ENGLISH TRANSLATION

OPERATIONAL GUIDELINES FOR DIGITAL TERRESTRIAL TELEVISION BROADCASTING

ARIB TECHNICAL REPORT

ARIB TR-B14 Version 3.8

(Fascicle 1) (2/2)

Established on January 24th, 2002
Revised on July 25th, 2002 Version 1.1
Revised on November 27th, 2002 Version 1.2
Revised on March 26th, 2003 Version 1.3
Revised on June 5th, 2003 Version 1.4
Revised on October 16th, 2003 Version 1.5
Revised on February 5th, 2004 Version 1.6
Revised on July 22nd, 2004 Version 2.0
Revised on September 28th, 2004 Version 2.1
Revised on December 14th, 2004 Version 2.2
Revised on March 24th, 2005 Version 2.3
Revised on May 26th, 2005 Version 2.4
Revised on September 29th, 2005 Version 2.5
Revised on November 30th, 2005 Version 2.6
Revised on March 14th, 2006 Version 2.7
Revised on May 29th, 2006 Version 2.8
Revised on September 28th, 2006 Version 2.9
Revised on December 12th, 2006 Version 3.0
Revised on March 14th, 2007 Version 3.1
Revised on May 29th, 2007 Version 3.2
Revised on September 26th, 2007 Version 3.3
Revised on December 12th, 2007 Version 3.4
Revised on March 19th, 2008 Version 3.5
Revised on June 6th, 2008 Version 3.6
Revised on September 25th, 2008 Version 3.7
Revised on December 12th, 2008 Version 3.8

Association of Radio Industries and Businesses

General Notes to the English translation of ARIB Standards and Technical Reports

1. The copyright of this document is ascribed to the Association of Radio Industries and Businesses (ARIB).

2. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, without the prior written permission of ARIB.

3. The ARIB Standards and ARIB Technical Reports are usually written in Japanese and approved by the ARIB Standard Assembly. This document is a translation into English of the approved document for the purpose of convenience of users. If there are any discrepancies in the content, expressions, etc., between the Japanese original and this translated document, the Japanese original shall prevail.

4. The establishment, revision and abolishment of ARIB Standards and Technical Reports are approved at the ARIB Standard Assembly, which meets several times a year. Approved ARIB Standards and Technical Reports, in their original language, are made publicly available in hard copy, CDs or through web posting, generally in about one month after the date of approval. The original document of this translation may have been further revised and therefore users are encouraged to check the latest version at an appropriate page under the following URL:
   http://www.arib.or.jp/english/index.html

Preface

The ARIB (Association of Radio Industries and Businesses) establishes the "ARIB Standards" and “ARIB Technical Reports” for the basic technical conditions such as standard specifications for a variety of radio communication equipment and broadcast transmission and reception equipment with the participation of broadcasting companies, broadcast equipment manufacturers, telecommunications carriers, radio communication equipment manufacturers and users.

This technical report encompasses materials related to “ARIB standards” which combine governmental technical standards and optional private sector standards.

This technical report stipulates provisions for general operations at broadcasting stations for digital terrestrial television broadcasting and functional specifications for digital terrestrial television receiver units.

We hope that this technical report will be put to practical use by broadcasting companies, broadcast equipment manufacturers, telecommunications carriers, radio equipment manufacturers and users.

<Intentionally blank>

Integrated Contents

Digital Terrestrial Television Broadcasting
General Information of the Technical Report ..... Fascicle 1- (1/2)
Volume 1 Digital Terrestrial Television Broadcasting
Provisions for Download Operations ..... Fascicle 1- (1/2)
Volume 2 Digital Terrestrial Television Broadcasting
Function Specifications for the Receiver..... Fascicle 1- (1/2)

Preface
Volume 3 Digital Terrestrial Television Broadcasting
Specifications for Data Broadcasting Operations ..... Fascicle 1-(2/2)

Preface
Volume 4 Digital Terrestrial Television Broadcasting
Provisions for PSI/SI Operations ..... Fascicle 2

Preface
Volume 5 Digital Terrestrial Television Broadcasting
Provisions for Conditional Access System(CAS)
Operations and Specifications on Receiver Units..... Fascicle 3
Volume 6 Digital Terrestrial Television Broadcasting
Provisions for Interactive Data Broadcasting Services..... Fascicle 3
Volume 7 Digital Terrestrial Television Broadcasting
Provisions for Carrier Operations ..... Fascicle 3
Volume 8 Digital Terrestrial Television Broadcasting
Provisions for Contents Protection ..... Fascicle 3
Volume 9 Digital Terrestrial Television Broadcasting
Provisions for Transmission Operations ..... Fascicle 3

<Intentionally blank>

ARIB TR-B14
Version 3.8-E1

Vol. 3

DIGITAL TERRESTRIAL TELEVISION BROADCASTING

Specifications for Data Broadcasting Operations

<Intentionally blank>

Contents

[Section 1] Common specifications for different types of reception....1

1 Introduction....1

2 References....1

3 Definitions....2

4 Type of operational specifications specified in Vol. 3....10
4.1 C-profile corresponding to different reception types....10
4.2 Relationship between assumed receiver units and each profile....10
4.3 Basic functions and optional functions....10

[Section 2] Operational provisions related to Profile A....11

1 Functions that basic receiver units should be equipped with to receive data broadcasting....11
1.1 Components of receiver units....11
1.1.1 Hardware components....11
1.1.2 Receiver unit reference model....12
1.2 Presentation functions....15
1.2.1 Pixel size/restrictions of each plane composing the display screen....15
1.2.2 Combination and restrictions of presentable planes....17
1.2.3 Relationship between mono-media encoding and the presentation plane....20
1.2.4 Audio playing function....24
1.2.5 Fonts....25
1.3 Remote control....26
1.3.1 Keys used in data broadcasting....26
1.3.2 Key masks....26
1.4 Memory that should be installed in receiver units....26
1.4.1 RAM....27
1.4.2 NVRAM....27
1.5 Communication functions....28
1.6 Character entry function....28
1.6.1 Function specifications....28
1.6.2 Character type....31

1.6.3 Kana Kanji conversion function 32
1.7 Bookmark list display function by receiver's native application (Optional) 33
1.7.1 Start/End bookmark list by receiver's native application 33
1.7.2 Bookmark list functions to be equipped by receiver's native applications 34
1.7.2.1 Display function of recorded bookmark titles 34
1.7.2.2 Functions that allow viewers to select desired bookmarks from a list and to select stations in accordance with specifications of the link destination URI 35
1.7.2.3 Delete management function 37
1.7.3 Startup of the bookmark list by receiver's native application by function 37
1.8 BML browser 37
1.8.1 Display by browser features 37
1.9 Registration transmission function (Optional) 38

2 Operation of the data transmission method 40
2.1 PSI/SI 40
2.1.1 Types of data broadcasting services 40
2.1.1.1 Data programs and TV programs 40
2.1.1.2 Types of data broadcasting service programs 40
2.1.1.3 "service_type" of channels that operate data broadcasting programs 41
2.1.2 Configuration of contents of data broadcasting service and operation of components 41
2.1.2.1 Contents and local contents 41
2.1.2.2 Relationship between local contents and ES 42
2.1.2.3 Operation of component tags 42
2.1.2.4 Identification of entry components 42
2.1.2.5 Restrictions on the number of ES 43
2.1.2.6 Detailed specifications of section data transmission 43
2.1.2.7 Default maximum bit rate of data broadcasting programs 43
2.1.2.8 Operation of video and audio components in data broadcasting services 43
2.1.3 Series reservation operation in data broadcasting services 45
2.1.4 Operation of the PMT specific to data broadcasting services 45
2.1.5 Operation of the Data Component Descriptor in the PMT 45
2.1.6 Operation of the Target Region Descriptor in the PMT 46
2.1.7 Operation of the H-EIT, M-EIT Data Contents Descriptor 46
2.1.8 Operation of the Data Contents Descriptors in H-EIT[schedule],M-EIT[p/f after] 49
2.1.9 Operation of EIT's specific to data broadcasting services 49

2.1.10 Performance of related receiver 50
2.1.10.1 BML-engine startup 50
2.1.10.2 Receiver operation when data broadcasting programs start 50
2.1.10.3 Receiver behaviour when updating the PMT 52
2.1.10.4 Handling of the dButton 53
2.1.10.5 Pixel size and aspect control in data broadcasting programs 55
2.1.10.6 Receiver operation when selecting stations 56
2.1.10.7 Reservation related behaviors of data broadcasting programs(Guidelines) 56
2.1.10.8 Specifications when the partial transport stream is output/input 57
2.1.10.9 the desired display of EPG, etc. 57
2.2 Operation of the independent PES transmission method 58
2.3 Data carousel transmission method/event message transmission method 58
2.3.1 Data carousel transmission operation 58
2.3.1.1 Introducton of data events and local contents 58
2.3.1.2 Operation of data events 58
2.3.1.3 Start/End of local contents 59
2.3.1.4 Introduction of the return flag 59
2.3.1.5 Local contents and the Data Contents Descriptor 59
2.3.1.6 Operation of empty carousels 61
2.3.1.7 Basic behavior of receiver during data broadcasting program presentation 62
2.3.1.8 Transmission of root certificates 64
2.3.2 Operation of “DownloadInfoIndication”(DII)message 65
2.3.3 Operation of “DownloadDataBlock”(DDB)message 67
2.3.4 Operation of event messages 68
2.3.4.1 Purpose for operating event messages 68
2.3.4.2 Transmission of event messages 68
2.3.4.3 Transmission of general event messages 69
2.3.4.4 Transmission of the NPT reference message 71
2.3.4.5 Event message process in receiver 73
2.3.4.6 Operation of the DSMCC_section( ) 77
2.3.4.7 Operation of General Event Descriptor 77
2.3.4.8 Operation of the NPT Reference Descriptor 77
2.3.5 Operation of IIT 78
2.3.6 Performance of related receiver 78
2.3.6.1 Filtering resources used when receiving data broadcasting 78
2.4 Temporary services and data broadcasting services 79

2.5 Multi-view operations and data broadcasting services 79
2.6 Interaction channel transmission protocol used for data broadcasting services....79

3 Operation of mono-media encoding 80
3.1 Video encoding 80
3.1.1 MPEG-1 Video 80
3.1.1.1 Restrictions of encoding parameters 80
3.1.1.2 Synchoronized playback with audio (MPEG-2 AAC) 80
3.1.1.3 Other restrictions 80
3.1.2 MPEG-2 Video 80
3.1.2.1 Restrictions of encoding parameters 81
3.1.2.2 Other restrictions 83
3.1.3 MPEG-4 Visual 83
3.2 Still images and bitmap figure encoding 83
3.2.1 JPEG 83
3.2.1.1 Encoding parameters 83
3.2.1.2 Scaling 84
3.2.1.3 Other restrictions 84
3.2.1.4 Operated markers and marker segments 84
3.2.2 PNG 85
3.2.2.1 Encoding parameters 85
3.2.2.2 Chunks operated by PNG 85
3.2.2.3 Other restrictions 85
3.2.3 MNG 85
3.2.3.1 Chunks operated by the MNG 85
3.2.3.2 Restrictions of MNG operation 87
3.3 Audio encoding 88
3.3.1 MPEG-2 AAC 88
3.3.1.1 Encoding parameters 88
3.3.1.2 Transmission of MPEG-2 AAC 88
3.3.1.3 Restrictions in data carousel distribution 88
3.3.1.4 Data format of the AAC audio file 88
3.3.2 AIFF-C 89
3.3.2.1 Encoding parameters 89
3.3.2.2 Maximum data volume 89
3.3.2.3 Other restrictions 89

3.3.3 MPEG-4 Audio 89
3.3.4 Additional sounds 89
3.3.5 Built-in sound 89
3.3.6 Audio synthesis by receivers 90
3.3.6.1 Mixing balance 90
3.3.6.2 Simultaneous playable encoding method 90
3.4 Character encoding 91
3.4.1 8-bit character codes (including EUC-JP) 91
3.4.1.1 Restrictions on the character coding function 91
3.4.1.2 Character sets used in data broadcasting 96
3.4.1.3 Initialization of character codes 97
3.4.2 Universal multi-octet coded Character Set 97
3.5 Description command figure encoding 97
3.5.1 Geometric 97

4 Operation of caption and superimpose encoding 98
4.1 Scope of service and definitions 98
4.2 Composition and transmission operation 98
4.2.1 Specification for composition and transmission 98
4.2.2 PES transmission methods used in caption 100
4.2.3 PES transmission methods used in superimpose 101
4.2.4 Operation of data groups 102
4.2.5 Operation of caption management data 103
4.2.5.1 Caption management data used in caption 103
4.2.5.2 Caption management data used in superimpose 104
4.2.6 Operation of caption statement data 104
4.2.7 Operation of data units 105
4.2.8 OPERATION OF PSI/SI 105
4.2.8.1 Operation of component tags 105
4.2.8.2 Operation of the PMT 105
4.2.8.3 Stream format identification 105
4.2.8.4 Descriptor operation 105
4.2.8.5 Data Component Descriptor 106
4.2.8.6 Target Region Descriptor 106
4.2.8.7 Data Contents Descriptor 106
4.3 Visual pixel size and display format of caption/superimpose 107

4.3.1 Display format 107
4.3.2 Display area 108
4.3.3 Initial operation location 108
4.4 Characters used in caption/superimpose 109
4.4.1 Character codes 109
4.4.2 Character fonts 109
4.4.3 Character size controls 109
4.4.4 Display zone 111
4.4.5 Non-spacing characters 122
4.5 Control codes used in caption/superimpose 123
4.5.1 Control codes 123
4.5.1.1 Operation of color specifications 127
4.5.2 Operation of flashing 132
4.5.2.1 Restricted matters 132
4.5.3 Raster color control 133
4.5.4 Operation of highlighting 133
4.5.5 Operation of underlines 133
4.5.6 Operation of outlines 134
4.5.7 Operation of scrolling 134
4.5.7.1 One line lateral direction scroll 134
4.5.7.2 Control codes 135
4.5.7.3 Scrolling speed 135
4.5.8 Priorities of the display function 135
4.6 Operation of the DRCS 135
4.7 Operation of initialization 136
4.7.1 Initialization by caption management 136
4.7.2 Initialization by caption statements 136
4.7.3 Initialization by data units in this document 136
4.7.4 Initialization by character control codes 137
4.8 Mono-media used in caption and superimpose 137
4.8.1 Operation of geometric 137
4.8.2 Operation of bit map data 137
4.8.3 Operation of alert sounds 137
4.8.4 Operation of additional sound 137
4.9 Expected performance of the receiver units 137
4.9.1 Start/end of caption display 138

4.9.2 Start/end of superimpose display 138
4.9.3 Set-up fields etc. in the receiver units 138
4.10 Roll-up mode (Optional) 139
4.10.1 Declaration of the roll-up mode 139
4.10.2 Operation of caption management data in the roll-up mode 139
4.10.3 Operation of caption statement data in the roll-up mode 139
4.10.3.1 Operation of the text data unit of caption statement data 139
4.10.4 Initialization of the roll-up mode 140
4.10.5 Presentation of the roll-up mode 140
4.10.6 Restrictions of PES packet transmission of roll-up mode 140
4.10.7 Compatibility with receiver units not equipped the roll-up mode 141
4.11 Caption outscreen display function (Optional) 141

5 Operation of multimedia encoding 142
5.1 Introduction 142
5.2 Operation of NVRAM used commonly in Multimedia services in Digital Terrestrial Television Broadcasting 142
5.2.1 Allocation of NVRAM area 143
5.2.1.1 Allocation of the A-profile memory area for the affiliation 143
5.2.1.2 Allocation of the A-profile memory area for the specified broadcaster and the A-profile memory area of communication purpose for the specified broadcaster 144
5.2.2 Identification of the A-profile memory area for the all broadcasters 145
5.2.3 Identification of the broadcaser affiliates area 145
5.2.4 Identification of the A-profile memory area for the specified broadcaster 146
5.2.5 Identification of the A-profile memory area of communication purpose for the specified broadcaster 146
5.2.6 Identification of the memory area for bookmark service 146
5.2.7 Use of the viewer residential area information from the Multimedia service 147
5.2.8 Operation of the A-profile memory area for the all broadcasters of receiver unit NVRAM 148
5.2.8.1 Purpose of the A-profile memory area for the all broadcasters 148
5.2.8.2 Format of the A-profile memory area for the all broadcasters 148
5.2.9 Specification relating to writing in the A-profile memory area for the all broadcasters 154
5.2.10 Specification relating to reading from the A-profile memory area for the all broadcasters 155
5.2.11 Specification regarding customer registration, changed contents 155

5.2.12 Specification when registering customer information to the center server 156
5.2.13 Contents description guidelines of NVRAM access 157
5.3 Use of remote control keys from Multimedia services 158
5.3.1 Values that used-key-list specification could take 158
5.3.2 Handling of remote control keys, key codes, and access keys 158
5.3.3 Guidelines of contents using selection by color keys 159
5.4 Operation of the BML version 159
5.5 Operation of character codes 159
5.5.1 Transmission of DRCS pattern data 159
5.6 Operation area of media type and mono-media 160
5.7 Operation of BML elements 160
5.7.1 Entity 160
5.7.2 Operational guidelines of BML elements 160
5.7.2.1 Restrictions on the order of elements within head elements 163
5.7.3 Attributes 164
5.7.4 Operation relating to beitem elements 167
5.7.5 Operation of the BML elements extension module (interruption event) 167
5.8 Operation of CSS 169
5.9 Operation guidelines relating to procedure description 169
5.9.1 Operation area of DOM 169
5.9.2 Operation area of embedded objects 169
5.9.3 Operation area of extended objects for broadcasting 169
5.9.4 Operation area of browser pseudo-objects 169
5.9.5 Operation range of navigator pseudo-objects 174
5.9.6 Extended functions that have been added in the digital terrestrial broadcasting 174
5.9.6.1 Print function (Optional) 174
5.9.6.2 Receiver application identification function (Optional) 174
5.9.6.3 Function to acquire root CA certificate information 174
5.9.6.4 Cache set-up function of communication contents (Optional) 174
5.9.6.5 Receiver’s native application startup function (Optional) 174
5.9.6.6 Specifiable values in arguments of “getBrowserSupport()” 175
5.10 Restrictions in the BML document description 175
5.11 Presentation control of BML document 175
5.12 Behavior of BML browsers 176
5.12.1 Operational guidelines relating to presentation 176
5.12.2 Operational guidelines relating to the performance of external characters 176

5.12.3 Behavior of the DOM 176
5.12.4 Operation of script language 177
5.12.5 Extended objects for broadcasting 177
5.12.6 Browser pseudo-objects 177
5.12.6.1 Operation of Ureg 177
5.12.6.2 Operation of Greg 177
5.12.6.3 Operation of EPG functions 178
5.12.6.4 Operation of series reservation function 179
5.12.6.5 Operation of the NVRAM functions 179
5.12.6.6 Interaction channel function-Operation of TCP/IP 179
5.12.6.7 Interaction channel function-Operation of the function that acquires the line connection status 179
5.12.6.8 Interaction channel function-Operation of large volume call acceptance service 179
5.12.6.9 Operation of operational control function 179
5.12.6.10 Operation of bookmark function 191
5.12.7 Embedded objects 192
5.12.8 Other specifications 192
5.13 Transmission of contents and name spaces 192
5.13.1 Mapping of scope to transmission systems 192
5.13.2 Restrictions related to reference of mono-media, etc. over media 192
5.13.3 Operation of name spaces 192
5.13.4 Operation relating to communication contents 195
5.14 Operation of communication contents 195
5.14.1 Operation guidelines relating to the presentation of communication contents 195
5.14.2 Guidelines relating to the performance of external characters of communication contents 195
5.14.3 Performance of DOM of communication contents 195
5.14.4 Operation of ECMA script execution for the script language of communication contents 195
5.14.5 Expansion object for the broadcasting of communication contents 195
5.14.6 Operation of browser pseudo-objects of communication contents 195
5.14.6.1 Ureg related functions 196
5.14.6.2 EPG functions 196
5.14.6.3 Series reservation function 196
5.14.6.4 NVRAM functions 197
5.14.6.5 Interaction channel function-Basic Procedure 197

5.14.6.6 Interaction channel function-TCP/IP 197
5.14.6.7 Interaction channel function/Function to acquire line connection status 203
5.14.6.8 Interaction channel function-Mass calls reception service 204
5.14.6.9 Operational control function 204
5.14.6.10 Receiver unit audio control 208
5.14.6.11 Timer function 209
5.14.6.12 External character function 209
5.14.6.13 Other functions 210
5.14.6.14 Closed caption display control function 210
5.14.6.15 Bookmark control function 211
5.14.6.16 Print related functions 211
5.14.7 Mapping the scope of communication contents 212
5.14.8 Operation guideline for transmission of communication contents 215
5.14.9 Restrictions related to mono-media reference from communication contents 221
5.14.9.1 Reference from broadcasting contents to communication contents 221
5.14.9.2 Reference from communication contents to broadcasting contents 221
5.14.10 Name space of communication contents 221
5.14.10.1 Restricted matters relating to URI 221
5.14.10.2 Operation of name spaces for servers 222
5.14.10.3 Operation of name spaces when referring to communication contents from broadcasting contents 222
5.14.10.4 Operation of name spaces when referring to broadcasting contents from communication contents 222
5.14.11 Guidelines of communication contents operation 224
5.14.11.1 Precautions during the operation of communication contents 224
5.14.11.2 Guidelines of communication contents descriptions 225
5.14.12 Guidelines of receiver unit performance when receiving communication contents 227
5.14.12.1 Guidelines of the receiver unit status transition performance 227
5.14.12.2 Guidelines for when receiver unit performance fails 231
5.14.12.3 Guidelines of receiver unit performance when receiving error responses 231
5.14.13 Guidelines for connecting/disconnecting 232
5.14.13.1 Receiver performance guidelines for disconnecting and re-connecting 232
5.14.13.2 Receiver unit performance guidelines when connecting by scripts 234
5.14.13.3 Receiver unit performance guidelines for auto-connection 235
5.14.13.4 Receiver unit performance guidelines when disconnected 236
5.14.13.5 Connection guidelines from receiver's native application 237

5.14.13.6 Display guidelines of confirmation messeages when connecting or sending data 237
5.14.14 Operation of root CA certificates 238
5.14.14.1 Operation of the memory area for root CA certificates 238
5.14.14.2 Storage of root CA certificates of general purpose to receiver units 239
5.14.14.3 Guidelines of receiver unit behavior for updating the root CA certificate of general purpose 240
5.14.14.4 Guidelines of receiver unit behavior when receiving root CA certificates of limited purpose by broadcasters 242
5.15 Operation of the bookmark service 244
5.15.1 Service type of the bookmark list service 244
5.15.2 Operation of the pre-list 244
5.15.3 Registration by bookmark type 244
5.15.4 Bookmark list display function by receiver's native application (Optional) 244
5.15.5 General address definitions relating to the bookmark service 244
5.15.6 Operation of the expansion data section of the memory area for bookmark service 245
5.16 Operation of registration transimission 245
5.16.1 Operation of the memory area for registration transmissions 245
5.16.2 Identification of the memory area for registration transmission 245
5.16.3 Operation of the bookmark function for registration transmission 246
5.16.3.1 Operation of the "writeBookmarkArray" function in registration transmission 246
5.16.3.2 Operation of the "readBookmarkArray" function in registration transmission 247
5.16.3.3 Operation of the "deleteBookmarkArray" function in registration transmission 247
5.16.3.4 Operation of the "lockBookmarkArray" function in registration transmission 247
5.16.3.5 Operation of the "unlockBookmarkArray" function in registration transmission 247
5.16.3.6 Operation of the "getBookmarkInfo2" function in registration transmission 247
5.16.3.7 Format of registration transmission information 248
5.16.3.8 Basic data section of registration transmission information 248
5.16.3.9 Expansion data section of registration transmission information 249
5.16.3.10 Operation of each field of registration information 251
5.16.4 Guidelines of registration transmission operation by contents 254
5.16.4.1 Operation when registration transmission is set up 254
5.16.4.2 Operation of registration transmission by contents 255
5.16.5 Registration transmission using registration transmission information by receiver's native application (Optional) 256

5.16.5.1 Registration of registration transmission information 256
5.16.5.2 Operation of registration transmission by receiver's native application 256
5.16.5.3 Reservation transmission function by receiver features 257
5.16.5.4 Guidelines for sending messages by receiver's native application 258

6 Operation of extended service (Optional) 260
6.1 Display of HTML contents (Optional) 260
6.2 Print function 260
6.2.1 Expansion API group 260
6.2.2 Print data format 261
6.2.3 Supplementary information regarding each print related API 262
6.2.4 Regarding presentation by receiver 263
6.3 Activation of the IPTV browser (Optional) 264
6.3.1 Activation and display of the IPTV browser 264
6.3.2 Returning to BML browser from IPTV browser 264
6.3.3 Confirmation of the IPTV browser function from BML content 265
6.3.4 Return values of the getResidentAppVersion() function 265
6.3.5 Positioning of the IPTV activation function startExtraBrowser() 265

Appendix-1 Receiver unit common fixed colors 266

Appendix-2 Module compression format 270

Appendix-3 Clip function in video plane 271

Appendix-4 Regarding the scaling of videos 273
Appendix 4-1. "Height", "width" attributes that are specifiable in object elements 273
Appendix 4-2. Display example of videos 275

Appendix-5 DTD for checking the operation area for basic services 280
Appendix 5-1 DTD 280
Appendix 5-2 Description of the DTD declaration section 285

Appendix-6 Precautions for Non-volatile memory access 286

Appendix-7 Root Certificate Descriptor 287

Appendix-8 getBookmarkInfo2 function 288

Appendix-9 Operation of argument of "getBrowserStatus()" 289

Appendix-10 Precautions at the time of switching video pixel size....290

Appendix-11 Operation guidelines of information on data broadcasting non-volatile memory 292

Appendix-12 Interaction channel communication based on connection status - Guidelines of return values for TCP/IP related functions 296

Appendix-13 Timing to initialize closed caption 297

Appendix-14 Arguments and return values for confirming whether the receiver implements the IPTV browser and its functions....298

[Section 3] Operational specifications related to profile B (T.B.D.)....299

[Section 4] Operational provisions related to C-profile 301

1 Introduction 301

2 (Unused number)....302

3 Functions that basic receiver units that receive video, audio and multimedia data 303
3.1 Configuration of the receiver units 303
3.1.1 Hardware configuration 303
3.1.2 Receiver unit reference model 304
3.2 Presentation function 306
3.2.1 Presentation technique of receiver units 306
3.2.2 Pixel sizes and restrictions of each plane configuring screen....307
3.2.3 Visual mono-media coding....308
3.2.4 Audio playback functions 310
3.2.5 Fonts 310
3.3 Buttons (remote controller) 310
3.3.1 Keys to be used in C-profile reception 310
3.3.2 Key masks....312
3.4 Memory capability of receiver units....312
3.4.1 RAM 312
3.4.2 NVRAM 313
3.5 Communication function 313
3.6 Character input function....314
3.6.1 Function specification....314
3.6.2 User interface....314

3.6.3 Character types 314
3.6.4 Kana Kanji conversion function 314
3.7 TVlink 315
3.7.1 Introduction of TVlink 315
3.7.2 What is a TVlink? 315
3.7.3 Changed points from profile A bookmarks 315
3.7.4 NVRAM in TVlink 316
3.7.4.1 Number of TVlink areas 316
3.7.4.2 Sharing with other medias 316
3.7.5 Format and details of TVlink 316
3.7.5.1 Format of TVlink 316
3.7.5.2 Details of each parameters 317
3.7.6 Operation of TVlink types 318
3.7.7 Guidelines for the functions for each TVlink type and receiver unit 320
3.7.8 Writing of TVlinks 322
3.7.9 Guidelines of TVlink list functions 323
3.7.9.1 Start and end of the TVlink list by receiver's native application 323
3.7.9.2 Display of TVlink information 324
3.7.9.3 TVlink selection and transition to the destination link 325
3.7.9.4 Deletion of TVlink information 325
3.8 BML browser 325
3.8.1 Browser specific display 325
3.9 Browser for the C-profile contents on the internet 326

4 Operation of data transfer format 329
4.1 PSI/SI 329
4.1.1 Type of data broadcasting services 329
4.1.1.1 Channel service type operated by data broadcasting programs 329
4.1.2 Configuration of the data broadcasting service contents and operation of the components 329
4.1.2.1 Contents and local contents 329
4.1.2.2 Relationship between local contents and ES 330
4.1.2.3 Operation of component tags 330
4.1.2.4 Configuration of the ES transmitted by 1 service 330
4.1.2.5 Identification of entry components 330
4.1.2.6 Identification of event message components 331

4.1.2.7 Detailed specifications for section data transmission 331
4.1.2.8 Default maximum bit rate for data broadcasting programs 331
4.1.3 Operation of PMT 331
4.1.3.1 PMT operation of specific data broadcasting services 331
4.1.3.2 Operation of the Data Component Descriptor in the PMT 332
4.1.3.3 Target Resion Descriptor 333
4.1.4 Operation of the L-EIT Data Contents Descriptor 333
4.1.5 Related receiver operation 333
4.1.5.1 Receiver operation at the beginning of data broadcasting 333
4.1.5.2 Receiver behavior during update of the PMT 334
4.1.5.3 Specifications on the partial transport stream output-input 334
4.2 Data carousel transmission operation 335
4.2.1 Data carousel transmission operation 335
4.2.1.1 Introduction of data event and local contents 335
4.2.2 Operation of data events 335
4.2.2.1 Operation of data events 335
4.2.2.2 Initiation and termination of local contents 336
4.2.2.3 Receiver basic behavior during data broadcasting program presentation 336
4.2.2.4 Operation when DII cannot be obtained for long time 336
4.2.3 Operation of empty carousels 337
4.2.4 Operation of the DownloadInfoIndication (DII) message 337
4.2.5 Operation of DownloadDataBlock (DDB) 340
4.3 Operation of event messages 340
4.3.1 Purpose for operating event messages 340
4.3.2 Transmission of general event messages 341
4.3.2.1 Operation of general event message data_event_id, event_msg_group_id 341
4.3.2.2 Regarding component specifications of general event messages from the BML document 343
4.3.2.3 Event message processing in receivers 343
4.3.3 Operation of DSMCC_section() 344
4.3.4 Operation of the General Event Descriptor 345
4.4 Receiver performance 345
4.4.1 Filtering resources used during data broadcasting reception 345
4.4.2 Desirable operation of the receiver in relation to the acquisition of C-profile modules 346
4.4.3 Receiver operation concerning updating the version 346
4.5 Auxiliary information function using general event message (Optional) 346

4.5.1 Operation of event_msg_id and general event message transmission 346
4.5.2 Transmission of the specified character string in private_data_byte 347
4.5.3 Operation of content.....349
4.5.4 Conditions for incorporation on the receiver 349
4.5.5 Conditions for presentation on the receiver 349
4.5.5.1 Presentation time.....350
4.5.5.2 Presentation.....350
4.5.5.3 Presentation update.....351
4.5.5.4 Deletion timing.....351
4.5.5.5 Character color, background color, character size and character type 351
4.5.5.6 Receiver operation in response to each specified character string 351

5 Operation of mono-media coding 355
5.1 Image coding 355
5.1.1 H.264|MPEG-4 AVC 355
5.1.1.1 Coding parameters 355
5.1.1.2 Picture format 355
5.1.1.3 Restrictions in the bit stream 355
5.1.1.4 Other restrictions 364
5.1.1.5 Identification of 16:9 pictures.....364
5.1.1.6 Operation of PanScan in the picture display area 364
5.2 Still picture and bitmap figure coding.....365
5.2.1 JPEG.....365
5.2.1.1 Coding parameters 365
5.2.1.2 Other restrictions 365
5.2.1.3 Operation markers and marker segments 366
5.2.2 GIF 366
5.2.2.1 Usable blocks 366
5.2.2.2 Header.....367
5.2.2.3 Trailer.....367
5.2.2.4 Logical Screen Descriptor.....367
5.2.2.5 Global Color Table.....367
5.2.2.6 Image Descriptor 367
5.2.2.7 Local Color Table.....368
5.2.2.8 Image Data 368
5.2.2.9 Graphic Control Extension.....368

5.2.2.10 Comment Extension 368
5.2.2.11 animation GIF operation restrictions 368
5.3 Audio coding 369
5.3.1 MPEG-2 AAC 369
5.3.1.1 Coding parameters 370
5.3.1.2 Transmission of MPEG-2 AAC 370
5.3.1.3 Limitations of data carousel transmission 370
5.3.1.4 Data format of MPEG-2 AAC files 370
5.3.2 AIFF-C 370
5.3.3 Built-in sound 370
5.3.4 MIDI 371
5.3.5 Audiosynthesis of receiver units 371
5.4 Character codes 372
5.4.1 8-bit character codes for C-profile 372
5.4.2 Shift JIS 372

6 Operation of closed caption coding (option) 372
6.1 Definition and range of service 372
6.2 Organization and transmission operation 372
6.2.1 Restrictions on organization and transmission 372
6.2.2 PES transmission method used in closed caption 373
6.2.3 Operation of data groups 374
6.2.4 Operation of closed caption management data 375
6.2.5 Operation of closed caption text data 376
6.2.6 Operation of data units 376
6.2.7 Operation of PSI/SI 377
6.2.7.1 Operation of component tags 377
6.2.7.2 Operation of the PMT 377
6.2.7.3 Stream format identification 377
6.2.7.4 Descriptor operation 377
6.2.7.5 Data Component Descriptor 377
6.2.7.6 Target Region Descriptor 377
6.2.7.7 Data Contents Descriptor 377
6.3 Display format of closed caption 378
6.3.1 Display format 378
6.4 Characters used in closed caption 379

6.4.1 Character entity 379
6.4.2 Character fonts 379
6.4.3 Character size 379
6.4.4 Display partition 381
6.4.5 Non-spacing characters 381
6.5 Control code used in closed caption 381
6.5.1 Control code 381
6.5.1.1 Operation of color specifications 383
6.5.2 Operation of flashing 383
6.5.3 Extension control code 383
6.6 Operation of the DRCS 384
6.7 Operation of the initialization operation 385
6.7.1 Initialization by switching classes in the data group 385
6.7.2 Initialization by closed caption text 386
6.7.3 Initialization by main text data unit 386
6.7.4 Initialization by character control codes 386
6.7.5 Initialization by receiver operation 386
6.8 Mono-media used in closed caption 386
6.8.1 Operation of geometric 386
6.8.2 Operation of bitmap data 386
6.8.3 Operation of built-in sound 386
6.8.4 Additional sound 386
6.9 Ideal receiver unit operation 386
6.9.1 Initialization and termination of closed caption display 387

7 Operation of multimedia encoding 387
7.1 Introduction 387
7.2 Operation of NVRAM in Digital Terrestrial Television C-profile broadcasting 387
7.2.1 Writing frequency of the NVRAM area 388
7.2.2 Allocation of NVRAM 388
7.2.2.1 Allocation of the C-profile area for the affiliation 388
7.2.2.2 Allocation of individual operator areas in the C-profile area for the affiliation 389
7.2.3 Deletion function of the C-profile area for the affiliation 390
7.2.4 Identification of the C-profile area for the affiliation 391
7.3 Operation of keys from the Multimedia service 392
7.3.1 Values handled for used-key-list characteristics 392

7.3.2 Correspondence of keys, key codes and access keys 392
7.4 Operation of the BML version 392
7.5 Operation of character coding schemes 393
7.6 Operation range of media type and monomedia 393
7.7 Operation of the BML elements 393
7.7.1 Declaration of XML and DOCTYPE 393
7.7.2 Operation of the BML elements 393
7.7.3 Attributes 396
7.7.3.1 Restrictions on the order of elements in the head element 402
7.7.3.2 Operation of bml:beitem element 402
7.7.3.3 Expansion of object elements 402
7.7.4 Entity 402
7.7.5 Operation of the BML element extended module (event) 402
7.8 Operation of CSS 403
7.8.1 Element applied each properties 406
7.8.2 Selector operation 407
7.8.3 Media type 407
7.8.4 Box model 407
7.8.5 Visual formatting model 409
7.8.6 Other visual effects 410
7.8.7 Colors and backgrounds 410
7.8.8 Fonts 411
7.8.9 Text 412
7.8.10 Default style sheets 412
7.8.11 length specification 412
7.9 extended property specification 413
7.10 Operations for procedural descriptions 415
7.10.1 Operations DOM 415
7.10.1.1 DOM Core Fundamental interfaces 415
7.10.1.2 DOM HTML interfaces 417
7.10.1.3 BMLCSS2Properties interface 428
7.10.2 Operation area of built-in objects 430
7.10.3 Operation area of extension objects for broadcasting 433
7.10.4 Operation area of Navigator pseudo-objects 434
7.10.5 Operation area of browser pseudo-objects 434
7.10.5.1 Operational rule of each extended function 438

7.10.5.2 Operation guidelines of function that generate communication....439
7.10.6 Extended functions provided by digital terrestrial broadcasting (1)....440
7.10.7 Extended function provided by digital terrestrial broadcasting (2)....440
7.10.8 Execution operation of the extended function....458
7.11 Restrictions in BML document descriptions....458
7.11.1 Restrictions of the script element....458
7.11.2 Positioning and restrictions....458
7.11.3 Vertical writing....460
7.11.4 Restrictions of element positioning....460
7.11.5 Operation related to object elements....461
7.11.5.1 Operation related to the presentation operation of animation GIF....461
7.11.6 Operation related to nested elements....461
7.11.7 Operation related to the p element....461
7.12 Operational guidelines related to presentation....462
7.12.1 Operational guidelines of object presentation....462
7.12.2 Presentation plane....462
7.12.3 Operation of monomedia presentation....462
7.12.3.1 Positioning by object elements and img elements....463
7.12.4 Guidelineson clipping....464
7.12.5 Range of values where CSS2 properties is applicable....464
7.12.6 Provision of box models....465
7.12.6.1 Attribute operation related to stream presentation behaviour....465
7.12.6.2 Presentation behaviour of animation GIF and operation of the streamstatus attribute....466

8 Guidelines for browser's behaviors....467
8.1 Guidelines for presentation....467
8.1.1 Principle of presentation in C-profile receivers....467
8.1.1.1 Principle of simultaneous presentation of video, audio and multimedia data....467
8.1.2 Presentation of video and audio....467
8.1.2.1 Presentation and control of broadcasting video and broadcasting audio....467
8.1.2.2 Display position and display size of video....467
8.1.2.3 Optimizing video display size....467
8.1.2.4 Display of QVGA 16:9 video....467
8.1.2.5 Changing the video display size....468
8.1.3 Operation during the startup of the TV reception function and tuning....468

8.1.3.1 Principle of display status when the TV reception function startups and during tuning 468
8.1.3.2 Principle of BML browser key acquisition 468
8.1.4 Prohibition of mixed display and simultaneous display 468
8.1.4.1 Princilple of mixed display prohibition 468
8.1.4.2 Simultaneous display of broadcasting screens and screens other than broadcasting (browsers for the contents on the internet, applications, etc.) 469
8.1.4.3 Various operations during simultaneous display 469
8.1.4.4 Operations of broadcasting content during simultaneous display 470
8.1.5 Prohibition of simultaneous display while displaying a part of broadcasting content (video audio only, audio only or BML browsers) 470
8.1.5.1 Principle of prohibition of simultaneous display while displaying a part of the broadcasting contents 470
8.1.5.2 Exceptions to simultaneous display prohibition when presenting only a part of the broadcasting contents 470
8.1.6 Simultaneous display of messages presented by receiver systems 471
8.1.7 Specifications regarding communication contents presentation 471
8.1.7.1 Contents presented in BML browsers 471
8.1.7.2 Activation of communication content other than C-profile linked content 471
8.1.7.3 Specifications to display receiver specific browsers for the contents on the internet 472
8.1.7.4 Common use of BML browsers and Browser for the C-profile contents on the internets 472
8.1.7.5 Browser to present with TVlink 472
8.1.7.6 Display of videos acquired from communication (optional) 473
8.1.7.7 Permission during communication generation and personal information dispatch 474
8.1.8 Presentation of multimedia data 475
8.1.8.1 Multimedia data display area 475
8.1.8.2 Exceptions in multimedia data display 475
8.1.8.3 Optimizing presentation size 475
8.1.8.4 Changing the size of BML browser display area 476
8.1.8.5 Image size possible for display 476
8.1.8.6 Character fonts and possible number of characters for display 476
8.1.8.7 Position of CSS and contents 477
8.1.9 Guidelines for contents creation 477
8.1.9.1 Pixel size assumed in contents creation 477

8.1.9.2 Character size controls and layout to be assumed in contents 477
8.1.9.3 Size of images used 477
8.1.9.4 CSS 478
8.1.9.5 Acquiring and releasing “*”, “#” and number keys 478
8.1.9.6 Notes on communication charges generated and information transmission 478
8.1.9.7 Permission when acquiring position information....478
8.1.9.8 Permission when acquiring receiver-unique identifiers or televiewer-unique identifiers....478
8.1.10 Display position for caption 479
8.1.11 Guidelines for DOM operation....479
8.1.12 Guidelines for external characters operation....479
8.1.13 Guidelines for ECMA script implementation....479
8.1.14 Guidelines for operation of extended objects for broadcasting 479
8.1.14.1 Operational specifications for BinaryTable objects....479
8.1.15 Guidelines for browser pseudo-object operations....479
8.1.15.1 Operations of Ureg 480
8.1.15.2 Operations of Greg 480
8.1.15.3 Operation of EPG functions....480
8.1.15.4 Interaction channel communication – Operation of TCP/IP....481
8.1.15.5 Operation of the operational control function....481
8.1.15.6 Non-volatile memory functions and operation of the data carousel storage function 493
8.1.15.7 Operation of functions related to TVlinks....494
8.1.15.8 Operation of special function for terrestrials 494
8.1.15.9 Operation of print functions (optional)....497
8.1.16 Built-in objects....499
8.1.17 Other restrictions....500
8.2 Transmission, reference and name space of content....500
8.2.1 Scope mapping to the transmission system....500
8.2.2 Restrictions when a mono-media is referred across a different media 500
8.2.3 Operations of name space....500
8.2.3.1 Restriction for the number of resources which are managed by receiver....501
8.2.3.2 Name space regarding multiple ES module locks 501
8.2.3.3 Name space regarding multiple ES module version watch 501
8.2.3.4 Reception operation and contents guidelines in #fragment operations....501

8.2.4 Reference guidelines of contents transmitted by components different from the BML document being presented 502
8.3 Operation of C-profile communicating content 503
8.3.1 Guidelines regarding presentation for C-profile communicating contents....503
8.3.2 Guidelines for operations of external fonts in C-profile communicating contents 503
8.3.3 Operation of DOM in C-profile communicating contents....503
8.3.4 Operation of ECMAScript implementation script in C-profile communicating contents...503
8.3.5 External objects for broadcasting in C-profile communicating contents....503
8.3.6 Operation of browser pseudo-objects in C-profile communicating contents 503
8.3.6.1 “Ureg” function....504
8.3.6.2 EPG functions....504
8.3.6.3 Interaction channel function -TCP/IP....504
8.3.6.4 Operational control functions....507
8.3.6.5 Receiver sound control....511
8.3.6.6 Timer functions....512
8.3.6.7 Other functions....512
8.3.6.8 Data carousel storage functions (T.B.D.)....513
8.3.6.9 Digital terrestrial broadcasting specific functions....513
8.3.6.10 Non-volatile memory functions....516
8.3.7 Communication contents scope mapping....516
8.3.8 Guidelines for C-profile linked contents transmission....520
8.3.9 Resource references between C-profile linked contents and broadcasting contents 520
8.3.9.1 Resource references from broadcasting contents to C-profile linked contents 520
8.3.9.2 Reference from C-profile linked contents to broadcasting contents 520
8.3.10 Name space in C-profile linked contents....521
8.3.10.1 Restriction for URI....521
8.3.10.2 Operation of name spaces in servers....521
8.3.10.3 Operation of name spaces upon transition from broacasting contents to C-profile linked contents....521
8.3.10.4 Operation of name spaces upon transition from C-profile link content to broadcasting content....522
8.3.11 Guidelines for operations of C-profile linked contents....522
8.3.11.1 Note on operations of C-profile linked contents....522
8.3.11.2 Guidelines for description of C-profile linked contents....522
8.3.11.3 Guidelines for receiver unit status transition operations....524
8.3.11.4 Guidelines for failure behaviour of receiver units....526

8.3.12 Guidelines for connection and disconnection 527
8.3.12.1 Guidelines for receiver upon automatic connection 527
8.3.12.2 Guidelines for receiver upon disconnection 527
8.3.13 Operation of certificates 528
8.3.14 Operation of User-Agent for identification of BML browsers and a browser for the C-profile contents on internet 528
8.3.15 Operation of X_DPA_getComBrowserUA() 530
8.3.16 Content transition and browser startup 531

9 Functions with which C-profile basic receiver unit that only receive video and audio 534
9.1 Configuration of the receiver unit 534
9.1.1 Hardware configuration 534
9.2 Presentation functions 535
9.2.1 Video presentation function 535
9.2.2 Audio playback function 535
9.3 Guidelines for reception operation 536
9.4 Guidelines for presentation 536
9.4.1 Presentation of video / audio 536
9.4.1.1 Display position and display size of video 536
9.4.1.2 Optimizing video display size 536
9.4.1.3 Display of QVGA 16:9 video 537
9.4.2 Presentation upon TV reception function startup and tuning 537
9.4.3 Restriction of mixed display of non-broadcasting screen and broadcasting screen and simultaneous display 537
9.4.3.1 Principle of mixed display prohibition 537
9.4.3.2 Prohibition of simultaneous display on receiver units that only receive video and audio 537
9.4.3.3 Simultaneous display of messages displayed by receiver unit 538
9.4.3.4 Operation of broadcasting contents upon simulaneous display 538

Appendix 1 Regards for NVRAM access 539

Appendix 2 Module compression format 539

Appendix 3 DTD for operation scope checking for basic service 539

Appendix 4 Default style sheet 546

Appendix 5 Operation scope for browser pseudo-object in Browser for the C-profile contents on the internets....547

Appendix 6 Operation of the getBrowserStatus() argument....553

Appendix 7 Communication restrictions for carriers....554

Appendix 8 Arrangement of elementary stream and empty carousel operations in the PMT....555

Appendix 9 Version update of the BML version....557

Appendix 10 Guideline concerning the operation of personal information in NVRAM and communications in data broadcasting....559

Appendix 11 Guideline concerning the cache function and URI history....560

Appendix 12 Details of the communication function....562

Appendix 13 Governing organization....566

[Section 5] Operational provisions of common digital receiver units for terrestrial/BS/wide bandwidth CS....567

1 Introduction....567

2 Functions required in common receiver units....567

2.1 RAM....567

2.2 NVRAM....567

3 Contents operation guidelines assuming common receiver units....569

3.1 Judgment of common receiver units....569

3.2 NVRAM access for other media....569

3.3 Channel selection of other media services....569

3.4 Viewing reservation and programmed recording of other media services....569

3.5 Information sharing between media....569

<Intentionally blank>

[Section 1] Common specifications for different types of reception

1 Introduction

Data broadcasting services for Digital Terrestrial Television Broadcasting will be executed in accordance with the Ministry of Public Management's ordinances/announcements(including the former Ministry of Posts and Telecommunications' ordinances/announcements)and standards outlined by the Association of Radio Industries and Businesses(ARIB)in the "Data coding and transmission specification for digital broadcasting"(ARIB STD-B24), and "Service information for digital broadcasting system"(ARIB STD-B10). However, in order to widely utilize such regulations, it will be necessary to create separate specifications regarding the operations in detail, so this document, "Digital Terrestrial Television Broadcasting Specifications for Data Broadcasting Operations", was created.

The operational specifications presented herein are based on preserving the flexibility of data broadcasting program arrangement for each respective broadcaster and expandability for data broadcasting service development in the future, and its' objective is to be a guideline for signal transmission and reception specifications for the sound operation of data broadcasting services in Digital Terrestrial Television Broadcasting.

Broadcasters of data broadcasting programs shall follow these transmission specifications for Digital Terrestrial Television.

Digital Terrestrial Television Broadcasting receiver units should be able to receive signals which are transmitted according to these guidelines. Also, adequate care should be taken to avoid malfunctions that might be caused by transmission of signals that stray from these guidelines.

2 References

The contents of this volume define the operation of data broadcasting in Digital Terrestrial Television Broadcasting based on stipulations specified in the following standards.

(1) "Receiver for digital broadcasting" standard ARIB STD-B21
(2) "Service information for digital broadcasting system" standard ARIB STD-B10
(3) "Transmission system for digital terrestrial television broadcasting" standard ARIB STD-B31
(4) "Data coding and transmission specification for digital broadcasting" standard ARIB STD-B24
(5) "Video coding, audio coding and multiplexing specifications for digital broadcasting" standard ARIB STD-B32
(6) "Conditional access system specifications for digital broadcasting" standard ARIB STD-B25

3 Definitions

16 : 9	Horizontal to vertical ratio of the display screen: 16 horizontal by 9 vertical
4 : 3	Horizontal to vertical ratio of the display screen: 4 horizontal by 3 vertical
8-bit character encoding	Lower overhead for character set switching compared to 7-bit codes; encoding system with improved transmission efficiency.
ARIB	Association of Radio Industries and Businesses (Corporation Aggregate). An association of broadcasters, telecommunications carriers and product manufacturers created to standardize use of radio-based technology within Japan.
BASIC PROCEDURE	Basic Mode Data Transmission Control Procedure: Communication procedure developed for basic hosting of data transmission control procedures and for terminal-terminal use. It features communication procedures to minimize inaccurate transmission of data.
CLUT	Color Look Up Table: Table to convert color information from an index value to a physical value.
DAVIC	Digital Audio-Visual Council: Name of the association whose objective was to define a standard method to transmit MPEG-digitized information reciprocally.
DRCS	Dynamically Redefinable Character Sets: Method to send external characters used in the standards of superimpose broadcasting and data broadcasting character encoding in patterns.
ES	Elementary Stream: : Basic stream. It corresponds to encoded video, audio, independent data in a PES packet. One ES is transmitted by the PES packet that has the same stream ID.
EUC-JP	Japanese character code encoded in accordance with ISO 2022
HTTP	HyperText Transfer Protocol: Application layer protocol. This protocol (RFC2616) is used for the transfer of data over the World Wide Web.
I frame	Intra Frame: Frames built from compressed data embedded within an initial frame.
IP	Internet protocol: Network layer protocol which defines Internet addressing and distribution processing of data.(RFC791)
ISO	International Organization for Standardization
Multimedia service	Data broadcasting service based on XML-based multimedia encoding methods.
MNG	Multiple-image Network Graphics: File format for animation graphics. It is pronounced “MING”. It includes multiple PNG images of which sequential displaying and repetition are possible.
MPEG-1	Moving Pictures Expert Group –1: MPEG1. Data compression coding technology including video and audio, which is standardized by the International Organization for Standardization (ISO/IEC 11172)

MPEG-2	Moving Pictures Expert Group –2: MPEG-2. Data compression coding technology including video and audio, which is standardized by the International Organization for Standardization (ISO/IEC 13818).
NPT	Normal Play Time: Absolute coordinates on time axis that show the positional relationship of events in a stream.
PES	Packetized Elementary Stream: Packetized stream. Packeted video, audio, and independent data of variable lengths.
PID	Packet Identifier: Packet ID (identifier). Shows the attributes of each individual stream of the corresponding packet in 13-bit stream identifying information.
PNG	Portable Network Graphics: Graphics file format succeeding GIF. It is pronounced “PING” and is capable of lossless compression. The file format is comprised of an 8-byte signature followed by a series of “chunks”.
RFC	Request for Comments: Technological information made public to the Internet community by the Network Working Group.
SBR	Spectral Band Replication: Low bit spectrum expansion technology by AAC.
TCP	Transmission Control Protocol: Protocol for the transport layer in end-end. Offers highly reliable connection transmission that includes error detection and correction. (RFC793)
TLS	Transport Layer Security: One of the protocols used to send/receive encrypted information via the Internet. This can prevent wire tapping, tampering of data, and Web spoofing by using a combination of security technologies such as public key encryption and secret key encryption, and digital certification.(RFC2246)
TS	Transport Stream: Transport stream standardized by the MPEG system standard(ISO/IEC 13818-1)
V.22bis	Modulation method for all double-layered telephone modems up to 2400bit/s specified by an ITU-T advisory
Kana-Kanji conversion	Process to convert input kana characters to appropriate kanji
Event	Program. Event in ARIB STD-B10
Entry component, entry carousel	Component whose component tag value is specified as 0x40 in the 2nd loop of the PMT is called “entry component”. Also, the data carousel transmitted in this component is called “entry carousel”.
Startup module	Module where moduleID=0
Startup document	Among all BML documents transmitted by the data carousel, this is the document that is presented first by default. Included in the startup module.
Stream format identification	Stream format identification specified in ISO/IEC 13818-1
Chunk	Blocked information found in the PNG and MNG file formats.
Data event	Period of time during which a BML document or a group of BML documents are transmitted in a component. Unrelated to SI events. Data events are switched based on the updating of the DII “data_event_id” transmitted in the component.
Data carousel	Method specified by ISO/IEC 13818-6 whose objective is to distribute data repeatedly in order to download various data via broadcasting.

TV programs with additional data	TV programs where additional data is broadcast along with an event in which video/audio are primary. Here, primarily audio programs are also considered “TV Programs”.
Visual component of TV programs with additional data	Parts other than data in TV programs with additional data.
BML-engine	Receiver software which receives and interprets multi-media data (BML documents) in order to present it to viewers.
Data broadcasting reception status	Status where receiver is receiving data broadcasting and playing it back.
Transport stream	Refer to the TS section.
Hash function	Mathematical function used to map large (very large in some cases) areas to smaller areas. Quality functions need to be simultaneously interaction channel and collision free.
Partial transport stream	Specially selected bit stream obtained by eliminating one or more transport packets not related the program from MPEG transport packets.
Font	Set of printed characters. Classified by style and size.
Font size	Same as design frame.
Bookmark list service	Service to broadcast “bookmark list display contents” which have the following features. Display bookmarked items as a list. When viewers select a desired bookmark from the list, and selects the channel to broadcast detailed information obtained in the URI from NVRAM bookmark information. Provides functions which manage bookmarks for viewers. The bookmark list service is not a service restricted by service type, but it refers to the service specified by bookmark list display contents.
Bookmark list data service	Service broadcast by service type 0xAA whose primary contents are “Bookmark list display contents”
Bookmark writing service	Service to broadcast “bookmark write contents” with the following features. Present bookmark icons with the timing that has been specified in the contents, following the broadcast contents. According to the buttons pressed by the viewer, write the URI etc. of the communication site providing related information to contents that are currently displayed to the bookmark area of NVRAM. In C-profile, refer to the “TVlink”.
Broadcasting bookmark *1	This is a function to write link information to a communication sites to provide information related to programs which have been included in BML contents according to buttons pressed by user, and is used in C-profile. It refers to “TVlink”.
Pre-list service	Service to broadcast “pre-list display contents” with the following features. List of services that provide the bookmark list service (organized channels). When the viewer selects the desired channel from the list, the specified service (composing channel) should be selected and the bookmark list service displayed.
Base URI directory	Identifier to recognize communication contents as one document group.
Multi-section	Format used to include and transmit more than one section into a single TS packet.

Multi-view	Multi-view TV. Multiple video and audio are broadcast within one service. Combinations of video/audio intended by broadcasting stations can be switched by using this method.
Real-time viewing	To acquire data for display from the data carousel in real time (i.e. while viewing).
Link state	Status where receiver units can receive/play both BML contents from a server managed by a broadcasting station and video/audio resources which are being broadcast, together.
Local Contents	Data carousel transmitted in a data event of a given component.
General header	Header added to messages, which is used with request/response messages.
VIDEO PES	Data component used to transmit encoded video as standardized in ISO/IEC 13818-1.
Response header	Header added to response messages, which is used only for response messages.
Response message	Unit of response received by HTTP/1.1 client.
Audio PES	Data component used to transmit encoded audio as standardized in ISO/IEC 13818-1.
Related data broadcast	Collective term for the additional data area of TV programs with additional data, and is a data broadcasting service assumed to be viewed alongside video.
Common key encryption	Also called secret key encryption/symmetrical encryption. Using common key encryption owned secretly by the sender/receiver, encryption on the sender side and decryption on the receiver side. The common key itself needs to be shared via other methods.
Common fixed color	Color specified as common on the receiver unit color palette for things such as logo display.
Public key encryption	Also called asymmetrical encryption. There are different keys for encryption (public key) and decryption (secret key). By managing the public key as public and secret key as secret, encrypted communication is possible without common secret information. Some public key encryption schemes (RSA encryption) have a signature function.
Viewing reservation	Used to make reservations to view programs by event, based on SI information.
Broadcaster color setup	Colors per broadcaster that can be set by CLUT index values and color combinations.
Closed caption	Service related to visual contents that display captions on top of the television broadcasting image.
Entity-header	Header added in order to indicate data attributes in case data is included in request/response messages.
Receiver dependent color	Colors per receiver that can be set by CLUT index values and color combinations.
Optional feature	When featured functions and performance depend on the receiver unit or product.
Uplink	Line used to connect to center equipment by using modems, etc.
2 displays of horizontal vertical picture element	Displays 1 pixel data in a 4 pixel area (horizontally and vertically).
Server	Web server capable of handling HTTP/1.1 on TCP/IP networks.

First (2nd, 3rd, 4th) level kanji	Standard levels of character codes. Specified by JIS X0208, etc.
Communication contents	BML contents located on servers
Independent data program	Data program principally involving multi-media data. In some cases, the video/audio components of TV programs are shared.
Unlinked status	Status where the receiver unit only receives and plays communication contents. In this state, receiver units cannot refer to broadcast video or audio resources.
Additional data	Data section of TV programs with additional data.
Superimpose	Closed caption service not synchronized with main video, audio and data. News flashes, component notification time tone, etc.
Request header	Header given to request messages used only for request messages.
Request message	Unit of request for HTTP/1.1 servers.
Temporary service	Service to create temporary SDTV channels by creating an extra bandwidth by lowering the bit rate of but keeping the fundamental component service. Emergency news services, etc.
Recording	To record broadcasting services in the transport stream or partial transport stream format on D-VHS or HDD. Recording function is optional for basic receiver units. (In case of referring to analog recording, write “analog recording” explicitly)
Programmed recording	To reserve (program) recording of programs by event unit, based on SI information.
Fringe area	Area located near the outer limits of the service area.
One-touch button	Key used for one-touch channel selection. Specifications specified in Volume II of this document.
Roll-up mode	Closed captionervice to display additional, prepared closed caption data sent as page data into areas of 3 lines, gradually line-by-line. Rolls up vertically at line breaks.
Closed caption out-screen display function	Function to reduce the video size in order to prevent overlapping of closed caption with captions within the broadcast image.
Root certificate	When encrypting communication by TLS, format pursuant to X.509 which is used to send and store public keys. In particular, certificates to authenticate the authorizer are called root certificates.
General-purpose root certificate	Root certificate that is stored in the receiver unit and is sent by a data carousel. When performing encrypted communication, the receiver unit refers to the general-purpose root certificate.
Root certificate exclusively for broadcasters	Root certificate transmitted by a data carousel which is used temporarily by receiver units.
Module for storing root certificates	Modularised root certificate used when transmitted by a data carousel. In the module for root certificate storage, a maximum of two root certificates are stored.
General-purpose root certificate storage area	Area allocated to NVRAM of receiver units to store general-purpose root certificates.

General-purpose root certificate storage number	Number to specify storage location of general-purpose root certificates and is location specified for storing general-purpose root certificates transmitted by data carousels.
General-purpose root certificate ID	ID to recognize the different types of general-purpose root certificates. The values are set by the certificate management group stated in Vol. 6.
General-purpose root certificate version	Value to indicate the version of general-purpose root certificates. The values are set by certificate management group stated in Vol. 6.
Registration transmission	Function to store/save telegrams in receiver units and send stored/saved telegrams per user instruction.
Registration transmission information	Term to refer to telegrams stored/saved in registration transmission and related information.
Reservation transmission	A function of registration transmission. Sends telegrams stored/saved by the registration transmission function at the time specified by the receiver unit. Registration (programming) of reservation transmission is done by the viewer.
BML browser	Data broadcasting browser which works based on this volume. In Profile A, it works when receiving data broadcasting and in linked, unlinked statuses, and received contents are presented via communication lines. In C-profile, it works when receiving data broadcasting and in linked status, and received contents are presented via broadcasting signals and communication lines.
Communication browser	General term to refer to the browser for the C-profile contents on the internet and the carrier's proprietary browsers. Mixed display with broadcast contents and execution of some scripts will be restricted.
C-profile communication browser	Communication browser which works based on the C-profile specifications in this volume. Execution of some scripts will be prohibited.
Carrier's proprietary browser	Browser implemented based on specifications defined by the carrier.
HTML browser	Browser with a feature to view HTML contents.
Broadcast contents	Broadcasting materials of video, audio, and data received via broadcasting signals.
8-bit character codes for C-profile	The character encoding method for closed caption used in C-profile, this is subset by restricting the use of control codes based on the "8-bit character codes" used in Profile A.
C-profile communication contents	General term for BML contents located on servers specifiedd in C-profile. Classified as C-profile linked contents and C-profile unlinked contents.
C-profile linked contents	Communication contents written under the assumption that a receiver unit in accordance with C-profile and with a linked status will present them.
C-profile unlinked contents	Communication contents written under the assumption that a receiver unit in accordance with C-profile and with an unlinked status will present them from a Browser for the C-profile contents on the internet.
Broadcaster contents	General term for broadcast contents and C-profile linked contents.
Communication carrier specification contents	Contents located on a server, scripted based on specifications specified by the communication carrier.

Mobile phone	A "specific mobile terminal equipment" as prescribed by the ministerial ordinance (Regulations for Enforcement of the. Telecommunications Business Law) provided by telecommunications carriers. (Note: From the viewpoint of protecting personal information, mobile phone is assumed to be a terminal compatible with TLS/SSL and root certificates.)
Low-frame-rate and low-resolution picture	Compared to those used for television services, this video's pixel size and frame rate is much lower.
Virtual plane	Plane to locate contents of data broadcasting. The virtual plane is defined by specifications and it is not required to be implemented by receiver units.
Color formatter	A method for converting pixel color information (such as converting YCbCr components into RGB).
H.264|MPEG-4 AVC	Advanced encoding/decrypting technology co-developed by the Moving Pictures Expert Group (MPEG) of the International Organization for Standardization/International Electro-technical Commission U.S., Inc.(ISO/IEC) and the Video Encoding Expert Group (VCEG) of the International Telecommunication Union(ITU).
NVRAM	NVRAM is Non-Volatile Random Access Memory. It is called a non-volatile memory area and it is an area that does not get turned off even when the power is turned off.
Affiliation	Network created in order for broadcasters to interoperate programs. It consists of a broadcaster called "key station" and multiple broadcasters called "net stations". A net station with multiple parent key stations is called a "crossnet station".
NIT	Network Information Table: Sends information related to transmission path information such as frequency and arranged channel, and all of arranged channel ID numbers included in one distribution system are addressed here.
PI	Processing Instruction. This is used to include instructions for applications in documents.
Mono-media	Independently presented media such as video, audio, characters, still images, etc.
Multi-media	Presented media composed from multiple mono-media. Involves mutual interaction between mono-media and often involves interaction with users.
DTD	Document Type Definition. Defines what kind of elements and attributes are used in a document.
Focus	Indicates the focal point of users in data broadcasting. Generally, a palette is highlighted or boxed or underlined to indicate where the focal point of the user is.
Module	One of the data units transmitted after being divided into blocks by a data carousel.
Resource	Resource often refers to mono-media, which is a component element of multi-media. However, it is called a resource when referred to as a raw material.
Selector(CSS)	Format to select for which element corresponding CSS characteristics are used.
Default style sheet	Refers to the CSS characteristics list which is implicitly assumed to have values specified within the receiver units. It has the lowest priority and it gets overwritten by values specified by external style sheets and style attributes.

Event	Refers to events in the computer world. This is differentiated by the word “event” in broadcasting, which refers to programs.
Simultaneous display	To display multiple different contents and statuses, etc. on one screen simultaneously.
Mixed display	When a content provider is intentionally associating its contents with other contents from a different provider and displaying them simultaneously, to have influences to display etc., in order to display contents from multiple different providers as if they are from the same source and thereby misleading the viewer.
Full screen view	Status where a part of a broadcaster's contents is displayed in full screen without displaying any other contents at the same time. Method to switch to full screen view is different for each terminal.
User-Agent	One of the HTTP Request headers that are sent to communication destinations from terminals that have communication functions.

(*1) Trademark rights for "BOOKMARK/ブックマーク" (Japanese Trademark No. 3281163) belong to Sony Corporation. Contact the governing organization described in [Section 4] Appendix 13 regarding the terms and conditions for the use of these trademarks.

4 Type of operational specifications specified in Vol. 3

4.1 C-profile corresponding to different reception types

In Digital Terrestrial Television Broadcasting, car TVs and portable type receiver units are considered as targets in addition to stationary type TVs and STBs. In each service, there are different presentation models of receiver units and different operational styles of mono-media as the contents target. Therefore, in this chapter, profiles are specified by the service type, and detailed specifications are specified in relation to each profile.

Table 4-1 Profiles for different types of receiver units

Profile	Contents
Profile A	Basic operation profile mainly targeting fixed receiver units (Stationary TVs, STBs, Portable TVs, etc.)
Profile B (T.B.D)	Basic operation profile of data broadcasting services mainly targeting transportable receiver units(car TVs, portable TVs, PDAs, etc.)
C-profile	Basic operation profile of data broadcasting services mainly targeting portable receiver units(mobile devices, etc.)

4.2 Relationship between assumed receiver units and each profile

Table 4-2 Relationship between profiles and receiver units

	Profile A	Profile B	C-profile	Comments
Fixed receiver units	O	Δ	Δ	
Transportable receiver units	T.B.D.	T.B.D.	T.B.D.	
Mobile receiver units	X	X	O	

O: Presentation is required for receiver units that feature data broadcasting functions

Δ: Optional

X: Reception is optional

4.3 Basic functions and optional functions

- Basic functions refer to the functions that receiver units with data broadcasting playback functions should be equipped with.
- All the functions other than the basic functions are optional; however, minimized operational specifications that should be followed in case the optional functions are realized will also be specified in this document.
- In Vol. 3, when it is written “should”, it refers to implementations (specifications) that are “not required” but are “preferable” from the broadcaster point of view.

[Section 2] Operational provisions related to Profile A

1 Functions that basic receiver units should be equipped with to receive data broadcasting

In this chapter, the functions that basic receiver units should be equipped with to receive data broadcasting are defined.

1.1 Components of receiver units

Basic receiver units are in compliance with Class A indicated in ”2. Example of receiver construction” in ARIB STD-B24 ”Data coding and transmission specification for digital broadcasting” Vol. 1 Part 1, Informative explanation. The provisions for each process part of basic receiver units from the hardware component perspective are defined as well as resources inside the receiver units by the reference decoder.

1.1.1 Hardware components

Basic receiver unit hardware components are shown in figure 1-1.

The digital broadcasting signal input into a basic receiver unit is converted into a transport stream by a tuner and demodulation. The demodulated transport stream is divided into video, audio, and other data by a transport stream decoding process, after which the video stream is output to a video decoding process and the audio stream is output to an audio decoding process. By going through the above process, playback of normal video and audio are performed by basic receiver units.

When receiving data broadcasting, data needs to be transferred to main memory or nonvolatile memory once to be processed by the CPU. Also, in addition to the playback process of normal video and audio, when presenting characters, data in main memory may be transferred to the video and audio decoder at the same time to perform the playback process of video and audio. Also, more interactive performance is expected compared to traditional TV viewing with the use of an uploading connection, etc. From the perspective of process performance of the above hardware, the following provisions need to be stated.

(1) Data receivable by transport decoders
(2) Playback of the stream system and accumulated audio data
(3) Playback of the stream system and accumulated video data.
(4) Presentation of video, still pictures, texts and graphics, etc.
(5) Interaction channel communication function using communication line
(6) Data size that can be saved indefinitely

(7) Data size that should be equipped in receiver units such as fonts.

(8) Memory size for obtaining and decoding data.

(9) Guidelines regarding remote control operation

In particular, (1) is defined as a TS decoder function, (2),(3),(4),(7) are defined as presentation functions, (5) is defined as a communication function,(6),(7),(8) are defined as memory volume, and (9) is defined as a remote control function.

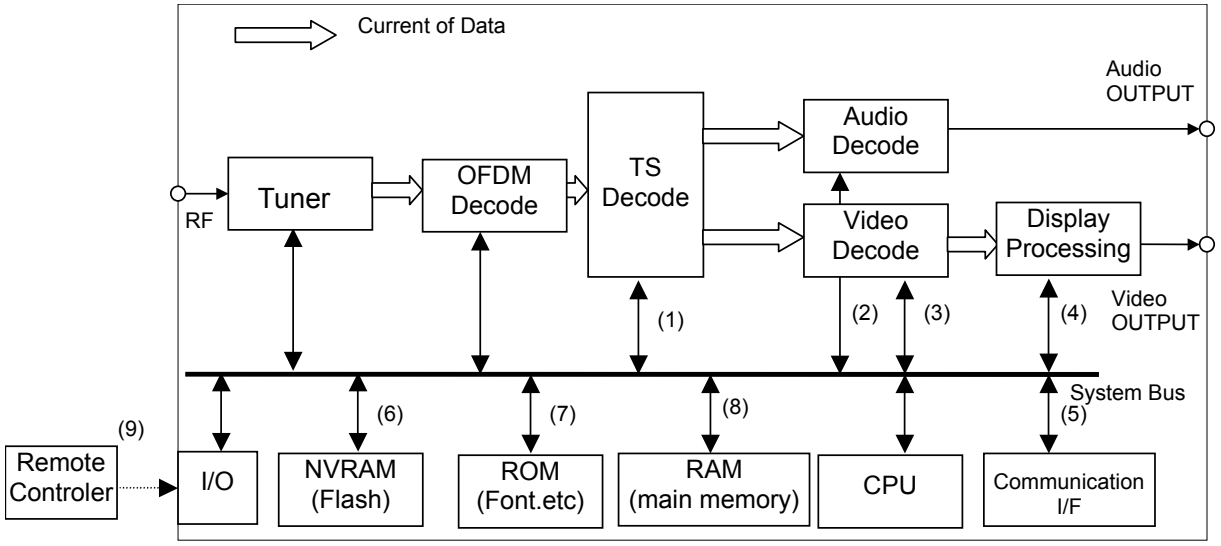


Figure 1-1 Hardware components of basic receiver units

1.1.2 Receiver unit reference model

By clarifying the resources inside the receiver units with a receiver unit reference model, the receivable limit of data broadcasting in a basic receiver unit is defined. The receiver unit reference model is shown in figure 1-2.

This reference model was created in compliance with provision items of this operation specification based on the decoder model indicated in DAVIC 1.4 Part 9. All units should be in compliance with definitions in DAVIC unless otherwise specified in subsequent sections of this document.

The received MPEG-2 TS is classified in each ES using the PID filter and elementary video and audio streams transmitted through the PES is stored in Bn, which is the primary buffer, through TBn, which is the transport buffer. On the other hand, elementary stream of MM contents transmitted by the data carousel is “section filtered” and stored in Bcontents through TBn after PID filtering.

Multimedia contents data received in this way are started by the receiver unit user, etc. The Multimedia engine follows this startup instruction and reads data inside Bcontents and performs execution of MM contents with Bwork as the memory for execution. In such case, mono-media content transmitted by a data carousel is given to each decoder from Bcontents, and mono-media content transmitted by streaming is given to each decoder from Bn. Mono-media content for audio systems is presented through speakers, which are the presentation device after decoding. On the other hand, after mono-media contents for video systems are decoded separately for video, still pictures, texts and graphics, and caption, they are buffered and combined into plane display memory once and presented through the monitor, which is the presentation device. Combinations of each plane are modeled as follows. In other words, effective pixels are switched separately for the video area and still picture area. This combined image of video and still pictures are mixed with the text and graphic plane by multiplication of the alpha value and its output is mixed with the subtitle plane by multiplication of the alpha value in the same way to compose the presentation screen.

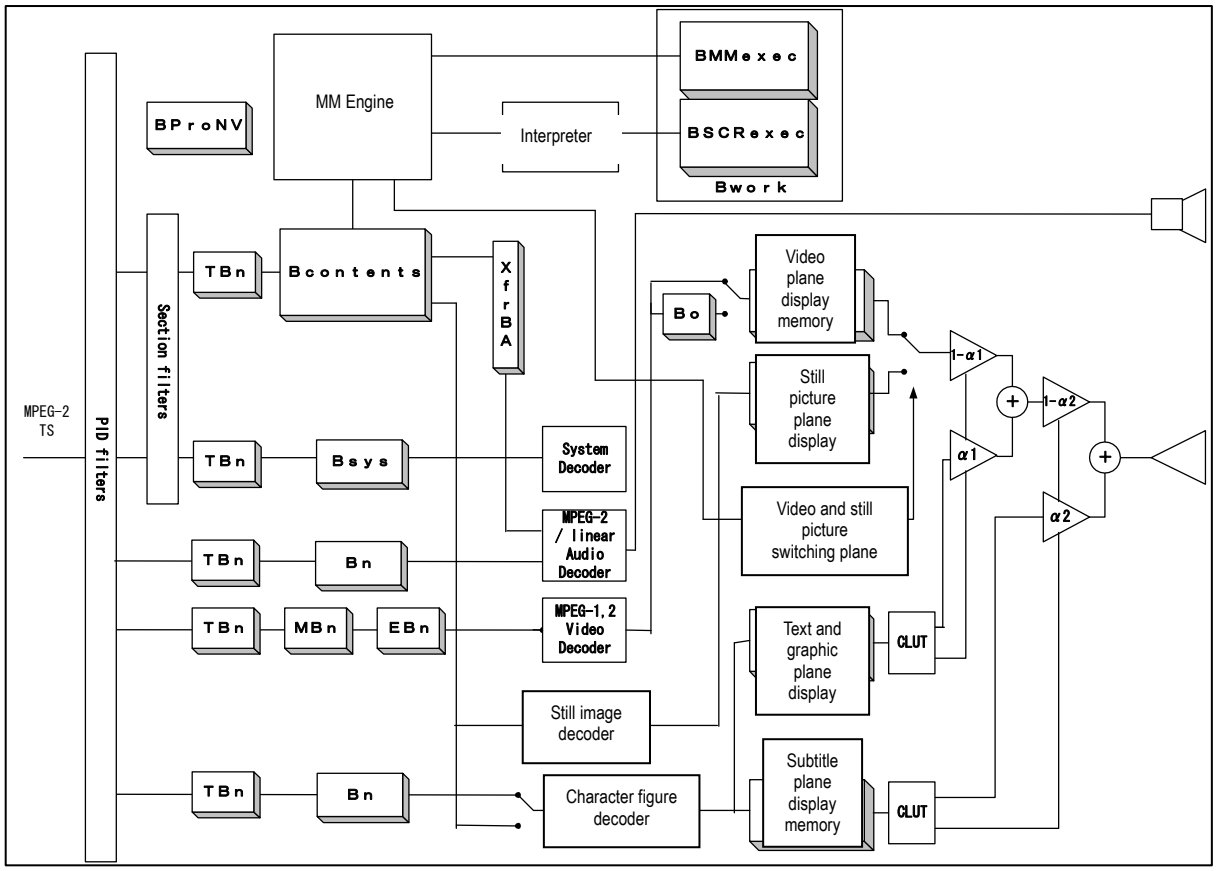


Figure 1-2 Reference model of basic receiver units

TBn	Transport buffer of the elementary stream *n*
Bn	Primary buffer of the elementary stream inside the decoder.
Bcontents	Buffer of multi-media contents data transmitted by the data carousel. When the module is transmitted with compression (Compression Type Descriptor of DII is placed), both data before and after compression (compressed, expanded) are buffered.
XfrBA	Buffer for transferring audio contents in file format to the audio decoder.
BMMexec	Execution memory of multi-media code
BSCRexec	Execution memory of script.
Bwork	Multi-media contents execution memory that is a combination of BMMexec and BSCRexec
BproNV	Non-volatile memory to store unique information for each receiver user and broadcaster.
Other buffers	Refer to ISO/IEC13818-1, DAVIC 1.4 Specification Part 9 for definitions.

1.2 Presentation functions

Presentation functions of a basic receiver unit are in compliance with presentation functions of receiver units indicated in Vol. 1, Part 1 of ARIB STD-B24 ”Data coding and transmission specification for digital broadcasting”.

1.2.1 Pixel size/restrictions of each plane composing the display screen

Provisions relating to the pixel size of each plane composing the display screen are shown in table 1-1.

Table 1-1 Pixel size of screen planes

Item	Description of provision	
Video plane	pixel size	1920x1080x16, YCbCr(4:2:2), 16:9
		720x 480x16, YCbCr(4:2:2), 16:9
		720x 480x16, YCbCr(4:2:2), 4:3
Still picture plane	pixel size	1920x1080x16, YCbCr(4:2:2), 16:9
		720x 480x16, YCbCr(4:2:2), 16:9
		720x 480x16, YCbCr(4:2:2), 4:3
text and graphic plane	pixel size	960x 540x8, 16:9 (*)Display size is 1920x1080(each pixel is rendered twice both horizontally and vertically)
		720x 480x8, 16:9
		720x 480x8, 4:3
	CLUT	CLUT Number:1 Common fixed colors :17 colors(Appendix-1 index value 0-16) Broadcaster setup colors :207 colors (Index value 17-223) Receiver unit dependent colors:32 colors (Index value 224-255) However, regarding index value of 0-127, each time of BML document presentation, the value indicated in Appendix-I should be set as the initial value by the receiver unit.
	Presentation	Present 8-bit index value of CLUT after converting it to α value of 4 bits and YCbCr(4:2:2). (*1)
Subtitle plane	pixel size	960x 540x8, 16:9 (*)Display size is 1920x1080(each pixel is rendered twice both horizontally and vertically)
		720x 480x8, 16:9
		720x 480x8, 4:3
	CLUT	CLUT Number:1 Common fixed colors :128 colors(Refer to Appendix-1) Receiver unit dependent colors:32 colors
	Presentation	Present 8-bit index value of CLUT after converting it to α value of 4 bits and YCbCr(4:2:2). (*1)
Video and still picture switching plane	pixel size	960x 540x1, 16:9 (*) Size is 1920x1080(picture element is divided into four equal-sized zones)

		360x 240x1, 16:9 (*) Size is 720x 480(picture element is divided into four equal-sized zones)
		360x 240x1, 4:3 (*) Size is 720x 480(picture element is divided into four equal-sized zones)

(*1) When setting a color map with 8-bit alpha value to the CLUT, the mosit significant 4 bits of the alpha value in the color map should be mapped to the 4 bits of the alpha value in the CLUT.
For a combination display of the text and graphic plane and the subtitle plane, nonlinearity in alpha blending is allowed. However, the objective is to allow a degree of flexibility in the design of blending circuit, and it is required to realize the presentation effect according to the 16 levels of alpha value.

Provisions relating to presentable mono-media code, the presentation position of mono-media content, and size, etc. are shown in table 1-2 as restrictions relating to each plane.

Table 1-2 Presentation restrictions on the screen plane

Item	Description of provision	
Video plane	Presentable mono-media code	MPEG-2 (*)Number of videos presented at once is 1 regardless of the encoding method.
		MPEG-1 (*)Number of videos presented at once is 1 regardless of encoding method.
	Presentation position	From even number picture elements to odd number picture elements of planes for both x and y coordinates. (*1)
	Size	Even number picture element for both x and y coordinates.
	Overlapping	Videos do not overlap with each other.
	Clipping	Only possible in the direction of x coordinates. (refer to appendix-3)
Still picture plane	Presentable mono-media code	JPEG
	Presentation position	From even number picture elements to odd number picture elements of planes for both x and y coordinates.
	Size	Even number picture element for both x and y coordinates
	Overlapping	No restrictions (*2)
Text and graphic plane	Presentable mono-media code	8-bit character codes including (*)EUC-JP
		PNG
		MNG
	Presentation position	From any picture element to any picture element for both x and y coordinates
	Size	Any picture element count for both x and y coordinates
	Overlapping	No restrictions(*2)
Subtitle plane	Presentable mono-media code	8-bit character codes
		Bit map data(*3)
	Presentation position	From any picture element to any picture element for both x and y coordinates
	Size	Any picture element count for both x and y coordinates

	Overlapping	Display operation to avoid the overlap of caption and superimpose as a general rule, but overlapping operations are possible. Refer to 4.9 for details.
Video and still picture switching plane	Switching position	From any picture element for both x and y coordinates.
	Size	Any picture element count for both x and y coordinates
Presentation switching effect	Process of presentation switching effect is implementation dependent.	

(*1) ARIB STD-B24 is followed for the definition of picture element.

(*2) Production of applications with no (or little) need to re-render still pictures, texts and graphics due to changes and moving of overlapping sequence is recommended. For receiver units, however, indication should not fail due to re-rendering. Please refer to section 1.2.2 for restrictions on overlapping of still pictures.

(*3) Refer to ARIB STD-B24 Vol. 1 Chapter 6 Presentable combinations of planes and restrictions

1.2.2 Combination and restrictions of presentable planes

As indicated in the reference model, the presentation screen is composed of a combination of various planes. Provisions for the combination of each plane are shown in table 1-3.

Table 1-3 Combination and restrictions of presentable planes

Item	Description of provision
Pixel size	For presentation of combinations of planes of video, still picture, text and graphic, subtitle, only those with the same pixel size and the same aspect ratio can be presented in combination. The text and graphic/subtitle plane in 960 x 540 is, however, it is recognized as 1920 X 1080. (*1)(*2)(*3)
	In the video and still picture switching plane, video to be switched/still picture plane pixel size in 1/2x1/2 pixel size is recognized as the same pixel size as the switching target plane. (*4)
Area specification in video and still picture switching plane	Area specification of either video or still pictures is rectangular. (*5)
Maximum area set up of video, still picture	When the rectangular area is video, the maximum area setup number is 1. When the rectangular area is all still pictures, the maximum area setup number is 4. (*5)

(*1) For presentation of contents with only video as the display target, a receiver unit that can perform presentation without composing the combination of above planes is conceivable. For such receiver units, the video display format indicated in Part 1 Chapter 5 of ARIB STD-B32 should be enforced for the transparent contents defined below. However, this will not interfere with the ability of receiver units to uniquely choose the effective areas of this provision. For example, in case the pixel size specified by the document is 960x540 16:9 and the pixel size of the video is 720x480 16:9, then in order to avoid blackouts at the time of transition from transparent contents to non-transparent, choosing to display transparent contents to be up-converted and presented as 1920x1080 16:9 video should be allowed. “Transparent contents” refer to BML documents where the “invisible” attribute of body elements are set as “invisible”. Even in such cases, “Precautions when switching video pixel size” indicated in Appendix-10 should be taken into consideration.

(*2) Even when aspect ratio of video is changed suddenly due to insertion of emergency broadcasting, etc., and the status becomes different from the data broadcasting pixel size, a combined display of TV

broadcasting and data broadcasting should be continued. Display behavior here is implementation dependent. Refer to Chapter 2 – "Operation of the data transmission method" for guidelines on receiver unit operation.

(*3) Combinations of text and graphic, subtitle, video, and still picture planes that can be used in data broadcasting simultaneously are called data broadcasting pixel size, and are defined as follows.

		Data broadcasting pixel size		
		960x540 (16:9) (*)	720x480 (16:9)	720x480 (4:3)
Text and graphic plane pixel size	960x540 (16:9)	O		
	720x480 (16:9)		O	
	720x480 (4:3)			O
Subtitle plane pixel size	960x540 (16:9)	O		
	720x480 (16:9)		O	
	720x480 (4:3)			O
Video plane pixel size	1920x1080 (16:9)	O		
	720x480 (16:9)		O	
	720x480 (4:3)			O
Still picture plane pixel size	1920x1080 (16:9)	O		
	720x480 (16:9)		O	
	720x480 (4:3)			O

(*) Text and graphic plane in data broadcasting pixel size 960x540 (16:9) is overlapped with video and still picture respectively with renderering each pixel twice horizontally and rendering each line twice vertically.

- Data broadcasting pixel size that can be combined with 720p video should be 960x540(16:9).
- In case new MPEG pixel size is defined along with the revisions of ARIB STD-B21, then inspection will take place as needed.

(*4) Full pixel size for both video and still pictures is YCbCr(4:2:2) and the effective unit for switching is 2 picture elements. This is to execute switching of video/still pictures by a video/still picture switching plane of 1/2x1/2 pixel size of video, still picture plane.

(*5) There are two visual patterns for display combinations of video plane and still picture plane.
The first pattern is the pattern where video is placed over a still picture as shown in figure 1-3. In this case, the rectangular area is a video area and the number of areas that can be set is 1.
The second pattern is where still pictures are placed over a full screen view video as shown in figure 1-4. In this case, the rectangular area is a still picture area and the number of areas that can be set is 4 or less. Also, as an example where multiple still picture areas are set as shown in figure 1-6, a case where the area is set with still picture 1 and 2 overlapping each other is conceivable. In this case, however, the area is not rectangular, so it is against these provisions and therefore not implementable. On the other hand, still pictures 3 and 4 in figure 1-5 do not form a rectangle, but in this case, this is understood as two rectangular areas that are sharing a border, and is recognized as in compliance with these provisions.

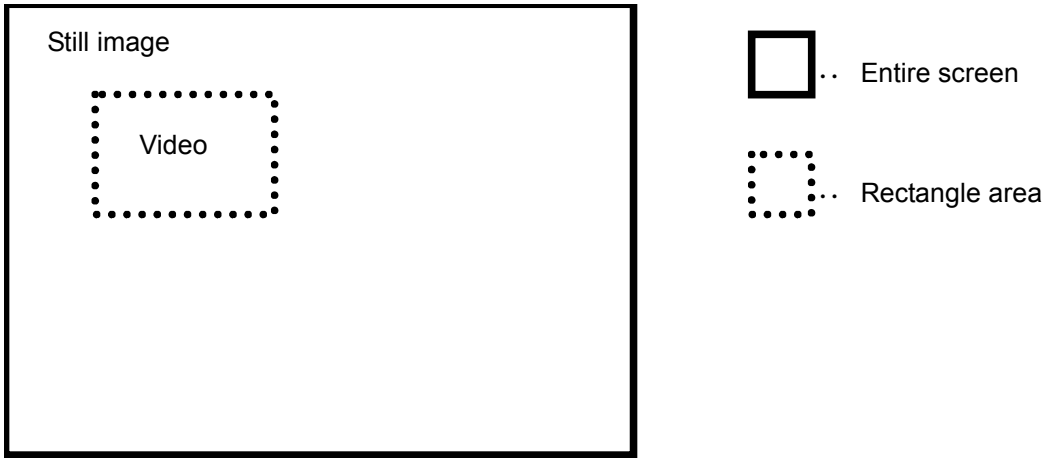


Figure 1-3 Display example 1 of possible combination of video and still pictures

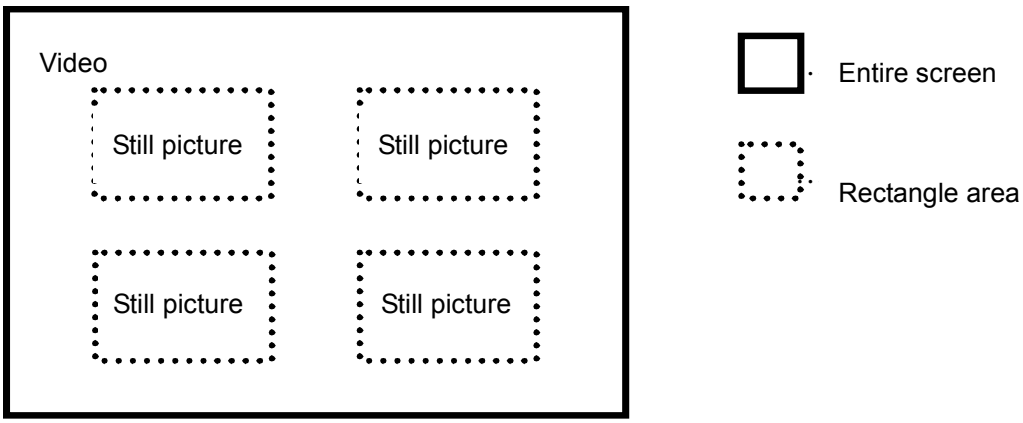


Figure 1-4 Display example 2 of a possible combination of video and still pictures

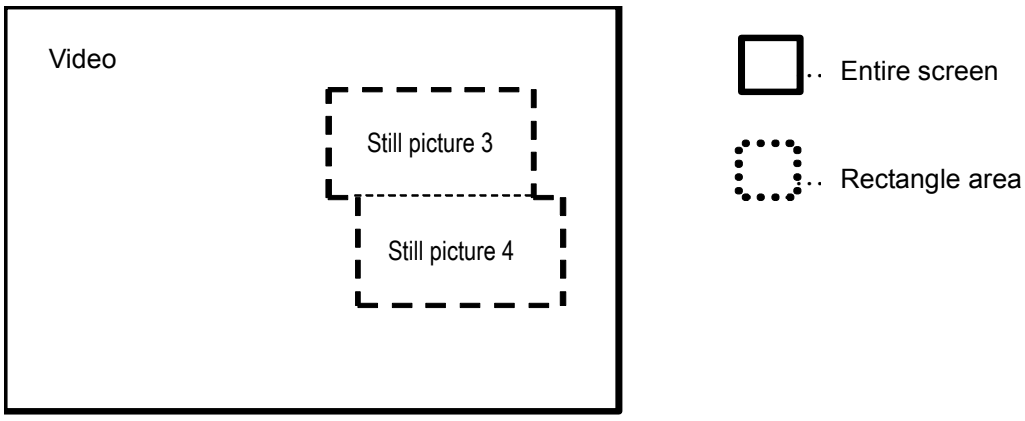


Figure 1-5 Display example 3 of a possible combination of video and still pictures

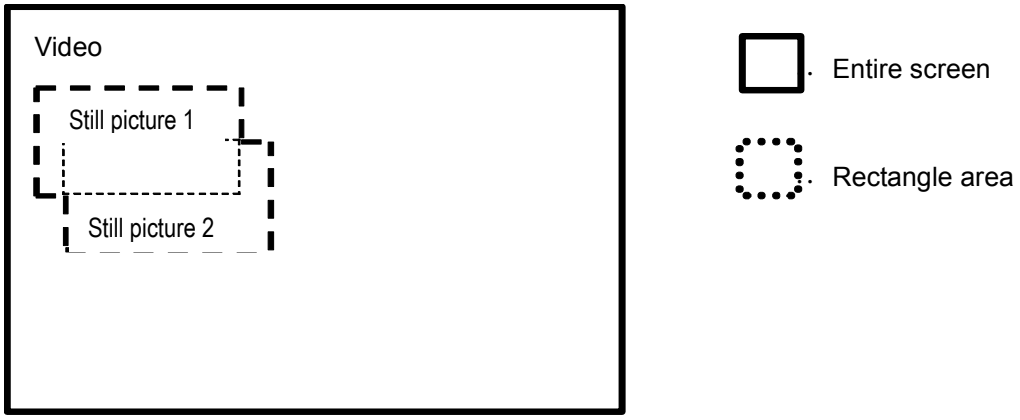


Figure 1-6 Display example of a impossible combination of video and still pictures

1.2.3 Relationship between mono-media encoding and the presentation plane

An overview of restriction conditions for mono-media coding presented in each plane of the above is described in Table 1-4. This is assuming the broadcaster will not be sending mono-media encoding that is not included in these provisions, or mono-media data encoded in non-standard operations. A detailed description of contents of provisions regarding each respective encoding method will be defined in Chapter 3.

Table 1-4 Overview of restriction conditions for mono-media encoding presented in each screen plane

Encoding method		Description of provision		
Video encoding	MPEG-2 (*1)	Transmission method	Screen image PES (*2);Stream format identification = 0x02	
		Image size(*3)	1920x1080 (16:9), 1440x1080 (16:9) 1280x720(16:9), 1920x1080 (4:3), 1440x1080 (4:3) 1280x720(4:3), 720x480 (16:9), 544x480 (16:9), 480x480 (16:9), 352x240 (16:9), 720x480 (4:3), 544x480 (4:3), 480x480 (4:3), 352x240 (4:3)	
		Scaling	256/128,192/128,160/128,128/128,112/128,96/128,80/128,64/128, 48/128,32/128(*4)	
	MPEG-1	Transmission method	Screen image PES (*2);Stream format identification =0x01	
		Image size	352x240(4:3, 16:9),176x120(4:3, 16:9) (*3)	
		Scaling	256/128,192/128,160/128,128/128,112/128,96/128,80/128,64/128, 48/128,32/128(*4)	
Still picture encoding	JPEG	Transmission method	Data carousel; Stream format identification = 0x0D	
		Image size	Any desired size from a horizontal vertical 16 picture element image to a full size image	
		Scaling	128/128(*5)	
		Others	Presentation in 4:2:0 pixel sizes is assumed. However, receiver unit indication should not fail by 4:2:2 input.	
Text and graphic encoding	PNG	Transmission method	MM ENCODING	Data carousel; Stream format identification = 0x0D
			Closed caption	Independent PES; Stream format identification = 0x06
		Image size	Any desired size from a horizontal vertical 2 picture element image to a full size image.	
		Scaling	128/128	
	MNG	Transmission method	Data carousel; Stream format identification = 0x0D	
		Image size	Any desired size from a horizontal vertical 2 picture element image to a full size image.	
		Scaling	128/128	
	8-bit character codes(*) Including EUC-JP	Transmission method	Independent PES;Stream format identification = 0x06 Data carousel; Stream format identification = 0x0D	

(*1) Possible to pseudo- display still pictures by transmitting only intra-frames. Restrictions for all such cases are in the category of video encoding. Therefore, in this case, simultaneous decoding with other types of video (MPEG-1, 2) is not possible. Also, when transmitting intra-frames above, the placement of a Video Decode Control Descriptor is required. The operation of a Video Decode Control Descriptor is in compliance with Vol. 4.

(*2) Stream packetized by PES packets, which was multiplexed with MPEG-2 TS

(*3) The physical picture element of planes has been defined as pixel size in this document. In case screen size of mono-media is not in accordance with this, mapping the plane is performed according to the following policy.

MPEG pixel size should be defined from Vertical Size, Horizontal Size, and Aspect Ratio in the “sequence header” in data encoded by MPEG. MPEG pixel size of video data presented in each respective video plane pixel size is as follows. The aspect ratio 4:3 of HD video data indicates the value of sent stream in the sequence header, and there is no difference from the 16:9 aspect ratio in appearance of the display.

MPEG VIDEO		MPEG pixel sizes	Video plane pixel sizes		
			1920x1080 (16:9)	720x480 (16:9)	720x480 (4:3)
HD video data	MPEG 2 VIDEO	1920x1080 (16:9)	O	X	X
		1440x1080 (16:9)	O	X	X
		1280x720 (16:9)	O	X	X
		1920x1080 (4:3)	O	X	X
		1440x1080 (4:3)	O	X	X
SD video data		1280x720 (4:3)	O	X	X
		720x480 (16:9)	O	O	X
		544x480 (16:9)	O	O	X
		480x480 (16:9)	O	O	X
		352x240 (16:9)	O	O	X
		720x480 (4:3)	O	X	O
		544x480 (4:3)	O	X	O
		480x480 (4:3)	O	X	O
		352x240 (4:3)	O	X	O
	MPEG 1 video	352x240 (16:9)	O	O	X
		176x120 (16:9)	O	O	X
		352x240 (4:3)	O	X	O
		176x120 (4:3)	O	X	O

(O:Presentable X:Not presentable)

(*4) Scaling factors are defined as follows by combinationsof MPEG pixel size and plane.(Appendix-4)

- HD video data presentation in video planes of 1920x1080 pixel size or scaling ratio when presenting SD video data in 720x480(16:9,4:3) = consideration of video data presentation by 128/128(100%) is defined as follows:

1) Quantize video data in each respective picture element of MPEG pixel size
2) Quantized video data is presented in the following number of horizontal/vertical pixels on the video plane:
   – Number of vertical direction picture elements is the same as vertical direction picture elements for video data.
   – The number of horizontal picture elements is the aspect ratio of the number of vertical picture elements x video data.

   However, video data with an MPEG pixel size of 352x240 gives side panels of 4 picture element size on both the left side and right side respectively, and is presented on the video plane with a horizontal direction total of 360 picture elements. Video data with an MPEG pixel size of 176 x 120

gives a 2 picture element size for side panels and is presented on the video plane with a horizontal total of 180 picture elements. When side panels are given by receiver units this way, the display format will be implementation dependent. Therefore, it is recommended to hide the side panel parts using another plane in multi-media content.

– Video data of 1280x720(16:9) will be a scaling ratio = 128/128 by mapping (vertical picture element count 1080, horizontal picture element count 1920) in the video plane of 1920x1080 (16:9) pixel size.

• HD video data presented in a video plane of 1920x1080 or consideration of video data presentation by a scaling ratio = n/128 in case of presenting SD video data in 720x480(16:9,4:3) is as follows.

1) Determine video plane pixel size. Determination method of the video plane pixel size is as follows;
   – Determine data broadcasting pixel size based on “document_resolution” specified by the BML document.
   – Determine video plane pixel sizes corresponding to data broadcasting pixel size
2) Quantize video data to the picture element of each respective MPEG pixel size.
3) In accordance with the logic written previously, calculate vertical/horizontal picture element count in case of scaling ratio = 128/128.
4) Present quantized video data in the following vertical horizontal picture element count on the video plane.
   – Vertical direction picture element count is the same as vertical picture element count of video data specified in height property for object elements in the BML document.
   – Horizontal picture element count is vertical picture element count x aspect ratio of video data. If the horizontal vertical direction becomes a non-even picture element count at this point, then the picture elements are rounded by truncating. Truncating is done from picture elements with a larger number of picture elements (lower right) for both horizontal and vertical. (Refer to ARIB STD-B24 for the picture element count) However, presentation where horizontal picture elements do not match the value specified in the width property of the object element is implementation dependent.

• Logic for scaling ratio when displaying SD video data on a video plane of 1920x1080 pixel size is defined as follows:
  Define with flexibility to change to a video pixel size of 960x540 data pixel size and also with consideration to secure image quality of the video.

Definition to secure flexibility to video pixel size change:

First of all, define the scaling ratio =128/128 as follows:

1) Quantize video data in each respective MPEG pixel size picture element.
2) Map the quantized video data for the vertical horizontal picture elements on the following video plane:
   – Vertical picture elements are converted to vertical picture elements on the video plane.
   – Horizontal picture elements are converted to a picture element count that maintains the roundness ratio against the vertical picture elements above.
     Next, in case the scaling ratio = n/128, use the above scaling ratio = 128/128 as the standard, and the vertical picture element count is the same as the vertical picture elements for video data specified by the height property for object elements in the BML document.
   – The horizontal picture elements are the vertical picture elements x aspect ratio of video data. If the horizontal vertical direction becomes non-even picture elements at this time, then the picture elements are rounded by truncating. Truncating is done from picture elements with a larger number of picture elements (lower right) for both horizontal and vertical. (Refer to ARIB STD-B24 for information on picture elements) However, the presentation of horizontal picture elements, which do not match the value specified in the width property of object elements is implementation dependent.

Definition with considerations to secure the picture quality of video:

First of all, define the scaling ratio =128/128 as follows:

1) Quantize video data to picture element of each MPEG pixel size.
2) Map quantized video data to the following vertical horizontal picture elements on the video plane.:

- Vertical picture elements are vertical picture elements on the video plane.
- The horizontal picture element count is converted to picture elements that maintain the roundness ratio against the above vertical picture elements.
  Next, in case the scaling ratio = n/128, then use the above scaling ratio = 128/128 as the standard and set the vertical picture element count to the same number as the vertical picture element count of video data specified by the height property for object elements in the BML document.
- The horizontal picture element count is the vertical picture element count x aspect ratio of video data. If the horizontal vertical direction becomes a non-even picture element count at this point, then the picture elements are rounded by truncating. Truncating is done from picture elements (lower right) with a larger number of picture elements for both horizontal and vertical. (Refer to ARIB STD-B24 for information on the picture element count) However, the presentation for when horizontal picture elements do not match the value specified in the width property of object elements is implementation dependent.

However, in either case, video data in 352x240 MPEG pixel size should be given a 4 picture element size side panel on both the left and right sides to make the horizontal size 360 picture elements, and also video data in 176x120 MPEG pixel size should be given a 2 picture element sized side panel on both the left and the right sides, and after editing the horizontal length total of 180 picture elements, apply the above definitions.

(*5) 256/128 scaling is operated only when transmitting pictures of a 960x540 pixel size and presenting it in a picture element divided into 2 equal-size displays as 1920x1080 size pictures.

1.2.4 Audio playing function

Regulations regarding playback of audio are shown in Table 1-5. This assumes that mono-media coding which is not described here, and mono-media data which is encoded by operations outside these provisions will not be sent or operated by broadcasters. Detailed descriptions of contents of the provisions regarding respective encoding methods are defined in Chapter 3.

Table 1-5 Audio playing function

Encoding method	Description of provision	
AAC-LC	Transmission method	Audio PES;Stream format identification = 0x0F Data carousel; Stream format identification = 0x0D
	Sampling frequency	48kHz,32kHz (*1)
	Maximum file size possible for continuous play	512KB
	Others	(*1)(*2)
AIFF-C	Transmission method	Data carousel; Stream format identification = 0x0D
	Sampling frequency	12kHz(*) 1/4 of main audio stream
	Maximum file size possible for continuous play	96KB
	Others	(*1)(*2)
Closed caption warning sounds	Transmission method	Built-in sound(*3)
	Sampling frequency	12kHz
	File size possible for continuous play	48KB

(*1) For restrictions, such as whether or not simultaneous decoding of video data and/or additional audio data is possible, see chapter 3.

(*2) For decoding file format data transmitted by data carousels, allows silent time when the sound repeats.

(*3) Size of total ROM for built-in sound is 480KB.

1.2.5 Fonts

Regarding fonts, considering the size of ROM featured in receiver units, restrictions have been added without affecting practicality. Font specifications are shown in Table 1-6.

Table 1-6 Fonts

Item	Description of provision
Number of font styles	Number of styles:3(round gothic, bold round gothic, angle gothic) (*)shared for 960x540 and 720x480 Proportional font: none
Character type	Kanji (1st, 2nd level), Hiragana, Katakana, alphanumeric, symbols, etc. (*1)
	External characters: YES
Character size controls (pixel)	Round gothic 16, 20, 24, 30, 36 Bold round gothic 30 Angle gothic 20, 24
Grayscale font	4 shades

(*1) Refer to 3.4.1.2 for details on character types

(*2) Character decoration such as hemming is assumed to be software implemented, but it is not mandatory. Details on implementation including whether or not to use exclusive fonts, is outside of the scope of this document.

1.3 Remote control

1.3.1 Keys used in data broadcasting

The types of remote control keys used in data broadcasting and guidelines for content production are shown in Table 1-7. In order to avoid user confusion, multiple meanings should not be assigned to one button. When assigning multiple meanings to one button, operation content should be explicitly explained to the user within the contents. Having a bookmark button is preferred for remote control used in data broadcasting. Featuring a bookmark button is optional.

Table 1-7 Remote control keys used for data broadcasting

Key type	Guidelines
↑、↓、←、→ (up, down, left, right keys)	To move up, down, left, right.
0 - 9 (number keys)	To input numbers
Enter	Separator of operation (enter)
Return	Cancel operation
	Back space of user input character (or bulk erase)
	Disconnection of a call to a communication server (*)During connection, receiver units will take the instruction; after connection, instruction is carried out in the contents. (A display to the effect that the connection will be terminated is desirable when the back key is pressed.)
	(*)It is okay to use BML documents for the purpose of going back. However, whether or not there is something after returning should be considered.
d	Data button. Switches display/non-display of multi-media data broadcasting.
Blue, red, green, yellow(color keys)	Selection of operation (execution) (*)Location of buttons on the remote control should be in order of blue, red, green, yellow from the left and each button should have the corresponding words “blue”, “red”, “green” and “yellow” displayed.
Bookmark (Optional)	Recording of bookmark.

1.3.2 Key masks

Multi-media contents are in compliance with ARIB STD-B24 and are able to perform masks for keys. However, keys related to selecting stations (one-touch select button, channel up/ddedicated button, screen image key) should not be masked by contents except during on-line communication. Masks on number keys (one-touch select button) should not be performed unless number input is necessary. Masks should be released once the input is over.

1.4 Memory that should be installed in receiver units

Refer to section 1.1.2 for the definition and location within the receiver units for each memory in basic receiver units

1.4.1 RAM

As shown in the reference model, a receiver unit will feature various memories. In this section, among all the memories that are assumed to use RAM, Bcontents, XfrBA will be defined in particular. Each RAM size is shown in Table 1-8. Refer to section 1.1.2 for details.

Table 1 -8 RAM

Item	Description of provision
Bcontents	More than 5MB
XfrBA	512KB

For the buffer size for caption and superimpose, refer to chapter 4.

1.4.2 NVRAM

As primary nonvolatile memory featured in receiver units related to data broadcasting, there is BproNV which stores unique information on each receiver unit user and broadcaster which is indicated in the reference model. The area type and capacity that BproNV should have is shown in Table 1-9.

Table 1-9 Type and capacity of BproNV

Type	Capacity
A-profile memory area for the all broadcasters	2KB
A-profile memory area for the affiliation	More than 32KB
A-profile memory area for the specified broadcaster	More than 48KB
A-profile memory area of communication purpose for the specified broadcaster	More than 24KB
Memory area for bookmark service	More than 16KB
Memory area for root CA certificates	24KB
Memory area for registration transmission	More than 4.5KB

For details of BproNV, refer to 5.2 or 5.15. The area indicated in Table 1-9 should, from the confidentiality perspective, be constructed so that viewers cannot read or write using other functions of receiver units not defined in standards of related documents or in provisions of this document or by using devices connected to receiver units. In order to satisfy the above restrictions, the corresponding areas should be located in the memory of the NVRAM, etc. receiver unit.

Refer to 1.1.2 “Receiver unit reference model” for details.

A device with limits on the number of times of writing for NVRAM equipped in receiver units is used. Function failure will occur on those devices once the writing times exceed its limit, and as a result, the lifespan of this receiver unit becomes shorter. Therefore, adequate attention should be paid to avoid exceeding the amount of writing to NVRAM. Regarding this issue, it is explained in Appendix-6.

1.5 Communication functions

The communication functions assumed in interaction channel communication for data broadcasting services are shown in Table 1-10.

Table 1-10 Communication functions

Item	Description of provision
Protocol	• TCP/IP communication protocol is used. Refer to Vol. 6, Chapter 6 for details on the TCP/IP communication protocol, • HTTP1.1(RFC2616) is used as the application layer. • Receiver units equipped with modems should support the BASIC system procedure (Code Independent Mode)
Security function	In case a scheme is specified by http: for BML contents, then TLS1.0 security should be followed. Refer to Vol. 6, Chapter 8 for details on the TLS operation.
Communication speed	Speed faster than 28.8kbit/s is preferred.

1.6 Character entry function

The character entry function, assuming there is a software keyboard, etc. for the purpose of supporting character entry to BML contents by viewer operation, is defined as a resident application. Functions that this application should be equipped with as standard are described in the following sections. Functions not defined in this document such as the *kana kanji* conversion function are implementation dependent.

1.6.1 Function specifications

This application is to support character entry to “input” elements in BML documents. An “input” element of which “readonly” attribute is not effective must be focused on for this application to start up, and this application must behave as specified by the attributes of the input element, described as below..

● “inputmode” attribute

Specifies whether or not text input frontend application should be started, and also specifies the format of the startup.

In case “direct” or “indirect” is specified in this attribute, whether or not “click”, “keyup”, “key down” events will occur by pressing the “enter” key is implementation dependent.

Table 1-11 “inputmode” attribute

Attribute value	Contents
direct	The “input” element is focused on and after application of the focus style and occurrence of the focus event, the text input frontend application is started immediately. (*1)(*2)
indirect	The “input” element is focused on and after application of the focus style and occurrence of the focus event, the text input frontend application is started upon pressing of the enter key. (*2)(*3)
none	Text input frontend application is not started.(Behaviour of the input element is in compliance with ARIB STD-B24)

(*1) In case a focus transition to the “input” element has occurred by pressing down a key, (i.e. the focus is moved according to "nav-index" property or the fouces() method is applied to the input element by “onkeydown”, “onclick” event handler)whether or not the processes defined in [6] of ARIB STD-B24 Vol. 2 Appendix 1 5.4.2.1 is executed is implementation dependent. The "onkeyup" event handler should not be specified for “input” elements for which the “inputmode" attribute is specified. For details, refer to section 5.7.3.

(*2) In case a focus transition to the “input” element has occurred by the access key for the “input” element, whether the key event processes defined after [6] of ARIB STD-B24 Vol. 2 Appendix 1 5.4.2.3 will occur or not is implementation dependent. In other words, in case “indirect” attribute is specified for the “input” element, there may exist receiver units starting up the application immediately by the pseudo "click" event, as well as receiver units that will not startup the application. Also, in case “direct” attribute is specified, the behaviour of receiver units relating to the key event after [6] of the above specification is implementation dependent. Therefore, the behaviour of receiver units in case the access key is specified for an “input” element for which “inputmode” attribute is specified is implementation dependent. For those reasons, contents should not set up the access key for “input” elements for which “inputmode” attribute is specified.

(*3) The above provision is not a restriction for the text input frontend application to be started by a key other than "enter". In other words, receiver units equipped with buttons exclusively for the character entry function for users to control the start and/or end of the text input frontend application may be designed to start the text input frontend application by one of those buttons, as long as startup by the “enter” key is supported. Whether or not to employ such buttons is outside of the scope of this document.

● “charactertype” attribute

Specifies the character type available for input.

This attribute is valid only when “direct” or “indirect” is specified in the “inputmode” attribute.

In case an external keyboard, etc. is equipped, it should be reflected in the operation. For details of character type, refer to the next section.

Also, guidelines for the character type specifications during contents writing is defined as follows.

When “password” is specified in the “type” attribute of the “input” element, either one of “number”, “alphabet”, “hankaku” should be specified in the “charactertype” attribute of the corresponding “input” element.

Table 1-12 “charactertype” attribute

Attribute value	Contents
all	No restrictions
number	Only single byte numbers
alphabet	Single byte(alphabet + symbols)
hankaku	Single byte characters(alphanumerical, symbols)
zenkaku	Double byte characters (Hiragana, katakana, alphanumerical, symbols)(*1)
katakana	Double byte characters (katakana + symbols)
hiragana	Double byte characters (hiragana + symbols)

(*1) For the selection of *kana*, it is recommended to make “*hiragana*” as the default.

For the above attributes, DOMBML interface is defined as the following in addition to ARIB STD-B24 Vol. 2 provisions.

Table 1-13 Operation of attributes in the DOMBML interface

Interface	Attribute/method	Operation	Comments
BML Input Element			
	style	-	
	normalStyle	O	note 1)
	focusStyle	O	note 1)
	activeStyle	O	note 1)
	inputMode	-	
	characterType	-	

Note 1) Refer to ARIB STD-B24 Vol. 2 Appendix 2

Ending of this application should be executed prior to the occurrence of the “change” event when focus is shifted from the corresponding “input” element. For details of focus control, refer to ARIB STD-B24 Appendix 1.

Furthermore, in case there are restrictions in character entry by dynamic changes of the “readonly” attribute, or in case invisible attributes becomes valid, or in case the browser finishes, this application ends. In these cases it is implementation dependent whether or not the text that has been entered to the application is implemented to the “value” attribute.

Texts entered by users must be implemented to the “value” attribute in EUC-JP encoding when this application is finished, before deciding on whether or not a change event occurs on the input element. It is implementation dependent whether or not incorporation to the “value” attribute while this application is running.

The event handling and script execution must be progressed while this application is running, except in the case of the provision of key event described later. Therefore, contents authers may expect that script executions by the event handler by bevent is processed even while this application is running. This means that, for example, processes from the disposal of

documents to re-presentation in case of the data event update defined in chapter 2 of this document must be executed by the browser even while this application is running.

Details on the user interface of text input frontend application , including graphical design, remote control operation and whether or an external keyboard is assumed or not, is outside of the scope of this document unless the following provisions are satisfied.

- Two startup methods in accordance with the value of “inputmode” attribute must be able to be selected for startup of the application.
- Type of input characters must be able to be regulated by the “charactertype” attribute
- All caracters described in the next section must be supported.
- All key events necessary to the application must be passed to it, regardless of the keygroup specified by the “used-key-list” property in the BML document that is currently presented, and those key events should not be used for purposes other than character entry.
- In case “password” is specified for the “type” attribute of the “input” element, it is recommended that the application displays the entered text in a non-plain form, just as in the “input” element itself.
- Sizeable display area of the application is, regardless of data broadcasting pixel size, the area where y coordinate value is 110 or greater. Outside of this areas, displays to interrupt contents presentation are not allowed.

1.6.2 Character type

Character types that the text input frontend application should be supported are described as follows. Supporting of character types other than those is implementation dependent.

Table 1-14 Character type

Items		Contents of provision	Comments(reference destination)
Single byte characters	Numbers	"0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" |"8" | "9"	Numbers defined in ARIB STD-B5 Table 8-9
	Alphabet	"a" | "b" | "c" | "d" | "e" | "f" | "g" | "h" | "i" | "j" | "k" | "l" | "m" | "n" | "o" | "p" | "q" | "r" | "s" | "t" | "u" | "v" | "w" | "x" | "y" | "z"| "A" | "B" | "C" | "D" | "E" | "F" | "G" | "H" | "I" | "J" | "K" | "L" | "M" | "N" | "O" | "P" | "Q" | "R" | "S" | "T" | "U" | "V" | "W" | "X" | "Y" | "Z"	Alphabets (both lower case and capital letters) defined in ARIB STD-B5 Table 8-9
	Symbols	"SP" | "!" | """ | "#" | "$" | "%" | "&" | "'" | "(" | ")" | "*" | "+" | "," | "-" | "." | "/" | ":" | ";" | "<" | "=" | ">" | "?" | "@" | "[" | "¥" | "]" | "^" | "_" | "`" | "{" | "|" | "}" | "~"	Symbols defined in ARIB STD-B5 Table 8-9, and space (SP) defined in Table 8-14
2 byte characters	*Hiragana*	”ぁ” | “あ” | “ぃ” | “い” | “ぅ” | “う” | “ぇ” | “え” | “ぉ” | “お” | “か” | “が” | “き” | “ぎ” | “く” | “ぐ” | “け” | “げ” | “こ” | “ご” | “さ” | “ざ” | “し” | “じ” | “す” | “ず” | “せ” | “ぜ” | “そ” | “ぞ” | “た” | “だ” | “ち” | “ぢ” | “っ” | “つ” | “づ” | “て” | “で” | “と” | “ど” | “な” | “に” | “ぬ” | “ね” | “の” | “は” | “ば” | “ぱ” | “ひ” | “び” | “ぴ” | “ふ” | “ぶ” | “ぷ” | “へ” | “べ” | “ぺ” | “ほ” | “ぼ” | “ぽ” | “ま” | “み” | “む” | “め” | “も” | “ゃ” | “や” | “ゅ” | “ゆ” | “ょ” | “よ” | “ら” | “り” | “る” | “れ” | “ろ” | “ゎ” | “わ” | “を” | “ん”	*Hiragana* defined in ARIB STD-B5 Table 8-5 and Table 8-6 (Row 4)
	katakana	”ァ” | “ア” | “ィ” | “イ” | “ゥ” | “ウ” | “ェ” | “エ” | “ォ” | “オ” | “カ” | “カ゛” | “キ” | “キ゛” | “ク” | “ク゛” | “ケ” | “ケ゛” | “コ” | “コ゛” | “サ” | “サ゛” | “シ” | “シ゛” | “ス” | “ス゛” | “セ” | “セ゛” | “ソ” | “ソ゛” | “タ” | “タ゛” | “チ” | “チ゛” | “ッ” | “ツ” | “ツ゛” | “テ” | “テ゛” | “ト” | “ト゛” | “ナ” | “ニ” | “ヌ” | “ネ” | “ノ” | “ハ” | “ハ゛” | “ハ゜” | “ヒ” | “ヒ゛” | “ヒ゜” | “フ” | “フ゛” | “フ゜” | “ヘ” | “ヘ゛” | “ヘ゜” | “ホ” | “ホ゛” | “ホ゜” | “マ” | “ミ” | “ム” | “メ” | “モ” | “ヤ” | “ヤ” | “ュ” | “ユ” | “ョ” | “ヨ” | “ラ” | “リ” | “ル” | “レ” | “ロ” | “ワ” | “ワ” | “ヲ” | “ン”	*Katakana* defined in ARIB STD-B5 Table 8-5 and Table 8-6 (Row 5)
	Numbers	"0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" |"8" | "9"	Numbers defined in ARIB STD-B5 Table 8-5(Row 3)
	alphabet	”A” | “B” | “C” | “D” | “E” | “F” | “G” | “H” | “I” | “J” | “K” | “L” | “M” | “N” | “O” | “P” | “Q” | “R” | “S” | “T” | “U” | “V” | “W” | “X” | “Y” | “Z” | “a” | “b” | “c” | “d” | “e” | “f” | “g” | “h” | “i” | “j” | “k” | “l” | “m” | “n” | “o” | “p” | “q” | “r” | “s” | “t” | “u” | “v” | “w” | “x” | “y” | “z”	Alphabets (lower case, capital letters) defined in ARIB STD-B5 Table 8-5 and 8-6(Row 3)
	Symbols	"(SP)" | “、” | “。” | “・” | “ー” | “―” | “「” | “」”	Symbols and space (SP) defined in ARIB STD-B5 Table 8-5 and Table 8-6(Row 1, Cell 1-3, 6, 28-29, 54-55)

1.6.3 Kana Kanji conversion function

Details on *Kana Kanji* conversion function is implementation dependent, including whether a receiver has the function or not. Note that the *kana kanji* conversion function can be effective only when the value of the “charactertype” attribute is “all” in accordance with the restrictions

of input characters by the “charactertype” attribute.

1.7 Bookmark list display function by receiver’s native application (Optional)

It is implementation dependent whether a receivier has the bookmark list function as a resident application. However, when the convenience of viewers who use bookmarks is considered, the bookmark list function for receiver’s native applications should be equipped as a simple selection method of communication contents assuming the TCP/IP communication protocol introduced in digital terrestrial broadcasting. The minimized functions assuming installation of the bookmark list function of receiver’s native application are indicated in the following. Functions and contents not defined in this document are implementation dependent.

Regarding the bookmark list by receiver’s native application, refer to ARIB STD-B24 Vol. 2 Appendix 1, “8.4 Guidelines on bookmarks“, “Informative Explanation 5 Bookmark services“.

(Note)Trademark rights of the “BOOKMARK/ブックマーク” regarding general electronics(registration of trademark in Japan No. 3281163)

1.7.1 Start/End bookmark list by receiver’s native application

- Startup of the bookmark list by receiver’s native application is done by executing the receiver features for remote control buttons, and functions of multi-media contents (startResidentBookmarkList). Regarding this function, it is explained later in this document.
- In case of starting the bookmark list by receiver’s native application during the presentation of multi-media contents, the display of multi-media contents can be terminated. However, the performance of multi-media contents should be continued.
- In the bookmark list screen display method, there are, for example, a method to display by shrinking the screen image and a method to α-blend and overlap the bookmark list screen on top of the screen image. In case of a data broadcasting service screen (excluding superimpose) rather than a screen image, then the graphic character display of the data broadcasting service and bookmark list could conflict in some cases. Therefore, for data broadcast services when the bookmark list screen is displayed, the display of the data broadcast service screen may be terminated in some cases within a reasonable degree.
- While bookmark list screen is displayed, remote control, etc. key input is obtained by the bookmark list.
- The bookmark list is closed at the time of transition to linked destination by selection of viewer.
- When presentation of multi-media contents is paused and the bookmark list is displayed, the bookmark is selected by the viewers and probably will not move to other contents, and when the bookmark list is closed, the paused multi-media contents for display are

re-displayed and performance should be continued. Key input also returns to multi-media contents.

- When the bookmark list is started during presentation of multi-media contents, since the data broadcasting browser continues its performance without synchronizing with the bookmark list, the screen presented when multi-media contents are re-displayed after the bookmark list is closed depends upon the performance of the data broadcasting browser executed during bookmark list presentation. For example, if the data event is changed during bookmark list display, then the display contents of data broadcasting and display contents at the time of closing the bookmark list are different. Even when the data screen is displayed, data broadcasting may not be presented after the bookmark list is closed in some cases.
- User interface and design are implementation dependent.

1.7.2 Bookmark list functions to be equipped by receiver's native applications

The bookmark list by receiver's native application should be in accordance with ARIB STD-B24 Vol. 2 Appendix 1 "8.4.3.5 Operation of bookmark list service".

In order to realize the bookmark list by receiver's native application, the following functions are necessary.

- Display function of recorded bookmark titles
- Function for viewers to select desired bookmarks from a list and to select stations and transit in accordance with specifications of the linked destination URI and URL.
- Delete management function

The minimum contents that receiver units should satisfy for respective functions are listed below. Contents that are not described in this section are implementation dependent.

1.7.2.1 Display function of recorded bookmark titles

The minimum items and contents displayed by the bookmark list for bookmark information recorded in the area for bookmark service of NVRAM are described in this section. Also, display of items other than the ones listed here, bookmark list, display method and user interface are implementation dependent.

The area for bookmark service of NVRAM is divided into a basic data section and an expansion data section. The minimum items that should be displayed in the bookmark list are the following items.

Table 1-15 Items that receiver application should present

Minimum items that should be presented	
Basic data section	Title
Expansion data section	Bookmark description("bookmark_information")
	Linkdestinationchannel information(”channel_information”) Or memorandum(”memorandum_information”)

Among the items above, explanation of bookmark, linked site information and memo information are assumed to be displayed after viewers select bookmarks from the list using buttons, however, in such an event, it is recommended that all information be displayed on the same screen.

The relationship between the bookmark type and items are as follows:

- Present the title of basic data section regardless of bookmark type.
- Bookmark type (bm Type) that presents the expansion data section is as follows.
  - Favorite channel B(bmType=001)
  - Favorite site B(bmType=003)
  - Related link service B (bmType=005)
  - Related link site B(bmType=007)
- Memorandum (bmType=008)
- Reserved area I for common bookmark types(bmType=009~099)
- Reserved area I for proprietary bookmark types(bmType=200~299)
- For a receiver's application(900 - 999)

In bookmark types that belong to reserved area I for common bookmark types or Reserved area I for proprietary bookmark types, the component of the sections equivalent to “bookmark description” and “link destination channel information” in expansion data section should be fixed.

Also, regarding reserved area II for common bookmark types(bmType=100~198), reserved area II for proprietary bookmark types(bmType=300~899), since components of the expansion data section are not fixed, the display of the expansion data section of bookmarks of this type by the resident bookmark list function is implementation dependent.

1.7.2.2 Functions that allow viewers to select desired bookmarks from a list and to select stations in accordance with specifications of the link destination URI

Based on bookmark information provided by the above display functions, selection of or transition to the service, component and communication contents (link state, non-link state) specified by the link destination URI (dstURI) of the basic data section of the area for

bookmark service is possible. The URI specified by the link destination URI (dstURI) may be up to module and resource specified, but the bookmark list function by receiver's native application interprets only up to component_tag as shown below. In case transition to contents of specification for the module below is desired, read the bookmark information in the startup document in the transit destination, and transition is possible.

- arib://<original_network_id>.<transport_stream_id>.<service_id>
- arib-dc://<original_network_id>.<transport_stream_id>.<service_id>
  [;<content_id>][.<event_id>]/<component_tag>
- Name space that begins with http, https

Selecting the station and transition performance for URI's other than the above are receiver unit-dependent.

The desired operations in receiver's native applications at the time of selecting stations and transition are indicated in Table 1-16.

Table 1-16 Transit action from the bookmark list

Transit destination	Desired performance
Service	In compliance with selecting station operation of the receiver units
Component	In compliance with selecting station operation of the receiver units
Communication contents of link state	Transition to communication contents is always in non-link state; therefore transition to communication contents from receiver's native applications in link state is not possible. (When a transition to link state is expected, it is done by processing on the contents side) For example, the destination of transition recorded in a bookmark is set as the component of data broadcasting, which is the originating link, and transit to desired communication contents from the startup document of the corresponding destination of transition in link state. In such an event, the URL of the desired communication contents obtains the bookmark number from the contents of Greg block 0 written by the receiver's native application, and reads and uses the URL information written in the bookmark.
Communication contents of non-link state	In case of reception condition of data broadcasting, the link state or non-link state, the performance is in compliance with 5.14.12 Guidelines of receiver unit performance when receiving communication contents and figure 5-6 transition D, transition F, transition H. Other cases will be receiver unit-dependent, however, presented status should be presented assuming non-link state.

1.7.2.3 Delete management function

Regarding the delete management function, functions indicated in ARIB STD-B24 Volume 2 Appendix 1 “8.4 Guidelines on bookmarks” need to be able to (1) perform manual deletion of bookmark information, (2) perform automatic deletion of bookmarks, (3) turn ON/OFF the flag to prohibit deletion by viewer’s instruction. For details, refer to ARIB STD-B24, Vol. 2 Appendix 1 “8.4.3.5 Operation of bookmark list service.”

1.7.3 Startup of the bookmark list by receiver’s native application by function

Regarding startup of the bookmark, other than starting by receiver units, it can be started by calling the receiver’s native application from contents using functions. Startup by receiver units is outside of the scope of this document, however, for startup by contents, the bookmark list can be directly started by executing “startResidentBookmarkList()”. A receiver's having the resident bookmark list function implies the support of this function , but it is optional, therefore, supporting of the function should be confirmed using getBrowserSupport( ) before calling this function. For details on this function, refer to 5.6.9 of this document.

1.8 BML browser

Regarding functions that BML browsers should be equipped with, hardware resources and software resources are defined in this chapter. Starting from the next chapter, transmission, mono-media, multimedia encoding will be defined, however, the specifications that are not included in those will be defined in this section.

1.8.1 Display by browser features

Browsers should not display anything that may possibly cover up the presentation that contents intend to be. This is a restriction to avoid missing a part of contents with display by browser features such as buttons for users to control the browser and/or the URL frame, which are common in browsers on PCs. However, this does not mean the restriction of the receiver’s resident application used for data broadcasting such as text input frontend at the time of starting the browser or displaying channel banners at the time of startup of the browser. Also, in case of presenting communication contents in non-link state (*1), browsers should display something of its own explicitly in the areas shown in figure 1-7 and 1-8 indicating that the receiver unit is in non-link state. Details on the expression and display methods are implementation dependent, but they should be obvious enough for viewers to understand that contents they are viewing are outside the responsible area of broadcasting contents.

(*1)For the non-link state, refer to the definitions and chapter 5.

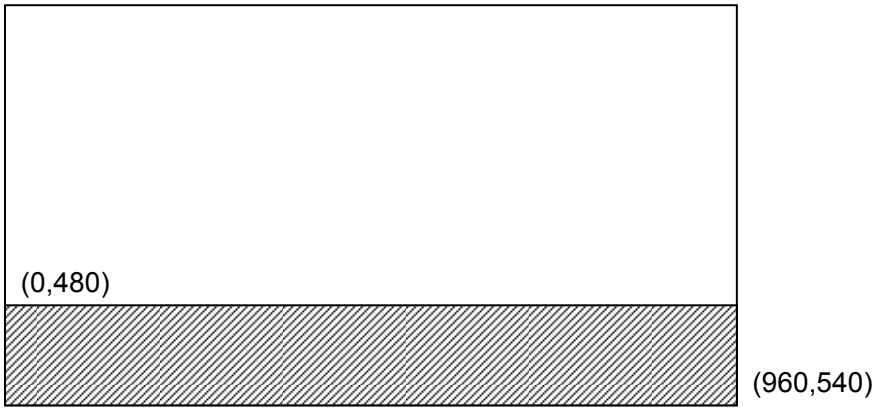


Figure 1-7 Display when the data broadcasting resolution is 960x540

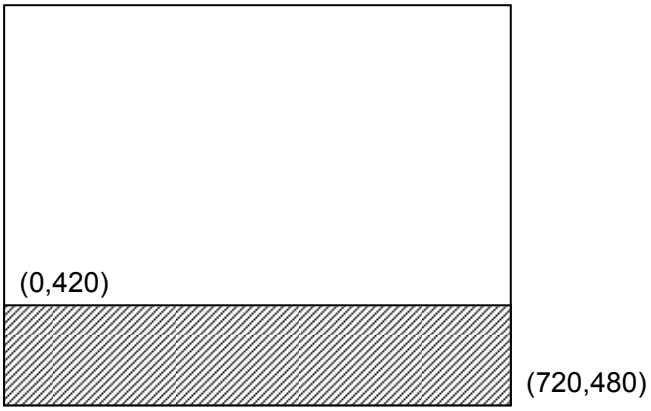


Figure 1-8 Display when the data broadcasting resolution is 720x480(16:9, 4:3)

1.9 Registration transmission function (Optional)

The registration transmission function is comprised of functions to send superimpose recorded in the area for registration transmission by Multimedia services and functions to set up superimpose in the area for registration transmission by Multimedia services, and contents which use registration transmission function or receiver's native applications with registration transmission function of Multimedia services in case superimpose that are necessary during programs that cannot be sent due to congestion or restrictions on the number of reception lines at the interaction channel center, or in case of sending superimpose in bulk in interaction channel programs. The details of provisions relating to receiver's native applications providing

registration transmission functions are defined in section 5.16.5 of this document.

The registration transmission function by receiver's native application is optional. However, it should be implemented for the following reasons.

- Since the registration transmission function by Multimedia service (contents) requires the viewer to be watching the registered channel at the time of specifying the call, and during program viewing, the viewer needs to operate the remote control for calling, the ratio of actual calls may be extremely limited, and it may be unavoidable for viewers to feel annoyance and dissatisfaction. Therefore, we should conclude that relying on only the registration transmission function by contents is not appropriate for the objective to have as many viewers as possible use the interaction channel function.
- Registration transmission by the Multimedia service is expected to be operated differently for each broadcasting station, and if there are a lot of cases where calling does not happen because of contents problems, then some trouble involving viewers is assumed.
- On the other hand, the registration transmission function by receiver features can be called by targeting the time zone the viewer is not viewing, and even if the call fails once, it is possible to retry by judgement of receiver unit, therefore, the possibility of a successful call is high and it fits the benefit of viewers. Also, operations by viewers at the time of the call are not necessary, and the usability for viewers is high as well. Furthermore, in case of calling by receiver features, it has nothing to do with defects of contents and there will be no problems caused by different operations of registration transmission by each transmission broadcaster.

2 Operation of the data transmission method

In this chapter, based on the contents of specifications in applicable documents listed below, for newly specified items, differences and restrictions for operation related to the transmission method of the XML-based multimedia encoding method will be mentioned. In this chapter, data broadcasting services refer to multimedia data broadcasting services by the XML-based multimedia encoding method unless specified otherwise. For items without a specific description, a description of specifications in applicable documents is applied as they are.

- ARIB STD-B10 ”Service information for digital broadcasting”
- ARIB STD-B24 ”Data coding and transmission specification for digital broadcasting”

2.1 PSI/SI

2.1.1 Types of data broadcasting services

2.1.1.1 Data programs and TV programs

Data programs and TV programs are distinguished as follows.

TV programs: Programs without a Data Component Descriptor located in the components, which is described first in PMT 2nd loop

Data programs: Programs with a Data Component Descriptor located in the components, which is described first in PMT 2nd loop

Furthermore, TV programs are classified as “TV program with additional data”, which has additional data in the same event and “normal TV program” without additional data. In this document, “TV program” refers to “normal TV program”.

Regarding data programs, only “independent data program”, which is meant to be viewed on its own is specified.

2.1.1.2 Types of data broadcasting service programs

Program types for programs to operate data broadcasting services are classified as follows.

Table 2-1 Data broadcasting service program types

Program types	Definitions
TV program with additional data	Among TV programs, programs with a Data Component Descriptor located in components other than the first PMT 2nd loop.
Independent data program	Broadcast as a service of service_type=0xC0(data service), 0xAA(bookmark list data service), and programs with a Data Component Descriptor located in the first PMT 2nd loop component.

2.1.1.3 “service_type” of channels that operate data broadcasting programs

“service_type” of service that broadcasts data programs is indicated below.

1) TV programs with additional data are broadcast by service_type=0x01 (TV service) channels.
2) Independent data programs are broadcast by service_type=0xC0, 0xAA(bookmark list data service)channels.

2.1.2 Configuration of contents of data broadcasting service and operation of components

2.1.2.1 Contents and local contents

The relationship between events and definitions of contents and local contents are shown in Table 2-2 and Figure 2-1.

Table 2-2 Definitions of contents and local contents

	Definitions	Operations
Contents	see the group of local contents that are transmitted during certain event periods in certain components. Identified by a Data Contents Descriptor.	Operation is never specified for the contents.
Local Contents	Group of BML documents that are transmitted in certain data events of certain components.	Sometimes they are broadcast by sequentially switching multiple local contents along with the passing time within a component.

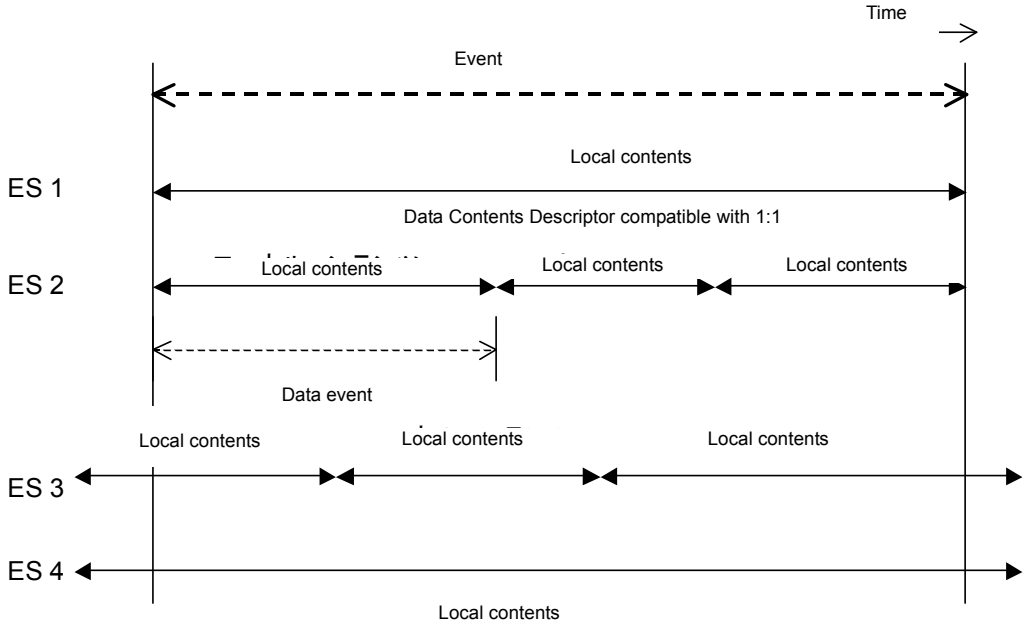


Figure 2-1 Local contents and events

2.1.2.2 Relationship between local contents and ES

One location contents is transmitted by one data carousel (ES) and local contents may refer to ES, which transmits video, audio, and event messages in some cases. The stream ID of components that transmit data carousels and event messages is always 0x0D.

2.1.2.3 Operation of component tags

Operation of the component tag value used in data broadcasting including closed caption and superimpose shall be in accordance with Vol. 7, section 5.1.1 (1). However, reference of the ACC audio component in a 24 kHz sampling frequency, which is transmitted by the partial reception layer, is optional.

2.1.2.4 Identification of entry components

The default component of data broadcasting programs is always transmitted by component with tag value of 0x40. The identification of entry components, which transmit startup documents on tuniing, is not done by the entry_point_flag of the Data Component Descriptor but it is done by this tag value. Entry components transmit the entry module (moduleId=0x0000) and the entry module always includes one startup document.

In case of multi-view programs, a component with a tag value of 0x40 is always the entry component.

2.1.2.5 Restrictions on the number of ES

Considering the restrictions on the hardware resources of receiver, the total number of ES's that can exist in one service (1 PMT) is 16 ES's. (For example, in case of video 1, audio 2, closed caption 1 and superimpose 1, then 11 can be used for the multi-media data component.) However, 16 ES's includes ES's that transmit PCR's as well.

Also, the number of ES's that transmits data carousels is 8 ES's and the maximum number of ES's that transmit only event messages is 4 ES's.

Also, in 1 service, the maximum number of ES's that have set a valid ECM PID is 12.

At the time of multi-view operation, this restriction is applied to each component group specified in the Component Group Descriptor.

2.1.2.6 Detailed specifications of section data transmission

- Multi section transmission (transmission of more than two sections in 1 TS packet) is not done in data carousel and event message transmission.
- The following standards are set for the transmission bit rate when transmitting data of each section of data carousels and event messages.
  - Do not send more than 6 packets continuously using the same PID.
  - The total bit rate of components received (maximum 7 PID, refer to 2.3.6.1) when receiving one content is less than 4Mbit/s. This includes DII, DDB and event messages.
  - Additionally, the maximum bit rate per one sub-table is less than 2Mbit/s (8KB±100% per 32 milliseconds)
  - When transmission exceeding those standards is executed, then the reception efficiency of sections may decrease for some receiver, and the time required to obtain necessary sections may become longer, so it is recommended that the above standards are not exceeded for transmission.

2.1.2.7 Default maximum bit rate of data broadcasting programs

Shall be in accordance with Vol. 7

2.1.2.8 Operation of video and audio components in data broadcasting services

In the following descriptions, video/audio streams in 1PMT are classified as follows.

TV video/audio:

The video and audio stream is specified in Vol. 7. It becomes the target for playback by EPG, etc. and switching by video/audio buttons. A component_tag of 0x00~0x0F(video)or 0x10~0x2F(audio)is given and when tuning an event that includes this stream, EPG, etc. will play those video/audio. TV video/audio may be used in independent data programs in some cases.

Video/audio referred to only from data broadcasting:

The video/audio stream is referred to and played only from data broadcasting contents, component_tag of 0x50~0x7F is given. EPG, etc. will not play and it is not a target of switching functions by the video and audio buttons.

- For either TV programs with additional data or independent data programs, as the AV stream referral method, methods to directly specify the component_tag value (direct tag specification) and to specify -1 as the component_tag and refer to the stream that EPG, etc. has chosen (default specification) are both operable.
- Based on the model indicated in ARIB STD-B24 Vol. 2 Informative Explanation 2, when referring to TV video/audio component from data broadcasting contents, the default specification is performed as a basic rule. However, in the following cases, TV video/audio switching operation by remote control key and such receiver functions would not contradict data broadcasting content specifications, therefore, referral by direct tag specification is possible.
  1) In case only one TV video component is included in event, direct tag specification can be referred to for the TV video component.
  2) In case only one TV audio component is included in an event and if it is not dual mono, then direct tag specification can be referred to for the TV audio component.
- Relating to the above, the following operational restrictions are instituted.
  1) At the time of starting the startup documents for independent data programs including TV programs with additional data and TV video/audio, only default specifications can be used. Reason: To prevent the BML-engine from switching video/audio that is being played by EPG, etc. at the time of starting the presentation of startup documents.
  2) Referral of TV video/audio from data broadcasting contents in multi-view programs is default specifications only. (Because multiple components of TV video are included)
  3) When change to TV video/audio status from the status where video/audio is being played by direct tag specification, both the default specification and direct tag specification can be used. In case there is dual mono TV audio, only default specifications are possible and since the selection status of dual mono of TV audio changed by default specifications is receiver dependent, the channel ID is specified if necessary.

- Operation of related receiver is specified as follows.
  1) In case TV video/audio exists, EPG, etc. performs video/audio playback process at tuning in the same way as TV programs with additional data for independent data programs.
  2) In case there is only one selection for TV audio, audio is not switched even if the audio button is pressed. It is the same for TV video.
  3) When the BML-engine ends, the default TV video/audio is played. However, if data broadcasting contents do not switch the video/audio by direct tag specification (in case of carrying out default specifications only), then the video/audio component that is being played at the time of completion will continue to be played.

2.1.3 Series reservation operation in data broadcasting services

- Refer to 5.9 for operation of the series reservation function from multi-media contents

2.1.4 Operation of the PMT specific to data broadcasting services

- Refer to 2.1.10.3 and 2.1.10.6 for receiver performance on tuning or updating the PMT.
- The only components equipped in the PMT that allow the non-existence of ES are the following 3.
  - Closed caption component
  - Superimpose component
  - Component that transmits event messages only

2.1.5 Operation of the Data Component Descriptor in the PMT

The Data Component Descriptor is located in the following components.

- Components that transmit closed caption and superimpose
- Refer to (chapter 4) the chapter on closed caption and superimpose for operation details of the Data Component Descriptor located in components of closed caption and superimpose.
- Components that transmit data carousels
- Do not locate the Data Component Descriptor in any other components. However, it can be located in components that transmit event messages only.
- Operation of Data Component Descriptor is shown in Table 2-3.

Table 2-3 Operation of the Data Component Descriptor

Flag	Operation
data_component_id	0x000C
Contents of additional_data_component_info(additional_arib_bxml_info( ))	
transmission_format	It is 00(Data carousel transmission method and event message transmission method).
entry_point_flag	• It is always 1 only for component of component_tag= 0x40 (component of component_tag= 0x40 that transmit modules including documents that should be started first when data broadcasting programs are selected.) It is 0 for any other components. • When receiver is tuned, obtain and present startup documents of data carousels transmitted by components of component_tag= 0x40.
auto_start_flag	It is operated. Follows the specifications of 2.1.10.2 and 2.1.10.3. It is always auto_start_flag=1 in independent data programs.
document_pixel size	The pixel size and aspect ratio of BML contents are displayed. Only the following 3 types of parameters are operated. 0011: 960X540 0100: 720X480(16:9) 0101: 720X480(4:3) However, refer to 2.1.10.5 for pixel size and aspect control of data broadcasting programs.
use_xml	It is 0.(XML using application dependent tags is not transmitted.)
default_version_flag	It is always 0.
independent_flag	It is always 1.
style_for_tv_flag	Operated as specified. If this flag is specified as 0 (does not have TV as the style and only includes contents that cannot be laid out by TV receiver) then, it is judged as “viewing is impossible”.
bml_major_version, bml_minor_version	Operated as specified. When these fields are arranged, basic receiver assess this as “viewing is possible” when bml_major_version is 3.
ondemand_retrieval_flag	It is always 1.
file_storable_flag	It is always 0.
Operation of additional_arib_carousel_info( )	
data_event_id	Not operated in the PMT. Value is fixed to 0xF(1111).
event_section_flag	It is always 1.

2.1.6 Operation of the Target Resion Descriptor in the PMT

The Target Resion Descriptor is not used.

2.1.7 Operation of the H-EIT, M-EIT Data Contents Descriptor

- Refer to chapter 4 for operation of the Data Contents Descriptor regarding closed caption and superimpose.
- In H-EIT, multiple Data Contents Descriptor may exist for one event.
- Even there is an event to execute multi-media data broadcasting services, sometimes the Data Contents Descriptor is not placed.

- Data Contents Descriptor is not always placed for all broadcasting contents.
- The maximum number of Data Contents Descriptors for one event in H-EIT is 32.

Operation of the H-EIT and M-EIT Data Contents Descriptor is shown in Table 2-4, and the relationship between contents and the ES is shown in Figure 2-2.

Table 2-4 Operation of Data Contents Descriptor

Field	Operation
entry_component	Specify components including startup module of corresponding local contents.
num_of_component_ref	"num_of_component_ref" of H-EIT[schedule]M-EIT[p/f after] does not mean anything to receiver, so it is not referred to.
component_ref	Describes component to transmit carousels other than the entry carousel and the component to transmit referred video/audio streams and event messages. H-EIT[p/f], M-EIT[p/f] have established values. Receiver do not have to refer to this field for H-EIT[schedule], M-EIT[p/f after] when recording reservations and viewing reservations are made. Execution is executed with the H-EIT[p/f], M-EIT[p/f] values.
ISO_639_language_code	Fixed in jpn(Japanese).
text_length	The maximum value is 80(bytes). However, in case of displaying the title of contents in receiver, then at least 40 bytes worth will be displayed.
text_char	Describes the title of contents displayed in EPG.
Operation of arib_bxml_info( )	
transmission_format	It is 00(Data carousel transmission method and event message transmission method).
auto_start_flag	"auto_start_flag" of EIT is not used. It is always 0.
document_pixel size	Presents the pixel size and aspect ratio of the corresponding contents. Only the following 3 types of parameters are used. 0011: 960X540 0100: 720X480(16:9) 0101: 720X480(4:3) However, for pixel size and aspect control of data broadcasting programs, refer to 2.1.10.5.
use_xml	It is 0. (XML using application dependent tags is not transmitted.)
default_version_flag	It is always 0.
independent_flag	It is always 1.
content_id_flag	Operated as specified. If "content_id" and "content_version" are located in this descriptor then 1, if not then 0.
associated_contents_flag	Operated as specified. When the receiver displays that data broadcasting associated with TV programs will be executed by EPG, etc., then execute only for the events where the Data Contents Descriptors, which are specified this flag as 1, exist.
style_for_tv_flag	Operated as specified. When this flag is specified as 0, then it is judged as impossible to view with a basic receiver. (does not have TV as style and only includes contents that cannot be laid out by TV receiver)

Field	Operation
update_flag	“update_flag“ is not operated. It is always 0.
ISO_639_language_code	Operated as specified.
content_id	Operated as specified.
content_version	“content_version” is used. However, “content_version” does not change during event time.
bml_major_version, bml_minor_version	Operated as specified. When those fields are arranged, and if “bml_major_version” is 3, then basic receiver will assess this as possible to view.
ondemand_retrieval_flag	It is always 1.
file_storable_flag	It is always 0.
Operation of arib_carousel_info( )	
num_of_carousels	When the number_of_carousels=0, then it shows that all of the following carousel information is undecided. Confirmed information (num_of_carousels≠0)is input in H-EIT[p/f], M-EIT[p/f]. This value is not referred to during recording reservation and viewing reservation.
component_tag	Operated as specified. When contents are selected from EPG, etc., the startup documents of the components specified by this tag value are obtained and presented.
event_section_flag	It is always 1.
component_size_flag	Operated as specified. If the “component_size” field is not located for reasons such as operation is not certain, then it is 0.
default_transaction_id_flag	It is 0(transaction_id is not encoded). That is, to obtain desired “transaction ID” without locating “transaction_id” in this descriptor.
default_timeoutDII_flag	Operated as specified. When this flag is 0, the “timeout_vaule_DII” field may not be located.
default_leak_rate_flag	It is 0(”leak_rate” field is not located). The “leak_rate” is not operated and transmission rate of data carousel is specified in 2.1.2.6.
component_size	Operated as specified and only the confirmed values are located. Confirmed values do not increase or decrease as a basic rule and the change in the direction of the value increase is rejected. When operation is not confirmed, then sometimes location is omitted.
transaction_id	(Not located)
timeout_value_DII	If the location is omitted, or 0xFFFFFFFF is specified (there’s no recommended time out value), then it should be interpreted as 5000 milliseconds is specified. If a carousel with a DII transmission period cycle exceeding 5000 milliseconds is included in this corresponding event, then locate the appropriate value that is suitable for the DII transmission period cycle for the carousel with the longest DII transmission period cycle. Also in some cases, the broadcaster sends out a DII at fixed period cycle regardless of the carousel length and operates timeout_value_DII value as fixed.
leak_rate	(Not located. For the transmission rate of section data, refer to 2.1.2.6)

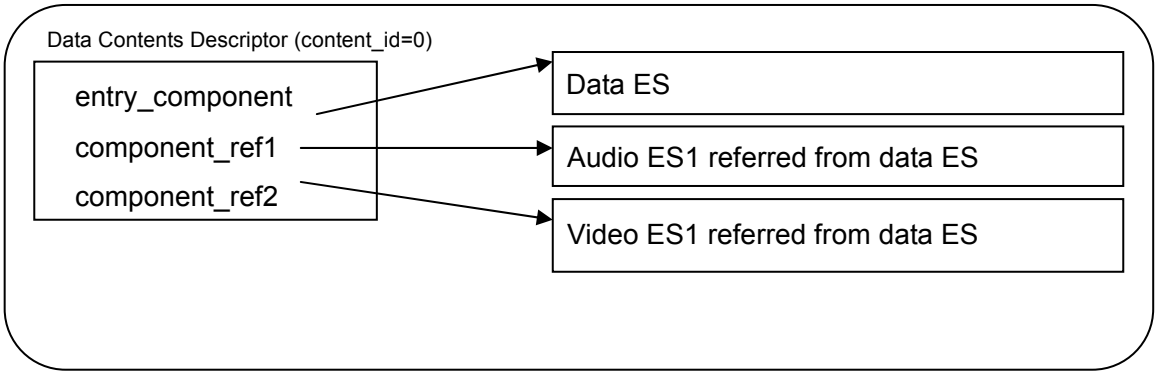


Figure 2-2 Relationship between contents and ES

2.1.8 Operation of the Data Contents Descriptors in H-EIT[schedule],M-EIT[p/f after]

- Since the num_of_component_ref in H-EIT[schedule],M-EIT[p/f after] may have unconfirmed values in some cases, do not refer to the receiver.
- In H-EIT[schedule],M-EIT[p/f after], num_of_carousel=0 may be set in some cases.

2.1.9 Operation of EIT's specific to data broadcasting services

- Information to select or to reserve contents is written in the Data Contents Descriptor. However, some programs that perform data broadcasting services do not have Data Contents Descriptor located.
- Descriptions regarding components of Data Contents Descriptors are fixed during the event time.

  For operation of various EIT, refer to Vol. 4.

2.1.10 Performance of related receiver

2.1.10.1 BML-engine startup

When the data broadcasting program is selected and the auto_start_flag of the Data Component Descriptor located in the PMT entry component (component_tag=0x40)is equal to 0(Only TV program with additional data is specifiable), then the broadcasting engine should not start immediately, and the startup process of the BML-engine starts when the viewer presses the d button. If the auto_start_flag=1, then the BML-engine startup process starts without waiting for the button to be pushed by the viewer. For the flow up to the start of the BML-engine, refer to the next section.

2.1.10.2 Receiver operation when data broadcasting programs start

Prior condition of data transmission operational specifications

- “data_event_id” located in the PMT Data Component Descriptor is fixed (0xF) and is not used.
- Component of the “component_tag=0x40” is set as an entry component, and data carousels transmitted by entry components are called entry carousels.
- An entry component exists at all times. PID values do not change except for exceptional cases. However, the following exceptions are assumed. Including cases other than the following example, performance of receiver when PID values are changed is specified in 2.1.10.3.
    - When changing from a multiple channel arrangement to a single channel arrangement in case of a spotted arrangement
- One entry component exists per one event.

Basic operation of receiver when selecting channels (Guidelines)

1) When ECM is specified in PMT 1st loop, this process is carried out. Perform the following process only if the program becomes viewable by confirming the contract status.
2) If TV video/audio are included in the PMT 2nd loop (TV video :component_tag=0x00-0x0F, TV audio :component_tag=0x10-0x2F), then it will be played.
3) Entry components(component of component_tag=0x40)are identified from components located in the PMT 2nd loop.
4) If the data_component_id of the Data Component Descriptor for the entry component is 0x000C, then it is judged as a data broadcasting program by the XML-based multimedia

encoding method, and the process below is followed. If it is a data encoding method that the receiver is not compatible with, then the data broadcasting service does not start.

5) Refer to the “auto_start_flag” of the Data Component Descriptor for the entry component and if it is 1, then execute the following process immediately. If the “auto_start_flag” is 0 then, the following process is done when the viewer presses the dButton.
6) Perform the following assessment process from 7 to 10 and judge whether or not the BML-engine startup should be performed.
7) Based on the BML / B-XML version number specified in the Data Component Descriptor of the entry component, judge whether or not the presentation of data is possible. If viewing is judged as impossible, then the presentation of the data broadcasting service is not carried out.
8) If the presentation of data broadcasting service is judged as possible in 7, then the BML-engine is started and after initialization of the BContents and Ureg is carried out, the startup documents of the entry component are obtained and presented. Also, the root certificate should be re-obtained.(Refer to 5.14.14.3)
9) However, after the BML-engine is started, if the entry component is an empty carousel, then perform the following. (Refer to 2.3.1.6 for empty carousels)
   - When the “auto_start_flag=0”, the engine start process is terminated as soon as the entry carousel is detected as empty, and it returns to standby to wait for the viewer to press the d Button. (Step 5 above)
   - When the “auto_start_flag=1”, then keep monitoring the data event switch of the entry component, and obtain and present the startup document as soon as the data event switch occurs and the startup module appears.

Receiver behaviour in case the component group (sub-channel) of multi-view programs is switched. (Guideline)

- Presentation of data broadcasting does not change even if the group is switched.
  (Presentation of the document that was presented shall continue even after switching)

Receiver behaviour when contents are selected from EPG, etc. and to view or to make a viewing reservation. (Guidelines)

1) Based on the Data Contents Descriptor of the EIT, EPG, etc. present content candidates to viewers. In such an event, based on the “on demand viewing possible/impossible information” located in the Data Contents Descriptor, whether the corresponding data broadcasting program is viewable or not by the receiver is judged, and when it is impossible to view, display an explanation and do not use it as a selection target.

2) In case of programs that are currently broadcasting, the component specified by the "entry_component" of contents selected by the viewer is set as the starting component and the viewing of data broadcasting program is started. In other words, the BML version and limited reception for corresponding components is judged and if the corresponding component can be viewed, the startup documents are obtained and presented.

3) In case of starting the viewing of programs that were reserved for viewing, the performance of 2 above from the start time of the program is executed.

2.1.10.3 Receiver behaviour when updating the PMT

Receiver behaviour when updating the PMT occurs during the viewing of data broadcasting programs:

- In case the components during viewing disappear.
  → Discard the documents that are currently presented and start the presentation of the startup document of the entry carousel.

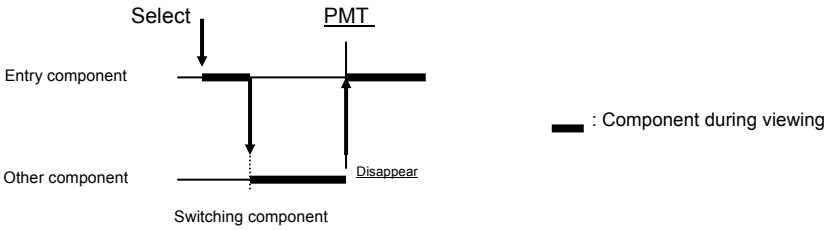


Figure 2-3 Behaviour when the components during viewing disappear due to the PMT update.

- In case the entry component has disappeared.
  → Since it is no longer data broadcasting program, the BML-engine ends.
- In case the PID of the component to transmit the data carousel (including empty carousel) during viewing or return flag monitoring has changed.
  → The receiver judges the contents of data program as changed and performs a process that is equivalent to re-selecting station for its own self-services described below.
  - Discard documents that are currently presented.
  - Delete all resources that are already acquired from BContents memory.
  - Re-acquisition and presentation of startup documents of the entry component.

- In case the value of the “auto_start_flag” is changed from 0 to 1.
  → If the BML-engine has not been started, then item 4 and the following items of “Basic operation of receiver when selecting channels” described in 2.1.10.2 should be performed.
  (*) However, depending on the display status of the receiver, the BML-engine may not start.
- In case the value of “auto_start_flag” is changed from 1 to 0.
  → The status of BML-engine does not change.

Receiver performance in case the update of the PMT occurs while viewing TV programs other than TV programs with additional data:

- In case the entry component appears
  → It is interpreted that the data broadcasting program has started, and item 4 and the following items of “Basic operation of receiver when selecting channels” described in 2.1.10.2 should be performed. However, depending on display status of the receiver, the BML-engine does not have to start.

2.1.10.4 Handling of the dButton

There are two patterns of use for the dButton which is equipped in remote controllers.

- BML-engine startup instruction
  For TV programs with additional data where a forced start is not specified, (auto_start_flag=0 is specified by the Data Component Descriptor of the entry component)it is used by viewers to start the data broadcasting service. When such a program is tuned, the BML-engine startup process is started by pressing the dButton.
- Contents control after BML-engine startup
  After BML-engine startup, events of pressing the dButton are all processed by the BML-engine and in BML documents, it is obtainable as an event (type attribute is “DataButtonPressed”) if needed. Processes when the dButton is pressed are scripted as multi-media contents. However, the usage of the dButton is as a basic rule a toggle operation for the ON/OFF display of data broadcasting services related to viewing TV programs. This is a restriction on contents production.
  For reasons listed above, the dButton cannot have functions to force exiting of the BML-engine. Whether or not such methods should be realized as a separate receiver features or remote control button depends on a model.

An example of using the assumed dButton is shown below:

(Example of assumed usage 1)

Contents of data broadcasting services are not displayed immediately after tuning, and contents viewing is started when the viewer presses the dButton.

1) In the Data Component Descriptor of the entry component, “auto_start_flag”=0 is specified
2) After selecting a TV program with additional data, the BML-engine startup process is not immediately performed
3) Once the pressing of the dButton by the viewer is detected, then the receiver performs the BML-engine startup process and after the BML-engine is started, obtains and presents the startup document.
4) In the startup document, the menu display of multi-media contents, etc. is carried out and the presentation of the data broadcasting service starts.
5) Pressing of the dButton during the presentation of the data broadcasting service is communicated to multi-media contents and all character shapes will be deleted by contents control or icons, etc. will be displayed.
6) When pressing of the dButton is detected again, it returns to the data broadcasting service presentation status before deletion by contents control.

(Example of assumed usage 2)

In case of displaying data broadcasting service contents immediately after the station is selected.

1) “auto_start_flag=1” should be specified in the Data Component Descriptor of the entry component.
2) After selecting the station of the data broadcasting program, the BML-engine startup process is performed and the startup document is obtained and presented.
3) In the startup document, the menu display of multi-media contents, etc. is executed and the presentation of data broadcasting service starts.
4) From here on, the equivalent processes for the above usage examples 1-5 and after are executed.

(Example of assumed usage 3)

In the case BML-engine is started immediately after selecting the station, but contents of the data broadcasting service are not displayed on the screen.

1) “auto_start_flag=1” should be specified in the Data Component Descriptor of the entry component.
2) After selecting the station of the data broadcasting program, the BML-engine startup process is executed and thestartup document is obtained and presented.
3) At the initial status of the startup document, only TV video and audio are presented in full size, and it enters standby to wait for the viewer to press the dButton.
4) Once the pressing of the dButton by the viewer is detected, the menu display of multi-media contents by contents control is performed and presentation of the data broadcasting service starts.
5) From here on, the equivalent processes of the above usage examples 1-5 and after are executed.

2.1.10.5 Pixel size and aspect control in data broadcasting programs

The receiver controls the pixel size and aspects based on “document_resolution” specified in the BML document instead of “document_resolution” of the Data Component Descriptor of the PMT or Data Contents Descriptor of the EIT. For details, follow 1.2.2 and 1.2.3.

2.1.10.6 Receiver operation when selecting stations

The receiver behavior list when tuning is shown in Table 2-5.

However, data broadcasting programs should be viewable by the judgment of the BML version.

Table 2-5 Receiver behavior when selecting stations

Selection method	Operation selection	Receiver performance
Channel selection of remote control	TV program with additional data is selected.	In case of “auto_start_flag=1”, presentation of the startup document of the entry component of TV programs with additional data is started. In case of “auto_start_flag=0”, presentation of the startup document is started after waiting for the viewer to press the dButton.
	Independent data program is selected.	The selected independent data program is started.
Select from EPG (Selection of current broadcasting program)	Specify the contents and select.	Regardless of the value of auto_start_flag, the specified contents are started. When displaying the title of the contents, display at least 40 bytes.
When executing viewing reservation	The same operation as channel selection.	

2.1.10.7 Reservation related behaviors of data broadcasting programs(Guidelines)

- When the viewing reservation is set

  In the same way as normal programs which do not conduct data broadcasting services, execute the reservation registration for viewing the start time, etc. of the event. In case the Data Contents Cescriptor is located in H-EIT[schedule]M-EIT[p/f after] at the time of the reservation, the following functions can be specified to the viewer.

  - Contents selection function

    Set up function to present selected contents first instead of the entry component of the PMT at the time of reservation execution.

- When the recording reservation is set

  The bit rate of required events in the mode selection of recording media or spectrum

securement is determined and based on specifications of 2.1.2.7. Others will be at the same time as when viewing reservation is setup.

- When viewing reservation and recording reservation are conducted

  At the starting time of the reserved event, the presentation and recording of data broadcasting based on PMT is started. If the contents are not specified at the time of setting up the viewing reservation, then presentation of startup document of the entry component is started. If specified, presentation of the startup document for specified contents starts.

2.1.10.8 Specifications when the partial transport stream is output/input

When the receiver output data components(component tag value 0x40~0x7F) to the partial transport stream, the descriptor shown in Table 2-6 should be output along with it. (As specified in Vol. 2)

Table 2-6 Descriptor outputting to partial TS

Descriptor name	Written table	Location standard
Data Component Descriptor	PMT 2nd loop	Required
Data Contents Descriptor	SIT 2nd loop	Required
Extended Broadcaster Descriptor	SIT 2nd loop	Required
Broadcast ID Descriptor	SIT 2nd loop	Required

- Operation when functions that require SI information which is not included in the partial transport stream while multi-media contents are being played with partial transport stream as input depend on models. For example, the following are examples of extended functions for broadcasting.
  - EPG functions: Selecting station(epgTune), reservations(epgReserve, epgRecReserve)
  - Operation control function: Confirm whether or not it is during broadcasting(isBeingBroadcast)

2.1.10.9 the desired display of EPG, etc.

- Whether or not the data program display will co-exist with the radio TV section or whether it will have an EPG exclusively for the data program depends on a model.
- The desired EPG display related to data broadcasting services is shown in Table 2-7.

Table 2-7 Desired EPG display

Display to show that related data broadcasting services exist.	Display of programs with additional data	Based on the associated_contents_flag located in Data Contents Descriptor of the EIT, display to indicate that there are additional data associated with TV programs in the TV program chart or display while stations are selected.

2.2 Operation of the independent PES transmission method

Refer to chapter 4 for the transmission method of closed caption and superimpose.

Independent PES transmission will not be executed for other components.

2.3 Data carousel transmission method/event message transmission method

2.3.1 Data carousel transmission operation

- Maximum number of modules sent per 1 data carousel is 256.
- Configuration of Module of data carousels may change during events with time. (Module may be increased/decreased) in some cases. When it changes, then the DII version is updated.
- Module that composes the data carousel may have a different transmission frequency depending on the module.

2.3.1.1 Introducton of data events and local contents

- The concept of data events which do not have any direct time relationships with events is introduced in order to make switching of contents possible regardless of during or in between programs, and the switching of contents presentation is performed in the data event unit. Contents transmitted within 1 data event are called local contents.(Refer to figure 2-5)
- Data events are identified by “data_event_id” of DII.

2.3.1.2 Operation of data events

- “data_event_id” is updated when local contents are switched. In other words, “data_event_id” is different before and after switching local contents. (It is not always increased by 1) (Figure 2-4)

- When ES is paused (Component Descriptor disappears from PMT), “data_event_id” is not always updated before or after the pause. Starting of ES means the beginning of new local contents. “data_event_id” before ES pause does not have to be recorded in receiver. (Figure 2-4)
- Manage/update “data_event_id” by each component. “data_event_id” is operated by values other than 15 (0xF).

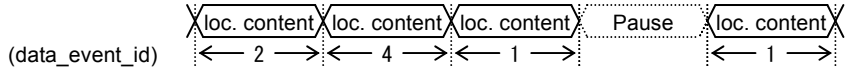


Figure 2-4 Local Contents and “data_event_id”

2.3.1.3 Start/End of local contents

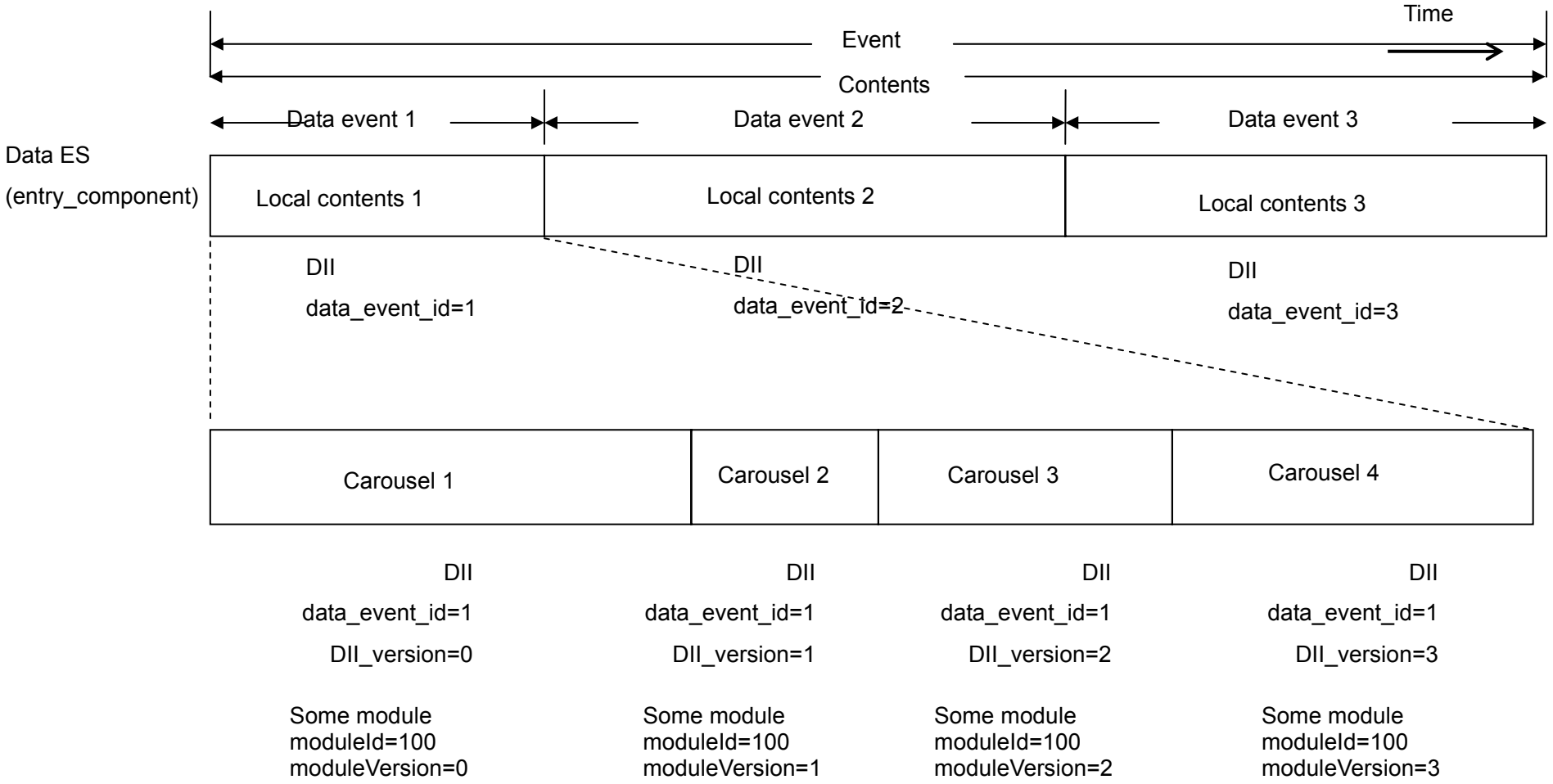
- When “data_event_id” of DII is updated, it is recognized as switching of local contents and the document that is currently presented is destroyed, and the startup document of new local contents is obtained and presented. (Refer to 2.3.1.7)
- Start and end of local contents occurs in sync with the update of “data_event_id” in the DII of the carousel that is currently being viewed as a basic rule.

2.3.1.4 Introduction of the return flag

- In sync with the local contents switch of the entry component, there are some cases in which it should be forced to move to local contents of entry regardless of viewing local contents. For that purpose, a flag (return flag: “return_to_entry_flag”)to notify this to the personal information area of the DII of the entry carousel should be located and receiver will monitor this regardless of which local contents of the component are presented.
- The return flag is not located in the DII of the data carousel transmitted by components other than the entry component.

2.3.1.5 Local contents and the Data Contents Descriptor

In case multiple contents are transmitted during the event period, the Data Contents Descriptor describe the information for a group of multiple local contents.



- If any one of the modules in DII is updated (module version is upgraded, or increased/decreased), then the version of DII is updated.
- Cut line of the data event does not always occur at the cut line of event.
- For local contents whose transmission period matches the event period, data contents descriptors are handled 1:1.

Figure 2-5 Data events and local contents

2.3.1.6 Operation of empty carousels

- Data carousels composed only of DII's whose "numberOfModules" field is 0, and do not include DDB's are called empty carousels.
- Empty carousels are used only to switch the data sending/stopping of components without changing the ES description in the PMT in case certain components are needed only for a specific event period etc.
- When non-empty data carousels are switched with empty carousels, the "data_event_id" is switched.
- The minimum transmission frequency specification of DII (section 2.3.2) is applied even when sending empty carousels.
- A version up of DII may occur while sending empty carousels. Also, an update of the data event may occur.
- When empty carousels are detected during tuning(when "auto_start_flag"=1) or by a switching data event, then the update of the DII is monitored without any errors, and the presentation of the startup document begins when the startup document appears. In case the entry component is an empty carousel when the BML-engine is started by the pressing of the dButton by a viewer in TV programs with additional data of "auto_start_flag"=0, then the BML-engine startup is aborted and returns to the standby status to wait for the dButton to be pressed.
- When carousels including documents that are currently being viewed are switched to empty carousels during the presentation of multi-media contents, an event ("DataEventChanged" event, identified by status=1) to indicate "to be changed to an empty carousel" will occur for the BML document. This performance is the same whether it's entry component or not, or regardless of the value of the auto_start_flag.
- When empty carousels in components transmitting specified BML documents are transmitted in the document launch where BML documents for the transition destination are specified just as launchDocument(), then they are processed as errors equivalent to when reference modules are not included in normal carousels.
- When empty carousels are transmitted by specified components in document transition where only components are specified without specifying the BML documents of transit destination just like the execution of "epgTuneToComponent()" or transition to entry component by return flag, then the same operation as detecting empty carousels by switching the data event is executed. (However, "DataEventChanged" does not occur.)

2.3.1.7 Basic behavior of receiver during data broadcasting program presentation

- Receiver monitor DII of carousels transmitted by the following component at all times.

  1) DII of components including modules currently being viewed.

  2) DII of entry components

  The DII of entry components is monitored at all times to judge whether or not forced process to return to the entry carousel should be performed at the time of switching local contents of the entry component.

  Receiver monitor the DII of carousels transmitted by the following components.

  3) DII of components transmitting modules specified in “lockModuleOnMemoryEx()”

  4) DII of components transmitting modules subscribing to “ModuleLocked” or “ModuleUpdated” event.

  Components that can become the target of module lock by “lockModuleOnMemoryEx()” are limited to components with a “component_tag” value of 0x40, 0x50, 0x60. (Refer to section 5.12.6.9) Also, components that can become the monitoring target of “ModuleLocked” / ”ModuleUpdated” event are limited to components that are currently presented and components with a “component_tag” value of 0x40, 0x50, 0x60. (Refer to section 5.13.3). Therefore, the receiver need to be able to obtain DII and modules in parallel with a maximum of 4ES (components with a “component_tag” value of 0x40, 0x50, 0x60 and components that are currently presented).

- Receiver processing when the data event that is currently viewed is switched.

  The BML-engine will make the data event message "DataEventChanged" occur for documents that are currently being presented. And then, documents that are currently presented are destroyed and the startup documents included in moduleID=0x0000 module are obtained and presented. When there is an empty carousel after the switching data event, then monitor the update of DII of corresponding components after discarding the document that is currently being presented, and as soon as it is no longer an empty carousel after the data event is switched again, the startup document is obtained and presented. However, synchronized/non-synchronized events that have occurred after the occurrence of "DataEventChanged" are discarded except for the “unload” event. Also, if “epgTune” / “epgTuneToComponent” / ”launchDocument” / ”launchDocumentRestricted” / “reloadActiveDocument” function is used in the "DataEventChanged" event handler then do not obtain the startup document and follow the specifications of the function.

- When “data_event_id” of the monitored entry component is updated while viewing the desired data component, then process as follows.
  - When the return flag of DII of entry component is 1, then execute the process to return to entry component. In other words, the module that’s currently presented is destroyed and the process to obtain and present the startup document included in the module of moduleID=0x0000 for the entry carousel is executed.(Figure 2-6) In such an event, the “DataEventChanged” event does not occur.
  - When the return flag of DII of the entry component is 0 then continue to present the document that is currently presented.(Figure 2-6)
  - In all cases, when viewing documents of the entry component, perform the “receiver process when the data event of components that are currently being viewed switches” in the above.

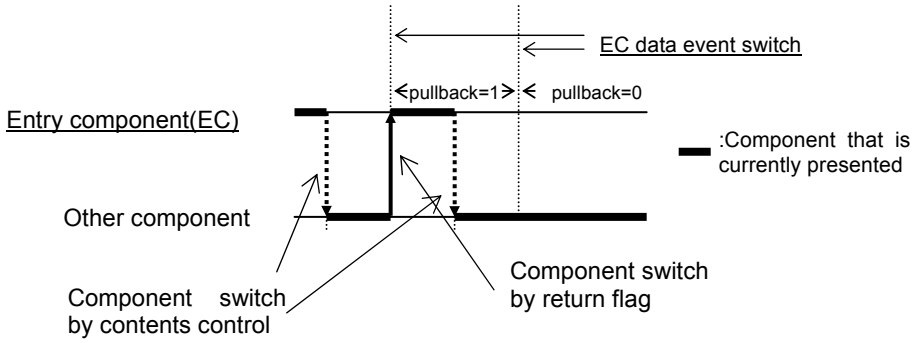


Figure 2-6 Update of “data_event_id” and return flag of entry components

2.3.1.8 Transmission of root certificates

Root certificates are transmitted by data carousels as one of the resources that comprise data broadcasting services. Figure 2-7 is a composition example of a data broadcasting service when transmitting a root certificate. The figure is an example of transmitting 2 root certificates, however, the number of root certificates can be 1.

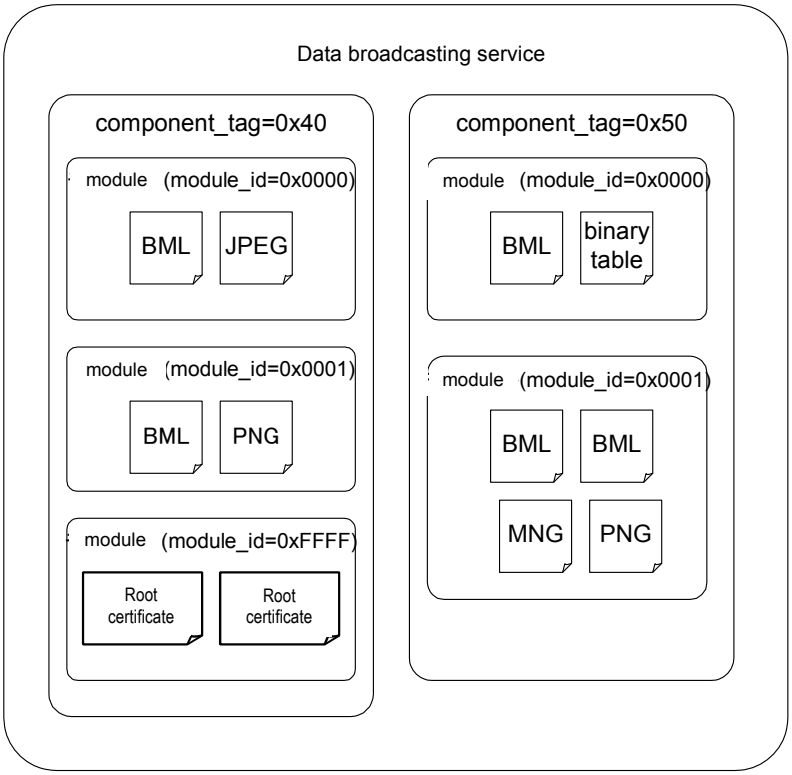


Figure 2-7 Transmission of root certificates

Broadcasters who provide programs that include interaction channel services using TLS or SSL should attach a root certificate at all times when broadcasting corresponding data broadcasting programs. Particularly for programs that register bookmarks or register calling using TLS or SSL should have root certificates attached for transmission. Also, it is recommended to check whether the required certificate is stored in receiver or not before registration. This is, however, except for cases where empty carousels are sent out by components of component_tag=0x40. Also, considering the receiver performance indicated in section 5.14.14.3, it is recommended to perform operations to store appropriate and the most recent general-purpose root certificates within receiver.

When transmitting root certificates, carry out the following steps.

- Use module_id=0xFFFF modules for component_tag=0x40 components to transmit root certificates. Store the resources which display route certifcates in this module in an entity format. For details on module components refer to Vol. 6.
- In order to make the loading volume of the filtering process of root certificates lighter, a Root Certificate Descriptor is located in the DII module information area(moduleInfoByte).

2.3.2 Operation of "DownloadInfoIndication"(DII)message

- Refer to figure 2-5 for the relationship between the DII and the local contents/data event.
- Due to restrictions on the processing ability of receiver, the minimum transmission interval of each DII component is constantly 100 milliseconds (excluding when the carousel is switched, etc.).
- "DSMCC_section" to transmit the DII is operated as specified.
- Module information stored in the DII message is stored in an ascending order for "moduleId". (However, "moduleId" is not always continuous.)
- Operation of "userNetworkMessage( )" is indicated in Table 2-8.

Table 2-8 DII: Operation of "userNetworkMessage( )"

Field	Operation	Comment
dsmccMessageHeader( )		
protocolDiscriminator	Operated as specified.	0x11
dsmccType	Operated as specified.	0x03
messageId	Operated as specified.	0x1002
transaction_id	Operated as specified. Transaction Number(the bottom 30 bits of "transaction_id") are updated in the following cases. ・ When data events are switched. ・ At least one module that composes the carousel is updated. ・ When the module number that compose the carousel is changed. (including vicinity of "numberOfModules=0") It is recommended that the value be incremented by 1. (When the value is changed, then the receiver should judge that the contents of the DII have been changed, this is not limited to when it is incremented by 1)	"contents_id" of the Data Contents Descriptor of the EIT is not always updated even when the transaction_id is changed.
dsmccAdaptionHeader( )	Not operated.	

Field	Operation	Comment
downloadId	Operated as specified. Updated at the time of switching the data event. bit31-28 data_event_id bit27-0 all 1 Operated for identification and switching of local contents and handling of event messages and local contents.	"Data_event_id" is operated to switch data events and to avoid the wrong reception of event messages in local contents that are next to each other time wise.
blockSize	Operated with a fixed value(4066).	4066
windowSize	Operated as specified.	0
ackPeriod	Operated as specified.	0
tcDownloadWindow	Operated as specified.	0
tcDownloadScenario	Operated. Among the modules that compose the carousel, the module with the longest sending cycle period is written.	The timeout time setup based on this value depends on a model.
compatibilityDescriptor( )	As specified in case of operating without contents.	compatibilityDescriptorLength=2 descriptorCount=0
numberOfModules	The maximum number of modules transmitted by 1 data carousel is 256. Also, to specify that this is an empty carousel, "numberOfModules=0" may be operated. For empty carousels, refer to 2.3.1.6.	
moduleId, moduleVersion	Operated as specified.	
moduleSize	The maximum value of the module size is 1MB. Refer to 2.3.3 for details.	
Module information area	Descriptor described later is located.	
Private data area	DII transmitted by the entry component of an event (tag value 0x40) may locate and operate the arib_bxml_private_data_descriptor( ) as specified. This descriptor is not located in DII's transmitted by other components. In case the (1) data event of the entry carousel is switched and (2)this descriptor exists and (3)"return_to_entry_flag" is 1, then the receiver should discard the document currently viewed and thestartup document of the entry carousel should be presented. Refer to 2.3.1.4.	descriptor_tag 0xF0 descriptor_length 1
Descriptor stored in the module information area		
Type Descriptor	In case 1 resource is mapped directly in 1 module, placement is required. It is not necessary for modules to store resources in an entity format.	
Name Descriptor	Not operated.	
Info Descriptor	Not operated.	
Module_Link Descriptor	Not operated.	
CRC Descriptor	Not operated.	

Field	Operation	Comment
Estimated Download Time Descriptor	Operated in some cases. When operated, specify the maximum transmission cycle period of the corresponding module.	Handling by basic receiver is optional.
Expire Descriptor	Not operated.	
ActivationTime Descriptor	Not operated.	
CompressionType Descriptor	Module may be compressed for transmission and in such case, it should be located. When module compression is not executed, then it is not located. Refer to 2.3.3 for details of module compression	
	Operation of compression_type	
	0:Compress module in the zlib format	
Control Descriptor	Operated as specified. When the BML version of the BML document matches the specification of the PMT, then this descriptor is not located. Also, use_xml is 0 at all times.	
RootCertificate Descriptor	Operated as specified.	Refer to Appendix-7

2.3.3 Operation of "DownloadDataBlock"(DDB)message

- The "DSMCC_section" that transmits DDB messages is operated as specified.
- Maximum size of modules transmitted by DDB messages is 1MB. To be more precise, the number of the "DSMCC_section" transmitting DDB is a maximum of 256.(Maximum module size is 4066 x 256=1040896 byte)
- Module may be compressed in zlib format for transmission. In this case, the "Compression Type Descriptor" is located in the module information area of the DII handling the corresponding module and the "compression_type" is set as 0. For details on the compression format, refer to Appendix-2.
- In case of compressed transmission of modules, the total module size before compression and after compression should not exceed the maximum module size above.
- Regarding the operation of the DDB(downloadDataMessage( )),
- Indicated in Table 2-9.

Table 2-9 DDB: Operation of "downloadDataMessage( )"

Field	Operation	Comments
dsmccDownloadDataHeader( )		
protocolDiscriminator	Operated as specified.	0x11
dsmccType	Operated as specified.	0x03
messageId	Operated as specified.	0x1003
downloadId	Operated as specified.	The same value as "downloadId" of DII is stored.

Field	Operation	Comments
adaptationLength	In case of a multi-media service, “dsmccAdaptationHeader( )” is not operated.	0
moduleId	The value of “moduleId” is not particularly specified.	
moduleVersion	Operated.	When it is updated, the value is not guaranteed to be incremented by 1.
blockNumber	Operated as specified.	Value obtained from moduleSize/blockSize

2.3.4 Operation of event messages

2.3.4.1 Purpose for operating event messages

- Event messages including the General Event Descriptor (hereinafter referred to as general event message) are operated to transmit data simultaneously attached to the event by generating events in multimedia contents specified by non-synchronized periods or time.
- In case of using an NPT as time specification in an general event message, or in case of using an NPT in multi-media contents, an event message including an NPT Reference Descriptor (referred to as NPT reference message from hereafter) is operated in order to notify the receiver of the relationship between the NPT and STC.
- Objective of NPT operation
  - Even when time shifts occur due to baseball game extensions or emergency news, etc., this is used this as a time specification that does not require correction/re-compilation of data broadcasting service contents. (It cannot be handled in relative time from the starting time or MJD)
  - It is the same when there are changes in CM duration during re-runs. (It cannot be handled in relative time from the starting time of the program or MJD)
  - Time precision may need to be 1/10 seconds for quiz shows, etc.(The relative time from the starting time of the program or MJD is not so precise.)
  - Time axis used in accumulated contents is necessary. (It cannot be handled by relative time from the starting time of the program or MJD)

2.3.4.2 Transmission of event messages

- General event messages are transmitted by components that transmit local contents, which use it in some cases, and may be transmitted by different components in other cases.
- NPT reference messages may be transmitted by multiple-components per one event.
- NPT reference messages and data carousels are not transmitted by the same component.

- “Event_section_flag” located in the Data Component Descriptor of the PMT and the Data Contents Descriptor of the EIT is not used. It is always 1.
- In “DSMCC_section” that transmits event messages, either the General Event Descriptor or NPT Reference Descriptor is located.
- The “last_section_number” of the “DSMCC_section” that transmits event messages is always 0. In other words, the subtable that transmits one event message is always transmitted by 1 section.

2.3.4.3 Transmission of general event messages

- From BML documents, general event messages transmitted by the same component as local contents which transmit corresponding documents or general event messages transmitted by different components can be used. The “component_tag“ of general event messages that can be specified by 1 BML document is up to 2 different types.
- “private_data_byte” is operated.
- The maximum number of General Event Descriptors that can be located in 1 DSMCC_section is 8.
- In case of executing setup to acquire the desired message_id while the message_id=255 or the message_id is omitted, the “message_id” is not set up at the same time as explicit specification of message_id(0~254). Also, in this case, (0~254) should be omitted (or 255 should be specified).
- In case of explicitly specifying the message_id, the maximum number of general event messages that can be subscribed simultaneously is 16.
- In order to avoid missed messages, a subtable with the same contents may be sent multiple times in general event messages.
- In this case, transmission intervals and the number of transmissions are not specified.
- In consideration of messages missed by receiver, it is recommended that the update interval of the DSMCC_section, which transmits general event messages within the same ES, be secured for more than 100ms. This indicates the recommended value of the interval between sending of the first DSMCC_section of any version to the DSMCC_section of next updated version. This is not to define the sending interval for the DSMCC_section of different versions that are adjacent to each other at the time of update.
- In general event messages that perform time specification by using the NPT, the sending of event message specified outside the effective period specified in the next section or after the NPT discontinue time specified as firing time NPT are not executed. Regarding NPT discontinuity, refer to the next section.

Operation of “data_event_id”, “event_msg_group_id” of general event messages

0 or 1 is used for the value of general event messages. The receiver will perform as follows.

- Regarding general event messages with the event_msg_group_id = 0, only when the value of the data_event_id of the event message is the same as the data_event_id of local contents that are currently presented, they are processed as a valid event message. If the data_event_id value is different, then the event message is ignored.
- Regarding general event messages with the event_msg_group_id = 1, regardless of the data_event_id value, they are processed as a valid event message.

The above definitions are applied in the same way when general event messages are transmitted by the same component as local contents, which it uses, or when they are transmitted by different components.

Normally, specifying the event_msg_group_id = 0 is recommended. This is, for example, just as in figure 2-8, when the event message is sent at a timing that is close to the switching of local contents, etc., adjoining local contents might receive it by mistake when event_msg_group_id=1 is specified.

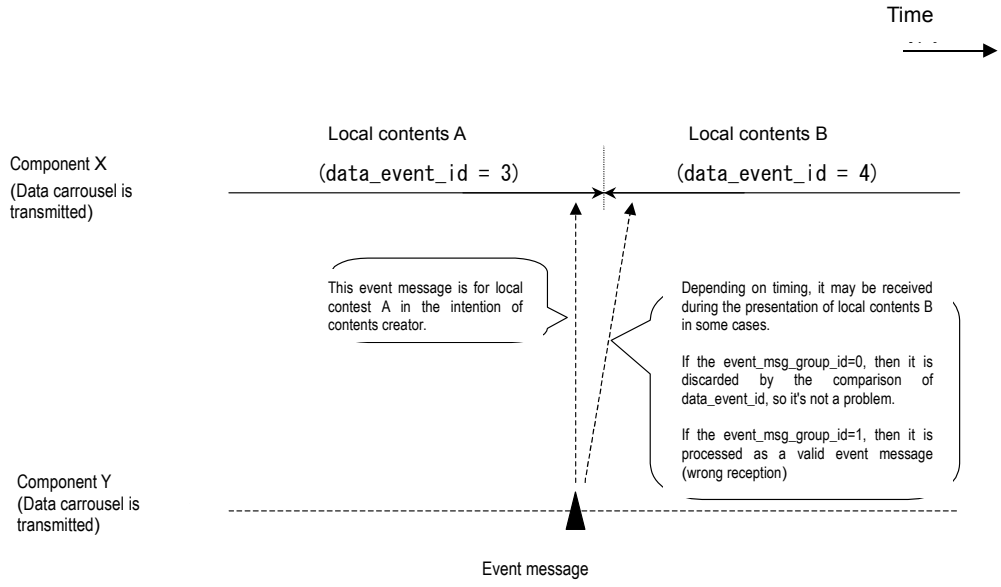


Figure 2-8 Receiving general event messages by mistake

Specification of “event_msg_group_id=1” is used when it is difficult to match the value of the “data_event_id” such as, for example, when the data carousel of the transmission station is used and the event message is sent by receiving station. (Figure 2-9)

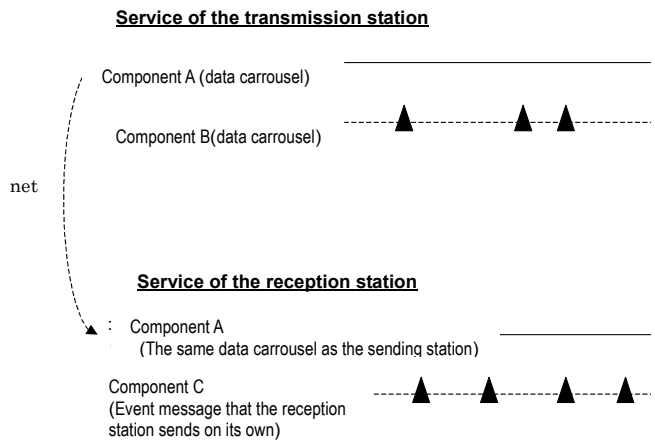


Figure 2-9 Example when the value of data_event_id cannot be matched.

2.3.4.4 Transmission of the NPT reference message

- The “table_id_extension” of “DSMCC_section” of the NPT reference message is fixed to 0xFFFF (“event_msg_group_id” = 0xFFF, “data_event_id” = 0xF). “event_msg_group_id”=0xFFF is used only for NPT reference messages.
- In case of sending NPT reference messages, the transmission interval is set as 1 second as specified, however, normally the minimum update interval of NPT reference messages is 5 seconds.
- In case of NPT stepping changes, an NPT reference message that notifies stepping changes is transmitted by 2 seconds before it happens.
- Specifications that make the NPT value go around in the middle of the data event should not be performed. However, sending NPT reference messages that make the NPT discontinue during data events is permitted. In this case, the timing of the discontinuity is assumed to be the time of event update, etc. and the frequent occurrence of discontinuity within a short time is not assumed. Regarding NPT related performance in case discontinuity occurs in referred NPT reference messages, refer to the next section.
- In case of using NPT in multi-media contents (in case of event firing by NPT or using getNPT), monitoring of the NPT reference message in multi-media contents is declared explicitly. (Written as event with an “NPTReferred” type attribute)
- Regarding transmission of the NPT reference message method, it is shown in figure 2-10.

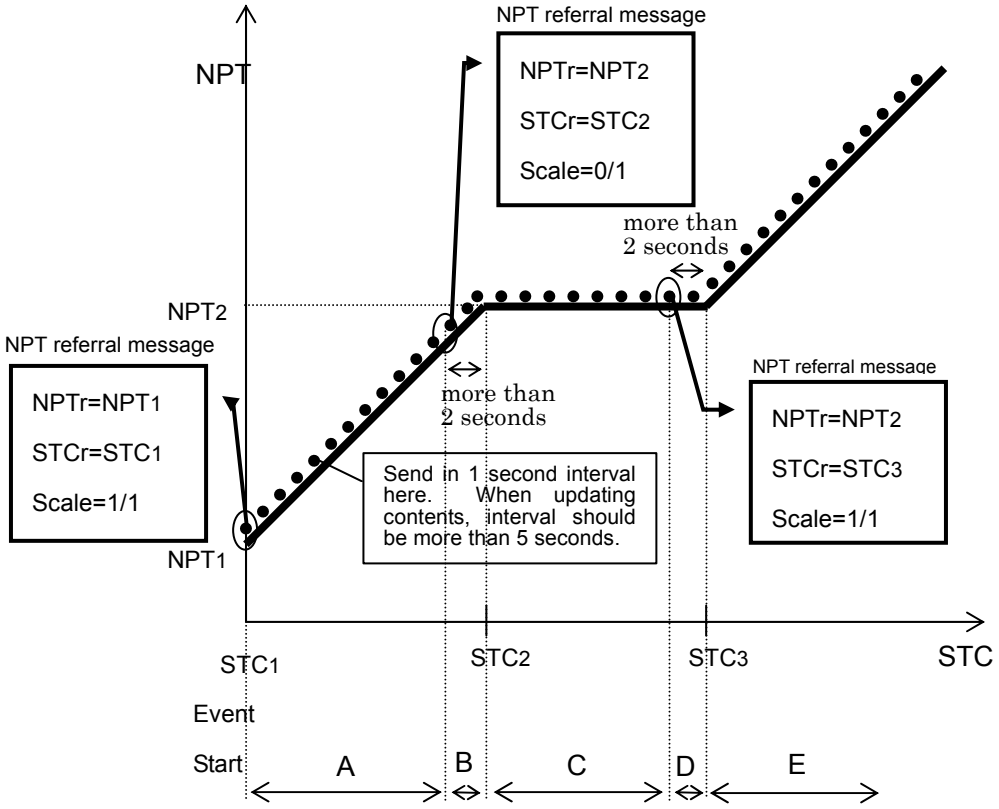


Figure 2-10 Transmission of the NPT reference message method

- Operations below are executed in order to handle 0 revolutions of the STC.
  Regarding the NPT Reference Descriptor to be sent or the NPT event message, send out the ones with STC value specified by the NPT Reference Descriptor or STC value corresponding to NPT that is specified by the NPT event message for the valid period range below at all times. (Refer to figure 2-11). This area is to specify the time range to the current STC value (STCc) and the valid period on the STC time axis is changed as needed with the following formula.

  Valid period: STCc + STC_PreWindow ~ STCc-STC_PostWindow

  However, on the actual STC time axis, the valid area will be STCc~STCmax and 0~(STCc + STC_PreWindow)-(STCmax+1) when the calculation value of (STCc + STC_PreWindow) exceeds the STC maximum value (STCmax), and 0~STCc and (STCmax+1)-(STC_PostWindow-STCc)~STCmax when calculation value of (STCc-STC_PostWindow) becomes negative.

  It is set as

  STC_PreWindow = 3888000000 (12 hours), STC_PostWindow = 3888000000(12 hours)

  (Hereafter, STC_PreWindow is also written as Wpre, and STC_PostWindow as Wpost)

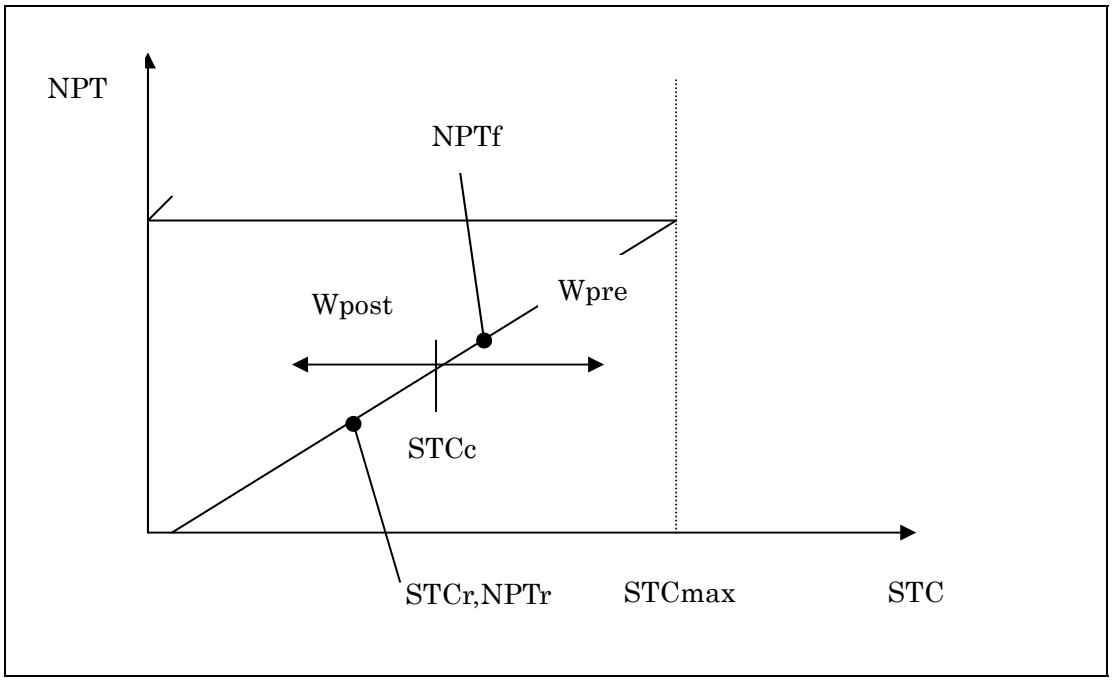


Figure 2-11 NPT valid period

2.3.4.5 Event message process in receiver

(1) General event messages

- When an event with a specification of “type” attribute as "EventMessageFired" is written in the BML document, then filtering of the DSMCC_section which transmits general event messages that match one of the following conditions should be carried out.
  (Condition 1)
  - “component_tag” specified in “es_ref” attribute
  - “event_msg_group_id”=0x000
  - “data_event_id” of local contents that are currently presented.

  (Condition 2)
  - “component_tag” specified in “es_ref” attribute
  - “event_msg_group_id”=0x001
- In case “es_ref” is omitted, it should be interpreted as a component of local contents that is currently presented.
- At the time of initial acquisition of the DSMCC_section or version upgrade detection of the DSMCC_section, the event as specified in the multimedia encoding method based on message_id/message_version specified in the BML document and event_msg_id of the General Event Descriptor included in the corresponding DSMCC_section is generated.

- Regarding operated time-mode
  0x00(immediate firing), 0x02(NPT time) are operated as the time mode specifying firing time.
- For firing times (if time mode 0x00, then event message receiving time) specified by general event messages, the receiver will generate events in multi-media contents as quickly as possible after the specified time. A rough standard is 100 milliseconds for both immediate firing and NPT specification.
- When multiple general event messages are received where the time mode is specified as 0 (immediate firing), then they will be fired in the order received. Timing for the process in this case is not specified.
- In receiver, queuing of a maximum of 8 general event messages before firing at the same time is possible.
- When multiple general event messages with immediate firing specifications that are transmitted by the DSMCC_section are the target, firing is in the order of the descriptors that are located in the DSMCC_section.
- When the specified NPT time is already passed at the time of reception, it will be fired immediately.
- The valid period of the general event message subscription is the period presented in the document that specified subscribe.

(2) NPT reference message

- When an interruption event with the “NPTReferred” attribute is described in the BML document, filtering of the”DSMCC_section” that transmits the NPT reference messages will be performed in the following condition.
  - “component_tag” specified by “es_ref” attribute
  - “event_msg_group_id”=0xFFF
  - “data_event_id”=0xF
- The above events are generated only when an NPT reference message is acquired for the first time, and events will not be generated even when a version upgrade of the DSMCC_section that transmits NPT reference messages takes place after that.
- When new NPT Reference Descriptors or General Event Descriptors specified by new NPT’s are incoming, then re-calculation of the valid period is done.
- When an NPT Reference Descriptor is received, check whether or not the STC value in the NPT Reference Descriptor is within the valid period. In case it is in the valid period, it is recognized as reception of the NPT Reference Descriptor. When receiving new NPT Reference Descriptors, re-calculation of the valid period is done.)

- After the event of "NPTReferred" is generated, if the NPT reference message that makes discontinuity of stepping for NPT stepping specified in the NPT reference message, which generated the same event was received, then the NPT related performance in the BML document depends on a model of receiver after that. Even when discontinuity is generated in NPT stepping as above, if an event with an “NPTReferred” attribute is subscribed after updating the BML documents, then execute the process specified in this section not as discontinuity of NPT stepping but as receiving new NPT reference messages. Therefore, for contents production, in order to secure stable performance of contents, transit of documents needs to be performed immediately after discontinuity occurs in NPT stepping. In case of using NPT related functions, NPTReferred interruption needs to be re-subscribed from the document after transition. As a method to transit documents after occurrence of the above NPT discontinuity, for example, event messages of immediate firing can be sent out immediately after occurrence of NPT discontinuity and reception of the event message written in the BML document and execution, etc. of transition along with the firing is assumed.
- For the calculation of NPT, a stepping calculation is done assuming STC 0 revolutions occur just once within the valid period.
- Firing process of NPT event message with firing time specified within the valid period is the same as it has been.(When new General Event Descriptors specified by NPT are incoming, then re-calculate the valid period)
- Regarding event message firing with specified NPT times and firing of events, while taking STC discontinuity into consideration, the implementation substituted by a timer process, etc. is possible.
- For the NPT value returned by getNPT, the return value corresponding to NPT stepping changes.
- Receiver record the most recently acquired NPT_Reference(NPTr), STC_Reference(STCr) and scale of NPT reference message as a set. At the time that the NPT value becomes necessary, based on this information and the STC value (STCc) acquired then, the NPT valid period and NPT value (NPTc) are calculated using the following algorithm.

NPT value calculation algorithm

```
STCc>STCr and STCc-STCr ≦ Wpost or
STCc<STCr and STCc+STCmax-STCr ≦ Wpost and
  Scale=1/1
      NPTc=mod((STCc+mod((STCmax+NPTr-STCr),STCmax)),STCmax)
                                          // Equivalent to sections A-E of Figure 2-10
  Scale=0/1
          NPTc=NPTr                       // Equivalent to section C of Figure 2-10

STCc>STCr and STCr+STCmax-STCc ≦ Wpre or
STCc<STCr and STCr-STCc ≦ Wpre and
  Scale=1/1
          NPTc=NPTr
                                          // Equivalent to section D of Figure 2-10
  Scale=0/1
       NPTc=mod((STCc+mod((STCmax+NPTr-STCr),STCmax)),STCmax)
                                          // Equivalent to section B of Figure 2-10

Only if STCmax=0x1FFFFFFF
```

- After presenting the document, the receiver performance in case the NPT related process is generated before the first NPT reference message is acquired(until the "NPTReferred" event is generated)is specified as follows.
  - At the reception time the general event message which performs the firing specification by NPT time
    The corresponding general event message is ignored and monitoring performance of the general event message is continued.
  - Event set up(bevent) by NPT time specification.
    Interruption event is not executed.
  - GetNPT
    Returns an error value
- The valid period of NPT reference message subscription is set as the period presented in the document specifying subscription.

2.3.4.6 Operation of the DSMCC_section( )

Table 2-10 Operation of DSMCC_section( ) transmitting event message

Field	Operation	Comments
table_id	As specified.	0x3D
section_syntax_indicator	As specified.	1
private_indicator	As specified.	0
dsmcc_section_length	As specified.	
data_event_id	General event message: In case the event_msg_group_id=0x000, the value is the same as the data_event_id of corresponding local contents. In case the event_msg_group_id=0x001, it is 0x0. NPT reference message: it is 0xF.	
event_msg_group_id	General event message: 0x000 or 0x001. NPT reference message: It is 0xFFF.	
version_number	In the same subtable which is identified by the combination of table_id, data_event_id, event_msg_group_id, the continuity of +1 is guaranteed.	
current_next_indicator	Always 1.	

2.3.4.7 Operation of General Event Descriptor

Table 2-11 Operation of the General Event Descriptor

Field	Operation	Comment
event_msg_group_id	The same value as the event_msg_group_id of the DSMCC_section.(0x000 or 0x001)	
time_mode	Operate the following modes 0x00: Event is fired immediately after reception 0x02: Event is fired in accordance with NPT time data	
event_msg_type	Always 1.	
event_msg_id(message_id, message_version)	Operated as specified.	
private_data_byte	Operated as specified.	Maximum 244bytes

2.3.4.8 Operation of the NPT Reference Descriptor

Table 2-12 Operation of the NPT Reference Descriptor

Field	Operation	Comment
postDiscontinuityIndicator	Only 0 is operated.	
dsm_contented	Not operated.(It is always 0.)	
scaleNumerator/scaleDenominator	“0/1” and “1/1” are operated.	

2.3.5 Operation of IIT

Still image transmission (still image carousel) by video PES whose objective is interactive playback is not operated. Therefore IIT is not operated.

2.3.6 Performance of related receiver

2.3.6.1 Filtering resources used when receiving data broadcasting

Necessary filtering resource for data carousels

Prior condition of transmission operation

- DII of the entry carousel in addition to the DII of the carousel that is currently presented is monitored. (Refer to section 2.3.1.4)
- The entry ES, and ES with a component_tag value of 0x50 or 0x60 can be locked by “lockModuleOnMemoryEx()”.(Refer to section 5.12.6.9.)

Table 2-13 Necessary filtering resources for data carousels

Target ES	Number of PID filters	Number of section filters	
		DII	DDB
ES that is currently presented	1	1	1
Entry ES	1	1	1
ES with component_tag=0x50,0x60	2	2	2
Cash target ES	1	1	1

Necessary filtering resources for event messages

Assumed transmission operation

- General event messages are transmitted by a maximum of 2 ES other than the ES that is currently presented. Also, since an “event_msg_group_id” with a value of 1 and 0 can be sent by each ES, 2 section filters are necessary respectively.
- A NPT reference message referred from a single BML document is transmitted with only one ES.

Table 2-14 Necessary filtering resources for event messages

Target ES	Number of PID filters	Number of section filters
General event message	2	4
NPT reference message	1	1

2.4 Temporary services and data broadcasting services

- Regarding the operation of temporary data broadcasting services, refer to Vol. 7.

2.5 Multi-view operations and data broadcasting services

- Broadcasting of TV programs with additional data is possible even during multi-view operation. However, it is data broadcasting services that are common for all component groups.
- The combinations of video, audio, data are described using the Component Group Descriptor.
- All component groups should include the same data broadcasting components. (Component transmitting data carousel or event message) In other words, all data broadcasting components should be included in all component groups during multi-view operation.
- Presentation of data broadcasting will not change even if the component group is switched. (Presentation of documents that were presented before switching will be continued after switching)
- ES limit number specified in 2.1.2.5 is applied to each component group.
- Return flag may be operated during multi-view. Refer to 2.3.1.7.
- When referring to the AV stream from the BML document in multi-view programs, only “-1” can be specified as the component_tag. (Refer to 2.1.2.8.)

2.6 Interaction channel transmission protocol used for data broadcasting services

- When performing interaction channel communication in data broadcasting services, the TCP/IP communication protocol is used. For TCP/IP communication protocol details, conform with Vol. 6.
- HTTP1.1 (RFC2616) is operated as an application layer. For operation of HTTP1.1, refer to 5.14.8.
- Basic system procedure is operated in receiver equipped with modems.

3 Operation of mono-media encoding

3.1 Video encoding

3.1.1 MPEG-1 Video

3.1.1.1 Restrictions of encoding parameters

Table 3-1 MPEG-1 Video Encoding parameter restriction

"Sequence Header" restrictions				Other parameters
vertical_size	horizontal_size	pel_aspect_ratio	picture_rate	
240	352	6, 12	4	Condition of "constrained parameters"
120	176			

Meaning of each code number for the MPEG-1Encoding parameters in Table 3-1	
pel_aspect_ratio	6 = 16:9 display (525line) 12 = 4:3 display (525line)
picture_rate	4 = 30/1.001 Hz

3.1.1.2 Synchoronized playback with audio (MPEG-2 AAC)

Synchronized playback of MPEG-1 transmitted by video PES and MPEG-2 AAC transmitted by PES is possible.

3.1.1.3 Other restrictions

MPEG-1Video is transmitted by video PES(stream_type 0x01).

3.1.2 MPEG-2 Video

- Vol. 7 section 4.1 shall be followed.
- It is possible to pseudo-display still images by transmitting intraframes only, however, restrictions in such case are specified in the video encoding category and is written as MPEG-2I.

3.1.2.1 Restrictions of encoding parameters

Table 3-2 MPEG-2 Video encoding parameter restrictions

Restrictions of the "Sequence Header" which are restrictions for the encoding parameter				Sequence extension restrictions	"Sequence display extension" restrictions			Other parameters
vertical_ size_value	Horizontal_size_ Value	Aspect_ ratio_ information	frame_ rate_ code	progressive_ sequence	color_ primaries	transfer_ characteristics	matrix_ coefficients	
1080	1440, 1920	2,3	4	0	1	1	1	Value specified for MP@HL
720	1280	2,3	7	1				
480	720	2,3	7	1				Value specified for MP@H14L
480	480, 544,720	2, 3	4	0				Value specified for MP@ML
240	352			1				Value specified for MP@LL

Meaning of each code number of MPEG-2 video encoding parameters in Table 3-2	
Aspect_ratio_information	1 = square pixel 2 = 4:3 display 3 = 16:9 display
Frame_rate_code	4 = 30/1.001 Hz, 7 = 60/1.001 Hz
Progressive_sequence	0 = interlaced scanning system 1 = progressive scanning system
Color_primaries	1 = Specified value of Rec.ITU-R BT.709(BT.1361)
Transfer_characteristics	1 = Specified value of Rec.ITU-R BT.709(BT.1361)
Matrix_coefficients	1 = Specified value of Rec.ITU-R BT.709(BT.1361)

Table 3-3 MPEG-2 I restrictions

"Sequence Header" restrictions				"Sequence extension" restrictions		"Sequence display extension" restrictions			Other parameters (note5)
vertical_ size_value	horizontal_ size_value	aspect_ ratio_ information	frame_ rate_code (note2)	Progressive_ sequence	low_ delay	color_ primaries	transfer_ characteristics	matrix_ coefficients	
1080 (note 1)	1920	2,3	4	0 (note 3)	1 (note 4)	1	1	1	Value specified for MP@HL
480	720	2,3	7	1					Value specified for MP@H14L
		2, 3	4	0 (note 3)					Value specified for MP@ML

Meaning of each code number of MPEG-2 I frame encoding parameters in Table 3-3	
aspect_ratio_information	1 = square pixel 2 = 4:3 display 3 = 16:9 display
frame_rate_code	4 = 30/1.001 Hz 7 = 60/1.001 Hz
Progressive_sequence	0 = interlaced scanning system 1 = progressive scanning system
low_delay	1 = B picture is not included.
color_primaries	1 = Rec.ITU-R BT.709(BT.1361)
Transfer_characteristics	1 = Rec.ITU-R BT.709(BT.1361)
matrix_coefficients	1 = Rec.ITU-R BT.709(BT.1361)

(note 1) 1088 is actually encoded in the MPEG-2 encoding method(ITU-T H.262). 8 lines worth of imaginary video data (dummy data) is added under the effective line in the encoder, and it will be encoded and processed as 1088 video data in reality. In the decoder, among the 1088 video data, the first 1080, in other words, the 1080 video signals with effective lines excluding dummy data are output.

(note 2) Timing of decoding and displaying is controlled by the time stamp value in the PES header and the vbv_delay value is 0xFFFF.

(note 3) In case the "progressive_frame=0"(there is a timing difference in 2 fields within the frame caused by the interlaced scan), then a freeze screen of the field should be displayed, and in case the progressive_frame=1(2 fields within the frame is the same timing), a freeze screen of the frame should be displayed.

(note 4) In case the low_delay=1, the time stamp for decoding and displaying will be the same value (DTS=PTS). Only PTS can be attached to a still image I frame.

(note 5) In case the sequence_display_extension is not transmitted, then each value for "color_primaries", "transfer_characteristics", "matrix_coefficients" are assumed to be equal to "1" and processed on the receiver side. The values specified in ISO/IEC 13818-2 for each level of the main profile are adapted to as the values of "vbv_buffer_size_value", etc. However, the "bit_rate_value" is the max value of each level and the maximum

capacity of MP@LL is 4Mbit/s, for MP@ML is 15Mbit/s, and for MP@H14L and MP@HL it is the maximum capacity that can be transmitted in Digital Terrestrial Television Broadcasting.

3.1.2.2 Other restrictions

- MPEG-2 Video is transmitted by video PES(stream_type 0x02)
- Scaling of MPEG-2I is not conducted.
- The following guidelines show the relationship in presentation timing control of MPEG -2I frame. Sending of the *n*-th +1 frame data shall start before the expected time to start presentation of *n*-th frame.

(Supplementary explanation)

"lowdelay" mode is specified when transmitting MPEG-2I.

The timing for starting display in receivers when in the "lowdelay" mode shall be in accordance with the PTS as a basic rule, however, for cases where the "sequence_end_code" is received after the PTS time, then the reception time of the "sequence_end_code" becomes the time to start the display. From the same perspective, the time expecting the above presentation can be understood as the PTS time or "sequence_end_code" receiving time. However, considering the general component of the receiver decoder, data from succeeding frames needs to be sent in order for this "sequence_end_code" to be recognized by the decoder (data is pushed out from the receiving buffer and arrives at the decoder itself). In this case, since the depth of this buffer is implementation dependent, data should be transmitted in bursting way without affecting the buffer control.

Also, the same process is needed for the presentation of the final frame, so after transmitting the final frame, data shall be transmitted at the expected timing for the start of presentation. In the meantime, considering the fact that above buffer size is different for each receiver, the same data as the final frame should be transmitted.

3.1.3 MPEG-4 Visual

Encoding methods using MPEG-4 Visual are not operated.

3.2 Still images and bitmap figure encoding

3.2.1 JPEG

- Shall be in compliance with the baseline method of ISO/IEC 10918-1(ITU-T T.81).
- Regarding the colorimetryof JPEG, B-24 Vol. 1, Part 1 7.2 "Colorimetry" is applied.

3.2.1.1 Encoding parameters

- Encoding sequence:Interleave method is used.
- Baseline method with loss.

- Sampling factor:It is set as YC_BC_R=4:2:0. However, it shall not collapse when receiving the 4:2:2 format. Also, the JPEG for the 4:2:0 format refers to the ones with the values of (2, 2), (1, 1), (1, 1) for the SOF0 marker information (H1,V1), (H2,V2), (H3,V3) respectively, and similarly the 4:2:2 format only refers to the ones with (2, 1), (1, 1), (1, 1) as the values of (H1,V1), (H2,V2), (H3,V3) respectively.

3.2.1.2 Scaling

Scaling is 128/128 only.

However, only when 960x540 pixel size images are transmitted and presented as 1920x1080 size images by expanding twice,horizontally and vertically on the receiver side, is 256/128 scaling operated.

3.2.1.3 Other restrictions

Presented screen size is less than the full screen size of a still picture plane.

Progressive mode is not operated.

3.2.1.4 Operated markers and marker segments

Operated markers and marker segments by JPEG are shown in Table 3-4.

Table 3-4 Marker/marker segments operated by JPEG

Marker	Comments	Receiver process
SOI	Starting of image	Normal process
DQT	Definition of the quantizing table	Normal process
DRI	Definition of the restart interval	Normal process(note 1)
SOFn	Starting of the frame Only SOF0(FFC0) becomes the target for decoding.	Normal process
DHT	Definition of the Huffman table.	Normal process
SOS	Start the scan.	Normal process
RSTm	End of the restart interval.	Normal process(note 1)
EOI	End of the image.	Normal process
COM	Comment	Normal process
APPn	To be used in the application	Ignore
DNL	Special size specification	Ignore

(note 1) Handling of DRI and RSTm when an error has occurred is implementation dependent.

Only up to the above SOI-DNL will appear in the baseline method, markers other than the above are handled as errors.

3.2.2 PNG

The PNG shall be in compliance with the specifications of ARIB STD-B24 Vol. 1 Part 2 5.3.

3.2.2.1 Encoding parameters

colortype=3

bitdepth=1, 2, 4, 8

When operating bitdepth 1, 2, 4, only the area of index 0-1, 0-3, 0-15 (fixed colors) can be specified for the color specifications of CLUT.

Image compression type =0 (zlib)

3.2.2.2 Chunks operated by PNG

Chunks operated by PNG are shown in Table 3-5. When chunks other than Table 3-5 are operated, the receivers will ignore them.

Table 3-5 Chunks operated by PNG

Chunk names	Operation details	
IHDR	Bit depth Color type Compression method Filtering method Interlace method	1, 2, 4, 8 3 only (palette specification) 0 only (deflate/inflate less than 32KB) 0 only 0 only (no interlace)
IDAT	Filter type	0 only (none)
IEND		

- PLTE is not operated.
- PLTE of the PNG chunk is not used for the specification of the color palette. Refer to the CLUT.

3.2.2.3 Other restrictions

The data storage format is non-interlace.

The presented figure size is less than the full screen size of the presented character shapes plane.

3.2.3 MNG

The MNG shall be in compliance with the specifications of ARIB STD-B24 Vol. 1 Part 2 5.4.

3.2.3.1 Chunks operated by the MNG

Chunks operated by the MNG are shown in Table 3-6. When other chunks are used, the receivers will ignore them.

Table 3-6 Chunk operated by MNG

Chunk names	Operation details.
MHDR	Required.
MEND	Required.
IHDR,PNG chunks,IEND	In compliance with operational specifications of PNG.
TERM	In compliance with ARIB STD-B24.
FRAM	In compliance with ARIB STD-B24.
DEFI	In compliance with ARIB STD-B24.

3.2.3.2 Restrictions of MNG operation

(1) Total data size

256KB

The total data size of the MNG file means the total data volume when different MNG's are deployed. Also, the data volume of each respective MNG is the horizontal picture element count x vertical picture element count x bit depth x PNG number.

(2) Total number of PNG images

64 images

The total number of PNG images refers to the total number of PNG images that are possessed by the MNG.

(3) PNG update cycle specification value

Minimum : 100 milliseconds

Maximum :5000 milliseconds

Specification unit:100 milliseconds

(4) Repeat

- In case of specifying an infinite repeat, then 0x7FFFFFFF shall be specified as the repeat count.
- In case of specifying a limited number, then the PNG cycling period x number of PNG image x repeat count shall not exceed 120 seconds.

(5) Display size

The maximum total PNG data size in 1 screen: 256KB

Total display area in one second shall not exceed 256 KB.

(6) Others

- The display location of the MNG cannot be changed during playback. (When streamStatusAttribute=play)
- PNG object size during MNG does not change.
- Location on the CLUT of the complete range of transparent colors is fixed.
- Even when there is a delay in the PNG update that should occur at the same time with other imaging executions, pixel skipping of PNG image is not executed and the display of PNG in order is executed. In case the "framing mode 0" is specified in the first frame, it shall be handled as "framing mode 1" regardless of the number of cyclic times.

3.3 Audio encoding

3.3.1 MPEG-2 AAC

In accordance with Vol.7, section 4.2.

3.3.1.1 Encoding parameters

Table 3-7 Encoding parameters of MPEG-2 AAC

Sampling frequency	Bit length
48kHz, 32kHz	16 bit

3.3.1.2 Transmission of MPEG-2 AAC

- Audio encoded in MPEG-2 AAC is transmitted by the audio PES(stream_type 0x0F)and the data carousel(stream_type 0x0D).
- Refer to 3.3.1.4 for the file format when transmitting by data carousel.

3.3.1.3 Restrictions in data carousel distribution

- File size is 512KB or less.
- When the stop control is specified, the audio currently being played can be stopped.
- Simultaneous playback with MPEG-2Video is not possible.

3.3.1.4 Data format of the AAC audio file

- It is in MPEG-2 AAC Elementary Stream Format.
- As shown in Figure 3-1, audio frames composed of audio data corresponding to the ADTS header are considered as “one unit”, and it is composed from this single unit or multiple units of those. (1 audio frame is 1024 sampling unit in PCM, so in 48kHz sampling, it is approximately 21.3 milliseconds)

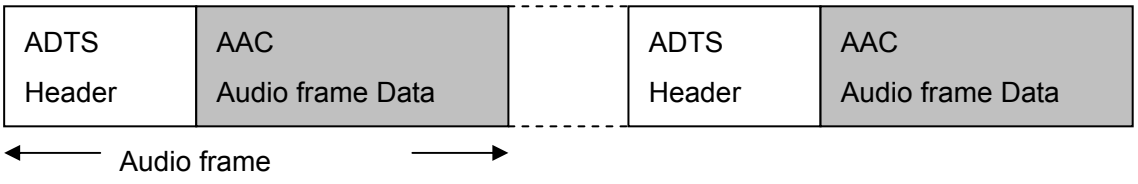


Figure 3-1 Data format of the AAC audio file

3.3.2 AIFF-C

3.3.2.1 Encoding parameters

Table 3-8 Encoding parameters of AIFF-C

Sampling frequency	Bit length
12kHz	16 bit

3.3.2.2 Maximum data volume

The maximum capacity is 96KB or less.

3.3.2.3 Other restrictions

- Audio encoded in AIFF-C is transmitted by data carousels(stream_type 0x0D).
- Basic receivers do not have to handle “Private_chunk”. (Chunks other than “Format_Version_Chunk”, “Extended_Common_Chunk” or “Sound_Data_Chunk”.)
- Number of channels is 1 channel.
- It does not have to support the repeat playback function. (Seamless play is not possible.)

3.3.3 MPEG-4 Audio

Audio encoding method using MPEG-4 is not operated.

3.3.4 Additional sounds

Additional sounds are not operated.

3.3.5 Built-in sound

The encoding method for sound built-in receivers is AIFF-C. The specifications of 3.3.2.1 shall be followed. However, depending on the implementation of the receiver, other encoding methods for equivalent functions can be used. Built-in sound sources are allocated in Table 3-9.

Table 3-9 Allocation of sound built-into receivers

0:Quick report chime 1	1: Quick report chime 2	2: Quick report chime 3	3: Quick report chime 4
4: Quick report chime 5	5: Button operation sound 1	6: Button operation sound 2	7: Button operation sound 3
8: Button operation sound 4	9: Button operation sound 5	10:Button operation sound 6	11:Button operation sound 7
12:Button operation sound 8	13:Alert sound	14:	15:

Numbers in the table indicate the sound_id in case they are specified from multi-media code, and they indicate the built-in sound specification in case of using expansion control codes PRA of 8-bit character codes.

The total capacity of receivers that use built-in sound sources is 480KB.

3.3.6 Audio synthesis by receivers

3.3.6.1 Mixing balance

When mixing audio distributed in different codes, the volume should be mixed in the ratio of 1:1.

3.3.6.2 Simultaneous playable encoding method

Simultaneous playback of multiple audio is only possible for the combination of O symbols in Table 3-10. ( ) indicates audio with a higher priority to be played when simultaneous playback is not possible. “AIFF-C file (quick report)” and “quick report super sound” refer to the built-in sound played by superimpose that specify automatic display.

If the quick report super sounds and other sounds are duplicated, then the quick report super sound has priority. (The quick report super sound shall be played continuously.)

If duplicated playback of AIFF-C is specified, then whichever is specified later has priority as a basic rule.

If the playback of MPEG-2 AAC files and AIFF-C are specified at the same time, then MPEG-2 AAC has priority for play.

Table 3-10 Audio encoding methods that support simultaneous play

	AAC-LC Stream (Main line)	AAC-LC file (storage)	AIFF-C file (storage)	AIFF-C file (Built-in sound)	AIFF-C file (Quick report)
AAC-LC Stream (Main line)	X	X (AAC stream is prioritized)	O (1)(2)(3)	O (1)(2)(3)	O (1)(2)(3)
AAC-LC File (storage)		X (Whichever is written later is prioritized)	X (AAC is prioritized)	X (AAC is prioritized)	X (Quick report is prioritized)
AIFF-C file (storage)			X (whichever is written later is prioritized)	X (whichever is written later is prioritized)	X (Quick report is prioritized)
AIFF-C file (built-in sound)				X (Whichever is written later is prioritized)	X (Quick report is prioritized)
AIFF-C file Quick report)					X (Whichever is written later is prioritized)

(1) While performing composite output of MPEG-2 AAC audio PES and AIFF-C, if the MPEG-2 AAC audio PES disappears, then playback of AIFF-C is not guaranteed.

(2) When AIFF-C is being played independently (without composite), it cannot be combined with the MPEG-2 AAC audio PES in the middle of playback.

(3) When the sampling frequency of MPEG-2AAC-LC (AAC-LC stream main line) that is transmitted by audio PES is 32kHz, simultaneous playback is not possible. Playback of MPEG-2AAC_LC transmitted by the audio PES is prioritized.

3.4 Character encoding

3.4.1 8-bit character codes (including EUC-JP)

JIS compatible *kanji* first-plane group, second-plane group, additional symbol group are not used.

3.4.1.1 Restrictions on the character coding function

Whether or not the character coding function used as mono-media which is referred from multi-media code are usable/non-usable is shown in Table 3-11. To find out if the character code function in caption/superimpose is usable/non-usable, refer to 4.5 ”Control codes used in caption/ superimpose” in this document.

Table 3-11 Character code function usable/non-usable

(1) C0 control groups

C0 control codes	Control function	Usable in multi-media codes	Restricted matters and supplements
NUL	Empty	O	
BEL	Bell	X	
APB	Active position backwards	O	
APF	Active positionforwards	O	
APD	Active position down	O	
APU	Active position up	O	
APR	Active position return	O	
PAPF	Parameterized active position forward	O	
APS	Active position set	O	
CS	Clear screen	X	
CAN	Cancel	X	
ESC	Escape	O	
LS1	Locking shift 1	O	
LS0	Locking shift 0	O	
SS2	Single shift 2	O	
SS3	Single shift 3	O	
RS	Data separator	X	
US	Unit separator	X	

O:Usable Δ:Usable with some restrictions X:Non-usable

(2) C1 control groups

C1 control codes	Control function	Usable/non-usable in multi-media codes	Restricted matters and supplements
BKF	Foreground color is black and specify color map lower address.	O	0 is specified in color map lower address of foreground color.
RDF	Foreground color is red and specify color map lower address.	O	1 is specified in color map lower address of foreground color.
GRF	Foreground color is green and specify color map lower address.	O	2 is specified in color map lower address of foreground color.
YLF	Foreground color is yellow and specify color map lower address.	O	3 is specified in color map lower address of foreground color.

C1 control codes	Control function	Usable/non-usable in multi-media codes	Restricted matters and supplements
BLF	Foreground color is blue and specify color map lower address.	O	4 is specified in color map lower address of foreground color.
MGF	Foreground color is magenta and specify color map lower address.	O	5 is specified in color map lower address of foreground color.
CNF	Foreground color is cyan and specify color map lower address.	O	6 is specified in color map lower address of foreground color.
WHF	Foreground color is white and specify color map lower address.	O	7 is specified in color map lower address of foreground color.
COL	Color specification	Δ	17 colors of receivers common fixed color and 207 colors of broadcaster setup color 207 can be used. Specifications by COL specifies index value (0-223) of CLUT with upper 4 bit by palette number and specifies lower 4 bit by CMLA. Example: 1) palette number = 0, CMLA = 0, then index value of CLUT = 0. 2) Palette number = 10 and CMLA = 5 indicate that 0xA5 and index value of CLUT = 165.
POL	Pattern polarity	Δ	Only POL 04/0(normal polarity)and POL 04/1(inverse polarity1) are operated. The process when normal polarity is set as inverse polarity is, background color → foreground color, half foreground color → half backround color, half backround color → half foreground color, foreground color → background color.
SSZ	Small size	O	Not specified for DRCS.
MSZ	Medium size	O	Not specified for DRCS.
NSZ	Normal size	O	
SZX	Specification size	Δ	Enlargement by doubled vertical, doubled horizontal or doubled vertical/horizontal doubled can be used.
FLC	Flashing control	X	
CDC	Conceal control	X	
WMM	Change writing mode	X	
TIME	Time control	X	

C1 control codes	Control function	Usable/non-usable in multi-media codes	Restricted matters and supplements
MACRO	Macro specification	X	Macro specification of the control code is not used and only the default macro is used.
RPC	Character repeat	O	
STL	Start underline and mosaic separation	Δ	Only underline is operated. Refer to 4.5.5 for operation of underline.
SPL	Finish underline and mosaic separation	O	Only underline is operated.
HLC	Highlighting character block	Δ	Refer to 4.5.4 for operation of boxes.
CSI	Control sequence introducer	O	

O:Usable Δ:Usable with some restrictions X:Non-usable

(3) Expansion control codes(CSI)

Character	Control function	Usable/non-usable in multi-media codes	Restricted matters and supplements
SWF	Set writing format	Δ	Specification is possible only before appearance of characters that require display performance and control codes after initialization of the display screen. Only one code parameter is used and parameters that can be specified to P11…. Pli are 7(960x540, horizontal writing) and 8(960x540 vertical writing) when pixel size of the character shapes plane is 960x540, and 9(720x480 horizontal writing) and 10(720x480 vertical writing) when pixel size of the character figure is 720x480.
CCC	Composite character composition	X	
RCS	Raster color command	O	Specification is possible only before the appearance of characters that require display performance and control codes after initialization of the display screen. For type of colors, receiver common fixed color and broadcaster setup color (index value 0~223) can be specified. Refer to 4.5.3 for the operation of raster color control. However, areas that are raster color controlled as rectangles are specified in CSS style of the object element of BML.

Character	Control function	Usable/non-usable in multi-media codes	Restricted matters and supplements
ACPS	Active coordinate position set	X	
SDF	Set display format	X	
SDP	Set display position	X	
SSM	Character composition dot designation	Δ	Only 16x16,20x20,24x24,30x30,36x36 dots can be specified. (note 1)
PLD	Partial line down	X	
PLU	Partial line up	X	
SHS	Set Horizontal spacing	O	
SVS	Set vertical spacing	O	
GSM	Character deformation	X	
GAA	Coloring block	X	Colored area is full display zone only.
SRC	Raster color designation	X	
TCC	Switch control	X	
CFS	Character font setup	Δ	3 font styles specified in chapter 4, which are(1. round gothic(P1:03/01), 2. angle gothic(P1:03/02), 3. bold round gothic(P1:03/03)) can be specified. (note 1)
ORN	Ornament control	Δ	Character decoration cannot be used, only outlines can be used. Refer to 4.5.6 for the operation of outlines.
MDF	Font	X	
PRA	Built-in sound replay	X	
XCS	External character set	X	

O: Usable Δ:Usable with some restrictions X:Non-usable

(Note 1) Only the combination of character fonts and font sizes specified in Chapter 1 can be specified.

3.4.1.2 Character sets used in data broadcasting

Table3-12 Character sets used in data broadcasting

Character sets	BML Document *1	8-bit character codes to reference external BML documents *2	Closed caption/ Superimpose *3
Alphanumeric set (1byte code) specified in ARIB STD-B24	O	O	O
Hiragana set (1byte code) specified in ARIB STD-B24	X	O	O
Katakana set (1byte code) specified in ARIB STD-B24	X	O	O
Kanji set (2 byte code, Row 1-94) specified in ARIB STD-B24 *7	O	O	O
DRCS character set (1 byte code) specified in ARIB STD-B24	X	X	O
DRCS character group (2 byte code) specified in ARIB STD-B24	Δ*4	Δ*5	O
Macro code set (1byte code) specified in ARIB STD-B24	X	O*6	O*6
JIS compatible *kanji* first plane set specified in ARIB STD-B24	X	X	X
JIS compatible *kanji* second plane set specified in ARIB STD-B24	X	X	X
Additional symbol set specified in ARIB STD-B24	X	X	X

*1 ARIB STD-B24 Vol. 2 Appendix 2, EUC-JP is used as specified in 4.1Character codes.

*2 Code table, expansion techniques, invocation of code element, designation of graphic sets and final byte of various code sets are operated as specified in ARIB STD-B24 Vol. 1, Part 2, 7.1. However, the JIS compatible kanji first plane set, JIS compatible kanji second plane set, additional symbol set, and DRCS character sets are excluded. For specifications on the display format etc, refer to Table 4-14.

*3 Code table, expansion techniques, invocation of code element, designation of graphic sets and final byte of various code sets are operated as specified in ARIB STD-B24 Vol. 1, Part 2, 7.1. However, the JIS compatible kanji first plane set, JIS compatible kanji second plane set, and additional symbol set are

excluded. For specifications on the display format etc, refer to Table 4-14.

*4 As specified in the specifications of ARIB STD-B24 Vol. 2 Appendix 2 4.1 Character codes, Row 87 and Row 88 of the kanji set are used as the DRCS area.

*5 Operate in accordance with the specifications specified in ARIB STD-B24 Vol. 2 Appendix 2 4.1Character codes. BML documents and 8-bit character codes, which are externally referenced from BML documents share Row 87-88 of the kanji set as the DRCS area.

*6 Use default macros only.

*7 For details, ARIB STD-B3 Appendix-(3) should be followed. The recommended company for the fonts in Row 90-91 is a VICS implemented foundation.

3.4.1.3 Initialization of character codes

Operations to initialize character codes shall be executed when starting presentation for each object included in BML documents that refer to 8-bit character codes.

The initial value is written horizontally and pixel size is the same as BML documents. The initialization of invocation and designation of the code, operation, and state shall be in accordance with the specifications of ARIB STD-B24 Vol. 1, Part 3, Chapter 8 Initializing operation. The initial values of character size controls, interval between characters, interval between lines are in accordance with Table 4-15. Also, the initial values for the character font, foreground additive colors and backgrounds additive color are round gothic, index value = 15, index value = 30 for each. The initial raster value color is transparent (index = 8).

3.4.2 Universal multi-octet coded Character Set

Universal multi-octet coded Character Set is not operated.

3.5 Description command figure encoding

3.5.1 Geometric

Geometric is not operated.

4 Operation of caption and superimpose encoding

4.1 Scope of service and definitions

The following two services of caption and superimpose are performed in Digital Terrestrial Television Broadcasting.

Caption : Caption Services (i.e.: translation caption) synchronized with main video, audio, and data.

Superimpose : Superimpose services (i.e.: emergency news, programming notice, time tones, etc.)

There are normal services and roll-up mode services for caption. The roll-up mode is a mode where caption data sent as page data are additionally displayed in units of lines and roll-up performance is assumed each time a line is added. The roll-up mode is classified as a separate display mode from normal caption in caption management data, and it can only be written horizontally and scrolled vertically.

Implementation of the roll-up mode is optional; however, even for receiver units without implementation, it is required for display as normal mode.

Details of operation and limitations of the roll-up mode are described in 4.10.

4.2 Composition and transmission operation

4.2.1 Specification for composition and transmission

(1) Transmission methods

Caption/superimpose are transmitted by the independent PES Transmission method(stream format identification0x06).

(2) Composition

Each caption and superimpose is transmitted by a separate ES respectively. Also, they are transmitted by the same PMT at the same time with the main service, and caption data is not distributed within the same program or before the program starts.

(3) Number of ES

The number of ES's for caption and superimpose that can be transmitted simultaneously is one ES each and a total of 2 ES. However, in case of multi-view, the maximum number of ES's for caption is 1 and the maximum number of ES's for superimpose is 1 for each component group.

(4) ES in case of multi-view

The maximum number of ES's for caption is 1 and the maximum number of ES's for superimpose is 1 for each component group.

Caption/superimposed character ES that appear in the component_group_id=0(default component group that are played while selecting stations) are the default ES of caption and superimpose respectively. For caption/superimposed character ES that appear in component groups where the component_group_id is not 0, and do not appear in component groups where the component_group_id=0, a fixed operation is not executed.

(5) ES of temporary services

The maximum number of ES's for caption is 1 and for superimpose is 1 for each temporary service.

(6) Transmission of multiple languages.

The maximum number of languages that can be transmitted at the same time is 2 languages per one ES, Language identification is done by caption management data in the ES.

(7) Bit map data

Bit map data can be used for superimpose.

(8) Usable display mode

For caption, only the" Automatic display when received/selection of display when recording and playback" and "selection of display when received/selection of display when recording and playback" can be operated. For superimpose, only "Automatic display when received/automatic display when recording and playback" and "Automatic display when received/selection of display when recording and playback" and "selection of display when received/selection of display when recording and playback" can be operated. In case of transmitting multiple languages, display mode of those languages will be the same. On the other hand, receiver performance in case of transmitting against the above is implementation dependent, however, automatic display is prioritized.

(9) Operation of alert sounds/additional sounds

Limited to built-in sound of receiver units, alert sounds can be operated for caption/superimpose. Additional sounds are not operated for caption/superimpose.

(10) Data Contents Descriptor

The Data Contents Descriptor of the EIT is not described since superimpose are not associated with events. 1 descriptor is written for caption per 1 ES. However, in case parameters do not match the Data Component Descriptor of the PMT or the set up of caption management data, then the Data Component Descriptor of the PMT or the set up of caption management are prioritized for each parameter in the display mode, number of languages and language code in receiver unit performance.

(11) Transmission frequency of caption management data

Since the caption management data includes information that is necessary to display caption and superimpose, caption texts cannot be displayed until the caption management data is

received. Therefore, considering the time for the channel selection, the caption management data is sent in the following intervals at the time of sending normal caption/superimpose.

The maximum sending frequency: 1 time / 0.3 second

The minimum sending frequency: 1 time / 5.0 seconds

However, it may be interrupted due to commercials, etc.

4.2.2 PES transmission methods used in caption

A synchronous PES transmission method is applied and synchronization of timing is initiated by the PTS. Parameters set in PES packets are indicated in Table 4-1.

- Setup parameter : Refer to Table 4-1
- The maximum number of ES sent to the same layer simultaneously: 1ES
- The maximum number of languages per 1ES : 2 languages
- The constitutional unit of PES : 1 data group
- The maximum size of PES : 32KB
- The minimum sending interval of PES packets : 100 milli-seconds
- The maximum ES rate : 256Kbit/s
- Receiving buffer : More than or equal to 64KB(In both cases of 1 language and 2 languages)

Additionally, 16KB is necessary for DRCS, refer to 4.6.

Table 4-1 Setup parameters of PES packets for caption

Field	Operation
Stream_ID	0xBD(private_stream_1)
PES_packet_length	Continuing number of byte in PES packet *1
data_identifier	0x80
private_stream_id	0xFF
PES_data_packet_header_length	Indicates the field length of PES_data_private_data_byte. Normally, input 0x00.*2
PES_data_private_data_byte	This field may be skip read. *2
Synchronized_PES_data_byte	Data of data group of caption is stored.

*1 Insert 0 in this value to prohibit operations without a defined PES packet length.

*2 The current length of PES_data_private_data_byte should be specified in PES_data_packet_header_length in case of operating PES_data_private_data_byte.

The following restrictions are stipulated regarding the sending of PES packets.

- Sending sequence of PES packets and time sequence of the PTS should not be reversed.
- At PTS time of *n*th PES packet, the total information Vol. of the PES packet, which started sending after the *n*th PES packet should not exceed the receiving buffer capacity (64KB).

- The sending of PES packets should be completed before Td from PTS time. Td refers to the time between completion of receiving and completion of presentation. The rough standard for this is 0.5 seconds.
- For intervals of sending PES packets of the caption text data group, the interval between PTS time of the *n*th PES packet in the sending sequence and PTS time of the n-1st PES packet should be bigger than Td of *n*th data. When the total delay T of the video satisfies the following conditions, it is possible for the sender side to send in sync with video images.

  T > LX8 / R + Td

  L refers to the maximum PES packet length and R refers to ES bit rate in this section.
- Performance of receiver units in case it exceeds the receiving buffer is implementation dependent.

4.2.3 PES transmission methods used in superimpose

A non-synchronous PES transmission method is applied. Parameters set in the PES packet are indicated in Table 4-2.

- Setup parameters : Refer to Table 4-2
- The maximum number of ES transmitted to the same layer at one time: 1ES
- The maximum number of languages per 1 ES : 2 languages
- The constitutional unit of PES : 1 data group
- The maximum size of PES : 32KB
- The minimum sending interval of PES packets: 100 milli-seconds
- The maximum ES rate : 256Kbit/s
- Receiving buffer : More than or equal to 64KB(In both cases of 1 language and 2 languages)

Other than the above, 16KB is required for the DRCS, refer to 4.6.

Table 4-2 Setup parameters of PES packets for superimpose

Field	Operation
Stream_id	0xBF(private_stream_2)
PES_packet_length	Number of succeeding bytes during PES packet. *1
data_identifier	0x81
private_stream_id	0xFF
PES_data_packet_header_length	Indicates the field length of PES_data_private_data_byte. Normally 0x00 should be inserted. *2
PES_data_private_data_byte	This field can be skip-read. *2
Asynchronized_PES_data_byte	Data of data group of caption is stored.

*1 Insert 0 in this value to prohibit operation without a defined PES packet length.

*2 In case of operating PES_data_private_data_byte, the correct length of PES_data_private_data_byte should be specified in the PES_data_packet_header_length.

The following restrictions are stipulated regarding the sending of PES packets.

- Bit map data units can be used only when the TMD is free.
- Regarding the interval of PES packet sending for the caption/statement data group, the interval between *n*th PES packet in the sending sequence and n+1st PES packet should be bigger than the Td of *n*th data. The rough standard of time from completion of receiving to completion of presentation is 0.5 seconds for text only, in case of bit map data of 32KB, it is 3 seconds.
- For the completion time of sending the *n*th PES packet + Td time, the total information Vol. of the PES packet should not exceed the receiving buffer Vol. (64KB).
- Performance of receiver units in case the receiving buffer is exceeded is implementation dependent.

4.2.4 Operation of data groups

Along with the update of caption management data , a “data_group_id” should be sent after switching the data group from group A to group B and from group B to group A. However, in case caption management data is not sent for more than 3 minutes, either one of group A or group B is sent out regardless of which group was sent earlier. The “data_group_version” is not used. When the caption management data is group A, the receiver unit processes only the caption texts (this statement, bitmap data, DRCS) of group A and when the caption management data is group B, the receiver unit only processes caption texts of group B.

In case the caption management data is the same as the caption management data of the group already received, it is processed as caption management data that was re-sent and the initialization operation by caption management data is not executed. When receiving the same caption texts as already received caption management data multiple times, each caption text is processed as the new caption text.

Table 4-3 Data group parameters

Field	Operation
data_group_id	Operated as defined.
data_group_version	It is not operated.
data_group_link_number	0x00
last_data_group_link_number	0x00
data_group_size	Operated as defined. However, 1 PES packet should not exceed 32KB.
data_group_data_byte	Data group data (Caption management data、caption text data) is stored.
CRC_16	Error detection is operated by CRC16. When an error is detected, the receiver unit will discard the corresponding data group.

4.2.5 Operation of caption management data

Within the same caption management data, data units for the same or different data unit parameters can be located multiple times. In case multiple data units exist within the caption management data, then it is processed in the appearing sequence of the data unit.

However, data that can be written in this document is only the control codes of SWF, SDF, SDP, SSM, SHS, SVS and the character code group that requires screen display cannot be written.

Any changes in line intervals and character intervals and format setup will be done immediately after CS.

In case of specifying SVS, SHS and SWF in the texts of management data, SCS, SHS and SWF should be specified in the text body of the caption texts. Also, for the roll-up mode, the changes in line intervals and character intervals and format setup should be done immediately after changing the group from A/B.

4.2.5.1 Caption management data used in caption

Caption management data should be sent at least every 3 minutes. In case caption management data is not received for more than 3 minutes, then the initialization operation of receiver unit at the time of selecting stations is done. Parameters that can be specified in caption management data used in caption are shown in Table 4-4.

Table 4-4 Parameters of caption management data for caption

Fields	Operations
TMD	‘00’(Free)
num_languages	1-2
language_tag	0-1
DMF	‘0010’(Automatic display when received/Selection of display when recording and playback) ‘1010’(Selection of display when received/Selection of display when recording and playback)
ISO_639_language_code	Language codes to be used
Format	‘1000’(Written horizontally in 960x540) ‘1001’(Written vertically in 960x540) ‘1010’(Written horizontally in 720x480) ‘1011’(Written vertically in 720x480)
TCS	‘00’(8-bit character codes)
rollup_mode	‘00’ (Non-roll up) ‘01’ (Roll up) ‘10’ (Reserved) ‘11’ (Reserved)
data_unit_loop_length	Operated as defined. However, 1 PES packet should not exceed 32KB.
data_unit	Data unit(this sentence, DRCS)is stored.

4.2.5.2 Caption management data used in superimpose

Considering the superimposition of time, real time setup of the TMD on top of a free setup is possible to execute synchronization of time by STM. In case caption management data is not received for more than 3 minutes, then the initialization operation of the receiver unit while selecting stations is done. Parameters that can be specified in caption management data used in superimpose are indicated in Table 4-5.

Table 4-5 Parameters of caption management data for superimpose

Fields	Operations
TMD	’00’(Free) ’01’(Real time) Free and real time cannot be mixed when presenting.
num_languages	1-2
language_tag	0-1
DMF	‘0000’(Automatic display when received/Automatic display when recording and playback) ‘0010’(Automatic display when received/Selection of display when recording and playback) ‘1010’(Selection of display when received/Selection of display when recording and playback)
ISO_639_language_code	Language codes to be used.
Format	‘1000’(Written horizontally in 960x540) ‘1001’(Written vertically in 960x540) ‘1010’(Written horizontally in 720x480) ‘1011’(Written vertically in 720x480)
TCS	‘00’(8-bit character codes)
data_unit_loop_length	Operated as defined. However, 1 PES packet should not exceed 32KB.
data_unit	Data unit(this text and DRCS)is stored.

4.2.6 Operation of caption statement data

Multiple data units of the same or different data unit parameters can be located within the same caption statement data. In case multiple data units exit from the same caption statement data, it is processed in the appearance sequence of the data units.

Parameters that can be set in caption statement data are indicated in Table 4-6.

Table 4-6 Parameters of caption statement data

Field	Operation
TMD	’00’(Free) ’01’(Real time): Superimpose only However, the same value as caption management data should be set within the same program.
STM	Operated as defined. Valid only when, in Data Component Descriptor of PMT, timing=’10’(Time synchronization)
data_unit_loop_length	Operated as defined. However, 1 PES packet should not exceed 32 KB.
data_unit	Data unit(this text and DRCS、bit map data)is stored.

4.2.7 Operation of data units

Parameters that can be set in data units are indicated in Table 4-7.

Table 4-7 Parameters of data units

Fields	Operations
unit_separator	It is 0x1F as defined.
data_unit_parameter	0x20(Texts) 0x35(Bit map data) 0x30(1 byte DRCS) 0x31(2 byte DRCS)
data_unit_size	Operated as defined. However, 1 PES packet should not exceed 32KB.
data_unit_data_byte	Data unit data is stored.

4.2.8 OPERATION OF PSI/SI

4.2.8.1 Operation of component tags

Component tag value of caption ES is 0x30-0x37 in case of transmitting through layers other than the partial reception layer, in case of transmitting through the partial reception layer, it is 0x87. As for component tag value of superimpose ES, it is 0x38-0x3F when transmitting through layers other than the partial reception layer, in case of transmitting through the partial reception layer, set value within the range of 0x88. However, for component tag values of default ES of caption, set 0x30 or 0x87, for component tag value of default ES of superimpose, set 0x38 or 0x88.

4.2.8.2 Operation of the PMT

Updating the PMT is basically to add/delete ES information when starting/ending caption and superimpose, however, operations to write ES information at all times is possible, too.

4.2.8.3 Stream format identification

Stream_type of caption/superimpose ES is 0x06(Independent PES_packet)

4.2.8.4 Descriptor operation

Descriptor operation of the PMT and EIT for caption/superimpose is indicated in Table 4-8.

Table 4-8 Descriptor operation of the PMT and EIT

Descriptors	PMT	EIT
Stream Identifier Descriptor	Required	-
Data Component Descriptor	Required	-
Data Contents Descriptor	-	H-EIT[p/f]、 M-EIT[p/f]: Required H-EIT[schedule]、 M-EIT[p/f after]: Arbitrary (superimpose are not written)

4.2.8.5 Data Component Descriptor

The data_component_id of the Data Component Descriptor is 0x0008 for both caption and superimpose. Also, parameters set in additional information ID are indicated in Table 4-9.

Table 4-9 Setup parameters of additional information ID of the Data Component Descriptor

Field	Operations
DMF	’0011’
Timing	Caption :’01’(Program synchronization) Superimpose :‘00’(Non-synchronization) or ‘10’(Time synchronization)

4.2.8.6 Target Region Descriptor

The Target Region Descriptor is not used.

4.2.8.7 Data Contents Descriptor

Parameters that can be set in the Data Contents Descriptor and its selector area are indicated in Tables 4-10 and 4-11. However, in case the setup parameter value does not match the PMT Data Component Descriptor and caption management data during the same event, set values of Data Component Descriptor and caption management data are prioritized.

Table 4-10 Setup parameters of the Data Contents Descriptor for caption

Fields	Operations
data_component_id	0x0008
entry_component	“component_tag” value of corresponding caption ES.
num_of_component_ref	0 should be specified.
component_ref	Not necessary because “num_of_component_ref=0”.
ISO_639_language_code	Fixed to jpn(Japanese).
text_length	The maximum value is 16(byte).
text_char	Contents of caption displayed in EPG are written.

Table 4-11 Setup parameters in the selector area of the Data Contents Descriptor for caption.

Fields	Operations
num_languages	The same value as caption management data.
DMF	The same value as the Data Component Descriptor.
ISO_639_language_code	The same value as caption management data.

4.3 Visual pixel size and display format of caption/superimpose

4.3.1 Display format

The usable display format is 960x540 and 720x480 in horizontal and vertical writing. Also, pixel size and the display format of caption/superimpose of the video plane should be in the combinations in Table 4-12 for horizontal and vertical writing respectively. When displaying in 720x480, it should be in the same display format regardless of the aspect ratio of video, and in consideration of the aspect ratio for display, it is corrected by the sender side.

Table 4-12 Combinations of video display format and pixel size

Pixel size of video plane	Display format of caption and superimpose
1920X1080	960X540
720X480	720X480

For data broadcasting programs, it is the same as above.

Even in case of combining HD data broadcasting for SD video, the caption display should be continued. The display position in this case is implementation dependent.

4.3.2 Display area

The display area refers to the horizontal/vertical picture element count specified by the control code SDF and the area specified by coordinates from the upper-left corner of the caption plane specified by the control code SDF. (Refer to Figure 4-1) The starting point (0,0) of the coordinates of display area is the upper left of the “caption plane” regardless of the direction (vertical or horizontal) of writing.

For caption and superimpose, the display area that can be set simultaneously is 1. Also, the display area is valid for bit map data. The priority order of the display area is,

(1) Values specified by SDF and SDP in the texts of caption statement data

(2) Values specified by SDF and SDP in the texts of updated caption management data

(3) Initial value based on the display format specified by the updated caption management data header. The initial value of the display area (display composition dot) is as shown in Table 4-13. Also, the initial values of character size controls, intervals between characters and lines are specified in Table 4-15.

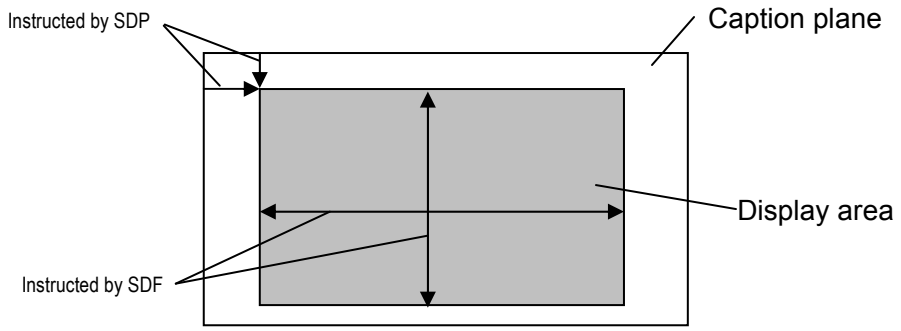


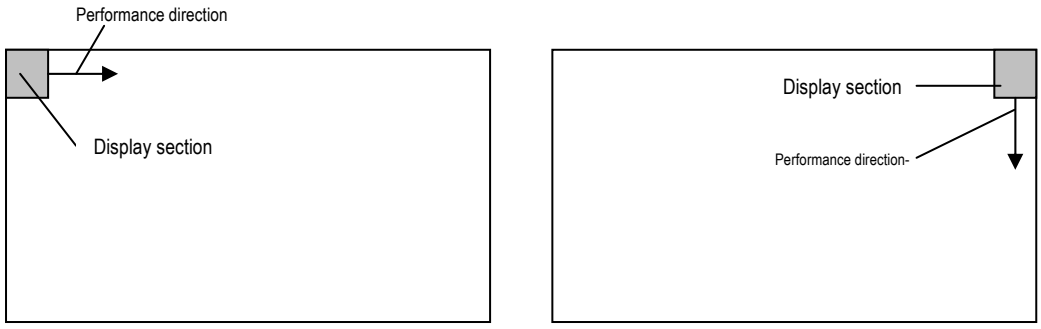
Figure 4-1 Caption plane and display area

Table 4-13 Initial values of the display area and display location

Display format	Display area	Display location
960X540	960X540	0, 0
720X480	720X480	0, 0

4.3.3 Initial operation location

The initial operation location is the first location of first line by the size of characters in the initial condition. The first location of the first line is the upper left corner of the display area in case of horizontal writing and the upper right corner of the display area in case of vertical writing. Also, the performance direction is towards the right for horizontal writing and downwards for vertical writing.



(a) In case of horizontal writing (b)In case of vertical writing

Figure 4-2 Initial operation locations and performance directions

4.4 Characters used in caption/superimpose

4.4.1 Character codes

The character encoding method used for caption/superimpose is 8-bit character codes. Refer to Chapter 3 mono-media for operating the code set.

4.4.2 Character fonts

Round gothic is preferred as the character font used for caption and superimpose.

4.4.3 Character size controls

There are 5 sizes of characters that can be displayed as caption/superimpose, which are 16 dots, 20 dots, 24 dots, 30 dots and 36 dots. For the specification of character size controls at the time of sending, the above mentioned sizes should be specified.

Also, standard, medium and small sizes can be used for each size. The definitions of standard, medium and small are as follows.

Normal size: Specified by the control code SSM.

Medium size: Only the size in lateral direction from standard is half the size.

Small size: Each size in the lateral direction and line direction from standard is half the size.

Restrictions related to character display are stipulated in Table 4-14

Table 4-14 Area of coding group that can be used for specification of display format and specification of character size controls

Characters, etc	2 byte code Row (Cell)	Horizontal/vertical writing		
		Standard Doubled vertical Doubled Horizontal Doubled Vertical/ horizontal	Medium	Small
Symbol	1, 2	O	O*2,*4	O
Alphanumeric	3*1	O	O*4	O
Hiragana	4*1	O	O*4	O
Katakana	5*1	O	O*4	O
Greek letters	6	O	O*4	
Russian letters	7	O		
Rule lines	8	O*3,*4		O*4
Kanji	16~84	O		
Additional symbol	90 (1~ 6) (8~11) (16~17) (20~40) (64~65)	O O O O O		
	91 (1~49)	O		
	92 (1~ 4) (5~12) (13~15) (16~25) (26~31) (32~41) (42~47) (48~52) (53~54) (55~91)	O O O O O O O O O	O*4 O*4 O*4 O*4 O*4 O*4	 O O O O
	93 (1~45) (48~91)	O O		
	94 (1~93)	O		
DRCS	Normal size picture element component Medium size picture element component Small size picture element component	O	O	O
Special codes (space/delete)		O	O	O

*1 Symbols have been added for 1 byte code sets.

*2 Excluding Row 2, Cell 94

*3 Standard only

*4 Not operated in vertical writing.

(1) If the display format is in vertical writing, the receiver unit will display the following characters in a different shape than in horizontal writing.

Row 1 : Cell 2, 3, 17, 18, 28~30, 33~37, 42~59, 65

Row 4 : Cell 1, 3, 5, 7, 9, 35, 67, 69, 71, 78

Row 5 : Cell 1, 3, 5, 7, 9, 35, 67, 69, 71, 78, 85, 86

Row 92 : Cell 48, 49, 50, 51

And, the *katakana* set and *hiragana* set correspond to the following codes.

katakana : 2/1, 2/3, 2/5, 2/7, 2/9, 4/3, 6/3, 6/5, 6/7, 6/14, 7/5, 7/6, 7/9~7/13

hiragana : 2/1, 2/3, 2/5, 2/7, 2/9, 4/3, 6/3, 6/5, 6/7, 6/14, 7/9~7/13

(2) Characters in Row 1, Cell 13-18 and Row 2 Cell 94 are non-spacing characters (refer to 4.4.5)

4.4.4 Display zone

Definitions of the display zone are as follows.

Size of lateral direction of the display zone=horizontal spacing/2+font size+ horizontal spacing/2

Size of line direction of the display zone=vertical spacing/2+font size+vertical spacing/2

* In case the horizontal(vertical) spacing is an odd number, then the value of the interval between characters on the front side of font/2 (interval between lines/2) shall be rounded down and the value of the interval between characters on the backside of font/2 (interval between lines/2) shall be rounded up. The relationship between the display section, font size and interval between characters and lines in case of horizontal writing is indicated from Figure 4-3 to Figure 4-8, in case of vertical writing is indicated from Figure 4-9 to Figure 4-14.
In case those intervals between characters are not multiples of 4, then the values of the interval between characters/4 (interval between lines/4) are rounded down.

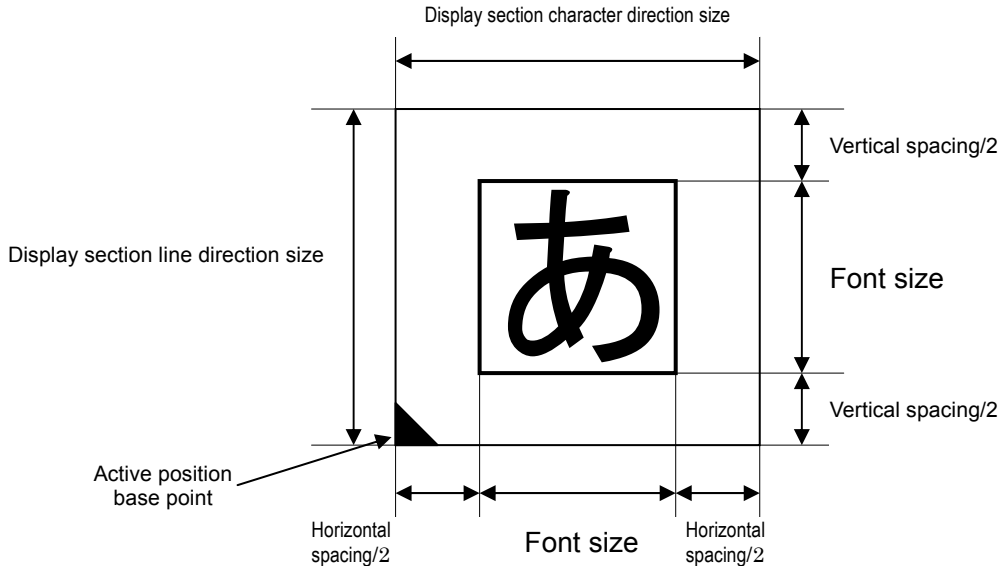


Figure 4-3　Display section and font size (standard) in horizontal writing and the relationship of the horizontal(vertical) spacing.

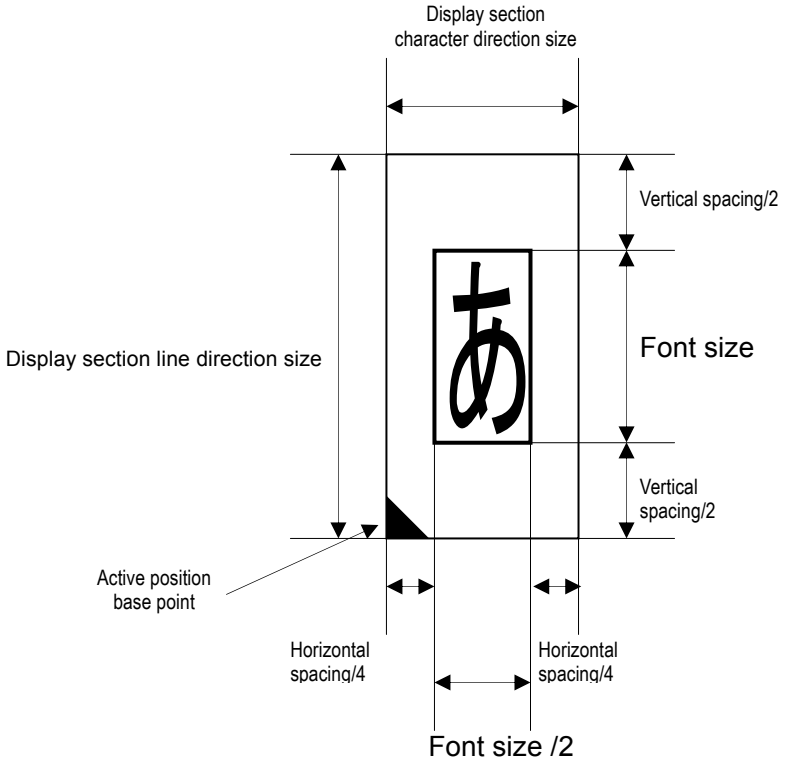


Figure 4-4　Display section and font size (medium size) in horizontal writing and the relationship of the horizontal(vertical) spacing.

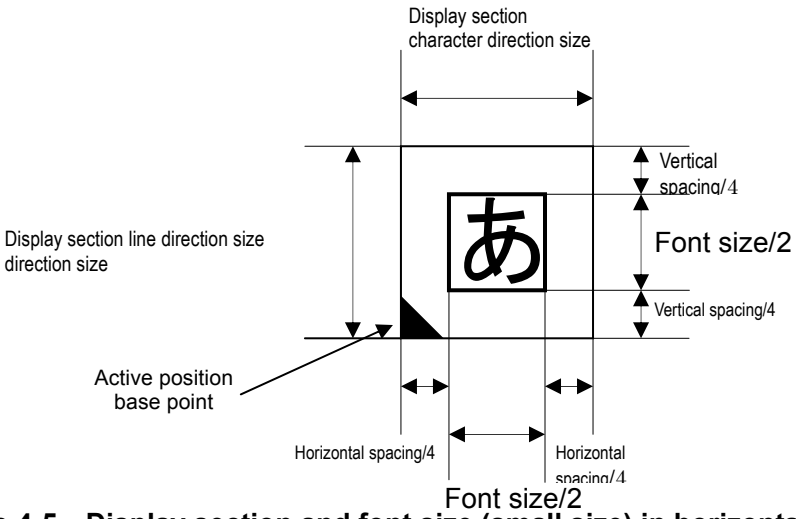


Figure 4-5 Display section and font size (small size) in horizontal writing and the relationship of the horizontal(vertical) spacing.

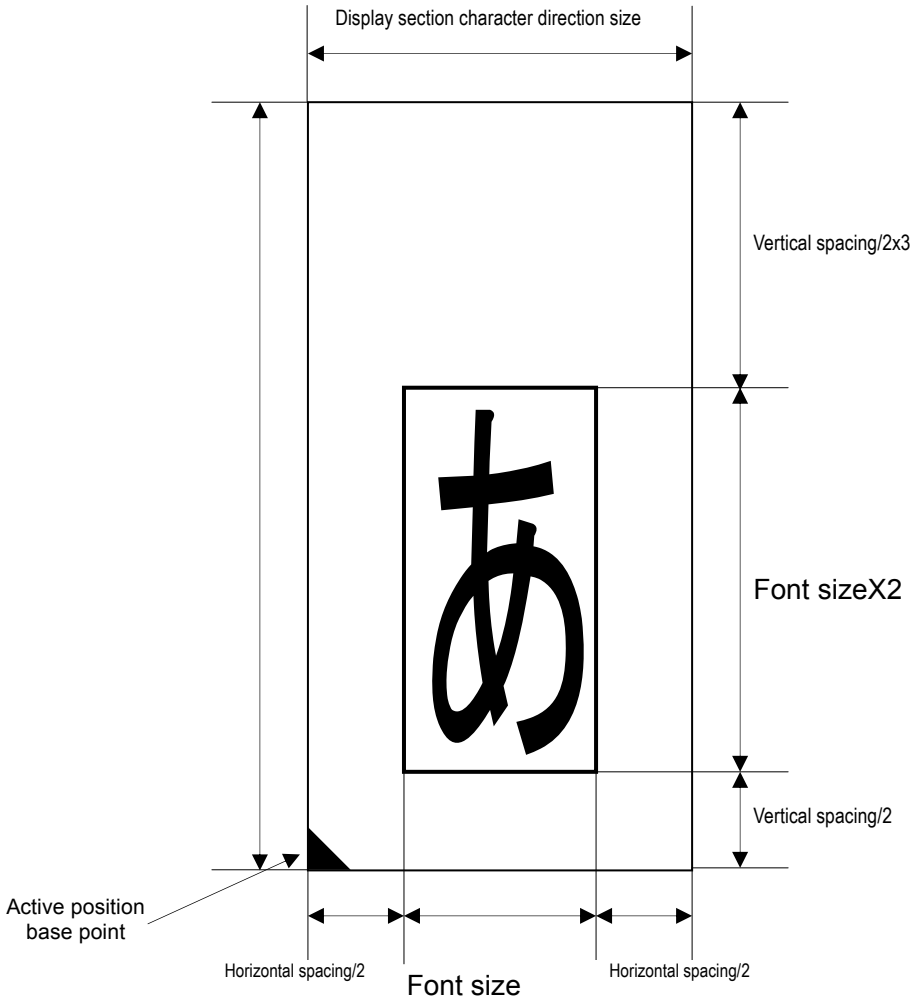


Figure 4-6 Display section and font size (doubled vertical) in horizontal writing and the relationship of the horizontal(vertical) spacing

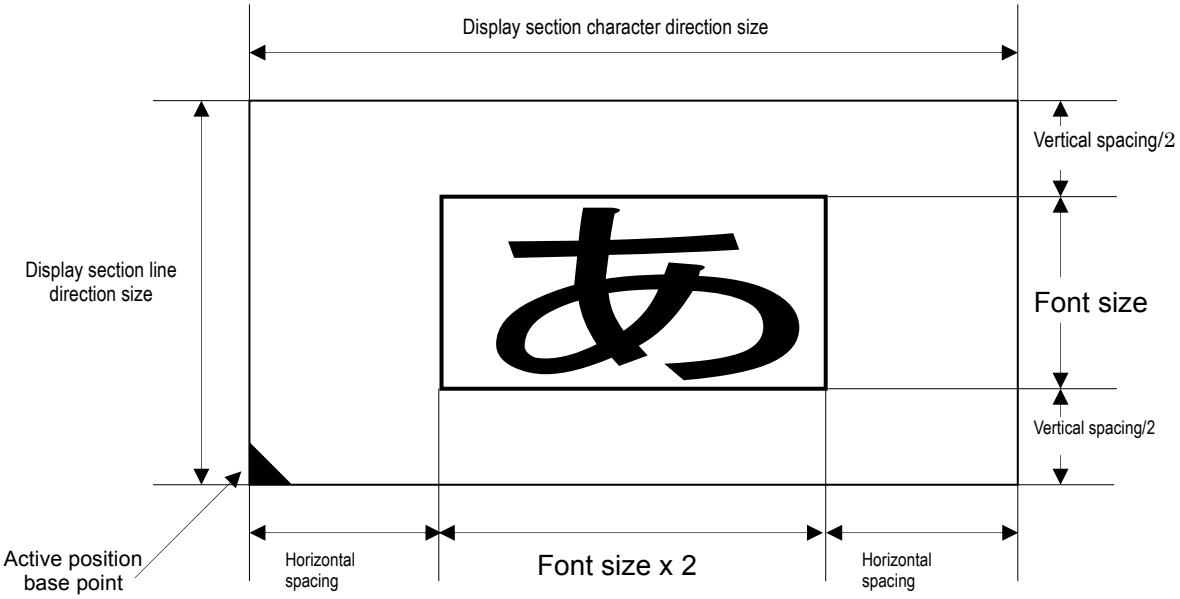


Figure 4-7 Display section and font size (doubled horizontal) in horizontal writing and the relationship of the horizontal(vertical) spacing.

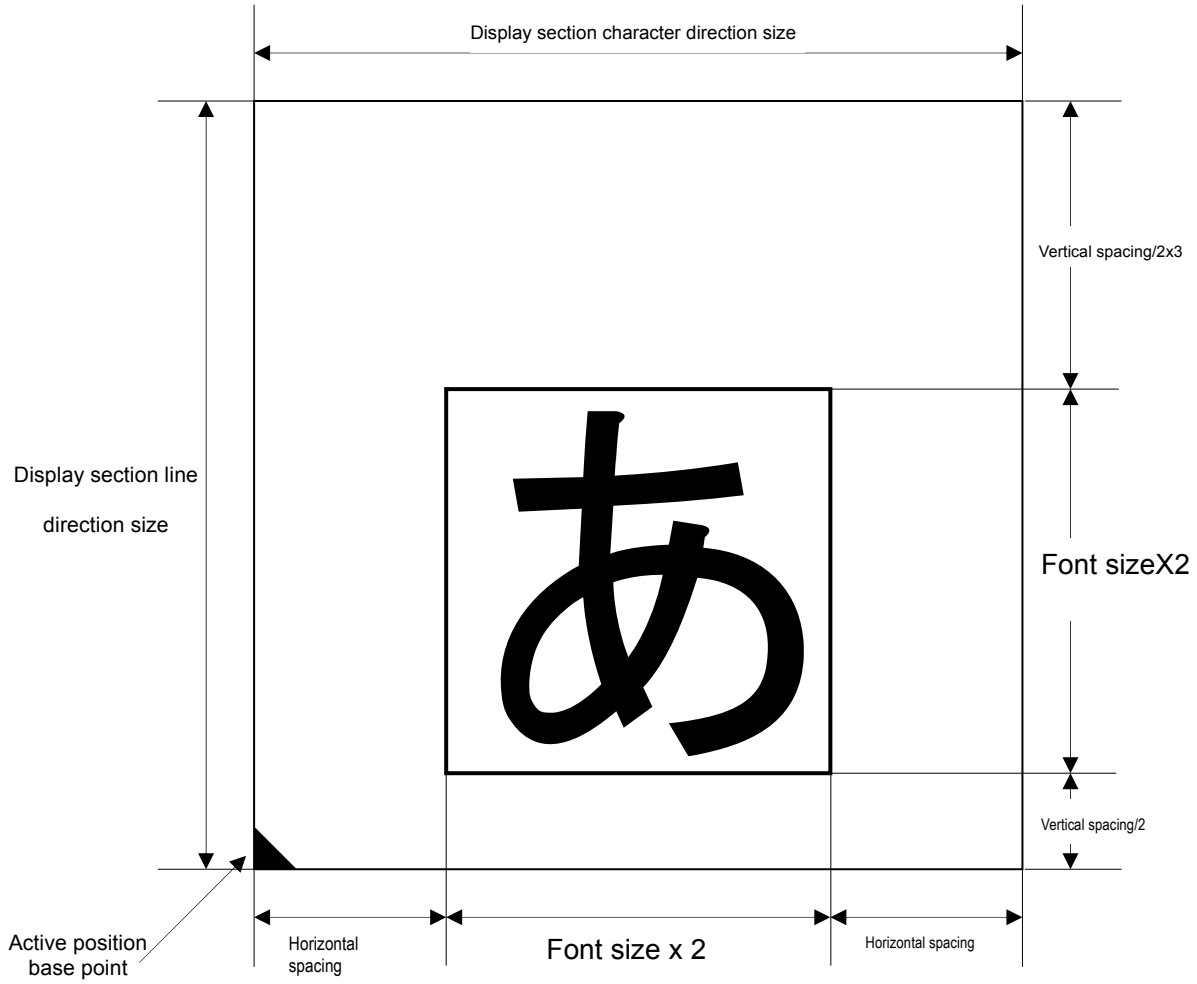


Figure 4-8　Display section and font size (doubled horizontal and doubled vertical) in horizontal writing and the relationship of the horizontal(vertical) spacing.

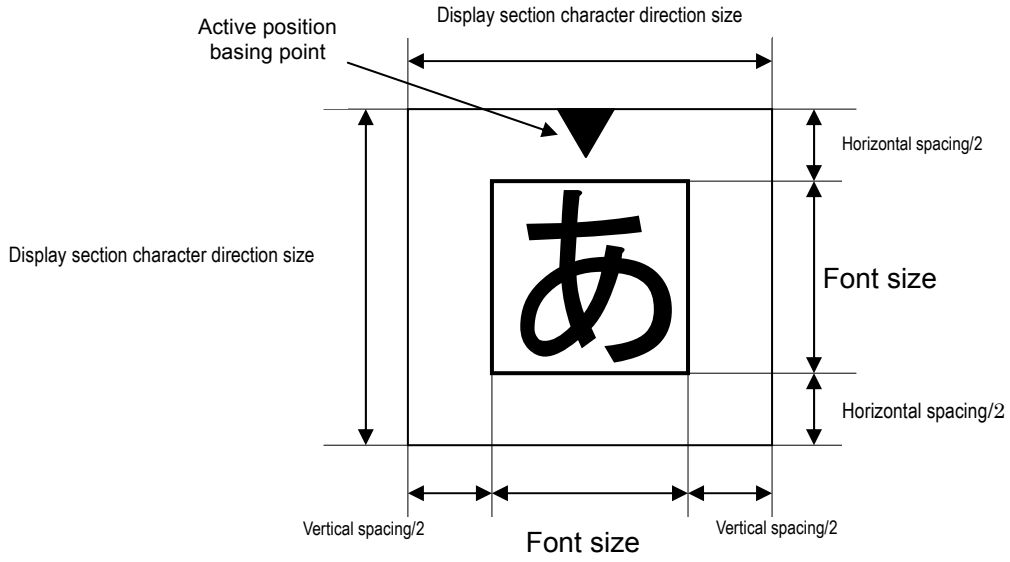


Figure 4-9　Display section and font size (standard) in vertical writing and the relationship of the vertical(horizontal) spacing.

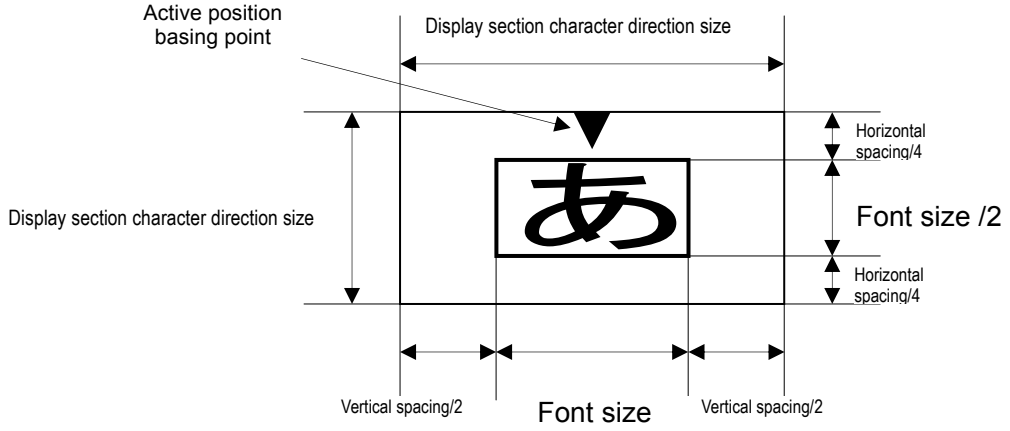


Figure 4-10　Display section and font size (medium size) in vertical writing and the relationship of the vertical(horizontal) spacing.

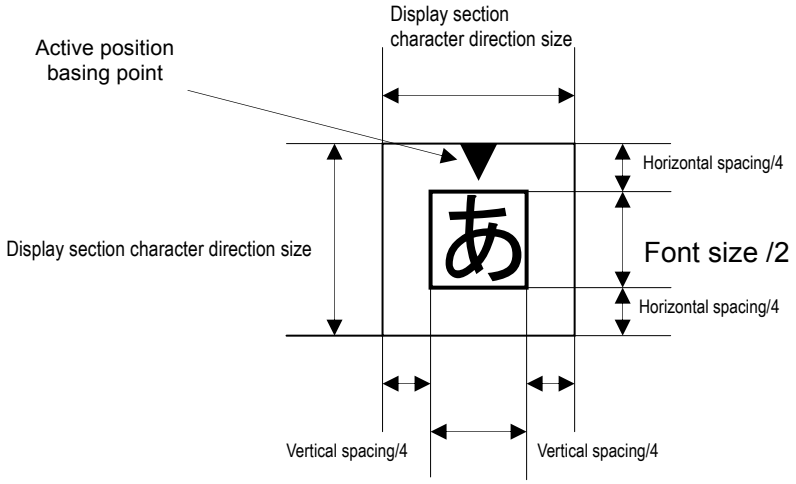


Figure 4-11　Display section and font size (small size) in vertical writing and the relationship of the vertical(horizontal) spacing.

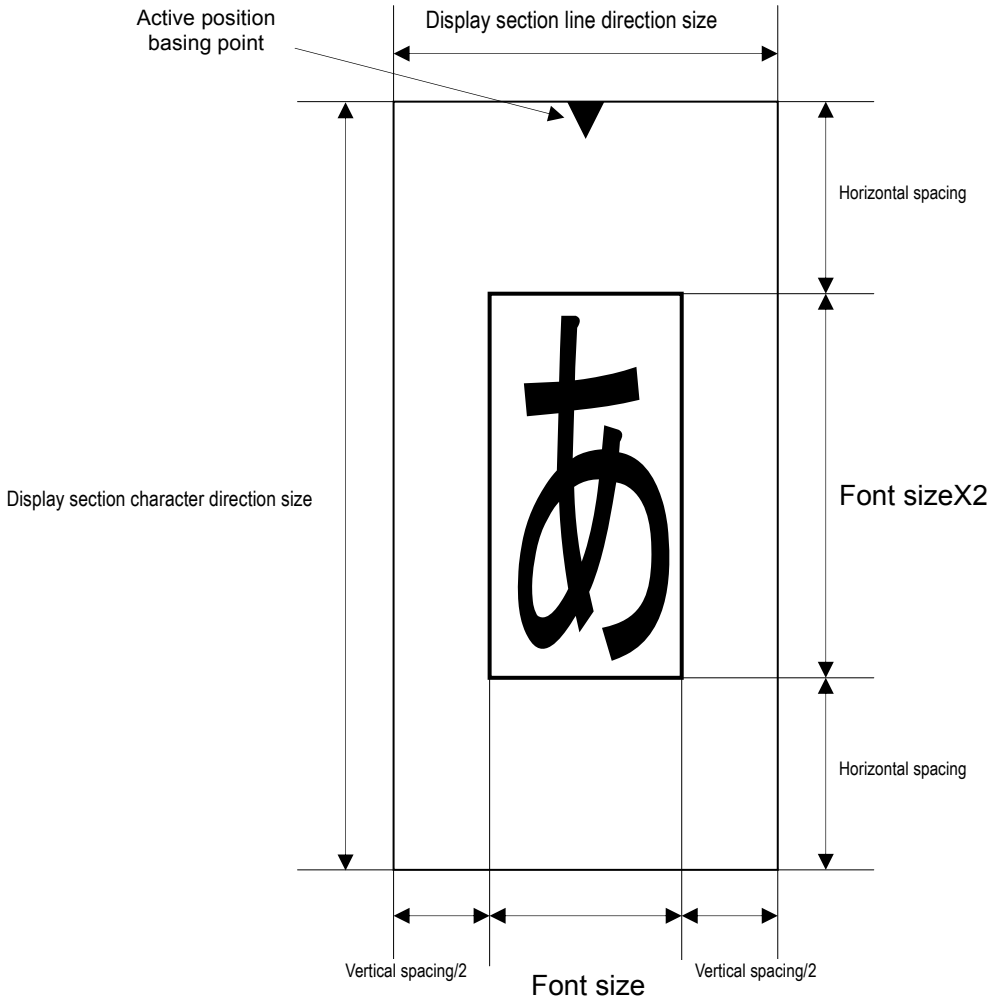


Figure 4-12　Display section and font size (doubled vertical) in vertical writing and the relationship of the vertical(horizontal) spacing.

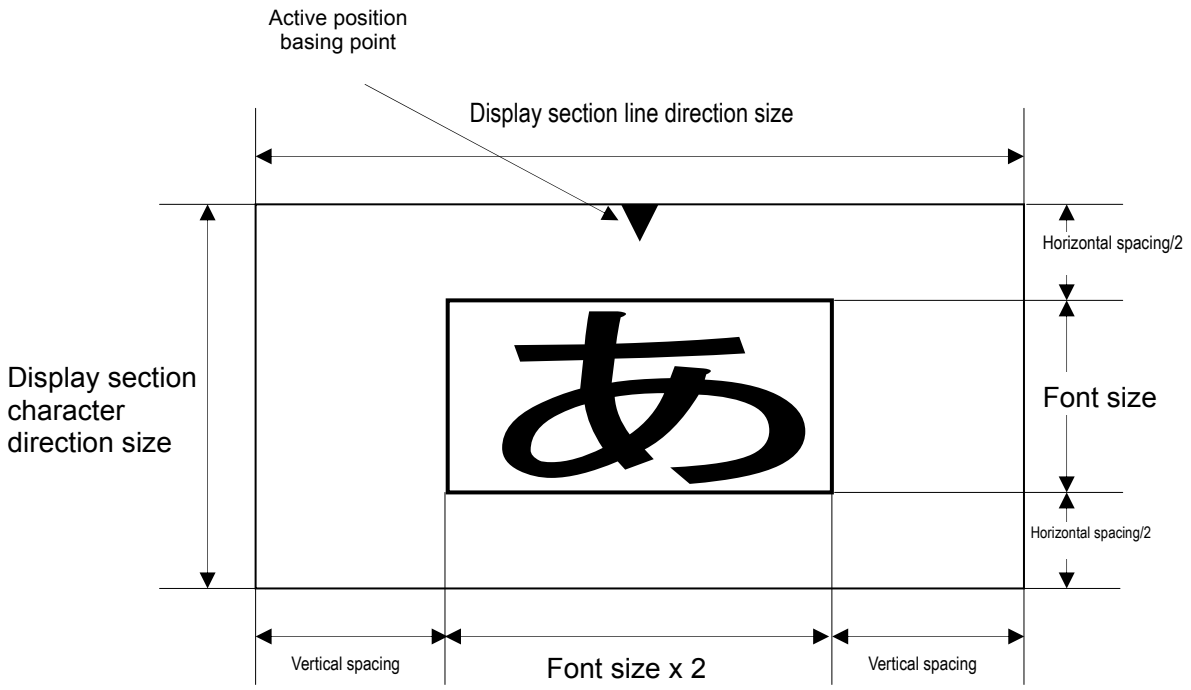


Figure 4-13 Display section and font size (doubled horizontal) in vertical writing and the relationship of the vertical(horizontal) spacing.

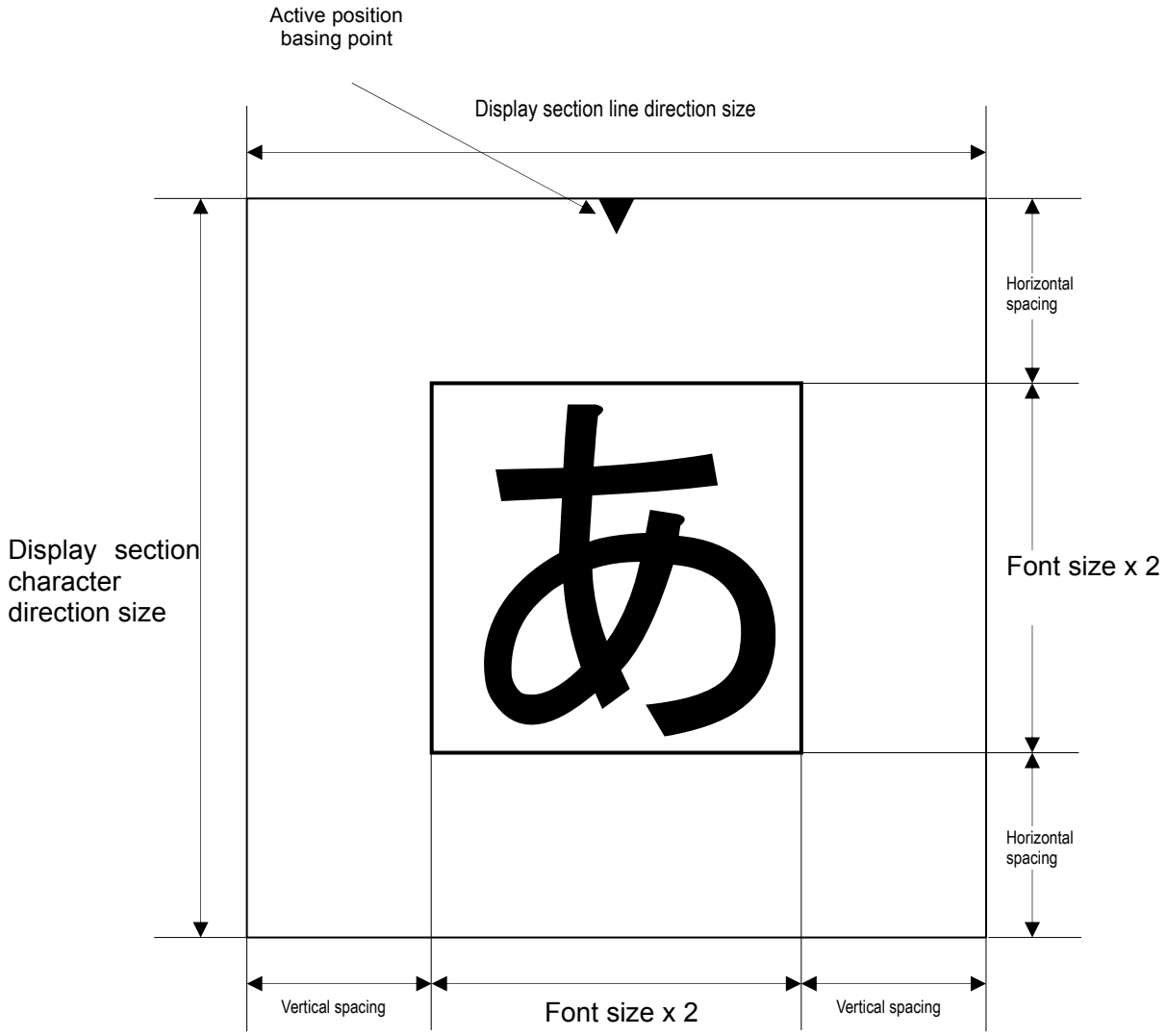


Figure 4-14 Display section and font size (doubled vertical and doubled horizontal) in vertical writing and the relationship of the vertical(horizontal) spacing.

The initial values of character size controls, horizontal and vertical spacing are indicated in Table 4-15.

Table 4-15 Initial values of character size controls, horizontal and vertical spacing

	960X540		720X480	
	Horizontal writing	Vertical writing	Horizontal writing	Vertical writing
Character size controls	36	36	36	36
Horizontal spacing	4	12	4	8
Vertical spacing	24	24	16	24
Display zone(W x H)	40x60	60x48	40x52	60x44

However, ruled line characters(Row 8)are displayed in all areas of the display zone only when the horizontal spacing and the vertical spacing are set in accordance with the horizontal writing of Table 4-16. (The function to display ruled lines in all areas of the display zone when the values other than the ones in Table 4-16 are set for the horizontal spacing and the vertical spacing is optional.)

Table 4-16 Display zone that can display ruled line characters[*1]/box /underline properly

	Character size controls	Horizontal spacing	Vertical spacing
960X540 horizontal writing	16	0, 1, 2	0, 5, 11
	20	0, 1, 2	0, 7, 13
	24	0, 1, 3	0, 8, 16
	30	0, 2, 3	0, 10, 20
	36	0, 2, 4	0, 12, 24
960X540 vertical writing	16	0, 3, 5	0, 5, 11
	20	0, 3, 7	0, 7, 13
	24	0, 4, 8	0, 8, 16
	30	0, 5, 10	0, 10, 20
	36	0, 6, 12	0, 12, 24
720X480 horizontal writing	16	0, 1, 2	0, 4, 7
	20	0, 1, 2	0, 4, 9
	24	0, 1, 3	0, 5, 11
	30	0, 2, 3	0, 7, 13
	36	0, 2, 4	0, 8, 16
720X480 vertical writing	16	0, 2, 4	0, 5, 11
	20	0, 2, 4	0, 7, 13
	24	0, 3, 5	0, 8, 16
	30	0, 3, 7	0, 10, 20
	36	0, 4, 8	0, 12, 24

*1 However, ruled line characters are applied only for horizontal writing.
(Reference) The above horizontal spacing value and vertical spacing value are calculated from the following calculation formulas.

Horizontal spacing ratio=initial horizontal spacing value/initial font size
Horizontal spacing value 1=Horizontal spacing ratioXfont size
Horizontal spacing value 2=(Horizontal spacing/2)Xfont size
Vertical spacing ratio =initial vertical spacing value/initial font size
Vertical spacing value 1=Vertical spacing ratio X font size
Vertical spacing value 2=(Vertical spacing ratio /2)Xfont size

The active position base point of the display zone is in the lower left corner of the display zone for horizontal writing and upper middle of the display zone for vertical writing. However, when the display zone size for the line direction in case of vertical writing is an even number, then the active position base point for the line direction is moved by 1 dot towards the line performance direction from the middle value. (For example, when the line direction display zone size = 68 dot then the line direction active position base point is the 35th dot.)

Regarding the active position, it is as follows;

(1) Active position is not moved by changing the size of the display zone.

(2) After executing display, the active position is moved forward automatically. However, non-spacing characters are not limited to this.

(3) When the display zone goes beyond the edges of the display area, display characters after changing lines. In such case, the active position is moved forward by the number of active position lines only by the display zone line direction side for the display zone of characters at the end of the line before changing the line.

4.4.5 Non-spacing characters

Non-spacing characters are displayed after being combined with characters or the space specified by succeeding codes. The characters and codes that can be used between characters and codes that are combined with non-spacing characters are as follows.

Blank space	: NULL
Expansion control	: Control codes of instruction or calling
Special function	: SP and DEL(Usable as the terminal-end)
Character codes group	: Spacing characters and external characters(Usable as the terminal-end), non-spacing characters

4.5 Control codes used in caption/superimpose

4.5.1 Control codes

Control codes used in caption are in compliance with ARIB STD-B24 Vol. 1 Part 2, 7.1.2.

However, operating regulations for Table 4-17, Table 4-18, Table 4-20 are to be stipulated.

Table 4-17 C0 control set

C0 control codes	Control function	Possible to use or not	Restricted matters and supplements
NUL	Blank	O	
BEL	Bell	X	
APB	Active position backward	O	
APF	Active position forward	O	
APD	Active position down	O	
APU	Active position up	O	
APR	Active position return	O	
PAPF	Parameterized active position forward	O	
APS	Active position set	O	
CS	Clear screen	O	
CAN	Cancel	X	
ESC	Escape	O	
LS1	Locking shift 1	O	
LS0	Locking shift 0	O	
SS2	Single shift 2	O	
SS3	Single shift 3	O	
RS	Record separator	X	
US	Unit separator	O	Used only to identify data units and is not used in 8-bit character codes character strings.

O: possible to use Δ: possible to use with restrictions X: impossible to use

Table 4-18 C1 Control set

C1 control codes	Control function	Possible to use or not	Restricted matters and supplements
BKF (CFLA0: TBD)	Foreground color Color map Lower address specifications	O	CMLA of foreground color is specified as 0.
RDF (CFLA1: TBD)	Foreground color Color map Lower address specifications	O	CMLA of foreground color is specified as 1.
GRF (CFLA2: TBD)	Foreground color Color map Lower address specifications	O	CMLA of foreground color is specified as 2.
YLF (CFLA3: TBD)	Foreground color Color map Lower address specifications	O	CMLA of foreground color is specified as 3.
BLF (CFLA4: TBD)	Foreground color Color map Lower address specifications	O	CMLA of foreground color is specified as 4.
MGF (CFLA5: TBD)	Foreground color Color map Lower address specifications	O	CMLA of foreground color is specified as 5.
CNF (CFLA6: TBD)	Foreground color Color map Lower address specifications	O	CMLA of foreground color is specified as 6.
WHF (CFLA7: TBD)	Foreground color Color map Lower address specifications	O	CMLA of foreground color is specified as 7.
COL	Color Controls	Δ	Color control COL P1(1byte) COL 04/8: CMLA of foreground color is specified as 8. COL 04/9: CMLA of foreground color is specified as 9. COL 04/10: CMLA of foreground color is specified as 10. COL 04/11: CMLA of foreground color is specified as 11. COL 04/12: CMLA of foreground color is specified as 12. COL 04/13: CMLA of foreground color is specified as 13. COL 04/14: CMLA of foreground color is specified as 14. COL 04/15: CMLA of foreground color is specified as 15.

C1 control codes	Control function	Possible to use or not	Restricted matters and supplements
			COL 05/0: CMLA of background color is specified as 0. COL 05/1: CMLA of background color is specified as 1. COL 05/2: CMLA of background color is specified as 2. COL 05/3: CMLA of background color is specified as 3. COL 05/4: CMLA of background color is specified as 4. COL 05/5: CMLA of background color is specified as 5. COL 05/6: CMLA of background color is specified as 6. COL 05/7: CMLA of background color is specified as 7. COL 05/8: CMLA of background color is specified as 8. COL 05/9: CMLA of background color is specified as 9. COL 05/10: CMLA of background color is specified as 10. COL 05/11: CMLA of background color is specified as 11. COL 05/12: CMLA of background color is specified as 12. COL 05/13: CMLA of background color is specified as 13. COL 05/14: CMLA of background color is specified as 14. COL 05/15: CMLA of background color is specified as 15. COL 06/0: CMLA of half foreground color is specified as 0. COL 06/1: CMLA of half foreground color is specified as 1. COL 06/2: CMLA of half foreground color is specified as 2. COL 06/3: CMLA of half foreground color is specified as 3. COL 06/4: CMLA of half foreground color is specified as 4. COL 06/5: CMLA of half foreground color is specified as 5. COL 06/6: CMLA of half foreground color is specified as 6. COL 06/7: CMLA of half foreground color is specified as 7. COL 06/8: CMLA of half foreground color is specified as 8. COL 06/9: CMLA of half foreground color is specified as 9.

C1 control codes	Control function	Possible to use or not	Restricted matters and supplements
			COL 06/10: CMLA of half foreground color is specified as 10. COL 06/11: CMLA of half foreground color is specified as 11. COL 06/12: CMLA of half foreground color is specified as 12. COL 06/13: CMLA of half foreground color is specified as 13. COL 06/14: CMLA of half foreground color is specified as 14. COL 06/15: CMLA of half foreground color is specified as 15. COL 07/0: CMLA of background neutral color is specified as 0. COL 07/1: CMLA of background neutral color is specified as 1. COL 07/2: CMLA of background neutral color is specified as 2. COL 07/3: CMLA of background neutral color is specified as 3. COL 07/4: CMLA of background neutral color is specified as 4. COL 07/5: CMLA of background neutral color is specified as 5. COL 07/6: CMLA of background neutral color is specified as 6. COL 07/7: CMLA of background neutral color is specified as 7. COL 07/8: CMLA of background neutral color is specified as 8. COL 07/9: CMLA of background neutral color is specified as 9. COL 07/10: CMLA of background neutral color is specified as 10. COL 07/11: CMLA of background neutral color is specified as 11. COL 07/12: CMLA of background neutral color is specified as 12. COL 07/13: CMLA of background neutral color is specified as 13. COL 07/14: CMLA of background neutral color is specified as 14. COL 07/15: CMLA of background neutral color is specified as 15. palette specification COL P1(1byte) P2(1byte) COL 02/0 04/0: palette number 0 is specified. COL 02/0 04/1: palette number 1 is specified. COL 02/0 04/2: palette number 2 is specified. COL 02/0 04/3: palette number 3 is specified. COL 02/0 04/4: palette number 4 is specified. COL 02/0 04/5: palette number 5 is specified. COL 02/0 04/6: palette number 6 is specified.

C1 control codes	Control function	Possible to use or not	Restricted matters and supplements
			COL 02/0 04/7: palette number 7 is specified.
POL	Pattern polarity Controls	Δ	Only POL 04/0(normal polarity) and POL 04/1(inverse polarity1) are used. The process in case the inverse polarity is set as normal polarity is; background color → foreground color, half foreground color → half background color, half background color → half foreground color, foreground color → background color.
SSZ	Small Size	O	Not specified for DRCS of standard, middle size.
MSZ	Middle Size	O	Not specified for DRCS of standard, small size.
NSZ	Normal Size	O	Not specified for DRCS of middle, small size.
SZX	Character Size Controls	Δ	Doubled vertical, doubled horizontal, doubled vertical and horizontal can be used.
FLC	Flashing control	Δ	Cycling period of flashing is 1 second, and the standard time ratio of On and Off is 1:1. Refer to 4.5.2 for operation of flashing.
CDC	Conceal Display Controls	X	
WMM	Writing Mode Modification	X	
TIME	Time Controls	Δ	Can be used only for process waiting
MACRO	Macro Command	X	Macro definition of control codes is not used, and only the default macro can be used.
RPC	Repeat Character	Δ	In the scroll character string, parameter P1 with "0" is not operated.
STL	Start Lining and mosaic separation	Δ	Only underline is operated. Refer to 4.5.5 for operation of underline.
SPL	Stop lining and mosaic separation.	O	Only underline is operated.
HLC	Highlighting Character Block	Δ	Refer to 4.5.4 for operation of highlighting.
CSI	Control Sequence Introducer	O	

O: possible to use Δ: possible to use with restrictions X: impossible to use

4.5.1.1 Operation of color specifications

Only 128 colors for receiver unit common fixed colors in Appendix-1 can be used for the color specification.

The specification of colors is specified using the following codes to indicate palette number, CMLA value in the palette and to which one of foreground, half foreground, half background that the colors are applied to.

- BKF-WHF is a name assuming the colors under palette0, therefore, it is referred to as CFLA0-CFLA7, which is another name for it in this specification in this volume.

- Palette specification COL 02/0 04/N (N is palette number :0-7)
  However, after the initialisation of caption, if palette0 continues to be used, then the palette specification can be omitted.
- In case the foreground color specification CMLA is 0-7, then CFLA0-CFLA7
  In case CMLA is 8 – 15, then COL 04/M (M is the CMLA number)
  After the initialisation of caption, if index value =7(0x07) continues to be used then both the palette0 specification and CMLA7 specification can be omitted.
- Background color specification COL 05/M (M is the CMLA number)
  After the initialisation of caption, if index value =8(0x08) continues to be used then, both the palette0 specification and CMLA8 specification can be omitted.
- Half foreground color specification COL 06/M (M is CMLA number)
  After the initialisation of caption, if index value =15(0x0F) continues to be used then, both the palette0 specification and CMLA15 specification can be omitted.
- Half background color specification COL 07/M (M is CMLA number)
  After the initialisation of caption, if index value =30(0x1E) continues to be used then, both the palette0 specification and CMLA14 specification can be omitted.

In case the index value of the 128 common colors indicated in appendix-1 are in hexadecimal numeral, then the upper 4 bits are handled by the palette number and the lower 4 bits are handled by the CMLA.

Example 1) In case of specifying the color of index value 0 to the background color.
Palette number =0 and CMLA=0, therefore COL 02/0 04/0 COL 05/0
After initializing, if anything other than palette0 is not specified, then it is only COL 05/0.

Example 2) In case of specifying the color of index value 0 as the foreground color.
Palette number =0, CMLA=0 , therefore COL 02/0 04/0 CFLA0
After initializing, if anything other than palette0 is not specified, then it is only CFLA0

Example 3) In case of specifying the color of index value 47(0x2F) as the foreground color.
Palette number =2 and CMLA=15, therefore COL 02/0 04/2 COL 04/15

Example 4) In case of specifying the color of index value 47(0x2F) as the half foreground color.
Palette number =2 and CMLA=15, therefore COL 02/0 04/2 COL 06/15

The meaning when the palette number is fixed as 0 and the half foreground colors and half backgrounds color are not specified is shown in Table 4-19 and thereafter.

Table 4-19 The meaning of color specifications when the palette number is 0

C1 control codes	Control function	Possible to use or not	Restricted matters and supplements
BKF (CFLA0: TBD)	Black Foreground	O	Foreground color is specified as black.
RDF (CFLA1: TBD)	Red Foreground	O	Foreground color is specified as red.
GRF (CFLA2: TBD)	Green Foreground	O	Foreground color is specified as green.
YLF (CFLA3: TBD)	Yellow Foreground	O	Foreground color is specified as yellow.
BLF (CFLA4: TBD)	Blue Foreground	O	Foreground color is specified as blue.
MGF (CFLA5: TBD)	Magenta Foreground	O	Foreground color is specified as magenta.
CNF (CFLA6: TBD)	Cyan Foreground	O	Foreground color is specified as cyan.
WHF (CFLA7: TBD)	White Foreground	O	Foreground color is specified as white.
COL	Color Controls	Δ	Color specification COL P1(1byte) COL 04/8: foreground color is specified as transparent. COL 04/9: foreground color is specified as half-brightness red. COL 04/10: foreground color is specified as half-brightness green. COL 04/11: foreground color is specified as half-brightness yellow. COL 04/12: foreground color is specified as half-brightness blue. COL 04/13: foreground color is specified as half-brightness magenta. COL 04/14: foreground color is specified as half-brightness cyan. COL 04/15: foreground color is specified as half-brightness white. COL 05/0: Background color is specified as black. COL 05/1: Background color is specified as red. COL 05/2: Background color is specified as green. COL 05/3: Background color is specified as yellow. COL 05/4: Background color is specified as blue. COL 05/5: Background color is specified as magenta. COL 05/6: Background color is specified as cyan. COL 05/7: Background color is specified as white. COL 05/8: Background color is specified as

			transparent. COL 05/9: Background color is specified as half-brightness red. COL 05/10: Background color is specified as half-brightness green. COL 05/11: Background color is specified as half-brightness yellow. COL 05/12: Background color is specified as half-brightness blue. COL 05/13: Background color is specified as half-brightness magenta. COL 05/14: Background color is specified as half-brightness cyan. COL 05/15: Background color is specified as half-brightness white. *1

*1 reference: In analog caption, there is an exceptional provision for background color, half brightness white to use half transparent half brightness, therefore, COL 05/15 is different from the analog provision. In order to create the same condition as analog COL 05/15, the index value of half transparent color in broadcasting station equipment should be determined and it should be specified in accordance with this provision.

Table 4-20 Expansion control codes(CSI)

Character	Control function	Possible to use or not	Restricted matters and supplements
SWF	Set Writing Format	Δ	After initialization of the display screen, characters that involve display of bit maps and display performance, only before the appearance of the control code can be specified. Only 1 code parameter is used and only parameters that can be specified to P11…P1i is 7 (horizontal writing in 960x540) or 8(vertical writing in 960x540) when pixel size of caption plane is 960x540, 9 (horizontal writing in 720x480) and 10(vertical writing in 720x480) when pixel size of caption plane is 720x480. When this control code is specified, it is prioritized from the format specification of caption management data.
CCC	Composit Character Control	X	
RCS	Raster Color Command	O	After initialization of the display screen, characters that involve the display of bit maps and display performance, only before appearance of the control code can be specified. As for types of colors, from 0 (black) to 15 (half brightness white) defined in ARIB STD-B24 on top of common fixed colors from 16 to 127 can be specified. Refer to 4.5.3 for the operation of raster color control.
ACPS	Active Coordinate Position Set	O	
SDF	Set Display Format	O	After initialization of the display screen, characters that involve the display of bit maps and display performance, only before appearance of the control code can be specified.
SDP	Set Display Position	O	After initialization of the display screen, characters that involve the display of bit maps and display performance, only before appearance of the control code can be specified.
SSM	Character composition dot designation	Δ	Only 16x16, 20x20, 24x24, 30x30, 36x36 dots can be specified.
PLD	Partial Line Down	X	
PLU	Partial Line Up	X	
SHS	Set Horizontal Spacing	O	
SVS	Set Vertical Spacing	O	

Character	Control function	Possible to use or not	Restricted matters and supplements
GSM	Character deformation	X	
GAA	Coloring block	X	Colored block is full display zone only.
SRC	Raster Color Designation	X	
TCC	Switch control	X	
CFS	Character Font Set	X	
ORN	Ornament Control	Δ	No character ornament, only outlines can be used. Refer to 4.5.6 for operation of outlines.
MDF	Font	X	
PRA	Built-in sound replay	O	
XCS	External Character Set alternative code string definition	X	
SCR	Scroll definition	Δ	It can be used only in case of horizontal writing. Regarding one line lateral direction scrolling(with or without rollout), only 1 place 1 line can be used.

O: possible to use Δ: possible to use with restrictions X: impossible to use

4.5.2 Operation of flashing

Flashing of the 8-bit character codes character string does the character flashing(foreground color 0.5 seconds, background color 0.5 seconds)to the foreground color (including half colors in case of 4 grayscale fonts)and background color, and flashing in the bit map data does bit map flashing (flashing color 0.5 seconds, raster color 0.5 seconds) to flashing color and raster color(transparent color if the raster color is not specified)specified by the FLC header defined in the bit map figure encoding of ARIB STD-B24 with a 1 second cycle as the standard. The timing to begin the flashing is when the character specified by the flashing is rendered first or when figure is rendered after the flashing color is specified in the FLC header in the bit map data. When the initialization operation for caption management at the time of update, or instruction to delete the display screen is sent by the CS, the flashing of corresponding characters or picture element ends.

4.5.2.1 Restricted matters

- Characters and bit map of caption and superimpose can be specified as desired within the common 128 colors, however, the color that does the flashing is assumed to be handled

separately from those 128 colors for implementation of the CLUT of receiver units, therefore the number of colors is limited as below.

- Flashing target colors of characters are valid foreground colors, half foregound colors and half background colors between [FLC 04/0] to [FLC 04/15], and there are less than 24 colors for one caption text data. Caption and superimpose can use 24 colors independently.
- Flashing target colors of bit map figures are the colors specified in the flc_header() and this is less than 16 colors for 1 caption statement data.
- Flashing is a positive phase only.
- Mixture with the border specification is prohibited.
- Mixture with scrolling specification is prohibited.

4.5.3 Raster color control

Boxes in the entire display area. The raster color controlled area is the rectangular area instructed by the SDF (Set Display Format) and SDP (Set Display Position). Specifications for the raster color control and index values of CLUT (refer to Appendix-1) are specified in P11 … P1i by RCS (raster color command).

4.5.4 Operation of highlighting

Highlighting perform the control to add frames made of 4 outer sides within a display zone. Only when the display zone is in accordance with Table 4-16 is the correct box rendered across the entire display zone. (Even when the horizontal spacing and the vertical spacing are changed freely, is the function to render boxes across the entire display zone optional). When the box specification is done for lateral directions or line directions, and if the extra interval between characters/lines is less than 1 dot, then rendering a correct box is not guaranteed. Gradation of highlighting is implementation dependent. (Receiver unit implementation to make the box gradation a 4 level gradation is not required.)

4.5.5 Operation of underlines

Underlines are rendered in a 1 dot width on the outer side in the display zone. In case of horizontal writing, it is attached to the side of the next line and in case of vertical writing, it is attached to the side of the previous line. Only when the display zone is in accordance with Table 4-16, is the underline across the entire display zone rendered correctly. (Even when the horizontal spacing and the vertical spacing are changed freely, is the function to render underlines across the entire display zone optional). When a value less than 2 dots are specified between lines, is the rendering of correct underlines not guaranteed. Gradation of underlines including implementation of 1dot width in 720x480 is implementation dependent. (Receiver unit implementation to make underlines a 4 level gradation is not required.)

4.5.6 Operation of outlines

In case of outlining, specify values bigger than 2 dots between characters and lines. When a value less than 2 dots is specified between characters and lines, is the correct outlining display not guaranteed. Since there is no half color specification for the gradation of outlining, it is implementation dependent. (Receiver unit implementation to make bordering a 4 level gradation is not required.)

4.5.7 Operation of scrolling

4.5.7.1 One line lateral direction scroll

It is prohibited to instruct the SCR multiple times in the same text. In case of scrolling, it should be transmitted as a different data unit (this statement) that specified one line worth of display area by SDF.

Receiver unit performance when specifying scrolling is as follows.

- Scrolling is performed in a rectangular area specified by SDF and SDP, and it is not rendered outside the rectangular area.
- Virtual area of 1 character worth on the right side of the first line of the display area is assumed to exist and once the scrolling specification (SCR) is specified, the active position is reset in the virtual writing area.
- Characters that were written in the display area before the scrolling specification are deleted after the scrolling specification.
- Display from the right side of the display from the first character.
- Scrolling is started by writing characters to the virtual writing area.
- If without rollout, after displaying the last character, scrolling is stopped.
- If with rollout, the scrolling is continued until there are no characters on the screen.
- In case data that should be displayed next is received during scrolling, wait until the scrolling is finished.
- In case the values between characters /between lines specified from the beginning of the scrolling instruction to the end of scrolling exceed the maximum values defined in Table 4-16, then the scrolling display is implementation dependent.

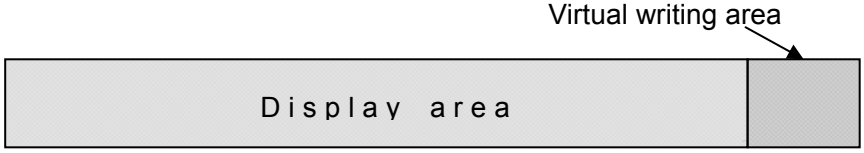


Figure 4-15 Display area of one line lateral direction scrolling and the virtual writing area

4.5.7.2 Control codes

The following control codes are not used from the beginning of instruction of scrolling and the end of scrolling.

Format specification	SWF, SDF, SDP
Size specification	SZX, SSM
Change of active position	APB, APR, APF, PAPF, APU, APD, APS, ACPS, SHS, SVS
Display effect	FLC, STL, SPL, HLC
Time control	TIME
Raster specification	RCS

APB, APR, APF, PAPF, APU, APD, APS, ACPS, FLC, STL, SPL, HLC and RCS specified before the scrolling instruction will be released at the start of the scrolling instruction.

4.5.7.3 Scrolling speed

Receiver units will start scrolling as quickly as possible after the scrolling specification, however, there may be some delay between the scrolling specification and the start of scrolling. As for the specification of scrolling speed, less than 3 characters per 1 second is recommended regarding one line lateral direction scroll. Dot unit movement at one time is implementation dependent.

4.5.8 Priorities of the display function

Display function by control codes is prioritized by the following order.

1) Size of characters
2) Outlining
3) Underline
4) Highlighting
5) Polarity Controls
6) Color specification, flashing

Note)Order for the ones with the same priority is as desired.

4.6 Operation of the DRCS

For DRCS calling, DRCS set calling of 1 byte or 2 bytes is used as defined in ARIB STD-B24 Vol. 1 Part 2 7.1.1.5. Also, for the encoding of DRCS patterns, only pattern transmission is operated, and operation by geometric is not executed. Also, specify 0 for the “fontId”. If anything other than 0 is specified, then the receiver unit is recognized as 0. Buffers secured by receiver units for the DRCS are 16KB in the DRCS for caption and 16KB in the DRCS for superimpose. The maximum number of DRCS used simultaneously for each caption and superimpose respectively is 188. (In case of placing the “patternData” of multiple sizes in 1 Character code, each one is counted as 1). “patternData” should send out 4 level gradation data for the design frame.

Table 4-21 Setup parameters of DRCS figure encoding

Field	Operation
NumberOfCode	Operated as defined.
CharacterCode	Operated as defined.
NumberOfFont	Operated as defined.
FontId	Only 0 can be specified. (Recognized as 0 even if anything other than 0 is specified.)
Mode	Only 0001 can be specified.
Depth	Only 2 can be specified.
Width	Operated as defined.*1
Height	Operated as defined.*1
patternData	Operated as defined.

*1 The size that is defined in 4.4.3.Character size controls should be specified.

4.7 Operation of initialization

Receiver units should perform in accordance with the timing and performance items of the initialization operation defined below and the performance contents should perform in accordance with ARIB STD-B24 Vol. 1 Part 3 Chapter 8 Initializing operation. However, regarding the initialization timing of character size controlss, it should be the same as “status instruction”. Also, when the selection of stations is performed, all initialization performance items relating to caption will be initialized. For the overall timing of the initialization of caption, refer to Appendix-13.

4.7.1 Initialization by caption management

Receiver units conduct the initialization performance for caption management at the time of update defined in ARIB STD-B24, when data groups of received caption management data is switched from group A to group B, or group B to group A.

At this time, the display area and display location are set in the values of Table 4-13. Also, the values for character size controls, horizontal spacing and vertical spacing are set in the values of Table 4-15.

4.7.2 Initialization by caption statements

Receiver units perform the initialization operation defined in ARIB STD-B24 when receiving the same caption statement data as data group and language during the presentation process.

4.7.3 Initialization by data units in this document

If data units of this text are included in the same caption text data as language and data group during the presentation process, the receiver unit performs the initialization operation defined in ARIB STD-B24 immediately before the presentation process of the receiver unit of data units of this text. Also, the default value at the time of initialization for the half foreground color is index value = 15, and for the half background color is index value = 30.

4.7.4 Initialization by character control codes

Receiver units perform the corresponding initialization operation defined in ARIB STD-B24 immediately before receiver unit execution of screen deletion (CS) and format selection (SWF). Also, the default value at the time of initialization for the half foreground color is index value = 15, and for the half background color is index value = 30.

4.8 Mono-media used in caption and superimpose

4.8.1 Operation of geometric

Geometric is not used for caption and superimpose.

4.8.2 Operation of bit map data

Bit map data can be operated in superimpose; however, the number of usable colors is 128 colors of receiver unit common fixed colors.

4.8.3 Operation of alert sounds

Can be operated by both caption and superimpose. However, it is limited to built-in sound.

4.8.4 Operation of additional sound

Neither caption nor superimpose will be operated.

4.9 Expected performance of the receiver units

- The number of caption and superimpose that can be displayed simultaneously is 1 caption and 1 superimpose for a total of 2.
- For receiver units, the presentation control of caption and superimpose should be controlled independently.
- As a basic rule, caption and superimpose are operated so that their display areas do not overlap with one another, however, when they do overlap, superimpose are prioritized and displayed before caption.
- When bit map data and text, and bit map data and another bit map data are overlapping each other, then what is written later has priority.
- The display size and position of caption and superimpose in data broadcasting programs are displayed with full screen area as the standard.
- Receiver units judge the existence of sending caption data by the existence of receiving caption management data. Displaying marks to notify the receiving of caption to viewers, display of caption, deletion are mainly executed using corresponding data as the standard. Considering the pauses for sending corresponding data between commercials, the timeout process should be executed after caption management data has not been received for

more than 3 minutes. Display control that is linked to other data such as EIT data is implementation dependent.

4.9.1 Start/end of caption display

The performance of receiver units at the time of starting/ending of caption is as written in Table 4-22. However, start means the “start of caption display specified by caption text” and ending means “deletion of caption text”.

Table 4-22 Operation of DMF in caption management data

DMF	When starting	When ending
Automatic display at the time of reception	After reception is completed, the corresponding text is displayed immediately. Cannot be deleted until reception is finished.	After reception is completed, the corresponding caption text is deleted immediately.
Automatic non-display at the time of reception	Nothing is done.	Nothing is done.
Select display at the time of reception	After reception is completed, some kind of information to indicate that there are caption and superimpose should be presented. Possible to display, or to delete by selection of viewer.	After reception is completed, if the corresponding caption text is displayed, then delete immediately. If it is not displayed, then nothing is done.

When playing the recordings, it is in compliance with the time of reception.

4.9.2 Start/end of superimpose display

The same as the start/end caption display.

4.9.3 Set-up fields etc. in the receiver units

- Receiver units display the caption and superimpose of the language selected immediately before. For example, when the second language caption is selected during viewing of a program and when another program with caption is started, then the second language is displayed.
- At the time of shipping of the receiver unit, the default set up of receiver unit displays the first language.
- Receiver units that can set up language codes such as Japanese/English display caption and superimpose of the set language code.

- When the language set in the receiver unit or the caption/superimpose of the language code is not being transmitted, then the receiver unit displays the caption/superimpose of the first language.

4.10 Roll-up mode (Optional)

The roll-up mode additionally displays caption data in units of lines, and it is a function to display while performing the roll-up towards the line direction at the time of line break. Implementation of receiver units is optional.

The roll-up mode is classified as a different display mode than normal caption in caption management data. Superimpose do not have the roll-up mode. Operation restrictions on the roll mode for normal caption are defined as below.

4.10.1 Declaration of the roll-up mode

If the “rollup_mode” field of caption management data is ‘01’ then the language is recognized as roll-up mode. Other than that, even codes that are yet to be defined are recognized as non-roll up.

4.10.2 Operation of caption management data in the roll-up mode

- Format field should be in horizontal writing.
- Only one text data unit can be placed within the same caption management data. DRCS is not operated.
- Data that can be scripted in text is limited to the control codes of SWF,SDF,SDP,SSM,SHS,SVS. In such cases, the dot number of the vertical direction of SDF (display component dot specification) is set as(font size + vertical spacing) times integer with considerations to smooth scrolling. Other control codes are the same as normal caption.

4.10.3 Operation of caption statement data in the roll-up mode

- Only one text data unit can be placed within the same caption management data. DRCS is not operated.

4.10.3.1 Operation of the text data unit of caption statement data

- 1 text data displays 1 line, but line break codes (APD, APR) are not placed.
- Considering the compatibility with the normal mode, CS should be placed at the start of the data unit.
- Only standard character and middle size characters of Table 4-14 can be placed for characters. DRCS is not operated.
- C0control codes: Only CS can be placed.
- C1control codes: Only CFLA0-CFLA7, COLORN, NSZ, MSZ can be placed.

4.10.4 Initialization of the roll-up mode

- Initialization of caption management data is the same as normal caption.
- Receiver units do not execute clear screen or initialization of active positions, character size controls, horizontal spacing or vertical spacing when receiving text for the same caption text as the language and data group during the presentation process, and only the instruction and calling of codes, color specifications, outlining, size of characters are initialized.
- Initialization by CS is not executed.

4.10.5 Presentation of the roll-up mode

- Receiver units perform line break execution automatically before displaying and after data unit for new caption text are received. However, when receiving for the first time after roll up declaration, line break should not be performed.
- Screen deletion by CS is not performed. If the text display goes beyond the right side of the display zone set by SDF, SDP, then characters are displayed after the execution of line break.
- Along with line break, if it goes beyond the display area in the vertical direction, then roll-up of the entire display area is by one line, and roll-up execution time is generally one second and it should be done in smooth manner.
- When displayed characters are in more than 2 lines in the display area, roll-up execution should be done for each line break.
- When the PTS time of PES packet of *n*+1 is exceeded at the time of completing the *n*th PES packet presentation by receiving continuous superimpose, the receiver units will start presentation execution (roll-up) of n+1 immediately.

4.10.6 Restrictions of PES packet transmission of roll-up mode

Changes in transmission methods of 4.2.2 are defined as below.

- In roll-up mode, the PTS standard is not the starting point of packet presentation, but it is at the time of the completion point of packet presentation, which is when the roll-up is started, and the Td is generally 0 seconds. At the time of completion of the nth PES packet presentation, the total information Vol. of the PES packets, which started sending after *n*th should not exceed the receiving buffer Vol. (64KB).
- For the interval of PES packet sending of the caption statement data group, the interval between the PTW time of the nth PES packet of sending sequence and the PTS time of the n+1-th PES packet should be bigger than the display lines of nth data x 1 second.

4.10.7 Compatibility with receiver units not equipped the roll-up mode

Implementation of the roll-up mode is optional, however receiver units without implementation should display data units of caption text as the normal caption. Because of this, each caption text of the roll-up mode has placed CS in the beginning and this CS is ignored in the roll-up mode.

4.11 Caption outscreen display function (Optional)

The function to display video in a reduced size for the purpose of viewing character telop in television broadcasting screens without overlapping is called the caption outscreen display function. Broadcasting stations will not transmit the control signal to announce that caption are assuming this kind of display. Therefore, receiver units will not automatically recognize the caption broadcasting assuming outscreen display. Receiver units should be in the following display setup by the pressing of one remote control key by the viewer.

In the outscreen mode, video is scaled and positioned in the value preset in the receiver unit (Table 4-23). The video location position can be selected from 2 locations, which are the top of screen or the bottom of the screen. The selection method is implementation dependent.

Table 4-23 Display method of video in the outscreen display function

Display format of caption	Video plane pixel size	Display method
960X540	1920X1080	Scale to 96/128 and display in (120X2, 0X2)-(840X2, 405X2) or (120X2, 135X2)-(840X2, 540X2)
720X480 (16:9)	720X480 (16:9)	Scale to 96/128 and display in (90,0)-(630,360) or (90,120)-(630,480)
720X480 (4:3)	720X480 (4:3)	Scale to 96/128 and display in (90,0)-(630,360) or (90,120)-(630,480)

- Any performances other than assumed above will be implementation dependent.

5 Operation of multimedia encoding

5.1 Introduction

Regarding the operation of multimedia encoding, as a basic rule, it should be in accordance with

・” Appendix 1　Operational guidelines”

・” Appendix 2　Operational guidelines for implementing basic services”

in ARIB STD-B24 Vol. 2 “XML-based multimedia coding scheme”

In this chapter, the items not specified in the above documents will be specified.

5.2 Operation of NVRAM used commonly in Multimedia services in Digital Terrestrial Television Broadcasting

The A-profile memory area for the all broadcasters and A-profile memory area for the affiliation and A-profile memory area for the specified broadcaster and A-profile memory area for the communication of the specified broadcaster and the memory area for bookmark service shown in Table 5-1 are established as allocation of NVRAM used for saving non-volatile information in Digital Terrestrial Television Broadcasting. Contents of data stored in the A-profile memory area for the all broadcasters, the A-profile memory area for the affiliation, the A-profile memory area of communication purpose for the specified broadcaster, the memory area for bookmark service are dependent on contents specified by all broadcasters or broadcaster independently of Digital Terrestrial Television Broadcasting.

Table 5-1 NVRAM used in Digital Terrestrial Television Broadcasting

Type	Meaning	NVRAM amount
A-profile memory area for the all broadcasters	Common area available for use by all digital terrestrial broadcasters.	2KB (Fixed length block of 64bytes * 32)
A-profile memory area for the affiliation	Common area available for use by broadcasters that belong to the same affiliation.	･4KB for 1 affiliation (Fixed length block of 64bytes * 64) ･Number of affiliations:　more than 8
A-profile memory area for the specified broadcaster	Area occupied by each broadcaster	･4KB for one broadcaster (Fixed length block of 64bytes * 64) ･Number of broadcasters that should be secured by receiver units simultaneously: more than 12
A-profile memory area of communication purpose for the specified broadcaster	Area to share information with broadcasting contents and communication contents	･2KB for 1 broadcaster. (Fixed length block of 64bytes *32) ･Number of broadcasters that should be secured by receiver simultaneously:　more than 12

Memory area for bookmark service	Area available to use for the bookmark service	･Total of more than 50 blocks of variable length block with maximum of 320 bytes.
Memory area for root CA certificates	Area to store root CA certificate of general purpose transmitted by carousels in memory.	･3KB for one certificate ･Quantity : 8
Memory area for registration transmission	Area to store messages that carry out registration transimission	･More than 3 blocks of variable length block with maximum of 1.5 KB.

NVRAM equipped in receiver units is a device with a limited number of writing. When writing times are exceeded, then a failure will occur and the lifespan of the receiver unit is shortened as a result. Therefore, it is recommended to pay attention so that exceeding the amount of writing is not done to NVRAM. This is explained in Appendix-6.

5.2.1 Allocation of NVRAM area

Numbers of each area for the A-profile memory area for the affiliation, the A-profile memory area for the specified broadcaster and the A-profile memory area of communication purpose for the specified broadcaster are specified by assuming receivable services in the receiving environment are able to be used as the standard. Therefore, in fringe areas, there are cases where it is possible to receive more services than this number of broadcasters and broadcaster affiliations by 1 receiver unit. Also, by changing the receiver setup location, broadcasters providing receivable services may change. Considering such situations, receiver units need to provide a confirmation method for the existence/non-existence of the NVRAM area allocation for broadcasters and broadcaster affiliations and methods to change the existence/non-existence that should be specified to users. Also, in the access from contents before/after allocation of the NVRAM area, a process in accordance with the rules of access limitations needs to be specified in this document.

5.2.1.1 Allocation of the A-profile memory area for the affiliation

The minimum number of affiliations that the A-profile memory area for the affiliation can be allocated to is 8. Allocation to affiliations where the affiliation identification (affiliation _id) specified in Vol. 7 with the following 8 values, is required.

Values of affiliation identification that require allocation of the A-profile memory area for the affiliation :

0, 1, 2, 3, 4, 5, 6, 7

Therefore, since the area allocation for the above 8 affiliations does not need to be changed, methods to change or initialize do not have to be equipped. Also, allocation to affiliations other

than the area 8 affiliations above in case the receiver unit has an area that exceeds that of the 8 affiliations depends on a model. However, when an area that is already allocated is being allocated to another affiliation, initializing should be executed. Initialization here refers to the status where an empty string (it is 0 when it is read as a numerical value) is returned at the time of reading.

5.2.1.2 Allocation of the A-profile memory area for the specified broadcaster and the A-profile memory area of communication purpose for the specified broadcaster

- In case there are 12 one-touch buttons placed in the remote control.
  The A-profile memory area for the specified broadcaster and the A-profile memory area of communication purpose for the specified broadcaster should be allocated to broadcasters providing service that have one-touch remote control buttons registered. Even when there are unregistered one-touch buttons, secure the area for registration in the future. However, keep in mind that one area is allocated to one broadcaster even when the same broadcaster is allocated to multiple one-touch buttons. The NVRAM area that is allocated once will continue to be allocated to the same broadcaster as long as the broadcaster's service is registered to any of the one-touch buttons. That is, even when the one touch button for services specified by one broadcaster which is registered changes from button number "1" to button number "2", allocation of NVRAM will continue and initialization will not be performed.
- In case 12 one-touch buttons are not placed in the remote control.
  The A-profile memory area for the specified broadcaster and the A-profile memory area of communication purpose for the specified broadcaster should be allocated to broadcasters providing services where the remote_control_key_id is "1"-"12" and the branch number is "0". Even when any of the remote_control_key_id's between "1"-"12" have not been received, the area should be secured for reception in the future. However, in case the remote_control_key_id is duplicated by different broadcasters, the broadcaster with the branch code "0" has the priority for allocation as a basic rule, but it is recommended to provide a selection method with another branch number service to the users.
- As the common element.
  In case of newly allocating a desired NVRAM area to broadcasters, initialization should take place along with allocation. As a basic rule, release of area allocation for broadcasters who could continue to be received may confuse the users, therefore, it is not preferred. In case of releasing allocation, an adequate announcement to the users is necessary. The allocation of the NVRAM area to broadcasters other than broadcasters allocated as required above, and the continuation or discontinuation of NVRAM area allocation to

broadcasters proving services that stray from the required conditions, depend on models, since the different processes are assumed depending on the number of NVRAM areas that the receiver units are equipped with. The timing of the process execution for the release of allocations and changes also depends on a model in the receiver units, but some attention should be paid so that the NVRAM access limit function to the original broadcaster will function without contradictions before and after registration.

In desired services, both the A-profile memory area for the specified broadcaster and the A-profile memory area of communication purpose for the specified broadcaster should be accessible or inaccessible together.

In case a re-scan (refer to Vol. 2) of receivable channels is generated, if a broadcaster who had allocated the NVRAM area before the re-scan is still receivable even after the re-scan, then the same area should be continued to be allocated to the broadcaster

5.2.2 Identification of the A-profile memory area for the all broadcasters

In case of reading and writing information for the A-profile memory area for the all broadcasters from the Multimedia service, one fixed length block is recognized as one file and the readPersistentArray( )/writePersistentArray( ) is executed. The reading and writing of information for the A-profile memory area for the all broadcasters from the Multimedia service is done in fixed length block units. To identify the fixed length block, use the URI shown below.

nvram://tr_common/<block number >

<block number> : 0~N

Refer to 5.2.8 for details on information stored in each block of the A-profile memory area for the all broadcasters.

5.2.3 Identification of the broadcaser affiliates area

In case of reading/writing information for the A-profile memory area for the affiliation from the Multimedia service, 1 fixed length block is recognized as 1 file and the readPersistentArray( )/writePersistentArray( ) is executed. The reading and writing of information for the A-profile memory area for the affiliation from the Multimedia service is done in fixed length block units. To identify the fixed length block, use the URI shown below.

nvram://<affiliation _id>;group/<block number>

<affiliation _id> : affiliation ID. Written in 2 digit hexadecimal
(when it is less than 2 digits then put 0 on the left side to make it 2 digits)

<block number> : 0~N

<affiliation_id> cannot be omitted.

In case the broadcaster that is broadcasting the contents that are currently playing does not

belong to the affiliation for the affiliation ID specified, then access to this area will fail. Refer to Vol. 4 Chapter 9 for the affiliation ID.

5.2.4 Identification of the A-profile memory area for the specified broadcaster

In case of reading/writing information for the A-profile memory area for the specified broadcaster from the Multimedia service, 1 fixed length block is recognized as 1 file and the readPersistentArray( )/writePersistentArray( ) is executed. The reading and writing of information for the A-profile memory area for the specified broadcaster from the Multimedia service is done in fixed length block units. To identify the fixed length block, use the URI shown below.

nvram://[<original_network_id>;]local/<block number>

<block number> : 0~N

<original_network_id> is always omitted, and it is regarded as the original_network_id of streams where the contents that are currently playing are transmitted is specified.

5.2.5 Identification of the A-profile memory area of communication purpose for the specified broadcaster

In case of reading/writing information for the broadcaster exlusive area for communications from the Multimedia service, 1 fixed length block is recognized as 1 file and the readPersistentArray( )/writePersistentArray( ) is executed. The reading and writing of information for the A-profile memory area of communication purpose for the specified broadcaster from the Multimedia service is done in fixed length block units. To identify the fixed length block, use the URI shown below.

nvram://[<original_network_id>;]local_web/<block number>

<original_network_id> is always omitted, and it is regarded as the original_network_id of streams where the contents that are currently playing is transmitted is specified.

5.2.6 Identification of the memory area for bookmark service

In case of reading/writing information for the memory area for bookmark service from the Multimedia service, 1 variable length block is recognized as 1 file and the “readBookmarkArray( )” / ”writeBookmarkArray( )” is executed. The reading and writing of information for the memory area for bookmark service from the Multimedia service is done in variable length block units. To identify the variable length block, use the URI shown below.

nvram://bookmark/<block number>

<block number> : 0~N(N should be greater than 49)

In case the names other than the above specification methods and the procedure function

for the memory area for bookmark service (readBookmarkArray( ),writeBookmarkArray( ))are executed, the reading/writing of NVRAM is not performed and the readBookmarkArray( ) will return null (failure) and writeBookmarkArray( ) will return NaN (failure) as return values.

Regarding the operation of the bookmark service, refer to section 5.15.

5.2.7 Use of the viewer residential area information from the Multimedia service

In case of reading/writing from the Multimedia service for the “viewer residential area information “, which is assumed to be recorded in NVRAM by the initial setup function of the receiver unit, use the URI below.

nvram://receiverinfo/<regiontype>

Specifiable character strings as <regiontype> are shown in Table 5-2. For details of the “viewer residential area information”, refer to Vol. 2.

Table 5-2 Types of viewer residential area information

<*regiontype*>	Type	Possible/impossible to Read/Write from the Multimedia service	Field type
Prefecture	Prefecture area code (bit location in the prefecture area specification bit map (ARIB STD-B10 Annex G Table G-2))	Only reading is possible as a numerical value.	U:1B
Regioncode	Region code corresponding to the emergency information signal(wireless facility regulation Article 9-3-5, regulations for wireless station operation article 138, region code corresponding to Ministry of Posts and Telecommunications announcement Sho 60 no. 405)	Only reading is possible as a numerical value.	U:2B
Zipcode	Zip code(7 digits)	Read/Write is possible as a fixed length character string with a length of 7. (ex: 5001234)	S:7B

- In case of specifying URI’s other than the identification method indicated in 5.2.2 – 5.2.7 to execute the readPersistentArray( ) and the writePersistentArray( ), then read/write of NVRAM is not performed, and the readPersistentArray( ) will return null(failure) and the writePersistentArray( ) will return NaN(failure)as return values.
- In case the viewer residential area information is not setup, and if the regiontype is prefecture then 255, if the regiontype is region code then 0, and if the regiontype is zip code then an empty string will be returned. In case values of other areas are not setup, then empty strings (0 if read as a numerical value) will be returned.

5.2.8 Operation of the A-profile memory area for the all broadcasters of receiver unit NVRAM

5.2.8.1 Purpose of the A-profile memory area for the all broadcasters

The information written in the A-profile memory area for the all broadcasters of the receiver unit NVRAM is used in order to improve the usability of viewers. To be more precise, it is used to supplement input information for viewers to avoid re-entering the same information that has been entered in other contents.

The corresponding information is only the input supplement information and it needs to be used under permission and confirmation of the viewer.

5.2.8.2 Format of the A-profile memory area for the all broadcasters

Allocation within the A-profile memory area for the all broadcasters is as in Table 5-3.

Table 5-3 Allocation of A-profile memory area for the all broadcasters blocks and the field structure.

Block number	Viewer information element	Example	Number of characters	Character type(note 8)	Field type
0	Date/time of the last update			EUC- JP	(Date/time of the last update) U:6b minute U:5b hour U:5b day U:4b month U:12b year S:4B, network identification
1	Name in hiragana(1)	やまだ＿たろう	2 byte character 15 characters (note 1)	EUC-JP Hiragana and symbols 2 byte code	S:30B, Name in hiragana S:24B, Name in kanji (Registration date) U:6b minute U:5b hour U:5b day U:4b month U:12b year S:4B, network identification
	Name in kanji(1)	山田＿太郎	2 byte character 12 characters (note 2)	EUC-JP 2 byte code	
2	Name in hiragana and name in kanji(2)		Same as above	Same as above	Same as above
3	Name in hiragana and name in kanji(3)				
4	Name in hiragana and name in kanji(4)				

Block number	Viewer information element	Example	Number of characters	Character type(note 8)	Field type
5	Name in hiragana and name in kanji(5)				
6	Name in hiragana and name in kanji(6)				
7	Name in hiragana and name in kanji(7)				
8	Name in hiragana and name in kanji(8)				
9	Hiragana address(1)	みなとくだいば 2-4-8 ふじてれび ほんしゃびる	2 byte character 28 characters (note 3)	EUC－JP Hiragana, alphanumerical and symbols 2 byte code	S:56B,Hiragana address(1) (registration date) U:6b minute U:5b hour U:5b day U:4b month U:12b year S:4B, network identification
10	Hiragana address(2)	かいはつきょく	2 byte character 24 characters (note 4)		S:48B, Hiragana address(2) (registration date) U:6b minute U:5b hour U:5b day U:4b month U:12b year S:4B, network identification
11	Kanji address(1)	港区台場 2-4-8 フジテレビ 本社ビル	2 byte character 28 characters (note 3)	EUC-JP 2 byte code	S:56B, kanji address(1) (registration date) U:6b minute U:5b hour U:5b day U:4b month U:12b year S:4B, network identification

Block number	Viewer information element	Example	Number of characters	Character type(note 8)	Field type
12	Kanji address(2)	開発局	2 byte character 24 characters (note 4)		S:48B, kanji address(2) (registration date) U:6b minute U:5b hour U:5b day U:4b month U:12b year S:4B, network identification
13	Zip code	1078006	Single byte 7 characters (note 5)	Number of EUC-JP0-9 1 Byte code	S:7B, address zip code(registration date) U:6b minute U:5b hour U:5b day U:4b month U:12b year S:4B, network identification
14	Phone number	03 1234 xxxx	Single byte 15 characters (note 6)	EUC-JP Numbers from 0-9 1 Byte code	S:15B, phone number(registration date) U:6b minute U:5b hour U:5b day U:4b month U:12b year S:4B, network identification
15	FAX NUMBER	03 1230 xxxx	Single byte 15 characters (note 6)		S:15B, FAX NUMBER(registration date) U:6b minute U:5b hour U:5b day U:4b month U:12b year S:4B, network identification
16	Mail address (1)	yama@ARIB.or.jp	Single byte 45 characters (note 7)	EUC-JP 1 Byte code	S:45B URI S:11B mobile phone number (registration date) U:6b minute U:5b hour U:5b day U:4b month U:12b year S:4B, network identification
	Mobile phone number(1)	0901234xxxx	Single byte 11characters	EUC-JP Numbers from 0-9 1 Byte code	

Block number	Viewer information element	Example	Number of characters	Character type(note 8)	Field type
17	Mail URI and mobile phone number(2)	Same as above	Same as above	Same as above	Same as above
18	Mail URI and mobile phone number(3)				
19	Mail URI and mobile phone number(4)				
20	Mail URI and mobile phone number(5)				
21	Mail URI and mobile phone number(6)				
22	Mail URI and mobile phone number(7)				
23	Mail URI and mobile phone number(8)				
24~31	RESERVED	-	-	-	-

（note 1） Separate the family name and first name with a 2 byte character space and store using hiragana or symbols.

（note 2） Separate the family name and first name with a 2 byte character space and store. Character type does not matter.

（note 3） Omit prefectures and store starting from city and ward.

（note 4） In case it cannot be stored in address (1), store the characters that cannot fit in (1) in address (2). When writing or updating the address in address (1), regardless of whether or not it fits within 28 characters, the block of address (2) should be re-written at the same time.

（note 5） When a hiragana address is input, then the input is required. When the hiragana address is updated, then the zip code should be updated along with it.

（note 6） Separate the area code, local office number, number with a single space and store.

（note 7） Do not record addresses with more than 45 characters.

（note 8） The definition of character types are as follows.

・Hiragana(2 byte code) :
Row 4 specified in ARIB STD B-24(Refer to Vol.1 Part 2 Table 7-4(1)(2))
・Alphanumerical(2 byte code) :
Row 3 specified in ARIB STD B-24(Refer to Vol.1 Part 2 Table 7-4(1)(2))
・Symbols(2 byte code) :
Row 1 and Row 2 specified in ARIB STD B-24(Refer to Vol.1 Part 2 Table 7-4(1)(2))
Except Row 1, Cell 13 to 18, and Row 2, Cell 94.
・Alphanumerical(1 byte code) :
alphanumerical group specified in ARIB STD B-24(Refer to Vol.1 Part 2 Table 7-5)

(1) The record length of each block of the A-profile memory area for the all broadcasters of NVRAM is a variable length.

- In case of executing the “readPersistentArray( )” / “writePersinstentArray( )”, specify the field type shown in Table 5-3 to the argument structure.

(2) Regarding names in hiragana and names in kanji.

- Writing is possible only when the input of kanji is possible in the kanji name field.
- In case of performing registration tasks from registration/change contents, the writing of names in hiragana is required and the writing of names in kanji is optional.
- As a basic rule, names in hiragana and names in kanji are written at the same time. In case of writing names in kanji later on for unavoidable reasons, names in hiragana need to be updated at the same time to avoid inconsistency of names in hiragana and names in kanji.
- Insert a 2 byte character space between the family name and first name for both names in the hiragana field and names in the kanji field.

(3) Regarding hiragana addresses and kanji addresses

- Input of kanji in the kanji address field is possible only when writing is possible.
- In case of performing registration tasks from registration/change contents, writing in the hiragana address is required and writing in the kanji address is optional.
- As a basic rule, hiragana addresses and kanji addresses are written at the same time. In case of writing kanji addresses later on for unavoidable reasons, hiragana addresses need to be updated at the same time to avoid inconsistency of hiragana addresses and kanji addresses.

(4) The relationship between zip codes and addresses

- Since prefecture names can be omitted in the address field, when registration/correction contents are writing in address field, make sure that the zip code is input at all times before execution.
- Writing of zip codes and addresses should be done at the same time.

(5) Regarding mail addresses

- The area to record mail addresses of individuals corresponds with the number of names in block 1-8. Contents should not record addresses that do not have corresponding names.
- Writing of mail addresses should be done with the consent of the viewer.

(6) Regarding mobile phone numbers

- The area to record mobile phone numbers of individuals corresponds with the number of names in block 1-8. Contents should not record phone numbers that do not have corresponding names.

- Writing of mobile phone numbers should be done with the consent of the viewer.

(7) Regarding character types of each block of the A-profile memory area for the all broadcasters

- Do not use "," (single byte commas), and ":" (single byte colons).
- Names in hiragana
  -Use hiragana and symbols.
- Names in kanji
  -No restrictions on character types.
  -Use 2 byte code.
- Hiragana addresses
  -Use hiragana, alphanumerical and symbols.
  -Use 2 byte code.
- Kanji addresses
  -No restrictions on character types.
  -Use 2 byte code.
- Zip codes
  -Use alphanumerical 1 byte code.
  -Use only numbers from 0-9
- Phone numbers, Fax numbers
  -Use alphanumerical 1 byte code.
  -Insert a single byte space between the area code /local code/numbers.
  -Use only numbers from 0-9.
- Mobile phone numbers
  -Use alphanumerical 1 byte code.
  -Do not insert single spaces between numbers.
  -Use only numbers from 0-9.
- URI
  -Use alphanumerical 1 byte code.

Definitions of hiragana, alphanumerical characters, and symbols in this section are in accordance with the definitions in Table 5-3 (note 8).

(8) Operation of writing history

In case of writing in each block of the A-profile memory area for the all broadcasters, the date of update and network identification of the broadcaster who executed the update should be written to each block in accordance with the format in Table 5-4.

Also, when writing/updating of even just 1 block was executed, then record and update the date/time of the last update and network identification of the broadcaster.

Table 5-4 Data structure of registration date

	Data format	Data type	Data length
Registration date, Date/time of the last update	(Year/month/day/hour/minute) U:12b,U:4b,U:5b:U:5b:U:6b(note 1)	All UnsignedInteger	Total 4Bytes
Network identification	From the display in hexadecimal character strings of "0xXXXX" format to original_network_id(note2) format without 0x.	Text	4Bytes

(note 1) In case of December 1st, 2000, 23:59, then 2000,12,1,23,59 is written
(note 2) For "original_network_id", data obtained in "getProgramID( )" is written.

5.2.9 Specification relating to writing in the A-profile memory area for the all broadcasters

(1) The conditions that writable BML contents should satisfy.

The following conditions are required for BML contents that execute the writing.

A Customer registration and changed contents of each broadcaster

- Broadcasters that writes/re-writes in the A-profile memory area for the all broadcasters should prepare customer registration, changed contents that satisfy this operational specification.

B Customer registration specified by each broadcaster and general contents with built-in documents for changes.

- In case there is the need to write or re-write in this area from general contents, it should be executed after embedding the customer registration, documents for changes specified by each broadcaster in the contents.
- Broadcasters need to prepare customer registrations and documents for changes that satisfy this operational specification and provide them to contents production companies.

(2) The conditions that BML contents that execute writing (re-writing) should satisfy.

When the BML contents specified above (1) is writing (overwriting) blocks, the following conditions are required to be satisfied.

- When information was updated by instruction of viewers from registration/changed contents (document) specified in (1).

- When client DB information is updated by instruction of viewers using methods other than the Internet on the back channel side of each broadcaster, and when information is registered or updated by instruction of viewers by pulling out the corresponding DB information from BML registration/changed contents (documents) by using some sort of method.

Other than when information is updated by the viewer, registration, changed contents should not overwrite fields in the A-profile memory area for the all broadcasters. (Prohibition to change, update without viewer operation)
…For example, when kanji is pulled out in the back channel side after the line is connected, receiver units(A-profile memory area for the all broadcasters of NVRAM)are not overwritten.

Personal information written is used to improve the usability of input supplements of viewers, and it is recommended to explicitly indicate to viewers that this information will not be diverted for other uses without permission from the viewer. Refer to the section “Specification when registering customer information to the center server” for details.

5.2.10 Specification relating to reading from the A-profile memory area for the all broadcasters

- Contents shall not leak information of the A-profile memory area for the all broadcasters using communication methods such as phone lines or LAN, etc. without permission from the viewer.
- Contents shall not copy information of the A-profile memory area for the all broadcasters from receiver units to external devices without permission from the viewer.

5.2.11 Specification regarding customer registration, changed contents

- Customer registration and changed contents (document) of each broadcaster should update the date/time of the last update, registration date and network identification only under the following circumstances.
  1) When the viewer updates information by using functions to change customer registration and changed contents information.
  2) In case incorrect information is recorded such as the registration date, and network identification of each field does not exist, and when the viewer is made to re-enter information in the field.

- When registering new client information, if there is no data written in the A-profile memory area for the all broadcasters, it is recommended that fields prepared in the A-profile

memory area for the all broadcasters (names in hiragana, hiragana addresses, zip codes, phone numbers)be written in the fields.

- Customer registration and changed contents should have operations and functions to switch the order of names.
- When a part of a name is deleted, the field is regarded as an empty field and even when names are registered in the numbers after the deleted number, the order will not be rolled forward.
- It is recommended to avoid registering the same person twice.
- If each broadcaster makes additional registrations on its own in the 8 name fields of the common area, it will cause a shortage of number of fields and confusion for the operation, therefore, when newly registering names, methods to avoid duplicating registration such as in the below example should be executed.
  1) Display the names (maximum of 8 people) of the common area in a list in the first step of new registration, and among them, only the names that are not registered to each center will be selectable, and the viewer needs to select the name of the person that the viewer is trying to register. In case the name of person who is not registered to the center is not in the name field, then display the direct input field. (Only when there are spaces in the name fields of the common area)
  2) In case the name of person to be registered is not in the list, the viewer needs to press the new name registration button to display the input field. (Only when there are spaces in the name fields of the common area)
  3) Normal registration sequence is as follows.

- To perform customer registration in the center side (by methods other than registration by broadcasting and updated contents), implementation of equipment to pull out center information from the registration and updated contents is desired.

5.2.12 Specification when registering customer information to the center server

Customer registration and changed contents (document) of broadcasters can be used as registration information to the client database of each broadcaster by reading personal information of the A-profile memory area for the all broadcasters when performing customer registration to the client data base of each broadcaster.

When registering personal information of the broadcaster common NVRAM, it is necessary to indicate the usage and purpose, and to obtain permission from the viewer before registration.

When registering, overwriting data written in the A-profile memory area for the all broadcasters is prohibited unless the information is updated by the viewer.

When registered information is irregular such as when the registration date or service identification of each field does not exist, information of the corresponding blocks is regarded as invalid and contents shall not be used for registration to the center's client DB.

When customer registration (no relationship with the receiver unit) is done in center of each broadcaster (client database side), then pulling out the corresponding information of the center and the writing of personal information to the A-profile memory area for the all broadcasters of NVRAM are permitted only when the viewer opens the registration/update contents of each broadcaster from the receiver unit to instruct registration within contents.

Registration and update contents link up multiple names of the A-profile memory area for the all broadcasters and information registered in the center server side in a ratio of 1:1, therefore, it is recommended to present multiple names to viewers and a screen (function) to prompt the linking operation should be specified.

Pulling out registration information of the center side should be done with an ID for the pull out presented to the registrant at the time as registration on the Internet, etc. and a password entered by the registrant.

5.2.13 Contents description guidelines of NVRAM access

- As written in section 5.2.1.2, if services by some broadcasters are not registered in the one-touch remote control buttons, then the A-profile memory area for the specified broadcaster and A-profile memory area of communication purpose for the specified broadcaster may not be used by the service of the broadcaster. Also, when the receiver unit cannot receive a BIT, (i.e.: immediately after power is turned on), whether or not the A-profile memory area for the affiliation is accessible cannot be judged and therefore access to the memory area may fail. Because of that, when accessing to the A-profile memory area for the specified broadcaster and A-profile memory area of communication purpose for the specified broadcaster, or the A-profile memory area for the affiliation, it is recommended that an error process in case of access failure be written each time and any unnatural performance be avoided.
- In case the broadcaster belongs to multiple affiliations, a multiple affiliation_id is written in the BIT Extended Broadcaster Descriptor. In such services by broadcasters, even if the contents created by some affiliation access the A-profile memory area for the affiliation of a different affiliation by mistake, the receiver unit will not judge this as an error, so this may destroy the information recorded by broadcasters of other affiliations. Therefore, when accessing the A-profile memory area for the affiliation, special attention needs to be

paid so that incorrect values are not written in the affiliation_id written in the URI.

5.3 Use of remote control keys from Multimedia services

5.3.1 Values that used-key-list specification could take

Value of <key-group> specifiable to style characteristics “used-key-list” and the handling of remote control keys to control exclusive processes of remote control keys in BML browsers and the station selection function are shown in the Table below.

Table 5-5 Values that key-group could take

<key-group>	Meanings
basic	Arrow keys (↑, ↓, →, ←), decide key, back key
data-button	Color key (blue, red, green, yellow), bookmark key
numeric-tuning	Numbers key for selecting stations (0 to 9, 10, 11, 12, etc)
other-tuning	Keys relating to selecting stations (one touch station selection button, channel up down key, video key, one touch selection of stations of other media, media selection button)

- Control of the “d” button by used-key-list characteristics is not possible and if the “d” button is pressed when the BML browser is active, then it is always processed by the BML browser.
- When there are keys with a selection station function other than the ones above, whether or not they should be included in “other-tuning” depends on a model.

5.3.2 Handling of remote control keys, key codes, and access keys

Remote control keys that are usable in the Multimedia service, and the mapping of characters specified as key codes and access keys are shown in Table 5-6.

Table 5-6 Handling of remote control keys, key codes, and access keys table

Remote control keys	Key codes	Access key characters
↑	1	N/A
↓	2	N/A
←	3	N/A
→	4	N/A
0~9,10,11,12	5-17	N/A
“Decide”	18	N/A
“Back”	19	‘X’
“d”	20	N/A
Color key (blue)	21	‘B’
Color key(red)	22	‘R’
Color key(green)	23	‘G’
Color key(yellow)	24	‘Y’

Remote control keys	Key codes	Access key characters
Bookmark key	100	N/A
Station selection related keys that belong to “other-tuning”.	150-	N/A

- When the “d” button is pressed, only the event “DataButtonPressed” will occur and the event “keydown” and “keyup” will not occur.
- Number selection station keys and station selection related keys other than 0 to 9 are not used for purposes other than to detect station selection operations in contents executing online communication. Also, key codes to allocate station selection related to keys depend on models.

5.3.3 Guidelines of contents using selection by color keys

In case of creating contents that do not have any other methods for selection except color keys, the color key that should be selected by the viewer should be identifiable by characters, etc. which indicate the corresponding color other than color distinction.

5.4 Operation of the BML version

It is set as “major_version=3”, “minor_version=0.”

5.5 Operation of character codes

Refer to ARIB STD-B24 Vol. 2 Appendix 2 ”4.1. Character codes”.

5.5.1 Transmission of DRCS pattern data

- Setup of pattern data of DRCS used in BML documents and 8-bit character codes that are externally referenced to are both done by “loadDRCS( )”.
- For 1 CharacterCode, a DRCS pattern of multiple fonts and multiple sizes may be transmitted in some cases.
- The maximum total size of the DRCS pattern used in a BML document is 64KB.
- The relationship between fontID and typestyle specified in the DRCS pattern data are as shown in Table 5-7. “fontID=0” is not operated.

Table 5-7 Handling of fontID and typestyle of DRCS

fontID	Typestyle
0	(Not operated)
1	Round gothic
2	Angle gothic
3	Bold round gothic

5.6 Operation area of media type and mono-media

Should be in accordance with ARIB STD-B24 Vol. 2 Appendix 2 ”4.2. Operational range of media types and monomedia” except for the following points.

- CSS data (media type "text/css")may appear in BML documents in some cases and may be transmitted as independent resources in other cases. CSS data transmitted as mono-media should be CSS on its own with a concluded description.
- ECMAScript data (media type "text/X-arib-ecmascript; charset='euc-jp'") may appear in BML documents in some cases and may be transmitted as independent resources in other cases. ECMAScript data transmitted as mono-media should be ECMAScript data on its own with concluded expression.

5.7 Operation of BML elements

5.7.1 Entity

Refer to ARIB STD-B24 Vol. 2 Appendix 2 ”4.3.1.Entity”

5.7.2 Operational guidelines of BML elements

Operational guidelines for basic service of elements used in BML are shown in Table 5-8. The meaning of “O”, “Δ”, “-” are in accordance with ARIB STD-B24 Vol. 2 Appendix 2 “Chapter 3 Legend”

Table 5-8 Operational guidelines of elements

Module		BML(Operation)	Elements
Core	Structure (required)	O	body
		O	head
		-	html
		O	title
	Text (required)	-	abbr
		-	acronym
		-	address
		-	blockquote
		O	br
		-	cite
		-	code
		-	dfn
		O	div
		-	em
		-	h1~h6
		-	kbd
		O	p
		-	pre
		-	q
		-	samp
		O	span
		-	strong
		-	var
	Hypertext (required)	O	a
	List (required)	-	dl
		-	dt
		-	dd
		-	ol
		-	ul
		-	li
Applet (deprecated)		-	applet
		-	param
Text Extension	Presentation	-	b
		-	big
		-	hr
		-	i
		-	small
		-	sub
		-	sup
		-	tt
	Edit	-	del
		-	ins
	Interaction channel	-	bdo
Form	Basic Forms	-	form
		-	input
		-	label
		-	select
		-	option
		-	textarea
	Forms	-	form
		O	input
		-	select
		-	option
		-	textarea
		-	button
		-	fieldset

Module		BML(Operation)	Elements
		-	label
		-	legend
		-	optgroup
Table	Basic Tables	-	superimpose
		-	table
		-	td
		-	th
		-	tr
	Tables	-	superimpose
		-	table
		-	td
		-	th
		-	tr
		-	col
		-	colgroup
		-	tbody
		-	thead
		-	tfoot
Image		-	img
CS Image Map		-	*a&*
		-	area
		-	*img&*
		-	map
		-	*object&*
SS Image Map		-	*img&*
Object		O	object
		-	param
Frames		-	frameset
		-	frame
		-	noframes
Target		-	*a&*
		-	*area&*
		-	*base&*
		-	*link&*
		-	*form&*
Iframe		-	iframe
Intrinsic Events		-	*a&*
		-	*area&*
		-	*form&*
		-	*body&*
		-	*label&*
		-	*input&*
		-	*select&*
		-	*textarea&*
		-	*button&*
Metainformation		O	meta
Scripting		-	noscript
		O	script
Stylesheet		O	style
Style Attribute		O	
Link		O	link
Base		-	base
Name Identification (deprecated)		-	*a&*
			applet&
			form&
			frame&
			iframe&
			img&
			map&

Module		BML(Operation)	Elements
Legacy (deprecated)		-	basefont
		-	center
		-	font
		-	s
		-	strike
		-	u
		-	*body&*
		-	*br&*
		-	*superimpose&*
		-	*div&*
		-	*h1-h6&*
		-	*ht&*
		-	*img&*
		-	*input&*
		-	*legend&*
		-	*li&*
		-	*ol&*
		-	*p&*
		-	*pre&*
		-	*script&*
		-	*table&*
		-	*tr&*
		-	*th&*
		-	*td&*
		-	*ul&*
BML	BML	O	*bml*
		O	*bevent*
		O	*beitem*
		O	*body&*
		O	*div&*
		O	*p&*
		O	*span&*
		-	*a&*
		-	*bdo&*
		O	*object&*
BML	Basic BML	-	*bevent*
		-	*beitem*
		-	*body&*
		-	*div&*
		-	*p&*
		-	*span&*
		-	*object&*

5.7.2.1 Restrictions on the order of elements within head elements

In a “head” element, there are a “title” element, a “style” element, a “link” element, a “script” element, and a “bevent” element, appearing in this order. A “title” element appears once and a “meta” element, a “style” element, a “link” element, a “bevent” element will appear 0 or 1 time. The “script” element appears 0 or 1 time without the “src” attribute specification, and the “script” element with the “src” attribute specification appears 0-2 times.

5.7.3 Attributes

Operational guidelines of attributes relating to elements specified for operation in section 5.7.2 are shown in Table 5-9. The meaning of “O”, “Δ”, “-” are in accordance with ARIB STD-B24 Vol. 2 Appendix 2 “Chapter 3 Legend.”

Table 5-9 Operational guidelines relating to the attributes of elements

Elements	Attribute	Operation	Restrictions for operation
Common Attributes			
Core Attributes			
	id	O	Character string with a maximum of 128 bytes
	class	O	
	title	-	
I18N Attributes			
	xml:lang	Δ	Fixed to "ja"
Events Attributes			
	onclick	O	
	ondbclick	-	
	onmousedown	-	
	onmouseup	-	
	onmouseover	-	
	onmousemove	-	
	onmouseout	-	
	onkeypress	-	
	onkeydown	O	
	onkeyup	O	
Style Attributes			
	style	O	
Core Modules			
Structure Module			
Body	%Common.attrib;		
	%Core.attrib;	O	
	%I18n.attrib;	Δ	
	%Events.attrib;	-	
	%Style.attrib;	O	
Head	%I18n.attrib;	Δ	
	profile	-	
Title	%I18n.attrib;	Δ	
Text Module			
Br	%Core.attrib;	O	
	%Style.attrib;	O	
Div	%Common.attrib;	O	
P	%Common.attrib;	O	
Span	%Common.attrib;	O	
Hypertext Module			
a	%Common.attrib;	O	
	accesskey	O	
	charset	Δ	Fixed to "EUC-JP"
	href	O	
	hreflang	-	
	rel	-	
	rev	-	
	tabindex	-	

Elements	Attribute	Operation	Restrictions for operation
	type	-	
Forms Modules			
Forms Module			
input	%Common.attrib;		
	%Core.attrib;	O	
	%I18n.attrib;	Δ	
	%Events.attrib;	O	Cannot be specified when “inputmode” attribute is "direct" or "indirect".
	%Style.attrib;	O	
	accesskey	O	
	checked	-	
	disabled	O	
	readonly	O	
	maxlength	O	From 1 to 40. note 1)
	alt		
	name	-	
	size	-	
	src	-	
	tabindex	-	
	accept	-	
	type	O	Either "text" or "password".
	value	O	
	inputmode	O	
	charactertype	O	
Client-side Image Map			
a&	corrds	-	
	shape	-	
input&	usemap	-	
object&	usemap	-	
Server-side Image Map			
input&	ismap	-	
Object Module			
object	%Common.attrib;	O	
	archive	-	
	classid	-	
	codebase	-	
	codetype	-	
	data	O	
	declare	-	
	height	-	
	name	-	
	standby	-	
	tabindex	-	
	type	O	
	width	-	
Target Module			
a&	target	-	
Intrinsic Events Module			
a&	onblur	O	
	onfocus	O	
body&	onload	O	
	onunload	O	Note 3)
input&	onfocus	O	
	onblur	O	
	onselect	-	

Elements	Attribute	Operation	Restrictions for operation
	onchange	O	Note 2)
Metainformation Module			
meta	%I18n.attrib;	Δ	
	http-equiv	-	
	name	O	
	content	O	
	scheme	-	
Scripting Module			
script	charset	Δ	Fixed to "EUC-JP".
	type	Δ	Fixed to "text/X-arib-ecmascript; charset="euc-jp&quot:".
	src	O	
	defer	-	
	xml:space	-	
Style Sheet Module			
style	%I18n.attrib;	Δ	
	type	Δ	Fixed to "text/css".
	media	Δ	Fixed to "tv".
	title	-	
	xml:space	-	
Link Module			
link	%Common.attrib;	-	
	charset	Δ	Fixed to "EUC-JP".
	href	O	
	hreflang	-	
	media	Δ	Fixed to "tv".
	rel	Δ	Fixed to "stylesheet".
	rev	-	
	type	Δ	Fixed to "text/css".
BMLmodule			
bml	%I18n.arrtib;	Δ	
	version	-	
	xmlns	-	
bevent	id	O	
beitem	id	O	
	type	O	One of the following is taken: "EventMessageFired", "ModuleUpdated", "ModuleLocked", "TimerFired", "DataEventChanged", "CCStatusChanged", "MainAudioStreamChanged", "NPTReferred", "MediaStopped", "DataButtonPressed","IPConnectionTerminated"
	onoccur	O	
	es_ref	O	
	message_group_id	O	It is "0" or ”1”. When omitted, specification of “0” is assumed.
	message_id	O	
	message_version	O	
	module_ref	O	
	language_tag	O	
	register_id	-	
	service_id	-	
	event_id	-	
	peripheral_ref	-	
	time_mode	O	One of the following is taken: "absolute", "origAbsolute", "NPT"

Elements	Attribute	Operation	Restrictions for operation
	time_value	O	
	object_id	O	Only the object element ID that indicates data transmitted by carousel and type attribute is either "audio/X-arib-aiff" or "audio/X-arib-mpeg2-aac".
	subscribe	O	
iframe&	align	-	
body&	invisible	O	
div&	accesskey	O	
	onfocus	O	
	onblur	O	
p&	accesskey	O	
	onfocus	O	
	onblur	O	
span&	accesskey	O	
	onfocus	O	
	onblur	O	
a&	effect	-	
bdo&	orientation	-	
object&	streamposition	O	The frame number is specified (type="image/X-arib-mng")when the mono-media that refers to the relevant object element is MNG. In case of other media, it is “0”.
	streamlooping	Δ	Fixed to "1".
	streampositionnumerator	-	
	streampositiondenominator	-	
	streamstatus	O	Initial value is specified by mono-media that corresponds to the relevant object element. Refer to ARIB-B24 Vol. 2 Appendix 2 “4.8.5.2.Operation of attributes regarding presentation of stream data”
	streamlevel	-	
	remain	O	Possible/impossible is determined by mono-media that refers to the relevant object element. Refer to ARIB-B24 Vol. 2 Appendix 2 “4.8.5.1. Application of remain attribute for each type”
	accesskey	O	
	onfocus	O	
	onblur	O	

Note 1) When the input exceeds the maximum length, then it is rounded down. If it goes beyond the frame, the exceeded part will not be displayed.

Note 2) Generated timing of the change event is when the focus is shifted to a different element.

Note 3) Only extended functions for broadcasting that can be used in the “onunload” event handler are “writePersistentArray()” and “unlockModuleOnMemory()”. Process contents should be limited to processes that end in a short time such as set up to Ureg where quick document transition is possible and judgement of simple condition, etc.

5.7.4 Operation relating to beitem elements

Refer to ARIB STD-B24 Vol. 2 Appendix 2 ”4.3.4. Operation of beitem element”.

5.7.5 Operation of the BML elements extension module (interruption event)

- The maximum number of ModuleUpdated events where the subscribe attribute can be set as subscribe simultaneously is 16.

- The maximum number of TimerFired events where the subscribe attribute can be set as subscribe simultaneously is 8. When absolute playback time/at the reception time/NPT time specified by TimerFired is time that is already passed at the time of BML document interpretation, then immediate firing will be executed.
- Performance when multiple event handlers are corresponded to for the same event depends on a model. The following descriptions, for example, fall under this category.
  - Multiple beitem elements whose type attribute is "DateEventChanged" are written.
  - Multiple beitem elements whose type attribute is “ModuleUpdated” are written and the same module is specified in the module_ref attribute.
- Even when the data event is updated in the ES that is not being presented, the DataEventChanged event will not occur.
- Regarding modules transmitted by the ES that is not being presented, when the ModuleUpdated event is subscribed and the data event is updated by the ES transmitting the module, and if the pullout that comes along with it does not happen, then the ModuleUpdated event will occur. As the status value of the ModuleUpdated event in this case, the following values will be newly operated.

Value of status	Meaning
4	The data event was updated in the ES where the corresponding module is transmitted. The corresponding module was not sent out before the update of the data event, and the corresponding module was sent out after the update of the data event.
5	The data event was updated in the ES where the corresponding module is transmitted. The corresponding module was sent out before the update of the data event, and the corresponding module was sent not out after the update of the data event.
6	The data event was updated in the ES where the corresponding module is transmitted. The corresponding module was sent out before and after the update of the data event.

- The following specifications are stipulated regarding the MainAudioStreamChanged event.
  - Factors for generating events where the status is -1 depend on models.
  - When the data attribute of object elements presenting audio streams in documents that are currently presented is other than "/-1" or the occurrence of the MainAudioStreamChanged event when overwriting of the data attribute of corresponding object elements occurs depends on a model of receiver.
  - When the stream status attribute of object elements presenting audio streams in documents that are currently presented is other than “play”, or the occurrence of

MainAudioStreamChanged event when overwriting of stream status attribute of corresponding object element occurs depends on a model of receiver units.

- The performance when "/-1" is specified to es_ref depends on a model.
- The performance in case the object element that refers to an audio stream does not exist in the document depends on a model of receiver units.
- When the channel ID is not specified to es_refAttribute, an event will not occur at the time of switching the main audio from/to primary/secondary.
- When es_ref is omitted in the MainAudioStreamChanged event, since specification for the targeted ES and channel ID is not possible, in which case an event should occur depends on a model of receiver units. Therefore, to operate contents safely, the es_ref should always be specified.

5.8 Operation of CSS

Refer to ARIB STD-B24 Vol. 2 Appendix 2 ”4.4. CSS-based style sheet”.

5.9 Operation guidelines relating to procedure description

5.9.1 Operation area of DOM

Refer to ARIB STD-B24 Vol. 2 Appendix 2 ”4.5.1. Operational guidelines of DOM”

5.9.2 Operation area of embedded objects

Refer to ARIB STD-B24 Vol. 2 Appendix 2 ”4.5.2.Operation range of build-in objects”

5.9.3 Operation area of extended objects for broadcasting

Refer to ARIB STD-B24 Vol. 2 Appendix 2 ”4.5.3.Operation range of extended objects for broadcasting”

5.9.4 Operation area of browser pseudo-objects

Operated in accordance with Table 5-10. The meaning of the “operation” fields are as follows.

“O”	Basic function in this specification.
“O(*)”	Optional function in this specification. Therefore, in case of using these functions in contents, inspect the possible/impossible of process function of the corresponding function in the “getBrowserSupport( )” function, and call the corresponding function only when the process is possible.
“-”	This is neither a basic or optional function in this specification.

Table 5-10 Operation area of browser pseudo-objects

	Functions	Operation	Comments
Ureg related functions			
	Ureg[]	O	
Greg related functions			
	Greg[]	O	
EPG functions			
	epgGetEventStartTime( )	O	
	epgGetEventDuration( )	O	
	epgTune( )	O	
	epgTuneToComponent( )	O	
	epgTuneToDocument( )	-	
	epgIsReserved( )	O	
	epgReserve( )	O	
	epgCancelReservation( )	O	
	epgRecIsReserved( )	O	
	epgRecReserve( )	O	
	epgRecCancelReservation( )	O	
Event group index functions			
	grpIsReserved( )	-	
	grpReserve( )	-	
	grpCancelReservation( )	-	
	grpRecIsReserved( )	-	
	grpRecReserve( )	-	
	grpRecCancelReservation( )	-	
	grpGetNodeEventList( )	-	
	grpGetERTNodeName( )	-	
	grpGetERTNodeDescription( )	-	
	epgXTune( )	-	
Series appointment functions			
	seriesIsReserved( )	O(*)	(note 1)
	seriesReserve( )	O(*)	(note 1)
	seriesCancelReservation( )	O(*)	(note 1)
	seriesRecIsReserved( )	O(*)	(note 1)
	seriesRecReserve( )	O(*)	(note 1)
	seriesRecCancelReservation( )	O(*)	(note 1)
NVRAM functions			
	readPersistentString( )	-	
	readPersistentNumber( )	-	
	readPersistentArray( )	O	
	writePersistentString( )	-	
	writePersistentNumber( )	-	
	writePersistentArray( )	O	
	copyPersistent( )	-	
	getPersistentInfoList( )	-	
	deletePersistent( )	-	
	getFreeSpace( )	-	
Functions for controlling access-controlled non-volatile memory areas			
	setAccessInfoOfPersistentArray( )	-	
	checkAccessInfoOfPersistentArray( )	-	
	writePersistentArrayWithAccessCheck( )	-	
	readPersistentArrayWithAccessCheck( )	-	
Interaction channel communication			
Interaction channel communication- delayed calling			

	Functions	Operation	Comments
	registerTransmission( )	-	
	registerTransmissionStatus( )	-	
	getTransmissionStatus( )	-	
	setDelayedTransmissionDataOverBasic( )	-	
Interaction channel communication -Basic Procedure			
	connect( )	O	(note 2)
	disconnect( )	O	(note 2)
	sendBinaryData( )	-	
	receiveBinaryData( )	-	
	sendTextData( )	O	(note 2)
	receiveTextData( )	O	(note 2)
Interaction channel communication-TCP/IP			
	setISPParams( )	O	
	getISPParams( )	O	
	connectPPP( )	O	
	connectPPPWithISPParams( )	O	
	disconnectPPP( )	O	
	getConnectionType( )	O	
	isIPConnected( )	O	
	saveHttpServerFileAs( )	-	
	saveHttpServerFile( )	-	
	sendHttpServerFileAs( )	-	
	saveFtpServerFileAs( )	-	
	saveFtpServerFile( )	-	
	sendFtpServerFileAs( )	-	
	sendTextMail( )	O(*)	
	sendMIMEMail( )	O(*)	
	transmitTextDataOverIP( )	O	
	setDelayedTransmissionData( )	-	
	setCacheResourceOverIP()	O(*)	
Interaction channel communication – Acquisition function of delayed calling status that is common for basic system procedures and IP connections.			
	getDelayedTransmissionStatus( )	-	
	getDelayedTransmissionResult( )	-	
Interaction channel communication-Functions to acquire line connection status			
	getPrefixNumber( )	O	
Interaction channel communication-Large Vol. call acceptance service			
	vote( )	O	(note 2)
Interaction channel communication-Enencrypted communication using CAS			
	startCASEncryption( )	-	
	transmitWithCASEncryption( )	-	
	endCASEncryption( )	-	
Interaction channel communication- Encrypted communication with a secret key not using CAS			
	setEncryptionKey( )	-	
	beginEncryption( )	-	
	endEncryption( )	-	
Operational control functions			
	reloadActiveDocument( )	O	
	getNPT( )	O	
	getProgramRelativeTime( )	O	
	isBeingBroadcast( )	O	
	lockExecution( )	-	
	unlockExecution( )	-	
	lockModuleOnMemory( )	O	

	Functions	Operation	Comments
	unlockModuleOnMemory( )	O	
	setCachePriority( )	O	
	getTuningLinkageSource( )	-	
	getTuningLinkageType( )	-	
	getLinkSourceServiceStr( )	-	
	getLinkSourceEventStr( )	-	
	getIRDID( )	O	
	getBrowserVersion( )	O	
	getProgramID( )	O	
	getActiveDocument( )	O	
	lockScreen( )	O	
	unlockScreen( )	O	
	getBrowserSupport( )	O	
	launchDocument( )	O	
	launchDocumentRestricted( )	O	
	quitDocument( )	O	
	launchExApp( )	O(*)	(note 3)
	getFreeContentsMemory( )	O	
	isSupportedMedia( )	O	
	detectComponent()	O	
	lockModuleOnMemoryEx()	O	
	unlockModuleOnMemoryEx()	O	
	unlockAllModulesOnMemory()	O	
	getLockedModuleInfo()	O	
	getBrowserStatus()	O	
	getResidentAppVersion()	O	
	isRootCertificateExisting()	O	
	getRootCertificateInfo()	O	
	startResidentApp()	O(*)	
	startExtraBrowser()	O(*)	
Receiver audio control			
	playRomSound( )	O	
Timer functions			
	sleep( )	O	
	setTimeout( )	-	
	setInterval( )	O	
	clearTimer( )	O	
	pauseTimer( )	O	
	resumeTimer( )	O	
	setCurrentDateMode( )	O	
External character functions			
	loadDRCS( )	O	
	unloadDRCS( )	-	
External device control functions			
	enumPeripherals( )	-	
	passXMLDocToPeripheral( )	-	
Other functions			
	random( )	O	
	subDate( )	O	
	addDate( )	O	
	formatNumber( )	O	
Closed caption display control functions			
	setCCStreamReference( )	-	
	getCCStreamReference( )	-	
	setCCDisplayStatus( )	O	
	getCCDisplayStatus( )	O	
	getCCLanguageStatus( )	O	
Directory operation functions			
	saveDir( )	-	

	Functions	Operation	Comments
	saveDirAs( )	-	
	createDir( )	-	
	getParentDirName( )	-	
	getDirNames( )	-	
	isDirExisting( )	-	
File operation functions			
	saveFile( )	-	
	saveFileAs( )	-	
	getFileNames( )	-	
	isFileExisting( )	-	
File input/output functions			
	writeArray( )	-	
	readArray( )	-	
Query functions			
	getDirInfo( )	-	
	getFileInfo( )	-	
	getCarouselInfo( )	-	
	getModuleInfo( )	-	
	getContentSource( )	-	
	getStorageInfo( )	-	
Data carousel accumulation functions			
	saveCarouselAs( )	-	
	saveCarousel( )	-	
	saveModuleAs( )	-	
	saveModule( )	-	
	saveResourceAs( )	-	
	saveResource( )	-	
Bookmark control functions			
	writeBookmarkArray( )	O	
	readBookmarkArray( )	O	
	deleteBookmark( )	O	
	lockBookmark( )	O	
	unlockBookmark( )	O	
	getBookmarkInfo( )	O	
	getBookmarkInfo2( )	O	
	startResidentBookmarkList()	O(*)	
Print related function API – basic print functions			
	getPrinterStatus()	O(*)	
	printFile()	O(*)	
	printTemplate()	O(*)	
	printUri()	O(*)	
	printStaticScreen()	O(*)	
Print related functionAPI – Memory card related			
	saveImageToMemoryCard()	O(*)	
	saveHttpServerImageToMemoryCard()	O(*)	
	saveStaticScreenToMemoryCard()	O(*)	

(note 1) Should be equipped on receiver units with the series appointment function.

(note 2) It is equipped on receiver units with modems.

(note 3) In accordance with ARIB STD-B24 Vol. 2 Appendix 1 even when using independent services.

5.9.5 Operation range of navigator pseudo-objects

It is not operated.

5.9.6 Extended functions that have been added in the digital terrestrial broadcasting

5.9.6.1 Print function (Optional)

Refer to section 6.2.

Refer to "7.6.17 Functions for Printing" in ARIB STD-B24 Vol.2 for the print-related functions.

5.9.6.2 Receiver application identification function (Optional)

Refer to "7.6.8 Operational control functions" in ARIB STD-B24 Vol.2 for getResidentAppVersion( ), the receiver application identification function.

Refer to "5.12.6.9 Operation of operational control function (11)" for the operation of getResidentAppVersion( ).

5.9.6.3 Function to acquire root CA certificate information

- Confirmation of root CA certificate

Refer to "7.6.7.9 Functions for operating root certificates for encrypted transmission" in ARIB STD-B24 Vol.2 for isRootCertificateExisting(), the confirmation function for the root certificate.

Refer to "5.12.6.9 Operation of operational control function (12)" for the operation of isRootCertificateExisting( ).

- Information acquisition of general-purpose root CA certificates

Refer to "7.6.7.9 Functions for operating root certificates for encrypted transmission" in ARIB STD-B24 Vol.2 for getRootCertificateInfo( ), the information acquisition function for the general-purpose root certificate.

5.9.6.4 Cache set-up function of communication contents (Optional)

Refer to "7.6.7.6 Communication functions assuming TCP/IP" in ARIB STD-B24 Vol.2 for setCacheResourceOverIP( ), the cache setting function for communication content.

Refer to "5.14.6.6 Interaction channel function-TCP/IP" for the details of the operation of setCacheResourceOverIP( ).

5.9.6.5 Receiver's native application startup function (Optional)

Refer to "7.6.8 Operational control functions" in ARIB STD-B24 Vol.2 for startResidentApp( ), the receiver application activation function.

Refer to "5.12.6.9 Operation of operational control function (13)" for the operation of startResidentApp( ).

5.9.6.6 Specifiable values in arguments of “getBrowserSupport()”

Refer to "5.12.6.9 Operation of operational control function (14) Operation of getBrowserSupport( )".

5.10 Restrictions in the BML document description

Should be in accordance with ARIB STD-B24 Vol. 2 Appendix 2 “4.7 Restrictions on description of BML documents” except for the following points.

- Script elements without the src attribute specification will appear 0 or 1 time, and script elements with the src attribute specification will appear 0-2 times. When writing the contents of script elements, put the entire element in <![CDATA[and]]>, and make it as one CDATA section.

5.11 Presentation control of BML document

Should be in accordance with ARIB STD-B24 Vol. 2 Appendix 2 ”4.8 Guidelines for presentation control of BML document” except for the following points.

(1) When the object elements whose type attribute value is "image/jpeg" (JPEG object) satisfy the following conditions, the remain attribute of the element can be made valid.

- Remain attribute of other JPEG objects of the same BML documents are not valid.
- Other JPEG objects and the display area do not overlap each other in the same BML document.
- The target JPEG object is set before the video image in z-order (including implicit z-order based of order of appearance of elements) and the width characteristics and height attribute of object elements for the video image are the same value as the horizontal pixel size and vertical pixel size indicated by the pixel size characteristics of body elements.
- The target JPEG object should be a sub element of the div element, which is the first sub element of the body element. Also, the sub elements of the above div element should be in the following combinations.
  - JPEG object where at most one remain attribute is valid.
  - Object element where at most the value of the type attribute is one is "video/X-arib-mpeg1" or "video/X-arib-mpeg2", and the remain attribute is valid.
  - Object element where at most the value of type attribute is one is "audio/X-arib-mpeg2-aac" and remain attribute is valid.
- In case the document is in transition while the remain attribute of the target JPEG object is valid, then the resource that is specified by the data attribute of the corresponding JPEG object should be locked in beforehand by “lockModuleOnMemory()” or “lockModuleOnMemoryEx()”.
- Background-image characteristics of the body element, which is the ancestor element of the target JPEG object should not be specified.

(2) When the stream status attribute is stopped for the file audio with the type attribute of "audio/X-arib-mpeg2-aac ", then playback of audio is stopped.

5.12 Behavior of BML browsers

5.12.1 Operational guidelines relating to presentation

Refer to ARIB STD-B24 Vol. 2 Appendix 2 "5.1.Operational guidelines of presentation" The following specifications, however, need to be added.

- In the case of the following, execute displays in accordance with TV video pixel size.
  - While receiving empty carousels
  - While acquiring BML documents immediately after selecting stations
- When switched to a non empty carousel by updating the data event while receiving empty carousels, execute displays in accordance with TV video pixel size until the BML document is displayed.
- In case the invisible attribute of body element of BML document is invisible, then execute displays in accordance with TV video pixel size just in the same way as when the BML document is not presented.
- During the performance of BML browser, if presentation of the BML document is needed, then
  - Document transition by "launchDocument()", etc.
  - Transition to entry component by pull flag
  - In case of transition to the startup document by an update of the data event during presentation, etc., during the period until the presentation of the document is started after completing the acquisition of new the BML document, (For example, the period after update of the date event happens and until the presentation ofstartup document actually starts) the presentation of the BML document that was presented immediately before is continued.
- Regarding the operation of CLUT, it should be in accordance with ARIB STD-B24 Vol. 2 Appendix 3, Table 5-4.

5.12.2 Operational guidelines relating to the performance of external characters

Refer to ARIB STD-B24 Vol. 2 Appendix 2 "5.2. Guidelines on behavior of management of external characters".

5.12.3 Behavior of the DOM

Refer to ARIB STD-B24 Vol. 2 Appendix 2 "5.3 Behavior of DOM". However, restrictions on the "size of character string set to NodeValue" written in section 5.3.5 Table 5-13 are not applied to CDATA section in script element and style element.

5.12.4 Operation of script language

Refer below. In BML documents where multiple script elements are scripted, the following restriction will be applied for the status where all scripts (scripts written in resources indicated by the src attribute of script elements, and the script written inside of script elements without the src attribute) are read.

- ARIB STD-B24 Vol. 2 Appendix 3 ”5.4.1.Operation of script working environment”
- ARIB STD-B24 Vol. 2 Appendix 2 ”5.4.2. Data type” “5.4.3 Effects on basic objects caused by data type restrictions”, “5.4.4. Operational general rule of implementation-dependent behavior”

5.12.5 Extended objects for broadcasting

Refer to ARIB STD-B24 Vol. 2 Appendix 2 ”5.5. Extended objects for broadcasting”.

5.12.6 Browser pseudo-objects

Operation of browser pseudo-objects is specified below. For those without particular specifications, they should be in accordance with ARIB STD-B24 Vol. 2 “XML-based multimedia coding scheme”.

- “Appendix 1 Operational guidelines”
- “Appendix 2 Operational guidelines for implementing basic services”

5.12.6.1 Operation of Ureg

Should be in accordance with ARIB STD-B24 Vol. 2 Appendix 2 “5.6.1. Operation of Ureg”. The timing for receiver units to write the contents group identifier (service_id in hexadecimal character string in “0xXXXX” format) to Ureg[0] is when the initial document of the contents group is acquired, in other words, when power is turned on, and when selecting stations.

Initialization of the Ureg value is done by the receiver unit when the browser is started or contents groups are changed (when selecting stations including when media is changed). The value read after initialization is an empty strings.

5.12.6.2 Operation of Greg

Should be in accordance with ARIB STD-B24 Vol. 2 “7.6.16 Greg pseudo object properties” The value of the Greg is initialized at the time of turning the power on, and the value is kept at all times while the power is ON. The value read after initialization is an empty string. Even when the communication contents that do not support Greg are presented, the Greg value is maintained. In case of presenting other media that do not support Greg, it is preferred for the Greg value to be maintained.

In case the Greg value cannot be maintained unavoidably while presenting other media, the

receiver unit initializes the value of Greg when presenting the first media that is supporting Greg after that.

5.12.6.3 Operation of EPG functions

Should be in accordance with ARIB STD-B24 Vol. 2 Appendix 2 ”5.6.2. Guideline for behavior of EPG functions” except for the additional specifications below.

(1) Operation of epgTune(), epgTuneToComponent()

Operation of epgTune()/epgTuneToComponent() are in accordance with ARIB STD-B24 Vol. 2 Appendix 1 ”8.5.1 Operation of operational control functions”, however, the performance when "arib://-1.-1.-1" or "arib-dc://-1.-1.-1" is specified as a service at transition destination by argument is specified as below. (Regarding the operation of service specification by the name of "arib://-1.-1.-1", also refer to section 5.13.3)

- Receiver units will finish BML browser once. (note1) And then,
  - In case of epgTune(), execute 4 and after the “Basic operation of receiver when selecting channels” written in section 2.1.10.2.
  - In case of epgTuneToComponent(), execute acquisition and presentation of thestartup document of specified components.

  (note 1) Regarding the implementation of epgTune()/epgTuneToComponent(), securing broadcasting program and contents uniqueness in particular should be carefully taken care of. (Refer to Vol. 2 9.3) For example, when 2 screens are displayed in the receiver features, and when epgTune()/epgTuneToComponent() is executed from the document presented by one of the 2 screens, the BML browser presenting the document that had called the function will end and presentation of video, audio, contents at the newly selected station specified by the function on the same screen is demanded. Implementation to present services specified by the same function to screens other than the screen presenting the document which executed the same function while not ending the original document and the screen presenting the original document will not be allowed.
- In case of TV programs with additional data, for TV video, TV audio, closed caption, the same ES will be played without pauses even after function execution.
- Display, etc. of channel banners along with selecting station performance is not executed.

(2) Operation of “epgReserve()” / “epgCancelReservation()” / ”epgRecReserve()” / “epgRecCancelReservation()”

After completing performance of the function, the succeeding script will be executed.

5.12.6.4 Operation of series reservation function

Operation is not required. In case of operation, it should be in accordance with ARIB STD-B24 Vol. 2 “7.6.3 Series reservation functions”. Also, the following specifications will be added.

- Whether or not receiver units can handle series reservation related to the API is identified by “getBrowserSupport()”.
- <series_scope_ref> used in identification of series is <network_id>/<terrestrial_broadcater_id>/<media_type>. In here,
  <network_id> Network ID (4 digits hexadecimal)
  <terrestrial_broadcaster_id> Broadcaster ID (4 digits hexadecimal)
  <media_type> media type
  1 TV type media type
  2 Audio type media type
  3 Data type media type
  For ID’s specified in hexadecimal, “0x” in the beginning or “h” at the end will not be attached.
- After completing performance of the function, the succeeding scripts will be executed.

5.12.6.5 Operation of the NVRAM functions

Should be in compliance with ARIB STD-B24 Vol. 2 Appendix 2 ” 5.6.3. Operation of non-volatile memory function”

5.12.6.6 Interaction channel function-Operation of TCP/IP

Refer to “5.14.6.6 Interaction channel function-TCP/IP”

5.12.6.7 Interaction channel function-Operation of the function that acquires the line connection status

Refer to “5.14.6.7 Interaction channel function/Function to acquire line connection status”

5.12.6.8 Interaction channel function-Operation of large volume call acceptance service

Refer to “5.14.6.8 Interaction channel function-Mass calls reception service”.

5.12.6.9 Operation of operational control function

For use from communication contents, refer to 5.14.6.9 “Operational control function”.

(1) Operation of “lockModuleOnMemory( )” and “unlockModuleOnMemory( )”.

- Receiver units should always have 1MB of free space to acquire modules from the carousel by execution of “lockModuleOnMemory( )”, and in case a memory area of 1MB

cannot secured, then the fixation process of the module should not be performed by this function.

- Even when the module version fixed to memory is updated, receiver units will not re-obtain the module automatically. Procedures to perform the detection of module update/lock release/re-fixation of modules to memory are written in the BML document.
- After “lockModuleOnMemory()” is executed, if transition to the document within the same ES document as the document that is currently presented is executed before the lock is completed, then lock performance is continued.
- Even if the module locked by “lockModuleOnMemory()” disappears from the carousel that is currently transmitted, the lock on the module continues.
- The module locked by “lockModuleOnMemory()” can be unlocked by “unlockModuleOnMemory()” regardless of carousel that is being transmitted is included in the module or not.

(2) Operation of “setCachePriority( )”

- Modules that can be specified with the argument of “setCachePriority()” are limited to modules transmitted by components included in the service that is currently being presented.
- For the priority of “setCachePriority()”, the value of n*10 (n= integer value between 1-12) and negative values can be specified. When a value between 0-120 is specified, receiver units should perform look-ahead caching of modules in the order starting from the larger priority value. However, depending on the performance status of the receiver unit, the expected cache performance may not be executed in some cases. When negative values are set, receiver units should not perform look-ahead caching of the corresponding module.
- The priority value of modules where priority is not specified by “setCachePriority()” is interpreted as 30.
- The priority specified by “setCachePriority()” is valid until the data event that is currently presented is finished.
- The maximum quantity of modules that can be specified by “setCachePriority()” simultaneously is 64.
- In case priority specification is performed multiple times by “setCachePriority()” for the same module, then the latest setup of the priority is valid.

(3) Operation of “launchDocumentRestricted( )”

Refer to 5.14.6.9. “Operational control function”

(4) Operation of “quitDocument( )”

Refer to “5.14.6.9. Operational control function”

(5) Operation of “getFreeContentsMemory( )”

This function is assumed to be used in combinations with “lockModuleOnMemory” and “lockModuleOnMemoryEx”. Before executing “lockModuleOnMemory”, this function is used in order to acquire free space of contents memory of receiver units to find out whether or not the module that is about to be locked can be stored.

However, in accordance with the operation of (1) Operation of “lockModuleOnMemory( )” in section 5.12.6.9, receiver units do not include 1MB area reserved for carousel acquisition when calculating “getFreeContentMemory( )”.

The number of resource modules read by “lockModuleOnMemory( )” to be executed is assumed to be written in the “number_of_resource” argument. The maximum value that can be set to “number_of_resource” is 768 in accordance with “5.13.3 Operation of name spaces”.

“number_of_resource” is used as hint information when calculating the remaining of contents memory of receiver units, and depending on the design of the receiver unit, corresponding arguments may be ignored. The maximum value of data size of modules is specified as 1MB in section 2.3.3, however, the return value does not follow this restriction, and it is in accordance with free space for the above mentioned contents memory.

In case a compressed module is transmitted, both the compressed module and the deployed module will be accumulated in contents memory. Therefore, when a compressed module is received, the value that is smaller than the contents memory total subtracted by carousel size is returned as free space of contents memory.

In addition, when transmitting compressed modules, specifications in section 2.3.3 should be followed and the total size of compressed modules and deployed modules should be in the range that does not exceed the maximum module size.

Free space of contents memory, which is a return value, may change depending on the processing status of the BML browser from status of performing “getFreeContentsMemory” to status of performing “lockModuleOnMemory”, “lockModuleOnMemoryEx”. The variation of contents memory at this time depends on a model, therefore, this should be used as a reference value to inspect whether or not the targeted module can be locked by the return value acquired by the corresponding API. Following the same logic, when this function is called before the lock of module is completed, there is no guarantee that the same value as when there is no lock request is returned. Contents should take into consideration that even when there is the required volume of free space in the return value, the lock of module may fail in some cases.

(6) Operation of “lockModuleOnMemoryEx( )”, “unlockModuleOnMemoryEx( )”

Using “lockModuleOnMemoryEx( )”, modules that are transmitted by different components than the document that is currently presented can be locked. This function is operated as follows. (Refer to section 5.14.7 for the meaning of linked status and unlinked status.)

- ES that can be locked by “lockModuleOnMemoryEx()” is limited to ES’s with a component_tag value of 0x40, 0x50, 0x60. The ES that is currently presented can be locked by “lockModuleOnMemoryEx()” only when component_tag value of the ES that is currently presented is 0x40, 0x50, or 0x60.
- Modules of 0x40, 0x50, 0x60 and the module of the ES that is currently presented may be instructed to be locked simultaneously in some cases, and receiver units need to be able to acquire modules of 4 ES in parallel. The performance to acquire 4 ES modules in order is not permitted. This is excluding cases where the total for the module size acquired simultaneously exceeds 1MB.
- Lock may be released by ”lockModuleOnMemoryEx( )” in the following cases.
  - When explicitly released by execution of “unlockModuleOnMemoryEx( )”, “unlockAllModulesOnMemory( )”.
  - When selecting stations for the service that is currently presented ends.
  - When the data event that is currently presented ends. (Including the disappearance of component that is currently presented.)
  - When the entry component disappears by PMT update.
  - When transition to unlinked status is occurs.
  - When transition to entry component occurs by executing “quitDocument()”.
- Lock by lockModuleOnMemoryEx( ) will continue in the following cases.
  - When the data event is updated in a component that is not currently presented.
  - When the locked module version is updated.
  - When the ES that was transmitting the locked module becomes an empty carousel.
  - When transition to a document of a different component within the same service as the document that is currently presented is executed.
  - When transition to entry component has occurred by a pull flag.
  - When transition to linked status occurs.
- Receiver units should secure free space of 1MB to acquire modules from carousels at all times, and in case a memory area of 1MB cannot be secured by the execution of “lockModuleOnMemoryEx( )”, then this function shall not perform the fixation process of the module.
- Receiver units should secure free space specified by the argument of this function at all times and in case a memory area for the corresponding volume cannot be secured by the execution of “lockModuleOnMemoryEx( )”, then this function shall not perform the

fixation process of the module. The volume specified in the argument does not include 1MB area indicated in the previous clause. When the argument is omitted, then it is interpreted as 0MB.

- When the fact that module specified by “lockModuleOnMemoryEx()” does not exist becomes clear from the PMT (i.e.: when the ES transmitting the corresponding module is not included in the PMT), then “lockModuleOnMemoryEx()” will end with return value of -3.
- When the module specified by lockModuleOnMemoryEx() does not actually exist, and in case that it could not be judged from the PMT, (i.e.: ES does exist, but the module does not exist, etc.), then “lockModuleOnMemoryEx()” will finish with a return value of 1 and when the target module is judged as non-existence later, a “ModuleLocked” event is generated with status = -2.
- Even when the module locked by “lockModuleOnMemoryEx( )” is updated, receiver units do not re-acquire the module automatically.
- After “lockModuleOnMemoryEx()” is executed, if transition is executed to the document within the same service as the document that is currently presented before lock is completed, then the lock performance will continue. In this case, the “ModuleLocked” event is generated as a basic rule when the corresponding lock performance is completed, however, depending on the timing of document transition, it may not be generated in some cases.
- After “lockModuleOnMemoryEx()” is executed, if data event of the lock target ES is updated before lock is completed, then the “ModuleLocked” event is generated with a status = -1, and locking is not executed.
- After “lockModuleOnMemoryEx()” is executed, if the lock target module is updated before lock is completed, the receiver unit will lock the updated module.
- Lock by “lockModuleOnMemory( )” and lock by “lockModuleOnMemoryEx( )” can be used in combination, however,
  - The module locked by “lockModuleOnMemory( )” cannot be released by “unlockModuleOnMemoryEx( )” or the module locked by “lockModuleOnMemoryEx( )” cannot be released by “unlockModuleOnMemory( )”. (Function will return error.)
  - The module locked by “lockModuleOnMemory( )” cannot be locked again by “lockModuleOnMemoryEx( )” or the module locked by “lockModuleOnMemoryEx( )” cannot be locked again by “lockModuleOnMemory( )”. (Function will return error.)
- As a method to update presentation at the time of updating the module that the object is referring to, a procedure using “lockModuleOnMemory()” is written in ARIB STD-B24

Vol. 2 Appendix 1 ”6.6.2 Relationship between update and module lock”, however, the equivalent effect can be obtained by using “lockModuleOnMemoryEx()”.

- As a method to update presentation at the time of updating the binary table that the BML document is referring to, a procedure using “lockModuleOnMemory()” is written in ARIB STD-B24 Vol. 2 Appendix 2 ” 5.5.2.2 Behavior of binary table”, however, the equivalent effect can be obtained by using “lockModuleOnMemoryEx().”
- As a method to share JPEG images, DRCS, binary tables between documents, a procedure using “lockModuleOnMemory()” is written in ARIB STD-B24 Vol. 2 Appendix 2 “4.8.5.1 Application of remain attribute for each type”, “5.2.2 Sharing of external characters among documents”, “5.5.2.3 Inter-document sharing” respectively, however, the equivalent effect can be obtained by using “lockModuleOnMemoryEx().” It is written that sharing is only valid in the respective document group, however, under this specification, sharing is valid within the same contents group. In case of sharing beyond the ES, the sharing target should be locked by using “lockModuleOnMemoryEx()”. Performance in case it is not locked shall be receiver unit dependent.
- Under the status of receiving broadcasting, in case of executing a transition to a BML document of another ES, or in case of executing transition to broadcasting contents under the linked status, the existence of a module can be confirmed by the locking module including documents in the transition destination by using “lockModuleOnMemoryEx()” beforehand.
- In case the ES transmitting module locked by “lockModuleOnMemoryEx()” disappears, whether or not the lock of the module will be continued depends on a model. The lock should be released explicitly from contents for such modules. Transition to such modules is not possible.
- The module locked by “lockModuleOnMemoryEx()” can be unlocked by “unlockModuleOnMemoryEx()” regardless of whether or not the module is included in the carousel that is currently in transmission.
- As mentioned above, lock by “lockModuleOnMemory( )” and lock by “lockModuleOnMemoryEx( )” have different conditions for unlocking. A comparison of the two is shown in Table 5-11.

Table 5-11 Comparison of lockModuleOnMemory and lockModuleOnMemoryEx

	Module locked by "lockModuleOnMemory"	Module locked by "lockModuleOnMemoryEx"
Update of the data event in the component that is currently presented.	Lock release	Lock release
Update of the data event in the component that is not currently presented.	No changes	No changes
Update of the locked module version	No changes	No changes
Transition to another service than the document currently in presentation.	Lock release	Lock release
Transition to another component within the same service as the document currently in presentation	Lock release	No changes
Transition to entry component by pull flag	Lock release	No changes
Transition to entry component by "quitDocument()"(Including when "quitDocument()" is executed during the viewing of entry components)	Lock release	Lock release
Transition to linked status	Lock release	No changes
Transition to unlinked status	Lock release	Lock release

Operation Example 1

Typical operation example of “lockModuleOnMemoryEx( )” is shown in Figure 5-1. In this example, service is comprised of 2 components, which are component A and Component B. In the figure, the diamond shaped symbols rendered on the straight line which indicate the component are indicating the points where data events are updated. The following is explained in accordance with the figure.

(a) Component A is presented in the beginning. The module of component B is locked to time t1 by “lockModuleOnMemoryEx( ).”

(b) Transition of the module of component B locked in (a) to time t2. Lock of the module continues.

(c) Module of component A is locked by “lockModuleOnMemoryEx( )” in time t3.

(d) Transition of module of the component A locked in (c) to time t4. The lock on the two locked modules continues.

(e) Since the data event of component A that is currently in presentation will be updated in time t5, all the modules locked by “lockModuleOnMemoryEx( )” up to this point will be unlocked.

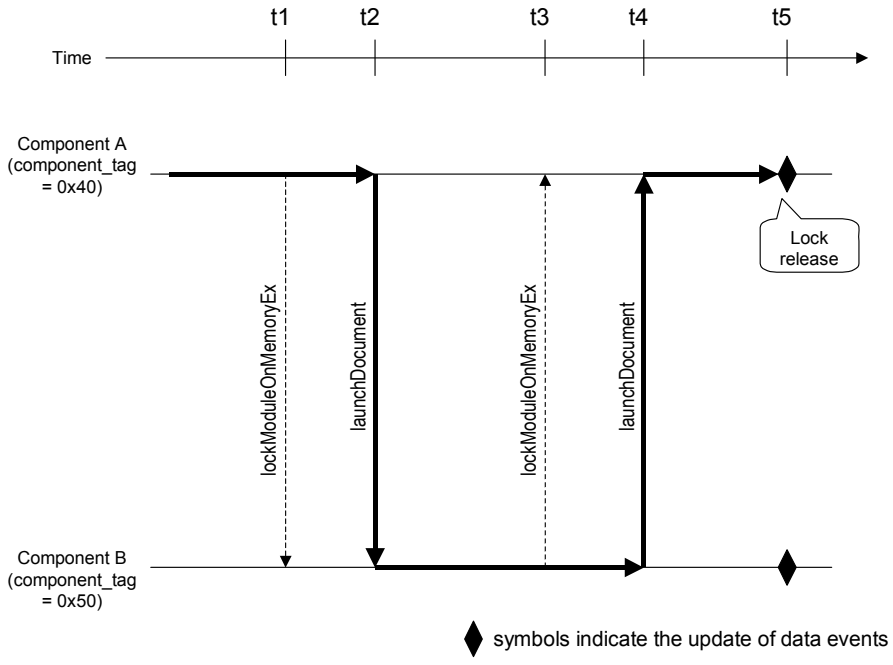


Figure 5-1 Operation example 1 of lockModuleOnMemoryEx

Operation example 2

A complicated operation example of “lockModuleOnMemoryEx( )” is indicated in Figure 5-2. The service is comprised of two components, which are component A and component B just the same way as Operation Example 1. The following is explained in accordance with the figure.

(a) Component A is presented in the beginning. The module that’s currently in presentation of component B is locked to time t1 by “lockModuleOnMemoryEx( ).”

(b) The module of component B is locked to time t2 by lockModuleOnMemoryEx( ).

(c) Transition to module of component B locked in (b). The locks on the two locked modules continue.

(d) Data event of component A is updated to time t4, but it does not have impact on modules that are already locked.

(e) Transition to module of component A that was previously presented to time t5. In component A, new data event is already transmitted, but document presented by this transition is document that was already transmitted at the time of locking in (a).

(f) Data event of component B is updated in time t6, however, it does not have an impact on modules that are already locked.

(g) Since the data event of component A that is currently presented is updated in time t7, locks on all modules locked by “lockModuleOnMemoryEx( )” up to this point will be unlocked.

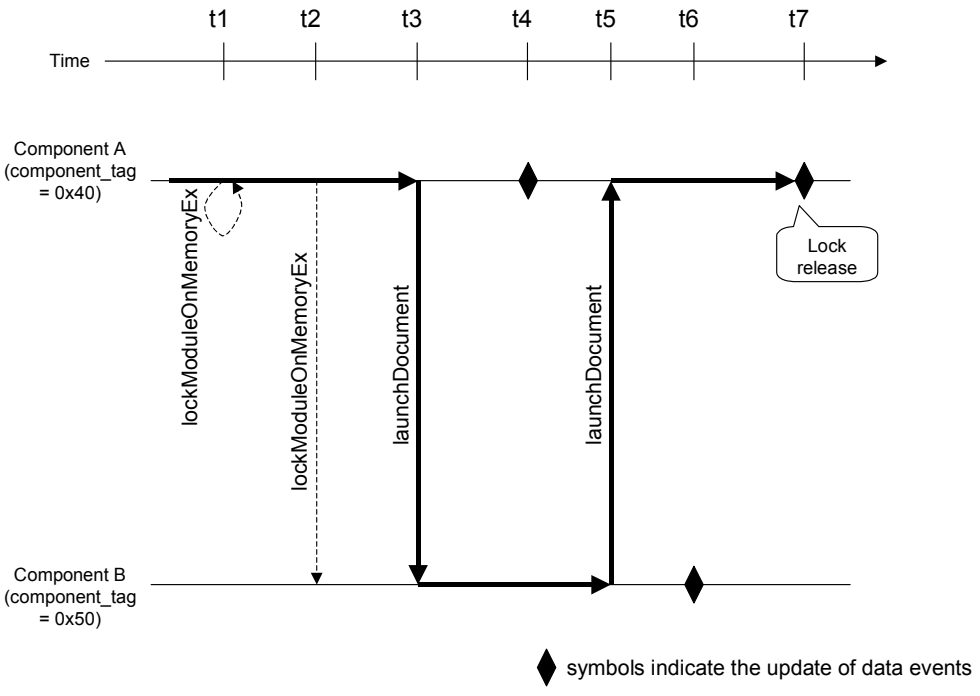


Figure 5-2 Operation example 2 of lockModuleOnMemoryEx

(7) Operation of “getIRDID( )”

When acquiring “cardID” as receiver unit unique identifier using “getIRDID()”, as specified in ARIB STD-B24, “CA_system_id” should be specified as type. The return value is a character string in hexadecimal. (“0x” is not attached to the front.) ReceiverID as hardware, MakerID used for downloads and the function to acquire ModelID are not operated.

(8) Operation of “getLockedModuleInfo( )”

Array[n][0](module name), a return value of the “getLockedModuleInfo” function is a String type and it is always stored in the following format.

/<component_tag>/<moduleName>

Return values, Array[n][1] and Array[n][2] are number types.

In case modules that are fixed or requesting to be fixed do not exist at all, array with 0 length is returned, but receiver can also return null.

For contents production, the fact that one of the return values is returned should be taken into consideration.

(9) Operation of the closed caption display control function

Should be in accordance with ARIB STD-B24 Vol. 2 Appendix 2 “5.6.4. Operation of subtitle presentation control”.

(10) Operation of “detectComponent( )”

Component specified as argument of “detectComponent( )” is limited to data components that are included in the service that is currently in presentation. When it is used in broadcasting contents, it should be in accordance with the specifications of section 5.13.3 and for URI specifying component, contracted forms starting with “/” shall be used. When it is used in communication contents, it should be in accordance with specifications in section 5.14.10.4, and for URI’s specifying URI, the absolute URI starting with "arib-dc://-1.-1.-1" shall be used.

(11) Operation of “getResidentAppVersion( )”

- The return value Array[0] shall be as defined in ARIB STD-B24.
- The return values Array[1]-[3] shall be appropriate values for the receiver as defined in ARIB STD-B24.
- The receiver shall return the details of receiver applications defined for each type of receiver application for return value Array[4]. The details for each receiver application are defined as follows. An empty string shall be returned if a receiver application not described below has been specified to argument appName.

(i) Operation of the receiver when "ReservedTransmission" has been specified to argument appName from the receiver applications described in additionalinfo when functionname in

Table 11-1 "Strings Available to Arguments of getBrowserSupport( )" in ARIB STD-B24 Vol. 2 Appendix 1 is “ResidentApp”

- For the return value Array[4], the following value shall be returned in String format.

  “active”: If the receiver has a reservation transmission function and it is enabled

  “inactive”: If the receiver has a reservation transmission function but it is disabled

  “none”: If the receiver does not have a reservation transmission function

(ii) Operation of the receiver when appName indicating the IPTV browser name as described in Appendix-14 is specified to argument appName

It is not possible to confirm using getBrowserSupport( ) whether the IPTV browser can support any IPTV functions specific to each service operator. However, it can be determined by the string of Array[4] obtained by getResidentAppVersion( ). The argument appName need to be specified appName indicating the IPTV browser name as described in Appendix-14.

- The return value Array[4] is String type and consists two kind of information. One is the item names describing the functions specific to each service operatior. Another is the string equivalent to UserAgent given by the IPTV browser. The return value shall be described in the following format:

  "Item name": "Item name": ...: "Item name" [: "UA=" "Character string equivalent to UserAgent"]

  - A separator ":" (a colon) shall be inserted between each item. ":" shall not be placed at the beginning of the first item.
  - The UserAgent string must always be allocated to the final field. (Allocation is arbitrary)
  - If UserAgent will be inserted, "UA=" shall always be added to the beginning of the character string.
  - Characters useable in item names and the character string equivalent to UserAgent shall be as defined in STD-B24 and TR-B14.
  - However, ":" and "UA=" shall be reserved words for item names.
  - Refer to Appendix-14 for the item names that can be described in Array[4].

  The content of the character string corresponding to UserAgent is implementation dependent and is therefore not defined here. However, it shall be desirable that the IPTV service operator or the vendor implementing the browser disclose the format of the character string to the broadcaster by some means.

It shall be desirable for item names to be names like "Service name_Function name" that allow the service names and functions provided by each service operator to be clearly identified.

Example: Dpa_Type1Service:acTvila_Full:▪▪▪▪▪▪:UA=Mozilla/4.0

(12) Operation of “isRootCertificateExisting ( )”

By confirming whether or not the relevant root certificate exists on the receiver in advance using this function, content can be developed in a way that viewers using a receiver on which the relevant root certificate is not stored will not be led to any content that require encrypted communication. root_certificate_id and root_certificate_version specified here shall be signed 32 bit values.

If 0 (general-purpose root certificate) has been specified as root_Certificate_Type, it shall return whether the specified root certificate exists in the general-purpose root certificate area.

In the event when 1 (operator-exclusive root certificate) has been specified as root_Certificate_Type, 1 shall be returned if the receiver has already acquired 1 or more operator-exclusive root certificates; and NaN shall be returned in all other cases.

(13) Operation of “startResidentApp( )”

- Combination of values that can be specified to arguments appName, showAV and Ex_info are as follows. Incorporation of each application shall be implementation dependent.

appName	showAV	Ex_info
HTMLBrowser	No restrictions	URI string to be passed to the HTML browser
ReservedTransmission	1 only	None
MailClient	No restrictions	4th argument: Title of the e-mail sent (corresponds to the Subject: header of the e-mail) 5th argument: Body text of the e-mail sent 6th argument: Destination address of the e-mail sent (corresponds to the To: header of the e-mail) 7th argument: Destination addresses of the copy of the e-mail sent (corresponds to the CC: header of the e-mail) These arguments shall be empty strings if not specified.
BookmarkList	1 only	None

- After the activated receiver application has been terminated, a process equivalent to station reselection for the service that executed this function shall be performed.

(14) Operation of “getBrowserSupport( )”

Values that can be specified as argument to getBrowserSupport() shall be added in the following combinations upon adding the values shown in Table 11-1 in ARIB STD-B24 Vol. 2 Appendix 1.

functionname	additionalinfo	Operation of getBrowserSupport()
ResidentApp	“JapaneseInput”	Returns 1 if the receiver implements a character input function (Refer to section 1.6) as one of its applications.
	A character string representing the browser as described in Table in Appendix-14 (2).	Returns 1 if the receiver implements the relevant browser.
Transmission Protocol	“datalink”, “PPP.modem”	Returns 1 if the browser implements a interactive communication function using PPP (i.e. PPP connection using a modem, not PPPoE).
ExtraBrowserFunction	A character string representing the browser function as described in Table in Appendix-14 (2).	Returns 1 if the receiver implements the relevant function.

Moreover, the following value shall be added to the extended function group specification (Refer to Table 11-2 in Appendix 1) to be specified as additionalinfo when functionname is "APIGroup".

API	Extended function group specification (additionalinfo argument)
getBookmarkInfo2()	Bookmark.Basic2
getBrowserStatus()	Ctrl.Status

(15) Operation of “startExtraBrowser( )”

Refer to Appendix 14 for values that can be specified to the browserName argument of startExtraBrowser().Refer to section 6.3 for the activation of the IPTV browser using this function.

5.12.6.10 Operation of bookmark function

Should be in accordance with ARIB STD-B24 Vol. 2 Appendix 1 “8.4 Guidelines on bookmarks” except for the additional following specifications.

(1) Operation of “startResidentBookmarkList()”

In case this function is called by receiver units where “startResidentBookmarkList()” is implemented and the bookmark list display function by receiver’s native application is not implemented, a NaN (failure) will be returned.

5.12.7 Embedded objects

- Time handled by data objects of ECMA Script is JST(UTC+9 hours) without adding the time offset of summer time. In case local time with considerations to summer time offset is needed to display the current time, etc., the local time is obtained by adding time offset by using “addDate()” for time obtained by “Date()” function in contents.
- Time handled in the date object during the presentation of communication contents should be in accordance with the time information that broadcasting contents were referring to immediately before the transition to communication contents.

5.12.8 Other specifications

Refer to ARIB STD-B24 Vol. 2 Appendix 3 ”5.8. Other restrictions”

5.13 Transmission of contents and name spaces

5.13.1 Mapping of scope to transmission systems

Should be in compliance with ARIB STD-B24 Vol. 2 Appendix 3 ”6.1. Scope mapping to transmission system”. Regarding the communication contents, refer to section 5.14.7.

5.13.2 Restrictions related to reference of mono-media, etc. over media

Should be in compliance with ARIB STD-B24 Vol. 2 Appendix 3 “6.2.2 Restriction on reference of mono-media and others across media types”. Regarding the communication contents, refer to section 5.14.9.

5.13.3 Operation of name spaces

Should be in compliance with ARIB STD-B24 Vol. 2 Appendix 3 ”6.3 Namespace”. Regarding the communication contents, refer to section 5.14.10.

Also, the following specifications are added for broadcasting contents.

- Reference to another service is only possible in the following extended functions for broadcasting.

  epgTune, epgIsReserved, epgReserve, epgCancelReservation,
  epgRecIsReserved,epgRecReserve, epgRecCancelReservation
  epgTuneToComponent

- Except for the following cases, contracted forms (Refer to ARIB STD-B24 Vol. 2 9.2)are used for the description of name spaces in case broadcasting contents are specified in BML documents.
  - Reference to another service.
  - Argument of the extended function for broadcasting with the event specification as the argument.
  - Reference of broadcasting contents from communication contents.
- Except for arguments to the extended function for broadcasting with the event specification as the argument, “event_id” is always omitted.
- Resource reference of mono-media, etc. referred from BML documents are operated as follows.
  - Resource transmitted by the same component as the document that is currently presented can be referred to at all times.
  - Regarding resources transmitted by different component than the document that is currently presented, it can be referred under the following regulations.
    - ✧ Resources referred to from BML documents should be locked by “lockModuleOnMemoryEx( )” before the presentation of the BML document. The performance in case of attempting to present BML documents that refer to unlocked resources depends on a model.
    - ✧ Performance in case of explicitly releasing resources that are referred from BML document that is currently presented depends on a model.
    - ✧ The resource that is required immediately after the start of BML presentation especially, such as CLUT and JPEG are referred to as the background-image of the body, since the lock by “lockModuleOnMemoryEx()” should be completed without failure before the presentation of the BML, some attention is required when creating the contents.
  - Regarding the reference of ECMAScript and CSS transmitted as independent resources, the following specifications are stipulated in particular.
    - ✧ In case of referring to ECMAScript and CSS from a different BML document component than the component by which they are transmitted, then the ECMAScript and CSS should be locked by “lockModuleOnMeoryEx( )” and transition to the corresponding BML document after the lock is completed should be performed. The performance in case document transition is performed without completing the lock depends on a model.
    - ✧ In case of referring to the ECMAScript and CSS from a different BML document component than the component by which they are transmitted, the locked CSS and ECMAScript cannot be referred to along with the disappearance of the ES, etc. in some cases. The above mentioned discordance can be prevented by

transmitting CSS data and ECMAScript data by entry component by the specifications of section 2.1.10.3.

- ✧ Even in case of referring to the CSS, ECMAScript transmitted by the component that is currently being viewed from the BML document, the ECMAScript and CSS should be locked before the transition of the corresponding BML document and transition should be performed after the completion of the lock.

- For the “href” attribute of “launchDocument( )”, “launchDocumentRestricted( )”, and “a” element, BML documents transmitted by components included in the same service as the document that is currently presented can be specified.
- The maximum number of resources (including both broadcasting contents and communication contents) that can be held in contents memory in receiver units by contents simultaneously is 768. As a method to realize this restriction, the total number of resources (resources with unique name spaces) within one date event period should be less than 768. However, in case the above mentioned restriction can be realized for certain on receiver units at the time of contents production, the total number of resources within a data event period can be more than 768, and this will not cause interference.

  However, in case the fixation of resources exceeding the above mentioned number is specified by “lockModuleOnMemory( )”, etc. against the above mentioned restriction, then receiver units may not perform this in some cases.

  The resources this section is referring to are the two resources below.
  - Resources that are directly mapped by modules.
  - Resources stored in modules in entity format of HTTP/1.1.
- The maximum number of modules that can be held in the contents memory of receiver units (including modules that are cached by specifications of “setCachePriority( )”) by contents simultaneously is 256.
- Modules where monitoring of module version update is possible is limited to the ES that is currently presented or ES’s with a component_tag value of 0x40,0x50,0x60.
- As indicated in ARIB STD-B24 Vol. 2 “9.2.5.1 Identification of currently selected broadcasting service on receiver”, in case "arib://-1.-1.-1" is specified as the service name, it is regarded as the service that is currently selected is specified. If during broadcasting reception, the service that is currently selected here refers to the broadcasting service that is currently being received, if during the playback of the partial TS, then it refers to the current service of the partial TS.
- In the BML document that performs transition by setting “remain” in the “remain” attribute of object element, in case the document in the originating point of transition and the document

in transition destination are included in different modules, then the specification of module number in the “data” attribute of the object element, which is the remain target, shall not be omitted; it should be specified. Also, in case the document in the transition originating point and the document in the transition destination are included in different components, then the component tag value in the “data” attribute of the object element, which is the remain target shall not be omitted; it should be specified.

5.13.4 Operation relating to communication contents

Refer to section 5.14.11, section 5.14.12, and section 5.14.13.

5.14 Operation of communication contents

5.14.1 Operation guidelines relating to the presentation of communication contents

It is the same as section ”5.12.1 Operation guidelines relating to presentation”.

5.14.2 Guidelines relating to the performance of external characters of communication contents

It is the same as section 5.12.2 “Operation guidelines relating to the performance of external characters”.

5.14.3 Performance of DOM of communication contents

It is the same as section 5.12.3 “Performance of the DOM”. However, there are operational restrictions in unlinked status. For details, refer to section 5.14.7.

5.14.4 Operation of ECMA script execution for the script language of communication contents

It is the same as section 5.12.4 ”Operation of script language”.

5.14.5 Expansion object for the broadcasting of communication contents

It is the same as section 5.12.5 “Expansion object for broadcasting”.

5.14.6 Operation of browser pseudo-objects of communication contents

The extended function for broadcasting that is equipped on browser pseudo-objects of communication contents may perform differently depending on the status (data broadcasting reception status, linked status or unlinked status) that the receiver unit is in. Regarding the basic idea of data broadcasting reception status, linked status or unlinked status, refer to section 5.14.7. In case the resource for broadcasting is referred to as an argument in unlinked status, it should be noted that performance of the extended function for broadcasting will fail. Table 5-16-48 shows the operation of substantial functions. The “O/X/Options” in the table has the following meanings, O- “performance will be successful”, X- “performance will fail”, Options

– “performance will be successful if optional functions are implemented.” For parts in browsers specified as “performance will be successful”, “performance will fail”, the receiver unit shall execute the performance specified in 5.14.12.2.

5.14.6.1 Ureg related functions

Operation of Ureg related functions in communication contents is indicated in Table 5-12. For the performance in data broadcasting reception status, refer to section 5.12.6.

Table 5-12 Performance of Ureg related functions in communication contents

	Linked status	Unlinked status
Ureg[ ]	O	X
Greg[ ]	O	O

5.14.6.2 EPG functions

Operation in communication contents of EPG functions is indicated in Table 5-13. For the performance of data broadcasting reception status, refer to section 5.12.6.

Table 5-13 Performance of EPG functions of communication contents

	Linked status	Unlinked status
epgGetEventStartTime( )	O	X
epgGetEventDuration( )	O	X
epgTune( )	O	O
epgTuneToComponent( )	O	O
epgIsReserved( )	O	O
epgReserve( )	O	O
epgCancelReservation( )	O	O
epgRecIsResreved( )	O	O
epgRecReserve( )	O	O
epgRecCancelReservation( )	O	O

5.14.6.3 Series reservation function

Operation of the series reservation function in communication contents is indicated in Table 5-14. For performance in data broadcasting reception status, refer to section 5.12.6. This function is optional. In case of using it, existence of the function should be confirmed by “getBrowserSupport( )” before use.

Table 5-14 Performance of the series reservation function of communication contents

	Linked status	Unlinked status
seriesIsReserved( )	Optional	Optional
seriesReserve( )	Optional	Optional
seriesCancelReservation( )	Optional	Optional
seriesRecIsReserved( )	Optional	Optional
seriesRecReserve( )	Optional	Optional
seriesRecCancelReservation( )	Optional	Optional

The performance in case it is operated as an option is the same as data broadcasting reception status.

5.14.6.4 NVRAM functions

Operation of the NVRAM functions in communication contents is indicated in Table 5.15. For performance in data broadcasting reception status, refer to section 5.12.6.

Table 5-15 Performance of the NVRAM functions in communication contents

	Linked status	Unlinked status
readPersisitentArray( )	O	X
wirtePersisitentArray( )	O	X

The only NVRAM area that can be used in linked status is the A-profile memory area of communication purpose for the specified broadcaster. Regarding name spaces, refer to section 5.2. In case other name spaces are specified, the performance of receiver units will fail.

5.14.6.5 Interaction channel function-Basic Procedure

Operation of the interaction channel function of basic procedure is shown in Table 5-16. Regarding this function, it is only applied to receiver units with equipped modems. In case of using it, the existence of the function should be confirmed by “getBrowserSupport( )” before use. For performance in data broadcasting reception status, refer to section 5.12.6.

Table 5-16 Performance of the interaction channel function of basic procedure of communication contents

	Data broadcasting reception status	Linked status	Unlinked status
connect()	O	X	X
disconnect()	O	X	X
sendTextData()	O	X	X
receiveTextData()	O	X	X

5.14.6.6 Interaction channel function-TCP/IP

Operation of the interaction channel function of TCP/IP is indicated in Table 5-17. The details of operation of each function are indicated after Table 5-17.

Table 5-17 Performance of TCP/IP Interaction channel function of communication contents

	Data broadcasting reception status	Linked status	Unlinked status
getConnectionType( )	O	O	O
isIPConnected( )	O	O	O
connectPPP( )	O	X	X

	Data broadcasting reception status	Linked status	Unlinked status
connectPPPWithISPParams ( )	O	O	X
disconnectPPP( )	O	O	X
setISPParams( )	O	X	X
getISPParams( )	O	X	X
sendTextMail( )	Optional	Optional	X
sendMIMEMail( )	Optional	Optional	X
transmitTextDataOverIP( )	O	O	O
setCacheResourceOverIP( )	Optional	Optional	Optional

- Function performance of “getConectionType( )”

  This can be performed in all situations.

  This function is used to obtain only the hint information in order to estimate the communication speed based on the information of the type of the connection lines.

  In ARIB STD-B24 Appendix 1, Informative Explanation 4, it is used as confirmation of different types of priority usage lines, however, the specifications of “getConnectionType( )” do not cover all types of connection lines, therefore, there are cases that do not fall into the sequences in Explanation 4, so keep in mind that this is used as nothing more than hint information.

  For details relating to the return value of the function, refer to ARIB STD-B24 Appendix 3 5.6.5.2. In environments where receiver units connect to a TCP/IP network using connection types that are not listed in Appendix 3, this function will return NaN.

- Function performance of “isIPConnected( ) “

  This can be performed in all situations.

  In this function, receiver units return values after judging whether or not the IP address is acquired or not, therefore, in case, for example, the receiver unit is connected to a dial-up router, then whether or not the router is connected to something else from there does not matter, the judgement is done based on whether or not an IP address is allocated (from DHCP, etc.) for the receiver unit.

  For details of operation of return value, refer to Appendix-12.

- Function performance of ”connectPPP( )”

  It functions only in data broadcasting reception status.

  Regarding phone number specification, name server specification specified in “connectPPP( )” should be in accordance with the specifications in ARIB STD-B24 Vol. 2 Appendix 3 5.6.5.4, 5.6.5.5.

  Empty strings cannot be specified in the character string specification of phone number of “conectPPP( )”. Significant phone number character strings should be specified.

This function is assuming to provide interaction channel communication to viewers without an environment to connect to ISP, for example, and is to be used as a method to realize free communication services related to broadcasting.

For the usage of this function, advance consent from the viewer in contents is assumed. For the startup of this function, the connection can be secured for certain by executing this function after disconnecting the communication by using “disconnectPPP” and the disconnect function in advance. However, consent from viewers should be obtained for disconnecting communication.

Contents should take the fact that there is already a connection set up in the receiver unit into consideration and confirm whether or not “getISPParams( )” is set up, and when ISP setup is done already in the receiver unit, “connectPPPWithISPParams( )” should be used.

In case connectPPP( ) is specified, even when ISP setup is done, the receiver units will connect parameters specified by “connectPPP( )” in higher priority.

In “connectPPP( )”, timeout can be set using the argument “idleTime” and this value is set in units of 100ms, and the range of value should be more than or equal to 60,000(one minute) and less than or equal to 1,200,000 (20 minutes). When there is no communication (no packets are sent from/to receiver units) during the setup timeout, the receiver units will generate a timeout and disconnect the communication. In such an event, “IPConnectionTerminated” event will be generated.

In case nameServer1 and nameServer2 of connectPPP() are both omitted, then the receiver units will use the name server specified by the dial-up server.

In receiver units with a constant connection environment (including PPPoE) such as Ethernet, and if this function is called by the “connectPPP()” function in the status where the connection is already established, the connecting destination should be switched based on the connection setup of the “connectPPP()” function.

In case a PPP connection is already established by automatic connection by the “connectPPPWithISPParams()” function or PPP, an error will be returned. In case a connection to the “connectPPP()” function is required, the connection should be disconnected once by the “disconnectPPP()” function and should be connected again by the “connectPPP()” function. In case the receiver unit does not have PPP functions, then an error will be returned based on the specifications in ARIB STD-B24 Appendix 3 5.6.5.2.

Connections established by “connectPPP( )” are disconnected by “disconnectPPP( )” or performance of selecting station or occurrence of a timeout.

For details on the operation of return values, refer to Appendix-12.

- Function performance of “connectPPPWithISPParams( )”

It is performed in data broadcasting reception status or linked status.

“connectPPPWithISPParams( )” is used to perform PPP connection using receiver unit setup in case the receiver unit is not connected by PPP. This function is used to connect from contents in case the return value of “isIPConnected( )” is 0. In case a PPP connection is established already based on each priority usage line types, then it will always fail based on the specifications in ARIB STD-B24, 7.6.7.6 and ARIB STD-B24 Appendix 3, 5.6.5.2. Also, when receiver units are not setup for each priority usage line type or do not have PPP functions, then there will be an error based on the same specifications.

The connections using “connectPPPWithISPParams( )” will be disconnected at the time of calling “disconnectPPP( )” and performance of selecting stations in normal performance, or in no communication status (status where not a single packet is sent or received) during the timeout time using time setup in receiver units.

At the time of a timeout occurrence, an “IPConnectionTerminated” event is generated.

For operation of return values, refer to Appendix-12.

- Function performance of “disconnectPPP( )”

It is performed in data broadcasting reception status and linked status.

For performance of disconnectPPP(), refer to the specifications in ARIB STD-B24 7.6.7.6, and 5.14.13. For details on the operation of return values, refer to Appendix-12.

- Function performance of setISPParams( )

It is performed only in data broadcasting reception status.

The following regulations are established for the function argument.

- First argument ispname is a maximum character string length of 64 digits (128 bytes).
- Empty string should be set up in case of omitting the 6th argument nameServer1.
- Empty string should be set up in case of omitting the 7th argument nameServer2.
- 11th argument status value is a broadcaster dependent operation using this function from an unconfigured status[1], and operational regulations are not established.
- Ethernet(401, 402, 403)should not be specified in the 12th argument lineType.

Regarding the details of function operations, refer to ARIB STD B24 Vol. 2 Appendix 3 section 5.6.6, and Vol. 6 section 7.3.1 “Transmission conditions of telephone numbers” and section 7.3.2 “Application functions”.

- Function performance of “getISPParams( )”

It is performed only in data broadcasting reception status.

The following regulations are established for this function.

- From the perspective of protecting personal information, contents should not send acquired information elements to the center.
- In case the return value array[4] is omitted, then empty strings should be returned.
- In case the return value array[5] is omitted, then empty strings should be returned.
- In case the receiver unit sets up the parameter, the return value array[9] should return status=2.
- The definitions of broadcaster identification information acquired in the return value array[10] are indicated in Table 5-18.

Table 5-18 The definitions of getISPParams( ) return value array[10]

Value(hexadecimal character string display)			Definitions
00	XXXX	XX	No setup status, or the setup at the time of deletion by receiver features.
8F	XXXX	XX	The setup at the time of setup by receiver features.
Other than the above			When it is set by a broadcaster that has original_network_id, broadcaster_id (note2) in the left.
FF	original_network_id	broadcaster_id	

Note 1 : ”X” means “don’t care”.

Note 2 : In case of digital terrestrial television broadcasting, it is fixed to FF.

- Function performance of “transmitTextDataOverIP( )”

It is performed in all situations.

This function is a function assuming that the sending and receiving of superimpose are performed in TCP/IP networks by BASIC procedure. Mainly it is used from broadcasting contents. The following restrictions are established for the argument of this function.

- There are two types of schemes that can be setup in the URI, which are "http" and "https".
- The maximum length of the character string that can be specified to argument text is 4096Bytes.
- The character code of sending text data is a fixed operation of EUC-JP only. Therefore, “EUC-JP” should be specified for the charset of the 3rd argument at all times. However, it should be noted that character codes such as external characters, etc. operated in the DRCS are not included in this sending text data.

When text is sent to servers, receiver units will send it using the POST method to the URI specified by the argument. When there is no text to be sent, then empty strings (“”) should be specified.

Receiver units should specify “application/x-www-form-urlencoded” of Content-Type in request message, "Denbun" should be specified as the “name” attribute. The encoding format of the text is in accordance with HTML4.01 “17.13.4 Form content types”. In such an event, Japanese character strings should be encoded as the character codes of EUC-JP. In case empty strings (“”) are specified, then character strings of "Denbun=" should be considered and specify 7 to the Content-Length.

The request message when sending texts “send denbun” to servers is indicated in Figure 5-3.

```
POST http://localhost/test.cgi HTTP/1.1
…header omitted…
Content-Type: application/x-www-form-urlencoded
Content-Length: 49

Denbun=%73%65%6e%64%20%64%65%6e%62%75%6e
```

Figure 5-3 An example of a request message issued by “transmitTextDataOverIP( )”

If there is text data to be passed to the receiver unit, the server, which requested the process, can attach text data on top of the response status. In such case, Content-Type is set as text/plane and EUC-JP should be specified in the Charset.

An example of a response message including text data "Reply texts" in the entity-body is indicated in Figure 5-4.

```
HTTP/1.1 200 OK
…header omitted…
Content-Length: 22
Content-Type: text/plain; Charset=EUC-JP

Reply texts
```

Figure 5-4 An example of response message received by “transmitTextDataOverIP( )”

The maximum size of the entity-body received by receiver units is 4096Bytes at this point. The receiving process of text in case text exceeding the maximum size is received by the receiver unit, it depends on a model. Character codes of text data is a fixed operation only in EUC-JP.

In case of unlinked status, a user interface to obtain permission from the viewer is presented by the receiver unit and consent from the viewer should be obtained before sending the text.

Failure performance of the user interface or in case permission could not be obtained from viewers is receiver unit dependent.

- Function performance of “setCacheResourceOverIP()”

“setCacheResouceOverIP” function sets the hint information of cacheable resources on the Internet stored in the Array of argument to receiver units. Cache performance of setup resources and the release performance depend on models. This function is performed in data broadcasting reception status, linked status and unlinked status. Even when cache performance of communication contents is specified by this function from contents in data broadcasting reception status or linked status, situations where the cache of communication contents is prevailed over the cache of broadcasting contents is not assumed by the service. Therefore, it is strongly recommended for receiver units to leave the cache of broadcasting contents by priority.

For receiver units performing communication by PPP connection, if this function is called in the status where a connection is not established, whether or not a PPP connection by automatic connection is used to get connected depends on a model. For using this function, contents should secure a PPP connection with “connectPPP” or “connectPPPWithISPParams” in advance.

Hint information is valid within the document. Accumulation of and usage of accumulation of hint information by history of document transition is not assumed as a service.

5.14.6.7 Interaction channel function/Function to acquire line connection status

The function performance of functions to acquire line connection status is indicated in Table 5-19. Regarding this function, it is only applied to receiver units with equipped modems. In case of using it, the existence of the function should be confirmed before use by “getBrowserSupport( )”.

Table 5-19 Performance of functions to acquire line connection status in communication contents

	Data broadcasting reception status	Linked status	Unlinked status
getPrefixNumber( )	O	X	X

The following regulations are established for the return value of getPrefixNumber( )

- Array[2] to Array[4] are operated for the return value.
- From the perspective of protecting personal information, contents should not send the return value as information to center.

For the details of function operation, refer to Vol. 6, section 7.3.1 “Transmission conditions of telephone numbers” and section 7.3.2 “Application functions”.

5.14.6.8 Interaction channel function-Mass calls reception service

The function performance of the mass calls reception service is indicated in Table 5-20. Regarding this function, it is only applied to receiver units with equipped modems. In case of using it, the existence of the function should be confirmed before use by "getBrowserSupport( )".

Table 5-20 Performance of mass calls reception service

	Data broadcasting reception status	Linked status	Unlinked status
vote( )	O	X	X

5.14.6.9 Operational control function

The performance function for communication contents of operational control functions is indicated in Table 5-21. The details of performance of each function are described in Table 5-21 and then on. Regarding the function performance of data broadcasting reception status, refer to section 5.12.6.

Table 5-21 Performance of the operational control function of communication contents

	Linked status	Unlinked status
reloadActiveDocument( )	O	O
getNPT ( )	O	X
getProgramRelativeTime( )	O	X
isBeingBroadcast( )	O	X
lockModuleOnMemory ( )	X	X
unlockModuleOnMemory( )	X	X
setCachePriority( )	X	X
getIRDID( )	X	X
getBrowserVersion( )	O	O
getProgramID( )	O	X
getActiveDocument( )	O	O
lockScreen( )	O	O
unlockScreen( )	O	O
getBrowserSupport( )	O	O
launchDocument( )	O	O
launchDocumentRestricted( )	O	O
quitDocument( )	O	O
launchExApp( )	Optional	Optional
getFreeContentsMemory( )	O	X
isSupportedMedia( )	O	X
detectComponent( )	O	X
lockModuleOnMemoryEx( )	O	X
unlockModuleOnMemoryEx( )	O	X
unlockAllModulesOnMemory( )	O	X
getLockedModuleInfo( )	O	X
getBrowserStatus( )	O	O
isRootCertificateExisting( )	O	O
getRootCertificateInfo( )	O	O

	Linked status	Unlinked status
getResidentAppVersion( )	O	O
startResidentApp( )	Optional	Optional
startExtraBrowser( )	Optional	X

- Function performance of reloadActiveDocument( )

It is performed in both linked status and unlinked status. When this function is called, the receiver unit should re-acquire the BML document and mono-media that are currently presented from the server. In case the cache within the receiver unit can be judged as the newest, then the display using the cache can be executed. In case of playing communication contents whose resource name cannot be specified such as when "/" is specified at the end of the URI or ?query is specified, the receiver unit will perform a re-acquisition of the communication contents that are currently presented by re-sending the URI specifying the corresponding BML document to the server.

For example, the performance in case the communication contents that had gone through transition in launchDocument("http://localhost/hoge/","cut") call "reloadActiveDocument" is the same as when the above mentioned "launchDocument" function is called.

- Function performance of getNPT( )

Performance in linked status is the same as data broadcasting reception status.

- Function performance of getProgramRelativeTime( )

Performance in linked status is the same as data broadcasting reception status.

- Function performance of isBeingBroadcast( )

Performance in linked status is the same as data broadcasting reception status.

- Function performance of getBrowserVersion( )

It is performed in both linked status and unlinked status. Performance is the same as data broadcasting reception status.

- Function performance of getProgramID( )

Performance in linked status is the same as data broadcasting reception status.

- Function performance of getActiveDocument( )

It is performed in both linked status and unlinked status. Return values are returned in the format where the scheme and host name are omitted and starts with / (abs_path specified in RFC1808).

In case "/" is specified at the end of URI, or "?query" is specified, or in case of playing communication contents whose resource name cannot be specified, the values excluding URI scheme that specified the corresponding BML document and host name should be returned.

For example, the return value when communication contents that had gone through transition in launchDocument("http://localhost/hoge/foo?query","cut") called

“getActiveDocument” function is ”/hoge/foo?query”.

- Function performance of lockScreen( )

It is performed in both linked status and unlinked status. Performance is the same as data broadcasting reception status.

- Function performance of unlockScreen( )

It is performed in both linked status and unlinked status. Performance is the same as data broadcasting reception status.

- Function performance of getBrowserSupport( )

It is performed in both linked status and unlinked status. Performance is the same as data broadcasting reception status.

- Function performance of launchDocument( )

It is performed upon stipulation of the following restrictions in linked status and unlinked status.

- Restrictions in linked status
  - In case of transition to broadcasting contents, an absolute URI starting with ”arib-dc:” should be specified. Refer to section 5.14.10 for details.
  - In case a BML document outside of base URI directory scope is specified, transition to unlinked status is executed. Refer to section 5.14.12 for details.
- Restrictions in unlinked status
  - Transition to a BML document of broadcasting contents cannot be executed. In case broadcasting contents are specified in unlinked status, the receiver unit will execute a failure performance.

- Function performance of launchDocumentRestricted( )

It is used to execute transition from data broadcasting reception status, linked status to unlinked status. For the details of operation, refer to section 5.14.12. In case this function is called in unlinked status, the performance will be equivalent to the performance of the “launchDocument( )” function. In case data broadcasting contents starting with “arib-dc” are specified in this function, then the failure performance, which is the same operation as re-selecting stations shall be performed.

- Function performance of quitDocument( )

It is performed in all situations. The “quitDocument( )” function will perform differently in data broadcasting reception status, linked status and unlinked status. Refer to Table 5-22. For details of the status transition, refer to section 5.14.12.

Table 5-22 Performance of quitDocument( )

Status	Performance of quitDocument( )
Data broadcasting reception status	The presentation of broadcasting contents that are currently presented ends and after executing 4 of “Basic operation of receiver when selecting channels” described in section 2.1.10.2, 6 and thereafter is executed, and transition to the startup document of the entry component is executed. At that time, all modules that are locked by the “lockModuleOnMemory” function and “lockModuleOnMemoryEx” functions are released. If connection by PPP is done at that time, the connection is disconnected in accordance with Figure 5-7 of 5.14.13.1.
Linked status	The presentation of broadcasting contents that are currently presented ends and after executing 4 of “Basic operation of receiver when selecting channels” described in section 2.1.10.2, 6 and thereafter is executed, and transition to the startup document of the entry component is executed. At that time, all modules that are locked by the “lockModuleOnMemory” function and “lockModuleOnMemoryEx” functions are released. If connection by PPP is done at that time, the connection is disconnected in accordance with Figure 5-7 of 5.14.13.1.
Unlinked status	The presentation of communication contents that are currently presented ends and the service that the communication contents had belonged to immediately beforeunlinked status is re-selected. At that time, if the connection by PPP has been executed, then the connection is disconnected in accordance with Figure 5-7 of 5.14.13.1.

- Function performance of launchExApp( )

It is performed in all situations. This function is optional. In case of usage, the existence of the function should be confirmed before use by “getBrowserSupport( )”. In receiver units that can use options, performance is the same as data broadcasting reception status.

- Function performance of getFreeContentsMemory( )

Performance in linked status is the same as data broadcasting reception status. Refer to section 5.12.6.

- Function performance of isSupportedMedia( )

Performance in linked status is the same as data broadcasting reception status. Refer to section 5.12.6.

- Function performance of “detectComponent( )”

Performance in linked status is the same as data broadcasting reception status. Refer to section 5.12.6.

- Function performance of “lockModuleOnMemoryEx( )”

Performance in linked status is the same as data broadcasting reception status. Refer to section 5.12.6. However, the receiver unit should release all modules secured by this function and the “lockModuleOnMemory( )” function at the time of transition to unlinked status.

- Function performance of unlockModuleOnMemoryEx( )

Performance in linked status is the same as data broadcasting reception status. Refer to section 5.12.6, section 5.13.3.

- Function performance of unlockAllModulesOnMemory( )

Performance in linked status is the same as data broadcasting reception status. Refer to section 5.12.6, section 5.13.3.

- Function performance of getLockedModuleInfo( )

Performance in linked status is the same as data broadcasting reception status. Refer to section 5.12.6, section 5.13.3.

- Function performance of getBrowserStatus( )

It is performed in both linked status and unlinked status. Performance is the same as data broadcasting reception status.

- Function performance of isRootCertificateExisting( )

It is performed in both linked status and unlinked status. Performance is the same as data broadcasting reception status.

- Function performance of getRootCertificateInfo ( )

It is performed in both linked status and unlinked status. Performance is the same as data broadcasting reception status.

- Function performance of getResidentAppVersion( )

It is performed in both linked status and unlinked status. Performance is the same as data broadcasting reception status.

- Operation of the startExtraBrowser() function

It shall only operate in linked status. The operation in linked status shall be the same as in the status while receiving data broadcasting. This function will fail if it is called in unlinked status.

5.14.6.10 Receiver unit audio control

The function performance of receiver unit audio control for communication contents is indicated in Table 5-23. Regarding the performance of data broadcasting reception status, refer to section 5.12.6.

Table 5-23 Performance of receiver unit audio control of communication contents

	Linked status	Unlinked status
playRomSound( )	O	O

It is performed in both linked status and unlinked status. Performance is the same as data broadcasting reception status.

5.14.6.11 Timer function

The function performance of the timer function in communication contents is indicated in Table 5-24. Regarding the performance of data broadcasting reception status, refer to section 5.12.6.

Table 5-24 Performance of the timer function in communication contents

	Linked status	Unlinked status
sleep ( )	O	O
setInterval( )	O	O
clearTimer( )	O	O
pauseTimer( )	O	O
resumeTimer( )	O	O
setCurrentDateMode( )	O	X

Excluding “setCurrentDateMode()”, the timer function is performed both in linked status and unlinked status. “setCurrentDateMode()” is performed only in linked status. Performance is the same as data broadcasting reception status.

5.14.6.12 External character function

The function performance of external characters in communication contents are indicated in Table 5-25. Regarding the operation of data broadcasting reception status, refer to section 5.12.6.

Table 5-25 Performance of the external character function in communication contents

	Linked status	Unlinked status
loadDRCS( )	O	O

The external character function is performed both in linked status and unlinked status. Performance is the same as data broadcasting reception status.

5.14.6.13 Other functions

The function performance of other functions in communication contents is indicated in Table 5-26. Regarding the operation of data broadcasting reception status, refer to section 5.12.6.

Table 5-26 Performance of other functions in communication contents

	Linked status	Unlinked status
random( )	O	O
subDate( )	O	O
addDate( )	O	O
formatNumber( )	O	O

Other functions are performed both in linked status and unlinked status. Performance is the same as data broadcasting reception status.

5.14.6.14 Closed caption display control function

The function performance of the closed caption control function in communication contents is indicated in Table 5-27. Regarding the performance of data broadcasting reception status, refer to section 5.12.6.

Table 5-27 Performance of the closed caption control function in communication contents

	Linked status	Unlinked status
setCCDisplayStatus( )	X	X
getCCDisplayStatus( )	X	X
getCCLanguageStatus( )	X	X

5.14.6.15 Bookmark control function

The function performance of the bookmark control function in communication contents is indicated in Table 5-28. Regarding the performance of data broadcasting reception status, refer to section 5.12.6.

Table 5-28 Performance of the bookmark control function in communication contents

	Linked status	Unlinked status
writeBookmarkArray ( )	X	X
readBookmarkArray( )	O	X
deleteBookmarkArray( )	X	X
lockBookmark( )	X	X
unlockBookmark( )	X	X
getBookmarkInfo( )	O	X
getBookmarkInfo2( )	O	X
startResidentBookmarkList( )	Optional	Optional

Performances of “readBookmarkArray( )”, “getBookmarkInfo( )” , “getBookmarkInfo2( )” are the same as data broadcasting reception status.

“startResidentBookmarkList()” is performed in both linked andunlinked status. Performance is the same as data broadcasting reception status. In case of using it, the existence of the function should be confirmed before use by “getBrowserSupport()”.

5.14.6.16 Print related functions

The performance function of print related functions in communication contents are indicated in Table 5-29. Regarding the performance of data broadcasting reception status, refer to section 5.12.6.

Table 5-29 Performance of print related function in communication contents

	Linked status	Unlinked status
getPrinterStatus()	Optional	Optional
printFile()	Optional	X
printTemplate()	Optional	X
printURI()	Optional	Optional
printStaticScreen()	Optional	Optional
saveImageToMemoryCard()	Optional	X
saveHttpServerImageToMemoryCard()	Optional	Optional
saveStaticScreenToMemoryCard()	Optional	Optional

5.14.7 Mapping the scope of communication contents

BML contents can also be located on servers, and receiver units can refer and present BML documents on servers (hereinafter referred to as communication contents).

There are 2 types of statuses in communication contents. One is linked status where broadcasting contents can be referred to and the other isunlinked status where communication contents cannot be referred to. For the reference relationship, refer to section 5.14.9.

Communication contents in the linked status act as one content of a content group. On the other hand, communication contents in unlinked status act as one content group. Communication contents in this status act as a special content group where data events do not exist.

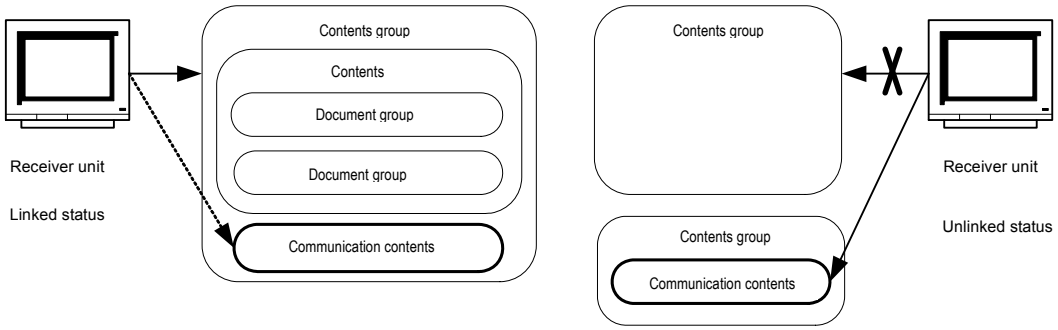


Figure 5-5 Linked status and unlinked status

.

- The restrictions of communication contents in linked status and unlinked status
- Extended functions for broadcasting that can be used are specifiedd. For operation, refer to section 5.14.6.
- Name spaces are different for BML documents for broadcasting and BML documents for communication. In case of specifying broadcasting contents from communication contents, a scheme with “arib:” or “arib-dc:” should be specified. In case of specifying communication contents from broadcasting contents, a scheme with “http:” or “https:” should be specified. It should be noted that handling of abbreviated forms is different as well. Regarding operation of name spaces in communication contents, refer to section 5.14.10. It is different from broadcasting contents and there is no concept of a startup document. However, it is possible for receiver units to specify the URI in a directory unit without directly specifying

the resource in the communication contents. In such an event, communication contents of specifications in accordance with the setup of servers are returned. (i.e.: index.bml, etc.)

- The restrictions of communication contents that are unique to linked status

- Since data events do not exist, the receivable event messages are limited only to the ones whose “event_message_group_id” is 1. In case the corresponding ID receives an event message with 0, then it is ignored. For operation of “event_message_group_id”, refer to section 2.3.4.
- In case pull back occurs during the playback of communication contents in linked status, the transition to entry component of broadcasting contents will be executed. If the receiver unit is connected to a TCP/IP network by a PPP connection using a modem, in such an event, the communication is disconnected.
- In case the transition to communication contents under the URI directory(hereinafter referred to as base URI directory), then a linked status is maintained, however, transition to unlinked status will be performed in case another URI is specified.*1 For the concept of base URI directory, refer to “Explanation of a base URI directory” in this section.
- Even in linked status, in case closed caption and superimpose are included in the broadcasting service that it belongs to, the presentation of closed caption and superimpose should be possible.

*1 In case the receiver unit receives a response in 300s and redirect is specified, and if the redirecting destination is under the base URI directory, then a linked status is maintained, however, if the redirecting destination is outside of the URI directory then transition to anunlinked status is executed.

- The restrictions of communication contents that are unique to unlinked status

- Among the extended functions for broadcasting that can use communication contents, since the communication contents in unlinked status is a service without a broadcasting stream, excluding EPG functions, the extended functions for broadcasting that refer to the broadcasting stream cannot be used. For the details of function performance being possible or impossible, refer to section 5.14.6.1.
- There are no restrictions on the URI specification of communication contents in unlinked status. However, the URI that specifies the reference of the broadcasting stream such as video, audio, closed caption, data carousel and event message, etc. cannot be specified. In case the receiver unit in this status is specified in the URI specifying the reference of those broadcasting streams, then the receiver unit will execute a failure performance. For details, refer to section 5.14.12.

- In unlinked status, receiver units should perform presentation, so that viewers can recognize that it is in unlinked status. For details, refer to section 1.8.1 “Display by browser features”.
- In unlinked status, the following regulations are established in addition to 5.7 “Operation of BML elements”.
  - “invisible” attribute of body element cannot be used.
  - “beitem” element cannot set values whose ”type” attribute is other than “TimerFired”, “MediaStopped”, “DataButtonPressed” and “IPConnectionTerminated”.

  In case it is used, receiver performance will depend on a model of receiver.
- In unlinked status, the following regulations are established in addition to 5.12.3 “Performance of the DOM”.
  - In object elements, “setMainAudioStream” method and “getMainAudioStream” method cannot be used.

  In case they are used, receiver unit performance will depend on a model of receiver units.
- In unlinked status, epgTune(), epgTuneToComponent() with specifications of "arib://-1.-1.-1" or "arib-dc://-1.-1.-1" as a transition destination service cannot be used. In case they are used, receiver unit performance will depend on a model.

- Explanation of a base URI directory

In linked status, a base URI directory is used as an identifier of document groups of communication contents. A base URI directory refers to the first host name and directory name that are specified by launchDocument( ) from broadcasting contents.

For example, the identifier that indicates the base URI directory of communication contents specified by “launchDocument”("http://localhost/hoge/index.bml", "cut") is "//localhost/hoge/".

The following specifications are stipulated for the judgement of base URI directories.

- Identifier indicating the base URI directory does not include a port number. For example, the base URI directory of communication contents specified by launchDocument("http://localhost:10080/hoge/index.bml", "cut") from the BML document of data broadcasting is ”//localhost/hoge/”. Therefore, even in the transition between different port numbers, if the base URI directory is matched, then a linked status will be maintained.
- The URL encoded name space should be handled as the equivalent to name spaces that are not encoded. For example, the base URI directory of ”http://localhost/%7Ehoge/index.bml” and http://localhost/~hoge/test.bml are handled assuming that they are matched.

・ Capital letters and lower case letters are not distinguished in host names. Capital letters and lower case letters are distinguished for directory names.

Resources directly under the base URI directory and resources stored under that directory are all regarded as within the document group of communication contents and a linked status will be maintained. In case the URI that does not match the base URI directory specified, then it will become an unlinked status. For details on the status transition, refer to section 5.14.12.

5.14.8 Operation guideline for transmission of communication contents

The transmission of communication contents is in accordance with HTTP/1.1 stipulated in RFC2616. Receiver units and servers execute communication by HTTP/1.1 on the port specified by the URI in case “http:” is specified in the URI, and in case “https:” is specified, then after establishing a connection by TLS1.0 and SSL3.0 on the port specified by the URI, encrypted communication by HTTP/1.1 is performed. In case the port number is not explicitly presented, then port number 80 in case of “http:” and port number 443 in case of “https:” are used as the default port numbers. However, handing ports might be different by firewall setup, etc. depending on the connection status. The connection status should be considered, and a default connection port can be set from the receiver unit. For operation details, refer to Vol. 6 and RFC2818.

This specification describes the minimum guidelines for operation and it does not by any means define the implementation range of receiver units.

- Operation guidelines of HTTP/1.1
  - Guidelines for the connection method of receiver units
    - Receiver units should satisfy the required conditions for constant connection and message transfer specified in RFC2616.
    - In case of receiving responses from servers, receiver units should be able to receive chunk format transfer encoding receiver specified in RFC2616.
    - In order to acquire the latest status of communication contents, connections based on the “expiration model” and “validation model” specified in RFC2616 should be performed.
  - For HTTP version, only HTTP/1.1 is operated. Operation of HTTP/1.0 by servers is not permitted as a basic rule. The performance process in case HTTP/1.0 is received by a receiver unit depends on a model.
  - Fixed length subset specified in RFC1123 is operated for the date/time format.

- For servers, only the date/time format of a fixed length subset specified in RFC1123 should be sent as the date format to the receiver units.
- The receiver unit should be able to interpret fixed length subsets specified in RFC1123 as the date/time format. In case dates in RFC1036 or ANSI C format are received, then the format of those dates should be interpreted, but they can be ignored as well.

  Examples are as follows:

  Sun, 06 Nov 1994 08:49:37 GMT ; RFC1123
  Sunday, 06-Nov-94 08:49:37 GMT ; RFC1036
  Sun Nov 6 08:49:37 1994 ; ANSI C

- As a character set, only “EUC-JP” is operated for all of request message/response message/entity. The performance in case the receiver unit receives another character set depends on a model.
- "Identity" is used for contents-coding. "deflate" is operated as an option. The performance in case “deflate” is received by receiver units that do not support “deflate”, or other values are received depends on a model.
- In case of specifying transfer-coding, “chunked” should be operated. The performance in case receiver units receive other values depends on a model.
- For product token, in case of specifying name of browser and version, it should be in alignment with the return value of “getBrowserVersion( )”.
- Operation of quality value depends on a model.
- "ja" is operated for the language tag. The performance in case other language tags are received depends on a model.
- The receiver shall implement "Keep-Alive".

- Operation of general headers
- Operation of Cache-Control

  Both request messages and response messages are operated.

  In services where end-to-end connection is required, receiver units should specify "no-cache" in the request message. In such an event, any type of field name attached to the no-cache identifier should not be included in the request message. For request messages, a proxy handling HTTP/1.0 on the transmission path should be considered and a Pragma general header should be located and “no-cache” should be specified.

  Servers should specify “no-cache” in the value of the Cache-Control header of the response message. In the same way as the request message, a Pragma general header can be located and “no-cache” can be specified, but this is not required. In case the receiver unit

receives a “no-store”, then the receiver unit should not cache the file specified by the HTTP session. The performance in case receiver units receive values other than “no-cache” or “no-store” depends on a model.

- Operation of Connection

  Both request messages and response messages are operated. “close” is operated for the connection option. In case of disconnecting constant connections, this header is given and “close” should be specified. The performance in case receiver units receive values other than “close” depends on a model.

- Operation of Date

  They are operated in the response message. Operation in request messages depends on a model.

- Operation of Pragma

  Both request messages and response messages are operated as options.

  Considering the fact that the proxy for HTTP/1.0 lies on the transmission path, receiver units should specify “no-cache” in the request header.

- Operation of Transfer-Encoding

  Response messages are operated. In case of responding using the chunk format transfer encoding, the server should be added to response header and “chunked” should be specified.

  It should be noted that Transfer-Encoding cannot be used simultaneously with Content-Length. For details, refer to RFC2616. The performance in case receiver units receive other values depends on a model.

- Other headers

  Regarding the operation of Upgrade, Trailer, Via and Warning, depends on a model.

- Operation of request headers
- Accept

  It is operated. Receiver units can specify the following media types in this header.

  text/plain, text/X-arib-bml, text/X-arib-jis8text,
  text/css, text/X-arib-ecmascript,image/jpeg, image/X-arib-png, image/X-arib-mng,
  audio/X-arib-mpeg2-aac, audio/X-arib-aiff,
  application/X-arib-bmlclut, application/X-arib-btable, application/X-arib-drcs

- Accept-Charset

  It is operated. In case of attaching this field, receiver units specify “EUC-JP” in this value.

- Accept-Encoding

  It is operated. Receiver units can only specify "identity" and "deflate" in this value.

- Accept-Language

  It is operated. Receiver units should specify “ja” in this value.

- Authorization

  It is operated. The presentation technique and specification of the user interface to input the user ID and password, etc. depends on a model of receiver units.

- Proxy-Authorization

  It is operated as an option. The presentation technique and specification of the user interface to input the user ID and password, etc. depends on a model of receiver units.

- Host

  It is operated.

- Operation of “If-Modified-Since”, “If-Unmodified-Since”

  It is operated as an option. When documents are reloaded by the extended function for broadcasting in communication contents, it can be used to judge whether or not the receiver unit should re-acquire the resources accumulated in Bcontents memory from the server. By using this header, receiver units can minimize the overhead of communication transactions.

- Referer

  It is operated as an option. By using this header, servers can use the URI information of the caller. The URI of broadcasting contents starting with ”arib-dc” can be set in the URI.

- User-Agent

  It is operated. Receiver units should be in alignment with the return values of ”getBrowserVersion( )”

- Other headers

The operation of “Expect”, “From”, “If-Match”, “If-None-Match”, “If-Range”, “Max-Forwards”, “Range” and “TE” depends on a model.

- Operation of response headers

• Location

It is operated.

• Proxy-Authenticate

It is operated as an option. The presentation technique and specification of the user interface for entering the user ID and password, etc. depend on models of the receiver unit.

• WWW-Authenticate

It is operated. The presentation technique and specification of the user interface for entering the user ID and password, etc. depend on models of the receiver unit.

• Other headers

The performance in case receiver units receive “Sever”, “Accept-Range”, “Age”, “ETag”, “Retry-After” and “Vary” depends on a model.

- Operation of entity-headers

• Content-Encoding

It is operated in the response message. For the value, "identity" can be specified. “deflate” is operated as an option. The performance in case receiver units that do not supporting “deflate” receives “deflate”, or in case receiver units receive other values, depends on a model.

• Content-Language

It is operated in the response message. “ja” can be specified as the value. The performance in case other values are received by receiver units depends on a model.

• Content-Length

It is operated both in the request message and response message. However, in case of using chunk format transfer encoding, it should be in accordance with RFC2616. The performance in case receiver units receive response messages that do not have this header attached depends on a model.

- Content-Location
  It is operated as an option in the response message.

- Content-Type
  Both request messages and response messages are operated.
  The media types that can be specified in a request message are as follows.

  application/x-www-form-urlencoded

  Media types that can be specified in response messages are as follows.

  text/plain, text/X-arib-bml, text/X-arib-jis8text,
  text/css, text/X-arib-ecmascript,
  image/jpeg, image/X-arib-png, image/X-arib-mng,
  audio/X-arib-mpeg2-aac, audio/X-arib-aiff,
  application/X-arib-bmlclut, application/X-arib-btable, application/X-arib-drcs

  The performance in case receiver units receive media types other than the above depends on a model.

- Last-Modified
  It is operated as an option. Receiver units can be used as the hint information of “If-Modified-Since” and “If-Unmodified-Since”. Unless there are special reasons, servers should send out this header.

- Other headers
  The performance in case receiver units receive “Allow”, “Content-MD5”, “Content-Range”, “Expire” depends on a model.

- Operation of methods
- Operation of “GET”
  It is operated.

- Operation of “POST”
  It is operated.

- Operation of “CONNECT”

  It is operated as an option. It is used to request tunnel performance in order to operate TLS1.0 and SSL3.0. In such an event, handling of multistage proxies should be considered. For the details of the migration process from TLS1.0 and SSL3.0 by HTTP/1.1, refer to RFC2817. Also, for details relating to the operation of TSL1.0 and SSL3.0, refer to Vol. 6.

- Other methods

  The operation of “OPTIONS”, “HEAD”, “PUT”, “DELETE”, “TRACE” depends on a model.

5.14.9 Restrictions related to mono-media reference from communication contents

5.14.9.1 Reference from broadcasting contents to communication contents

Reference of communication contents from broadcasting contents is not executed in accordance with ARIB STD-B24 Appendix 3 6.2 “Guideline on reference across media types”. Only the transition of BML documents from broadcasting contents to communication contents is possible.

5.14.9.2 Reference from communication contents to broadcasting contents

Occurrence of reference from communication contents to broadcasting contents is limited to when it is in linked status specified in section 5.14.7.

- When stream is referred to, the stream of video, audio and events within the same contents group (the same service) can be referred to. The ES in other services cannot be referred to. The restrictions of the ES that can be referred to simultaneously are in accordance with restrictions of broadcasting contents.
- In case of referring to mono-media of JPEG, PNG, binary tables, etc., if the specifications in section 5.14.7 are followed, then communication contents can be interpreted as it is in a different ES than broadcasting, therefore, only the resources within the module of the ES locked by using “lockModuleOnMemoryEx( )” can be referred to. For reference of other ES’s locked by “lockModuleOnMemoryEx( )”, refer to 5.13.3.
- The CSS style sheet file that can be specified by link elements sent in broadcasting contents and the ECMA Script file that can be specified by script elements are outside of the reference target from communication contents.

5.14.10 Name space of communication contents

5.14.10.1 Restricted matters relating to URI

Operation of name spaces should be in accordance with operation of ARIB STD-B24 ver3.2. Appendix 3 6.3. “Namespace”. However, when presenting BML communication contents

acquired from servers, if the BML document is in accordance with the specifications of ARIB STD B24 Appendix 2, the following restrictions are established in the name space.

- Multi-byte characters such as Japanese are not used in the URL.
- ”href” attribute of “a” element, and arguments of “launchDocument( )” and the URI that can be specified as arguments of “launchDocumentRestricted( )” should be a BML document.

5.14.10.2 Operation of name spaces for servers

If “/” is specified at the end of the URI at the time of acquiring contents from servers, then the servers should return the BML documents specified in accordance with the setup.

(Example : browser.launchDocument("http://localhost/hoge/", "cut");)

Receiver unit performance in case BML documents are not returned from servers depends on a model.

- In case an abbreviated URI format (relative URI) is used, then it is the same scheme as displayed contents, and it is handled as the relative format from the directory name where the displayed contents exist.
  (Example: In case the URI of the BML document is http://localhost/test/tmp/index.bml and if “../hoge.bml” is written within the document, then the URI will be http://localhost/test/hoge.bml.)

5.14.10.3 Operation of name spaces when referring to communication contents from broadcasting contents

Reference of communication contents from broadcasting contents is not performed by the specifications of section 5.14.9. When transition from broadcasting contents to communication contents is executed, then the absolute URI starting with “http:” and “https:” are used.

5.14.10.4 Operation of name spaces when referring to broadcasting contents from communication contents

- When referring to stream of broadcasting contents from communication contents, only the absolute URI starting with “arib://-1.-1.-1” is used for specification. The performance in case URI in any other format is specified depends on a model. Also, for the specification method of the “component_tag” value, section 2.1.2.8 should be followed.

  (Example: <object id="v" type="video/X-arib-mpeg2" data="arib://-1.-1.-1/-1" />)
- When referring to mono-media of broadcasting contents (not stream) from communication contents, only the absolute URI starting with “arib-dc://-1.-1.-1” should be used at all times.
  (Example: object id="i" type="image/jpeg" data="arib-dc://-1.-1.-1/40/0000/image.jpg" />)

- When performing transition from communication contents to BML documents of broadcasting, only the absolute URI starting with “arib-dc://-1.-1.-1” should be used at all times.
- When specifying “es_ref “ or “module_ref “ of the “beitem” element from communication contents, only the absolute URI starting with “arib://-1.-1.-1” or “arib-dc://-1.-1.-1” should be used.
  (Example: <beitem id="em" type="EventMessageFired" es_ref="arib-dc://-1.-1.-1/54" ‥ etc‥/>)
- In communication contents, the “es_ref” or “module_ref” of the “beitem” element should be specified. In case of specification, only the absolute URI starting with “arib://-1.-1.-1” or “arib-dc://-1.-1.-1” should be used.
  (Example: <beitem id="em" type="EventMessageFired" es_ref="arib-dc://-1.-1.-1/54" ‥ etc‥/>)
- In case “type” attribute of the object element is a character string starting with “video”, then the performance when anything other than ”arib:” is specified in the scheme depends on a model.
- When the “remain” attribute of the object element is set as “remain” and transition is executed from broadcasting contents to communication contents, then the absolute URI starting with either "arib://-1.-1.-1" or "arib-dc://-1.-1.-1" should be used for the object element that is the “remain” target of communication contents in the transition destination.
- When the ”remain” attribute of the object element is set as “remain” and in case of transition from broadcasting contents to communication contents, if the abbreviated form “/-1” is set for the AV stream in broadcasting contents, then the value of “data” attribute obtained in the DOM interface of the corresponding object element in communication contents after transition should return "arib://-1.-1.-1/-1".
- When the “remain” attribute of the object element is set as “remain” and in case of transition from broadcasting contents to communication contents, if, for example, the abbreviated form of "/50/0000/a.jpg" is set for still images, then the value of the “data” attribute obtained in the DOM interface of the corresponding object element in communication contents after transition should return "arib-dc://-1.-1.-1/50/0000/a.jpg".
- In case of transition from linked status to unlinked status, or transition from data broadcasting reception status to unlinked status, the “remain” attribute cannot be specified. For details, refer to section 5.14.12.
- JPEG images that exist in servers starting with “http:” or ”https” cannot set “remain” in the “remain” attribute of object elements.

- It is possible to limit the reference to services other than the service that is currently referred to by the following extended function for broadcasting. For arguments that can be used for functions at this time, they operate the same way as data broadcasting reception status. (Refer to 5.13.3)
  epgTune, epgIsReserved, epgReserve, epgCancelReservation,
  epgRecIsReserved,epgRecReserve, epgRecCancelReservation
  epgTuneToComponent

5.14.11 Guidelines of communication contents operation

5.14.11.1 Precautions during the operation of communication contents

Since the service provider providing communication contents have the attributes of TCP/IP networks which are employed for the distribution of communication contents, the fact that it is extremely difficult to present BML contents in the same quality as data broadcasting should be taken into consideration. The characteristics of TCP/IP networks are listed below.

- There is no guarantee that the transmission path spectrum is constant.
- Depending on the viewer's environment, the transmission path spectrum that can be used is different.
- Since the transmission path can take various shapes, a uniformed response to viewers cannot be guaranteed.
- Depending on the characteristics of the en route proxy, data transmission may be blocked.

Furthermore, by putting an excessive load on servers, the response may be lowered greatly. By those factors, the fact that the presentation of communication contents distributed by TCP/IP networks will not be done in uniformed manner to all viewers, and the presentation of receiver units may not be done in proper way, depending on the status of the network, should be taken into consideration.

Because of those factors, BML contents that fit the following descriptions should be distributed by data broadcasting instead of via communication.

- BML contents whose performance timing is important such as synchronization of video and audio.
- BML contents for which the proper presentation of all elements is important.
- BML contents that use a lot of mono-media such as still images.

5.14.11.2 Guidelines of communication contents descriptions

Precautions for writing communication contents are as follows.

- Communication contents are written with profiles in accordance with operation of section 5.12.
- It should be noted that name spaces are different for broadcasting contents and communication contents. Refer to section 5.14.10 for details.
- For communication contents assuming operation in linked status, it is strongly recommended that only HD pixel size be operated. In case of operating communication contents of SD pixel size in communication contents in linked status, if it is linked improperly from the data broadcasting contents of HD video, then it will be a combination of data in SD pixel size and video in HD pixel size, which may not be played by receiver units. In case of operating communication contents of SD pixel size in linked status, it should be operated with meticulous care, however, it should not be operated as a basic rule.
- For communication contents assuming operation in linked status, the specification of port numbers should not be performed.
- In case of specifying communication contents from directory specification, ”/” should be attached at the end. (Example: ”http://localhost” X: incorrect, http://localhost/ O: correct)
- Since the performance in case #fragment is attached to the URL specifying communication contents depends on a model, specification using #fragment should not be performed. (Example: ”http://localhost/test.bml#fragment X: incorrect)
- BML documents composing communication contents, binary tables, src attributes of “script” elements that are associated in the “ECMAScript” file, character code of the CSS file associated in the “link” element should be all written in EUC-JP.
- “BML version” written in communication contents should be 3.0, which is the same for broadcasting contents.
  (Example: <?bml bml-version =”3.0” ?>)
- In order to distinguish communication contents in linked status and unlinked status, extensions of the BML document composing communication contents should be:

  For BML documents assuming linked status: lbml(example: index.lbml)

  For BML documents assuming unlinked status: ibml(example: index.ibml)

  Receiver units should not judge linked status or unlinked status by this extension. Performance in accordance with status transition indicated in section 5.14.12.1 should be performed.
- By locating mono-media composing one screen within the same directory, the path searching process of receiver units can be decreased.
- The URI that can be specified in the “href” attribute of “a“ element is a BML document only.

- In communication contents, there should be either a link set up to another BML document or writing of a method to finish. This is because when receiver units encounter BML documents without links, the receiver units have no methods for transition to another status other than selecting stations.
- Resources in communication contents (mono-media such as JPEG or PNG) can be specified independently from the scope of the base URI directory. In case mono-media is linked independently from the scope of the base URI directory in linked status, then a linked status will be maintained.
- In case of locking modules with the lockModuleOnMemoryEx( ) function in linked status, attention needs to paid to whether or not the specified module exists for certain or the module about to be locked is intentional.
- Since the functions that can be used in linked status and unlinked status are different, the performances are different even for the same contents depending on the differences in status. Therefore, regarding the contents that are written assuming only linked status, it should be noted that the presentation in unlinked status is not guaranteed.
- In case of operating bookmark contents, the cases where it is started from the bookmark list function by receiver features (*1) should be considered and bookmark contents need be written limited to the extended function for broadcasting, which can be used in unlinked status. In case of writing bookmark contents with functions that can be used in linked status, viewers will not be able to reach the corresponding bookmark contents due to the failure performance of receiver units by calling the function of linked status in unlinked status.
- In case of specifying resources in communication contents in linked status, the names in abbreviated forms specified in ARIB STD-B24 Vol. 2 9.2.1.3 cannot be used. In case of referring to resources transmitted by broadcasting, the absolute URI starting with either “arib://-1.-1.-1” or “arib-dc://-1.-1.-1” should be used.
- Considering that the area appointed for module acquisition in the module is 1MByte, the total volume of resources in the communication contents that are referred to within the BML documents at one time should not exceed 1MByte.
- If descriptions using overwriting of the data attribute are done to perform switching of mono-media such as button shapes, then the receiver units will access the web server each time and the performance response may greatly be worsened. For the switching of shapes, if it can be realized by overwriting ”visibility”, then worsening of response can be avoided.
- For within document scope (during presentation of one BML document), the timing for receiver units to follow the update of resources on the web server depends on a model (the newest resource is acquired and implemented in the presentation). Therefore, if the

receiver unit wants to guarantee the acquisition of the newest resources, then along with document transition, consider the possibility that the corresponding resource has been referred to before and has been acquired already, Cache-Control: no-cache should be attached and transmitted for the corresponding resources.

- TLS1.0 / SSL3.0 shall be used when handling personal information in Interactive Data Broadcasting services and as well as communication content in linked status.

(*1) Refer to transition H of section 5.14.12.1.

5.14.12 Guidelines of receiver unit performance when receiving communication contents

5.14.12.1 Guidelines of the receiver unit status transition performance

Receiver units can take the following 3 statuses for receiving BML contents.

(1) “Data broadcasting reception status” which is receiving and playing data broadcasting and video/audio
(2) “Linked status” which is receiving and playing both communication contents and video/audio
(3) “Unlinked status” which is receiving and playing only the communication contents

In order for viewers to be able to distinguish linked statusunlinked status, receiver units should perform the display specified in 1.8.1 in unlinked status.

Regarding linked status and unlinked status, refer to section 5.14.7. Receiver units perform transitions between those 3 statuses upon instructions from contents or instructions from users. The structure of status transition is indicated in Figure 5-6.

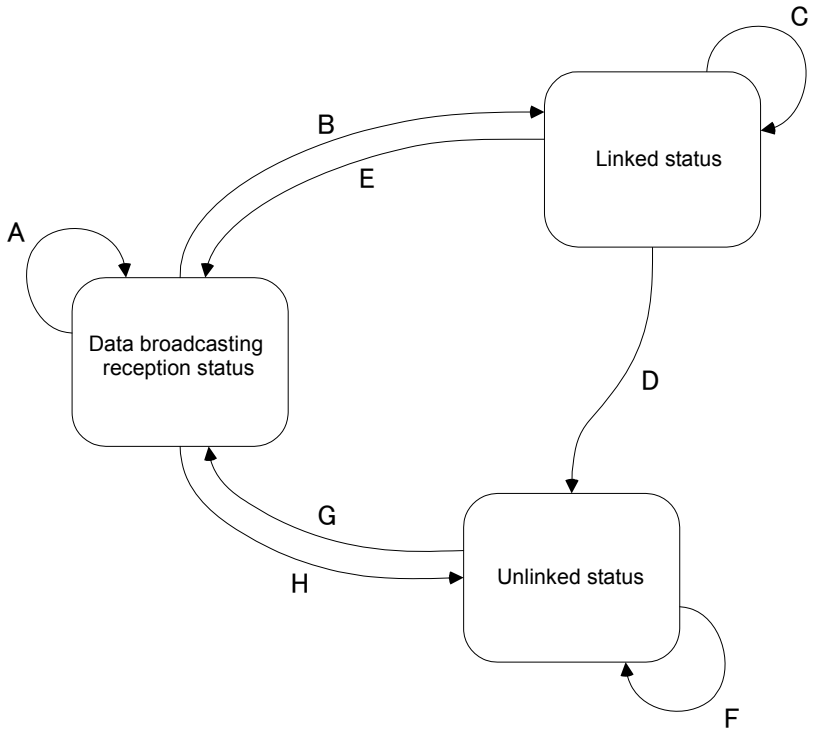


Figure 5-6 Status transition of receiver units when receiving communication contents

- Explanation of transition A (Data broadcasting reception status→Data broadcasting reception status)

Transition A is a transition from normal data broadcasting to data broadcasting. For the operation details, refer to Chapter 2.

- Explanation of transition B (Data broadcasting reception status→Linked status)

Transition to linked status from data broadcasting reception status is possible by “launchDocument( )” or “a” element. There are two types of schemes of the URI to specify communication contents, which are ”http:” and ”https:”. For the details of the name spaces of communication contents, refer to section 5.14.10. (Example: browser.launchDocument("http://localhost/hoge/index.bml", "cut");)

Communication contents can specify the directory without specifying resources. In case a directory is specified, servers will return specified communication contents in accordance with the setup. (For example, index.bml, etc.) In case of specifying directory, “/” at the end should be attached as an indication of directory.
(Example: browser.launchDocument("http://localhost/hoge/", "cut");)

- Explanation of transition C (Linked status→Linked status)

  The scope of the document group of communication contents in linked status will make the directories under the URI host address and directory (hereinafter referred to as base URI directory) the target area. It does not depend on scheme (protocol) specification.
  (Example: //localhost/hoge/)

  For example, in case of transition from "http://localhost/hoge/index.bml" to “http://localhost/hoge/foo/test.bml”, the linked status will be maintained. In case of transition to "https://localhost/hoge/foo/test.bml", the linked status will be maintained as well.

- Explanation of transition D (Linked status→Unlinked status)

  In case the transition of the base URI directory to a BML document of a different URI, for example "//localhost/hoge/", is performed, then the receiver unit should be in anunlinked status. For example, in case of transition to "http://otherhost/startup.bml" or "http://localhost/test/startup.bml", the receiver unit will be in anunlinked status.

  In case of transition to anunlinked status, the receiver unit should release all modules locked by “lockModuleOnMemoryEx( )”.

  In case a transition to anunlinked status is desired even though the transition destination is within the base URI directory scope, “launchDocumentRestricted( )” is used. By using this function explicitly, the reference to broadcasting contents in the directory within the base URI directory scope can be terminated. *1
  (Example: browser.launchDocumentRestricted("http://localhost/hoge/another.bml", "cut");)

- Explanation of transition E (Linked status→Data broadcasting reception status)

  Transition from a linked status to a data broadcasting reception status is possible by using “launchDocument( )”, “a” element or “epgTune()”, “epgTuneToComponent()”. Before the transition, contents should lock the BML documents in the transition destination by ”lockModuleOnMemoryEx( )” to confirm the existence of documents in the transition destination to perform the transition safely. The performance in case BML documents do not exist in the transition destination depends on a model.

  "arib-dc:" should be specified for the URI scheme to specify the broadcasting contents. In case of transition using “launchDocument”, it should be noted that only the current service such as "arib-dc://-1.-1.-1/50/0000/startup.bml" can be specified. In case of “epgTune( )”, “epgTuneToComponent( )”, "arib:" or ”arib-dc:” should be used in the URI scheme to specify the broadcasting contents, and in case of using those functions, services other than the current service can be specified.

(Example: borwser.launchDocument("arib-dc://-1.-1.-1/50/0000/startup.bml","cut");)

By using “quitDocument( )”, transition from a linked status to the data broadcasting reception status is possible. In such an event, transition of the receiver units to the entry component of broadcasting service that the receiver unit currently belongs to is executed.

The transition from linked status to data broadcasting reception status can be generated by switching the performance of the channel (service) by user. In such an event, if the receiver unit is performing PPP connection using a modem, the receiver unit will disconnect the connection in accordance with Figure 5.7 in 5.14.13.1. In such an event, performance to select stations should be performed after presentation of a message to notify disconnection, and the connection should be disconnected upon the approval of the viewer. *2

- Explanation of transition F (Unlinked status→Unlinked status)
  The concept of a base URI directory does not exist in theunlinked status and BML documents that exist in name spaces of ”http:” and ”https:” can be specified freely with the “launchDocument( )” argument or the “href” attribute of the “a” element.

For details of operational restrictions on the extended function for broadcasting in this status, refer to section 5.12.6.

As it is indicated in the transition figure, a transition of receiver units from anunlinked status to linked status is not possible.

- Explanation of transition G (Unlinked status→Data broadcasting reception status)
  The transition from an unlinked status to a data broadcasting reception status can be performed by using ”quitDocument( )”, “epgTune( )” and “epgTuneToComponent( )”. When “quitDocument( )” is specified in theunlinked status, the receiver unit will re-select the station in the broadcasting service that was referred to until immediately before the transition to unlinked status. Designers of receiver units should keep in mind that the performance of “quitDocument( )” is different from transition E.

  Transition from an unlinked status to a data broadcasting reception status can be generated by the switching performance of channels (service) by the user. In such an event, performance to select stations should be performed after presentation of a message to notify disconnection, and connection should be disconnected upon approval of the viewer. *2

- Explanation of transition H (Data broadcasting reception status→Unlinked status)
  In case of transition from a data broadcasting reception status to an unlinked status,

“launchDocumentRestricted( )” is used. By using this function explicitly, receiver units can be switched to anunlinked status at the time of transition from broadcasting contents to communications and the reference to broadcasting contents can be terminated. *1

(Example: browser.launchDocumentRestricted("http://localhost/hoge/index.bml", "cut");)

Even when receiver units with the function to specify the URI of communication contents directly refers to communication contents directly, transition to anunlinked status should be executed.

In case of transition to anunlinked status, receiver units should release all modules locked by “lockModuleOnMemoryEx( )”.

*1 Contents producers should keep in mind that in case of transition from a data broadcasting receptions stauts or a linked status to an unlinked status, the presentation of video/audio/still images is terminated. In other words, the “remain” attribute cannot be used during this transition. Receiver units should delete the still image plane at the time of transition and should guarantee that the mixed display of data broadcasting contents does not happen to contents in anunlinked status.

*2 Contents producers should keep in mind that receiver units may not present the message of disconnection in some cases for cases other than channel change operation. In case of disconnecting, when elements associated with “launchDocument( )” or “quitDocument( )” on the contents are activated, then a message of disconnection should be presented.

In receiver units with the bookmark list function by receiver features, even when transition to bookmark contents by communication contents is performed, transition of the receiver unit to anunlinked status should be performed.

5.14.12.2 Guidelines for when receiver unit performance fails

In case functions that cannot be used in status specified in 5.14.12.1 are used or broadcasting contents are referred in anunlinked status, receiver units will fail.

Receiver units will perform the following processes for the parts specified as “performance failure” or “failure performance” of browsers in 5.14.6, etc.

(1) Browser is finished.
(2) Along with it, in case the connection to the TCP/IP network was made by a PPP connection using a modem, it should be disconnected.
(3) In case it is in linked status, 4 and after of “Basic operation of receiver when selecting channels” described in section 2.1.10.2 are executed.
(4) In case it is in unlinked status, re-selecting of the service that it belonged to until immediately before the transition to unlinked status is executed.

5.14.12.3 Guidelines of receiver unit performance when receiving error responses

In case the receiver unit requests communication contents and resources composing the

communication contents in the server, and if the response result is an error response, the guidelines for performance are as follows.

- Receiver unit performance in case of receiving error responses in BML documents depends on a model.
- The presentation performance in case of receiving error responses in resources composing BML documents depends on a model.

*1 In case status codes are in 400s or 500s. For details, refer to RFC2616.

5.14.13 Guidelines for connecting/disconnecting

For the operation of the return value of the following extended functions for broadcasting relating to connection and disconnection, refer to Appendix-12. For details of status to be used in Appendix-12, refer to 5.14.13.1.

isIPConnected, connectPPP, connectPPPWithISPParams, disconnectPPP

5.14.13.1 Receiver performance guidelines for disconnecting and re-connecting

Using “connectPPP( )”, for example, when disconnected by reasons such as a timeout occurrence during the presentation of some contents on the server whose access is permitted only by certain providers, if the receiver unit performs an auto-connection, then there is concern that the contents cannot be received from there on.

In order to avoid such a situation, receiver units will temporarily store connection parameters specified by contents in the receiver unit at the time of connection. In this section, the area of temporary memory is called “connection parameter temporary memory area”. The value of the connection parameter recorded in this area is a value set by the argument in case of “connectPPP( )”, and a value set by the receiver unit in case of “connectPPPWithISPParams( )”. For details, refer to 5.14.13.2 and 5.14.13.3.

The performance of disconnection and re-connection of receiver units is indicated in Figure 5-7. Disconnected status in (1) is the initial status. Status transition performance in Figure 5-7 is a guideline of PPP connection regarding connection and re-connection. Receiver units in constant connection environments such as Ethernet connections are out of target.

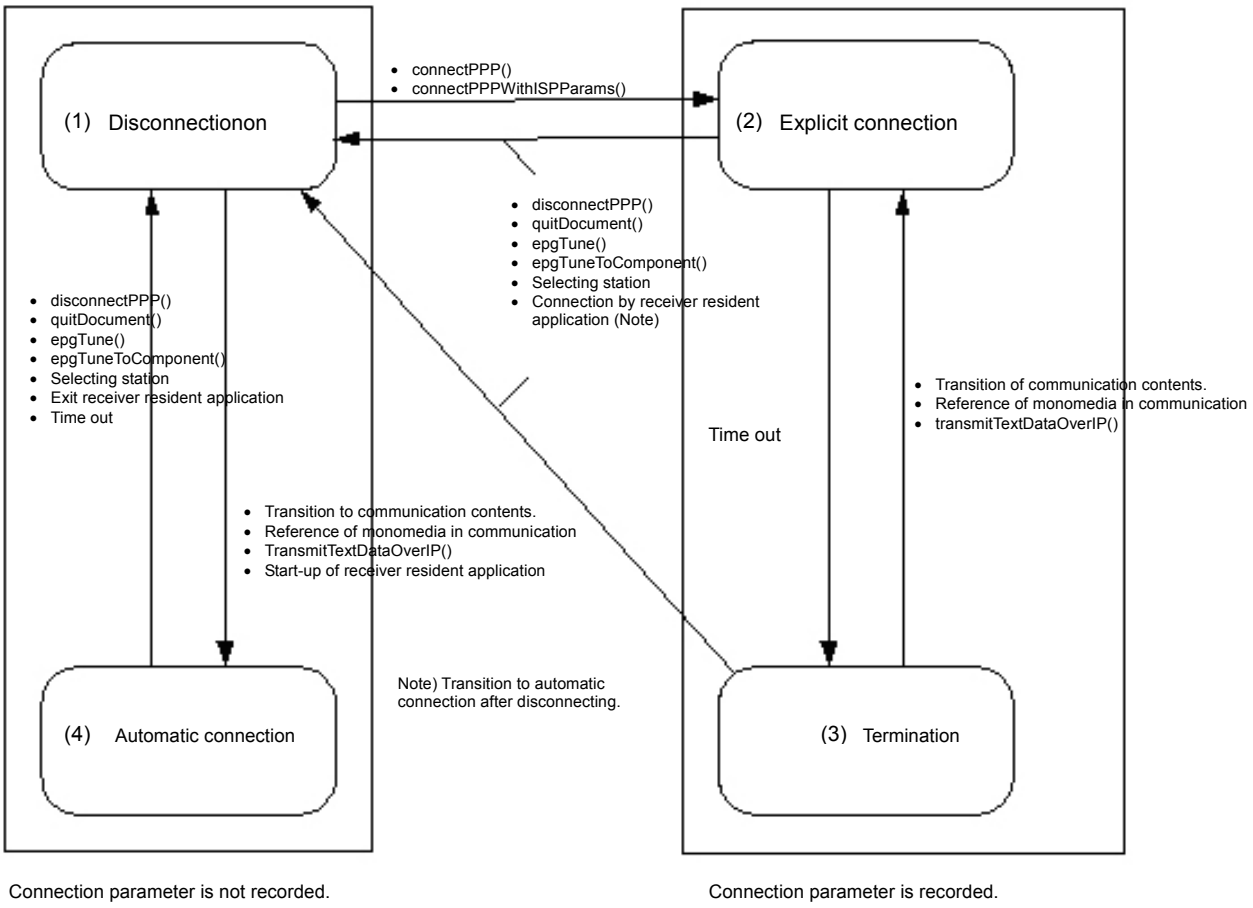


Figure 5-7 Status transition of receiver units in performance of disconnecting and re-connecting

- Status transition from 1

In case the script uses “connectPPP( )” or “connectPPPWithISPParams( )”, the receiver unit will store the connection parameter to the connection parameter temporary memory area (refer to 5.14.13.2) and connection is performed. If the connection is successful, transition to status 2 is executed.

In case of connecting to TCP/IP networks with factors other than above (refer to 5.14.3.3 for details), the receiver unit performs connection without recording the connection parameter to the connection parameter temporary memory area. If the connection is successful, transition to status 4 is executed.

- Status transition from 2

In case disconnection occurs for the following reasons, parameters are cleared and transition to status 1 is executed. “IPConnectionTerminated” event is not generated in this disconnection.

- Disconnection by “disconnectPPP( )”
- Transition to broadcasting by “quitDocument( )”
- Selecting station performance by users and selecting station performance by “epgTune( )”.

In case disconnection is made by other factors, the connection parameter is not cleared and transition to 3 is executed. In this disconnection, the “IPConnectionTerminated” event is generated.

For the details of disconnection performances, refer to 5.14.13.4.

- Status transition from 3.

In status of 3 where it is terminated already

- Transition to broadcasting by “quitDocument( )”
- In case the selecting station performance by the user or selecting station performance by “epgTune( )” is executed, the receiver unit will clear the connection parameter and transition to status 1 is executed.

For the details of process relating to disconnection and termination, refer to Appendix-12.

In case a connection to a TCP/IP network is generated by the following factors from the status of 3, then the receiver units will re-connect using parameters stored in the connection parameter temporary memory area.

- In case a mono-media reference on a server is generated from the BML document that is currently presented.
- In case “transmitTextDataOverIP( )” is called from the BML document that is currently presented.
- In case transition from the BML document that is currently presented to another BML document has occurred.

In case a connection by receiver resident application is generated, then the connection parameters are cleared and after transition to status 1, transition to automatic connection is done.

- Status transition from 4

While in the status of automatic connection, if disconnected by a station selection operation by a viewer or by calling the “epgTune()” function, “epgTuneToComponent()” function, “quitDocument()” function, “disconnectPPP()” function from BML contents, then a transition to status 1 is executed.

5.14.13.2 Receiver unit performance guidelines when connecting by scripts

It should be noted that connection by the “connectPPP( )” function can only be used when the receiver unit is in the status of viewing data broadcasting. (Refer to 5.14.6.6). When

connection is done, the receiver unit will store the “connectPPP( )” argument as a connection parameter in the connection parameter temporary memory area specified in 5.14.13.1.

When connection by script is done using “connectPPPWithISPParams( )”, the receiver unit will connect using the connection parameter that is preset in the receiver unit. For this performance, refer to 5.14.6.6.

When connection performance by the “connectPPP()” function or the “connectPPPWithISPParams()” function is executed, the receiver unit will set the connection parameter in the connection parameter temporary memory area and by maintaining the area, it can be distinguished from the automatic connection status in 5.14.13.3. This is transition performance of 1 → 2 in Figure 5-7.

5.14.13.3 Receiver unit performance guidelines for auto-connection

In case the receiver unit has not done an IP connection and the connection parameter is not recorded in the connection parameter temporary memory area specified in 5.14.13.1 (when in the status of 1), if the following events are generated, the receiver unit will automatically connect to the TCP/IP network.

- In case of transition to the BML document on the server is performed (Transition using “launchDocument( )”, “launchDocumentRestricted( )” and ”href” attribute of anchor element)
- In case of mono-media reference on the server is generated from the BML document.
- In case transmitTextDataOverIP( ) is called.
- In case of connecting to the TCP/IP network from a receiver’s native application.

In case of automatic connection, the connection parameters that are preset in the receiver unit should be used in accordance with the setup for each priority usage line type. In automatic connections, values will not be set in the connection parameter temporary memory area. Automatic connections can be distinguished from connections by “connectPPPWithISPParams( )” by existence/non-existence of values in the connection parameter temporary memory area.

In case connection parameters to the TCP/IP network are not set in the receiver unit, then automatic connection will fail and transition to status of 1 is executed. Also, when connection parameters to the TCP/IP network are set in the receiver unit, but if they are not a connection by PPP, then automatic connection to the TCP/IP network will succeed and transition to status of 4 is executed.

The performance when the receiver unit fails to connect to the TCP/IP network depends on a model. However, in case the receiver unit has multiple communication methods, automatic connection will be attempted in accordance with the set up for each priority usage line type.

5.14.13.4 Receiver unit performance guidelines when disconnected

When an explicit disconnection performance (*) is executed by the viewer or BML contents, then the “IPConnectionTerminated” event is not generated by the receiver unit. In such an event, if the connection parameter temporary area is set, then the contents will be cleared.

In case disconnection occurs by factors other than an explicit disconnection performance, in case of, for example, occurrence of a no communication timeout by the receiver unit, or external factors (disconnected from the ISP, etc.), the receiver unit will generate a “IPConnectionTerminated” event at the time of disconnection. Regarding connection parameters stored in the connection parameter temporary area at this time should follow the transition performance of 5.14.13.1.

(* Explicit disconnection performance: “disconnectPPP”, “quitDocument”, “epgTune”, “epgTuneToComponent”, selection of station, connection by receiver unit resident application)

In case “disconnectPPP( )” is called in the termination status (status 3 in Figure 5-7), value of success is returned and the contents of the connection parameter temporary area are cleared and a transition to the disconnected status in 1 is executed.

In disconnection of communication by “quitDocument()”, an ”IPConnectionTerminated event” will not be generated. Receiver units should complete disconnection performance before the startup document specified in 5.14.12 is displayed and the timing to disconnect after the calling of “quitDocument( )” to the display of the startup document depends on a model. Also, if the connection parameter temporary memory area is set, then the receiver unit will clear the contents at this timing.

The “IPConnectionTerminated” event is not generated at the disconnection of communication caused by the selecting station performance, ”epgTune()” function, “epgTuneToComponent()” function. Receiver units should complete the disconnection performance before completion of station selection and the timing to disconnect from after selecting station or calling of the “epgTune( )” function to completion of selecting station depends on a model. If the connection parameter temporary memory area is set, then the receiver unit will clear the connection parameters at this timing.

In the status of explicit connection, in case the receiver unit resident application with the communication function is started, then clearing of the contents of disconnection of the connection/connection parameter temporary memory area is executed, and after transition to a disconnected status is done, connect from automatic connection.

5.14.13.5 Connection guidelines from receiver's native application

In case the receiver's native application connects to the TCP/IP network, connection should be made using connection parameters that are set in the receiver unit, and connection using connection parameters used in "connectPPP()" stored in the connection parameter temporary memory area should not be executed.

Also, in case the receiver's native application performs communication while the receiver unit is in a connected status using "connectPPP()", then connection by "connectPPP()" is explicitly disconnected once and after transition to 1 is done, communication should be established. While connected by "connectPPPWithISPParams()" or during automatic connection, communication from the receiver's native application can be performed.

Receiver unit performance in case the receiver's native application attempts to perform communication while connected by "connectPPP()" depends on a model.

5.14.13.6 Display guidelines of confirmation messeages when connecting or sending data

Whether display of confirmation message to viewer by dialogue, etc. is required or not in case of executing connection performance of communication is as shown in Table 5-30. Also, whether the display of confirmation messages to viewers by dialogue, etc. is required or not in case of executing data sending performance is as shown in Table 5-31.

Table 5-30 Confirmation message display of receiver units relating to connection

	Connection factors		Status		
			Broadcasting	Linked	Unlinked
1	connect()		X	-	-
2	vote()		X	-	-
3	connectPPP()		X	-	-
4	connectPPPWithISPParams()		X	X	-
5	Automatic connection		O	O	O
6	Reconnection		O	O	O
7	transmitTextDataOverIP()	Dialup	O(*1)	O(*1)	O(*1) (*2)
8		Constant connection	X	X	X(*2)

O : Should be displayed(optional feature of receiver unit)
X : Not displayed(Obtaining permission from viewer in contents before execution is assumed)
- : Not operation (Execution error)
(*1) : In case of already connected status (regardless of explicit connection by function or automatic connection format), do not display confirmation messages.(This is because execution of "transmitTextDataOverIP()" is not a connection factor.)
(*2) : Confirmation message related to the sending of data should be displayed. (Refer to Table 5-31)

Table 5-31 Display of confirmation message of receiver units relating to sending of data

	Factor of sending of data		Status		
			Broadcasting	Linked	Unlinked
1	sendTextData( )		X	-	-
2	transmitTextDataOverIP()	Constant connection / Dialup	X	X	O

O : Should be displayed
X : Not displayed(Obtaining permission from viewer in contents before execution is assumed)
- : Not operation (Execution error)

5.14.14 Operation of root CA certificates

In encrypted communication using TLS1.0 and SSL3.0, root certificate to authenticate the broadcaster to operate the server is necessary. There are two types of certificates which are the root CA certificates of general purpose and the root CA certificates of limited purpose by broadcasters. Root CA certificates of general purpose are stored in the receiver units permanently; however, root CA certificates of limited purpose by broadcasters are effective only within the service operated individually by broadcasters. When root CA certificates is written in this chapter, it refers to the general term of root CA certificates, which does not differentiate between root CA certificates of general purpose and root CA certificates of limited purpose by broadcasters, and in case individual operation is necessary, then it is written explicitly as root CA certificates of general purpose and root CA certificates of limited purpose by broadcasters.

For details relating to the transmission of route certification documents, refer to Chapter 2. For details of encrypted communication and for details of the root CA certificate format, refer to Vol. 6.

5.14.14.1 Operation of the memory area for root CA certificates

Receiver units establish a memory area to store root CA certificates. If the root CA certificate type (root_certificate_type) in the Root Certificate Descriptor is 0, then the receiver unit handles it as a root CA certificate of general purpose.

The root CA certificates of general purpose acquired from the data carousel are stored permanently in the area indicated in Table 5-32. The receiver unit refers to the root CA certificate stored in this memory area when performing encrypted communication regardless of the receiver unit status (data broadcasting reception status/linked status/unlinked status).

Table 5-32 Memory area for root CA certificates used in Digital Terrestrial Television Broadcasting

Type	Meaning	Capacity of memory area
Memory area for root CA certificates	Area to store root CA certificates of general purpose transmitted by carousel	• 3KB per one certificate • Quantity: 8

Each block of each memory area for root CA certificates is indexed with a root CA certificate storage number of 0-7 corresponding to the line of “root_certificate_id” and “root_certificate_version” in the Root Certificate Descriptor. (Refer to Figure 5-8) If the “root_certificate_id” is 0xFFFFFFFF in the Root Certificate Descriptor, then it is indicating that root CA certificates of general purpose to be stored do not exist in the transmitted module for storing root CA certificates in transmission.

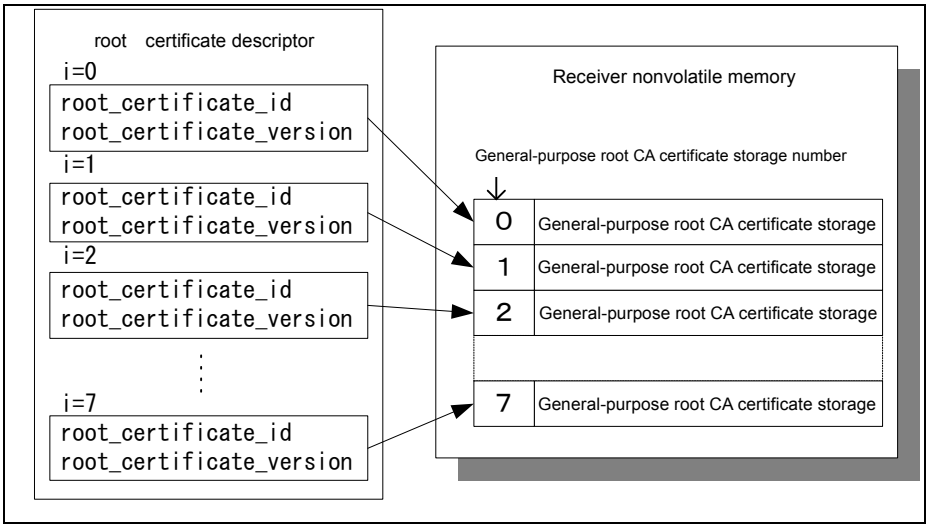


Figure 5-8 Corresponding relationship of the Root Certificate Descriptor and the memory area for root CA certificates

5.14.14.2 Storage of root CA certificates of general purpose to receiver units

When storing a root CA certificate of general purpose, the ID of the root CA certificate for general purpose and the version of the root CA certificate for general purpose(hereinafter referred to as the root CA certificate information) written in the module for root CA certificates correspond to the storage number of memory area for root CA certificates by receiver within the receiver units. (Refer to Figure 5-9) Regarding the format to store root CA certificate storage information, it depends on a model of receiver units. For details of the structure of module for storing root CA certificates, refer to Vol. 6. For details of receiver unit performance of root CA certificate of general purpose, refer to 5.14.14.3.

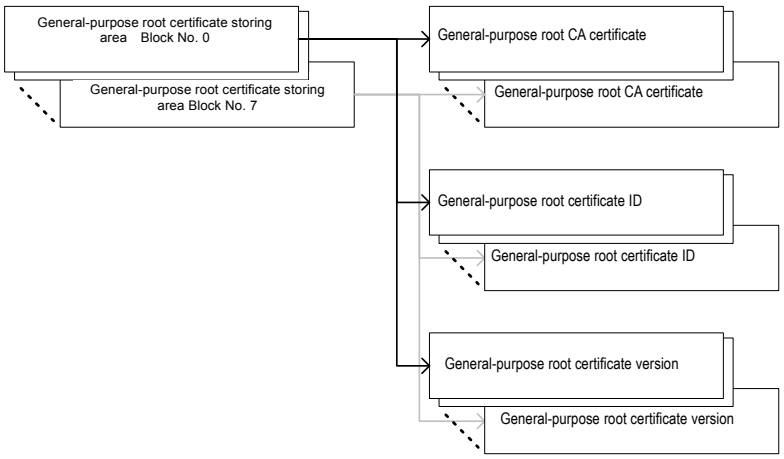


Figure 5-9 The relationship between storage numbers of memory area for root CA certificates and related root CA certificates information

For handling expired root CA certificates in the memory area for root CA certificates, it depends on a model of receiver units.

5.14.14.3 Guidelines of receiver unit behavior for updating the root CA certificate of general purpose

(1) Confirmation of the Root Certificate Descriptor

Receiver units check the existence of a Root Certificate Descriptor immediately after the selection of a station. When the Root Certificate Descriptor is received and if the root_certificate_type is 0, then it is judged as a root CA certificate of general purpose. After that, the “root_certificate_id” within the Root Certificate Descriptor is checked and if all of the “root_certificate_id” is 0xFFFFFFFF, then it is judged that there is no root CA certificates of general purpose to be stored, and succeeding processes will not be executed.

In case the receiver unit detects a “root_certificate_id” other than 0xFFFFFFFF, then it refers to the corresponding “root_certificate_version” to confirm the value is not 0xFFFFFFFF and the succeeding process is executed. If the “root_certificate_version” is 0xFFFFFFFF, then it is judged as an incorrect root CA certificates of general purpose, and succeeding processes will not be executed.

For the process in case the “root_certificate_type” = 1, refer to 5.14.14.4.

Confirmation of the root CA certificate is done immediately after the selection of a station or in case a process equivalent to re-selecting a station is executed (when the PID of the component that is currently viewed is changed, etc.), however, it should be noted that this is

not executed in any other cases (when the data event of the component that is currently viewed is changed or a pull back occurs). As for the performance of contents, in case a forced reacquisition of the root CA certificate is desired, it can be reacquired by executing “epgTune()” in the service that is currently viewed.

(2) Update confirmation of the root CA certificate of general purpose

The “root_certificate_id” of the Root Certificate Descriptor and the value of the root_certificate_version and the value of the root CA certificates information corresponding to this line are compared. This process is done for every “root_certificate_id” except for the detected 0xFFFFFFFF. *1

In case the contents of the Root Certificate Descriptor and root CA certificates information have the same values, it is judged as a certificate that is already stored, and the succeeding process is not executed.

In case a root CA certificate of general purpose is not stored in the specified storage area, or in case the Root Certificate Descriptor and root CA certificates information are different, proceed to the next process.

*1 Up to two root CA certificates can be stored in the module for storing root CA certificates, therefore the maximum number of “root_certificate_id” that can be detected other than 0xFFFFFFFF is 2 at most.

(3) Authentication inspection of modules for storing root CA certificates

In case the contents of the Root Certificate Descriptor and root CA certificates information corresponding to the line stored within the receiver unit have different values, then the receiver unit will judge that a new root CA certificate has been transmitted and a module for storing root CA certificate is received. The “root_certificate_id “ and “root_certificate_version“ in the Root Certificate Descriptor corresponding to the root CA certificate storage number that is written within the module, and the root certificate ID and root certificate version number that are written within module are compared. In case they do not match, it is judged as an incorrect module for storing the root CA certificate and the succeeding processes will not be performed. For the details on the module for storing root CA certificates, refer to Vol. 6.

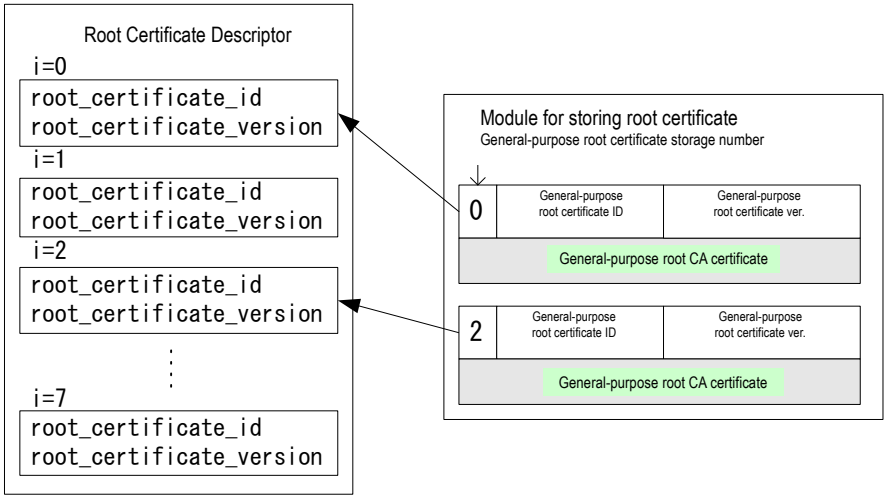


Figure 5-10 The relationship between the module for storing root CA certificates and the Root Certificate Descriptor

(4) Storage of root CA certificates of general purpose

In case the validity of the module for storing the root CA certificate is confirmed, the corresponding root CA certificate of general purpose will be stored in the storage number of the memory area for root CA certificates which is written in the module for root CA certificates. Receiver units will store the root CA certificate of general purpose, and at the same time, the root certificate version written in the module for root CA certificates are stored in the storage number of the memory area of root CA certificates.

5.14.14.4 Guidelines of receiver unit behavior when receiving root CA certificates of limited purpose by broadcasters

Broadcasters can operate independent root CA certificates. In case the “root_certificate_type“ of the Root Certificate Descriptor is 1, then the receiver unit will handle the root CA certificate transmitted by the carousel as a root CA certificates of limited purpose by broadcasters. In case of operating a root CA certificates of limited purpose by broadcasters,, broadcasters can use a root CA certificate of general purpose in addition to the root CA certificate of limited purpose by broadcasters

(1) Confirmation of the Root Certificate Descriptor

Receiver units check the existence of a Root Certificate Descriptor immediately after the selection of a station. In case the “root_certificate_type” of the Root Certificate Descriptor is 1, then the receiver unit will judge it as a root CA certificate of limited purpose by broadcasters and the succeeding process is performed.

(2) Confirmation of the validity of the module for storing the root CA certificate

Receiver units receive modules for storing root CA certificates, and if the first character of the storage number of the root CA certificates of limited purpose by broadcasters stored in the module for root CA certificate is '_', then it is judged as a correct module, and the root CA certificates of limited purpose by broadcasters is stored and it proceeds on to the next process. If it's other than '_', then the receiver unit will judge it as an incorrect module for storing root CA certificates, and the succeeding process will not be performed.

(3) Storage of root CA certificates of limited purpose by broadcasters

After validity of root CA certificates of limited purpose by broadcasters is confirmed, the receiver unit will store it temporarily. In case of encrypted communication, the root CA certificate of limited purpose by broadcasters and root CA certificate of general purpose are referred to.

Since the attribute of the root CA certificate of limited purpose by broadcasters is determined by the "root_certificate_type" in the Root Certificate Descriptor in DII, referring of a carousel is necessary. For this purpose, it can be used only in a data broadcasting reception status or linked status. The root CA certificate of limited purpose by broadcasters is effective only within the service.*1

(4) Discarding of root CA certificates of limited purpose by broadcasters

As soon as the performance of selecting a station or transition to an unlinked status of the receiver unit occurs, the stored root CA certificate of limited purpose by broadcasters is no longer valid.

5.15 Operation of the bookmark service

The bookmark service is operated. Regarding the operation of the bookmark service, refer to ARIB STD-B24 Vol. 2 Appendix 1.

5.15.1 Service type of the bookmark list service

As a basic rule, the bookmark list service is broadcast as a service type of the “bookmark list service” (service_type=0xAA). However, broadcasters who do not have a license for (independent) data services may broadcast the bookmark list contents in other the service type in some cases.

5.15.2 Operation of the pre-list

Prelist by multi-media contents is not operated as a basic rule. However, this is not to interfere with broadcasting of the prelist by the desired broadcaster. In case of operation, the statement regarding the prelist in ARIB STD-B24 Vol. 2 Appendix 1 should be followed.

5.15.3 Registration by bookmark type

- Registration of broadcaster individual operation type

Registration by bookmark types of new broadcaster individual operation type by broadcaster. The operation will be specified separately.

5.15.4 Bookmark list display function by receiver’s native application (Optional)

The bookmark list display service is assumed to be realized in contents, however, considering the convenience of viewers, the receiver unit should have a bookmark list display function that is not dependent on broadcasting contents as a receiver’s native application for the reasons listed below. Also, this function should be used by a simple operation such as by an exclusive key, etc. on a remote control. Regarding the operation of implementation of this function, refer to ARIB STD-B24 Vol. 2 Appendix 1.

The specification for writing time to Greg while selecting stations from the bookmark list by receiver features is written in ARIB STD-B24 Vol. 2 Appendix 1 “8.4.3.5 Operation of bookmark list service”, however, if the receiver unit is in situation where it cannot acquire the time, then "000000000000" as a decimal number character string of 12 bytes indicating the time should be written.

5.15.5 General address definitions relating to the bookmark service

Address definitions relating to the bookmark service are specified as follows.

Table 5-33 Address definitions relating to bookmarks

Item	Address term
Bookmark service	Bookmark service
Bookmark key	Bookmark button

5.15.6 Operation of the expansion data section of the memory area for bookmark service

Among the expansion data section in the memory area for bookmark service, in case of omitting items that are specified as omittable, the following values should be written in the corresponding fields.

- In case of omitting variable length character strings, set 0 to the length byte.
- In case of omitting fixed length character strings, then 0x00 is written in all of them.
- In case of omitting numerical values of a fixed length, then 0 is written in all of them.

5.16 Operation of registration transimission

Registration transmission is assumed to be performed by instruction from viewers using contents or receiver's native applications. The operation by contents is written in section 5.16.4, and the operation of registration transmission by receiver's native application is written in section 5.16.5. Registration transmission by receiver's native application is optional.

5.16.1 Operation of the memory area for registration transmissions

In order to store superimpose that could not be sent, receiver units will establish a memory area for registration transmission in NVRAM. Broadcasters can store superimpose to perform registration transmission in the memory area for registration transmission. For data to be written in the memory area for registration transmission, the specifications in section 5.16.3 should be followed.

5.16.2 Identification of the memory area for registration transmission

In case of performing reading/writing of information for the memory area for registration transmission from the Multimedia service, 1 variable length block should be regarded as one file and "readBookmarkArray()/writeBookmarkArray()" is executed. Reading/writing of information for the memory area for registration transmission from the Multimedia service is done in units of variable length blocks. The following URI is used to identify variable length blocks.

nvram:// denbun/<block number>
<block number>:0-N (N is bigger than or equal to 2)

In case the extended function for broadcasting(readBookmarkArray(),writeBookmarkArray())is executed for the memory area for registration transmission after names other than above mentioned specification methods are specified, then the reading/writing of NVRAM is not performed and "readBookmarkArray()" will return null (failure) and "writeBookmarkArray()" will return NaN (failure) as return values.

5.16.3 Operation of the bookmark function for registration transmission

5.16.3.1 Operation of the "writeBookmarkArray" function in registration transmission

The performance of this function is in accordance with specifications of ARIB STD-B24, however, there is a limit for arguments in operation of registration transmission. The restricted matters are specified in Table 5-34. This function has a process to set the time that writing basic data section of registration transmission information is performed, however, if acquisition of the time fails, then writing returns a failure.

Table 5-34 Argument operation of "writeBookmarkArray function " in registration transmission

Argument	Values to be set	Comments
filename	URI to identify the memory area for registration transmission is specified.	Regarding the identification of the memory area for registration transmission, refer to section 5.16.2.
title	Program name that had set the superimpose of the registration transmission is set	It is recorded in S:1V format, however, length is not specified in the argument.
dstURI	Server of the sending destination	It is recorded in S:1V format, however, length is not specified in the argument. In case of BASIC procedure, specification is not necessary.
expire_str	Expiration date of registration transmission superimpose is set.	YYYYMMDDHH 10 digits should be specified for the expiration date. If the digit number of the specification is not 10 digits, then the writing is regarded as a failure.
bmType	199 is specified.	EUC-JP character strings
linkMedia	0 is specified.	
usageFlag	0 is specified.	
extendedStructure	Format of registration transmission information is specified.	Refer to section 5.16.3.9.
extendedData	Registration transmission superimpose data is established.	

5.16.3.2 Operation of the “readBookmarkArray” function in registration transmission

The performance of this function is in accordance with the specifications in ARIB STD-B24, however, there is a limit for arguments in operation of registration transmission. The restricted matters are specified in Table 5-35.

Table 5-35 Argument operation of the “readBookmarkArray” function in registration transmission

Argument	Values to be set	Comments
filename	URI to identify the memory area for registrasion transmission is specified.	Regarding the identification of the memory area for registration transmission, refer to section 5.16.2.
bmType	199 is specified.	
extendedStructure	Format of the registration transmission information is specified.	Refer to section 5.16.3.9.

Among the return values of the “readBookmarkArray” function, Array[6] and Array[7] are operated. Expansion data is stored in Array[8].

5.16.3.3 Operation of the “deleteBookmarkArray” function in registration transmission

Performance of this function is in accordance with specifications in ARIB STD-B24, however, in operation of registration transmission, the URI specified in section 5.16.2 should be specified as the “filename” of the argument.

5.16.3.4 Operation of the “lockBookmarkArray” function in registration transmission

Performance of this function is in accordance with specifications in ARIB STD-B24, however, in operation of registration transmission, the URI specified in section 5.16.2 should be specified as the “filename” of the argument.

5.16.3.5 Operation of the “unlockBookmarkArray” function in registration transmission

Performance of this function is in accordance with specifications in ARIB STD-B24, however, in operation of registration transmission, the URI specified in section 5.16.2 should be specified as the “filename” of the argument.

5.16.3.6 Operation of the “getBookmarkInfo2” function in registration transmission

Performance of this function is in accordance with Appendix-8, however, the meaning of the return value is specified as the following in operation of registration transmission.

Array[0]: Number, all numbers of the implemented registration transmission area

Array[1]: Number, number of the remaining registration transmission superimpose that can be registered.

Array[2]: String, URI indicating the registration transmission area that can be newly registered.

5.16.3.7 Format of registration transmission information

For the maximum size of calling information stored in the registration transmission area, the total of the basic data section and expansion data section should be less than or equal to 1.5KB.

5.16.3.8 Basic data section of registration transmission information

The basic data section of registration transmission information should be in accordance with ARIB STD-B24 Appendix 1. Operation of registration transmission on its own is described below.

Argument of writing function (writeBookmarkArray())and values specified by receiver units are written in the basic data section.

Table 5-36 Operation of the basic data section of registration transmission information

	Property	ECMA type	Writing type	comments
Basic data section	title	String	S:1V(* Maximum 41Bytes)	EUC-JP character string * including length
	dstURI	String	S:1V(* Maximum 61Bytes)	EUC-JP character string * including length
	expire	String	S:10B	YYYYMMDDHH
	registerDate	String	S:10B	YYYYMMDDHH (set by the receiver unit)
	bmLock	String	S:1B	1 alphanumerical character (set from exclusive API)
	bmType	String	S:3B	Fixed to 199
	linkMedia	String	S:1B	Fixed to 0
	usageFlag	String	S:1B	Fixed to 0

- Operation of titles (title)

Program name that performed the registration of registration transmission is written. This field should be written by contents at the time of registration of registration transmission information.

For titles, the maximum is 40 bytes. (Not including length) In case it is exceeds 40 bytes, the receiver units will not store beyond the 41st byte. In case the 40th byte is the first byte of a double byte character, then the character should not be stored.

- Operation of a linked destination URI (dstURI)

For the URI of a linked destination, the maximum is 60 bytes. (Not including length) In case the sending destination is a server, then this field should be written by the contents.

In case it is exceeds 60 bytes, the receiver units will not perform writing and NaN will be returned. In case of a Basic procedure, an empty string with a length of 0 should be specified.

- Operation of expiration dates (expire)

In expiration dates, the last date/time that registered registration transmission information will remain effective until is written. This field should be written by contents at the time of registering registration issue information.

- Operation of writing date/times (registerDate)

Absolute time at the time that the writing function of registration transmission information is called is written. In case the acquisition of time fails, then a writing failure will be returned.

- Operation of the deletion prohibition flag(bmLock)

This flag indicates that deletion is prohibited by instructions of the viewer.

0: If immediately after bookmark is recorded or “unlockBookmark()” is called, it is 0.

1: 1 is stored if deletion is prohibited by “lockBookmark()”

- Operation of each bookmark type(bmType)

Fixed value of 199 is written. The performance in case other values are set depends on a model of receiver units.

- Operation of linked media identification(linkMedia)

Fixed value of 0 is written. The performance in case other values are set depends on a model of receiver units.

- Operation of each permission type(usageFlag)

Fixed value of 0 is written. The performance in case other values are set depends on a model of receiver units.

5.16.3.9 Expansion data section of registration transmission information

The format of information to be stored in the expansion data section of registration transmission information is indicated in Table 5-37.

Table 5-37 Format of data written in the memory area for registration transmission

Property	ECMA Type	Writing Type	Comments
RegisterNetwork	Number	U:2B	Network ID of the service that performed the writing
RegisterService	Number	U:2B	Service ID of the service that performed the writing
EventInfo1	String	S:1V	Supplemental information related to the program
EventInfo2	String	S:1V	Supplemental information related to the program
Information	String	S:1V	Announcement to viewers
ConnectRequest	String	S:12B	Desired date and time for the calling YYYYMMDDHHmm
ConnectServiceStartTime	String	S:4B	Time to start accepting the calls HHMM
ConnectServiceEndTime	String	S:4B	Time to finish accepting the calls HHMM
ServiceUtilityArea	String	S:1V	Utility area
Interactive	Number	U:1B	Flag to indicate calls by instruction of the viewer is possible or not outside of reception hours.
ConnectType	Number	U:1B	In case of 0, then connection using “receiverISP” setup In case of 1, then connection using “connectPPP” In case of 2, then connection using “Basic procedure”.
Tel	String	S:1V	*1*2*4*6
bProvider	Number	U:1B	*1*4*6
uid/hostNo	String	S:1V	*1*4*6 If “ConnectType” is 1, then the uid is set. If 2, then the ”hostNo” is set.
passwd	String	S:1V	*1*5*6
nameServer1	String	S:1V	*3*5*6
nameServer2	String	S:1V	*3*5*6
softCompression	Number	U:1B	*1*5*6
headerCompression/ speed	Number	U:1B	*1*4*6 If the “ConnectType” is 1, then the “headerCompression” is set. If 2, then the speed is set.
timeout/idleTime	Number	U:3B	*1*4*6*7
Denbun	String	S:2V (Maximum of 1024Bytes)	EUC-JP Character strings

*1 Required if “ConnectType” is 1.
*2 Even if “ConnectType” is 1, in case of PPPoE, it can be omitted. In case of omission, empty strings should be set.
*3 Even if “ConnectType” is 1, it can be omitted in case PPP driver can obtain value dynamically. Empty strings should be set in case of omission.
*4 Required if “ConnectType” is 2.
*5 If “ConnectType” is 2 and in case of String, then empty strings, in case of Number, then 0 should be set.
*6 If “ConnectType” is 0, and in case of String then empty strings, in case of Number, then 0 should be set.
*7 If “ConnectType” is 1, then timeout should be set, If 2, then idleTime should be set.

5.16.3.10 Operation of each field of registration information

- RegisterNetwork

The network ID of the service that performed the registration is written. Contents should fill in the network ID of their own service at the time of registering the calls.

- RegisterService

The service ID of service that performed the registration is written. Contents should fill in the service ID of their own service at the time of registering the calls.

- EventInfo1

Information related to the programs that performed registration. It can be used by contents displaying the list of registration transmissions to present information related to programs that performed registration for viewers. Writing in this field is recommended. In case it is not written in for unavoidable reasons, it should be an empty string with a length of 0. The maximum number of characters is 40 bytes and double byte codes should be used. Single byte code characters are not used.

- EventInfo2

Separately from “EventInfo1”, information related to programs is written. In case it is not necessary, this field should be an empty string with a length of 0. The maximum number of characters is 40 bytes and double byte codes should be used. Single byte code characters are not used.

- Information

Field that is used when a registered program wishes to communicate some sort of information to the viewers. Writing in this field is recommended. In case it is not written for unavoidable reasons, it should be an empty string with a length of 0. The maximum number of characters is 120 bytes and double byte codes should be used. Single byte code characters are not used.

- ConnectRequest

The desired time for contents to call programs is written. In case the writing in this field is not necessary, then a fixed character string ”FFFFFFFFFFFF” should be written.

- ConnectServiceStartTime

Time to start accepting calls is written. It should be written in a 24 hour time format. In case the writing in this field is not necessary, then a fixed character string ”FFFF” should be written. In case fixed character strings "FFFF" are set, then fixed character strings "FFFF" should be set in the ConnectServiceEndTime field as well.

- ConnectServiceEndTime

Time to finish accepting calls is written. It should be written in a 24 hour time format. Time

that straddles over days can be written from ConnectServiceStartTime.
(Example: ConnectServiceStartTime : 23:00 / ConnectServiceEndTime : 05:00)
In case writing in this field is not necessary, then a fixed character string ”FFFF” should be written.

- ServiceUtilityArea

The field to be used for passing parameters that should be set in registration transmission contents. In case writing in this field not necessary, this field should be an empty string with a length of 0. For the size of this field, the total of the basic data section of registration transmission information and expansion data section should be in a range not exceeding 1.5KB (1536 bytes).

- Interactive

Flag to indicate whether or not to accept calls by instructions from viewers outside of the reception hours for registration transmission indicated by “ConnectServiceStartTime” and “ConnectServiceEndTime”. In case the value of this field is ‘1’, it means that calls by instructions from viewers are accepted even outside of the reception hours for registration transmission. On the other hand, when it is ‘0’, then it means that calls by instructions from viewers cannot be accepted outside of the reception hours for registration transmission. In case the value of this field is ‘0’, and when a call is instructed by viewers outside of the reception hours for registration transmission then performance of calling should be restrained. Also, when fixed character strings ”FFFF” are set in “ConnectServiceStartTime” and “ConnectServiceEndTime”, then ’1’ should be set in this field.

- ConnectType

Specifies the type of calling function. In case it is 0, then connection is made by “connectPPPWithISPParam”or automatic connection. In case it is 1, then connection is made by “connectPPP”. In case it is 2, then it means the connection is made by basic procedure. In case of registering calls, this field should be written by contents.

- Tel

In case of using the “connectPPP()” function or “connect()” function, then specifying phone number should be written. In case of the “connectPPPWithISPParams()” function or automatic calling, then it should be an empty string with a length of 0. The maximum number of characters is 30 bytes.

- bProvider

In case of using the “connectPPP()” function or “connect()” function, network specification identification should be written. In case of connecting by the “connectPPPWithISPParams()” function or automatic connection, value 0 should be written.

- uid/hostNo

In case of using the “connectPPP()” function, then the user ID should be written, and in case of using the “connect()” function, the host number should be written. In case of connecting by “connectPPPWithISPParams()” or automatic connection, then values do not have to be written. In such case, it should be an empty string with a length of 0. The maximum number of characters is 64 bytes.

- passwd

In case of using the “connectPPP()” function, then the password for user authentication should be written. In case of connecting by the “connect()” function, “connectPPPWithISPParams()” function or automatic connection, then values do not have to be written. In such case, it should be an empty string with a length of 0. The maximum number of characters is 32 bytes.

- nameServer1

It should be written in case of using “connectPPP()”. In case of connecting by the “connect()” function, “connectPPPWithISPParams()” function or automatic connection, then values do not have to be written. In such case, it should be an empty string with a length of 0. The maximum number of characters is 15 bytes.

- nameServer2

It should be written in case of using “connectPPP()”. In case of connecting by the “connect()” function, “connectPPPWithISPParams()” function or automatic connection, then values do not have to be written. In such case, it should be an empty string with a length of 0. The maximum number of characters is 15 bytes.

- softCompression

In case of using the “connectPPP()” function, then a flag to indicate whether or not the compression of software is required or not should be written. In case it is compressed then ‘1’, and in case it is not compressed then ‘0’ should be set. In case of connecting by the “connect()” function, “connectPPPWithISPParams()” function or automatic connection, then values do not have to be written.

- headerCompression/speed

In case of using the “connectPPP()” function and in case it is compressed then ‘1’ should be set, and in case it is not compressed then ‘0’ should be set as a flag to indicate whether header compression is used or not. In case of using the “connectPPP()” function, the specification value of line speed to be connected should be written. In case of connecting by the “connectPPPWithISPParams()” function or automatic connection, then value 0 should be written.

- timeout/idleTime

In case of using the “connectPPP()” function, continuous time of no communication status (idleTime) before disconnecting should be written, and in case of using the “connect()” function, time to recognize it as timeout (timeout) should be written. In case of connecting by the “connectPPPWithISPParams()” function or automatic connection then value 0 should be written.

- Denbun

Superimpose that should be sent is set. In case of using the registration transmission function, this should be written. The maximum byte number of superimpose is 1024 bytes and the character string should be EUC-JP.

5.16.4 Guidelines of registration transmission operation by contents

5.16.4.1 Operation when registration transmission is set up

Contents that set up registration transmission should store registration transmission information in the memory area for registration transmission upon consent from the viewers. Contents that perform registration transmission setup that depend on time elapse should write contents in order not to set up the registration transmission during the playback of recording.

Regarding the writing of registration transmission information, contents should use the “getBookmarkInfo2()” function to obtain the URI for the memory area for registration transmission and the existence of free space should be confirmed. In case free space is confirmed, registration transmission information can be stored in the URI obtained by the above mentioned function.

As a basic rule, new registration transmission information should not be overwritten on stored mamory areas for registration transmission.

In case there is no free space, writing can be done after deleting stored registration transmission information. The registration transmission information that can be deleted is as follows.

- Expired. In such case, the contents to be deleted should present expired registration transmission information to viewers *1, and explicitly announce deletion due to expiration.
- The oldest among the registration transmission information without deletion prohibition flag. In such case, registration transmission information of deletion candidates is presented *1, and contents to be deleted should be deleted upon consent for deletion from the viewers.

*1 For the registration transmission information that should be displayed, refer to Table 5-38.

5.16.4.2 Operation of registration transmission by contents

Contents that perform registration transmission should be equipped with the following functions.

(1) Function to display a list of registration transmission information that has set stored registration transmission. (hereinafter referred to as, registration transmission information list function)

For the registration transmission information that should be displayed, refer to Table 5-38.

(2) Function that allows viewers to select desired registration transmission from a registration transmission information list and to send superimpose to desired site of the registration transmission information.

However, in case expired registration transmission information is selected, contents should explicitly inform the viewer of expiration and superimpose is not sent out and the registration transmission information is deleted. The operation of deletion of registration transmission information refers to the following.

(3) Delete management function for registration transmission information

Deletion management should have the following functions.

- Manual deletion of registration transmission information

  Function to delete desired registration transmission information by operation of the viewer.

- Automatic deletion of registration transmission information

  Registration transmission information can be automatically deleted without instruction from viewers in the following cases.

  - Expired registration transmission information. However, viewers should be explicitly informed of the deletion of corresponding registration transmission information at the time of deletion.

  - In case the communication of registration transmission information is a success.

- Deletion refusal specification of registration transmission information

  Function to turn ON/OFF the flag to delete the desired registration transmission information by operation of the viewer.

  In case the flag to prohibit deletion is ON, then it cannot be deleted even if it was expired. In such case, contents should explicitly inform viewers that even though it is expired, it cannot be deleted because the flag is ON.

Table 5-38 Registration transmission information that should be displayed

Fields	Meaning	Comments
title	Program name that has set the superimpose of registration transmission	
expire_date	Expiration date of registration transmission superimpose	
RegisterService	Service ID that has set the superimpose of registration transmission	Presentation of a service logo etc., is possible.
EventInfo1	Supplementary information regarding programs - Part I	
Information	Announcement to viewer	
ConnectRequest	Desired time for calling	
ConnectServiceStartTime	Desired time to start accepting calls	
ConnectServiceEndTime	Desired time to finish accepting calls	

5.16.5 Registration transmission using registration transmission information by receiver's native application (Optional)

Registration transmission information can be realized by receiver's native applications as well. In this section, requirements that registration transmission function should fulfill are explained.

5.16.5.1 Registration of registration transmission information

Registration transmission information is assumed to be registered by contents. For details, refer to section 5.16.4.1.

5.16.5.2 Operation of registration transmission by receiver's native application

Registration transmission function by receiver's native application basically depends on a model. An example of a registration transmission function by receiver features is shown in this section.

Registration transmission function by receiver features has the following functions.

(1) Function to display a list of registration transmission information that has set stored registration transmission. (hereinafter referred to as registration transmission information list function) Registration transmission information that should be displayed is in accordance with Table 5-38.

(2) Function that allows viewers to select the desired registration transmission from the registration transmission information list and to send superimpose to the specified site of

the registration transmission information.
However, in case expired registration transmission information is selected, the receiver's native application should explicitly inform the viewer of expiration and the superimpose is not sent out and the registration transmission information is deleted. For the operation of deletion of the registration transmission information, refer to the following.

(3) Delete management function of registration transmission information

Deletion management should have the following functions.

- Manual deletion of registration transmission information

  Function to delete the desired registration transmission information by operation of the viewer.

- Automatic deletion of registration transmission information

  Function to delete registration transmission information automatically in case of the following.

  - Expired registration transmission information. However, viewers should be explicitly informed of the deletion of corresponding registration transmission information at the time of deletion.

  - In case the communication of registration transmission information is a success.

- Deletion refusal specification of registration transmission information

  Function to turn ON/OFF the flag to delete the desired registration transmission information by operation of the viewer.

  In case the flag to prohibit deletion is ON, then it cannot be deleted even if it has expired.

(4) Startup of the registration transmission function by the bookmark button

For receiver units equipped with the bookmark list function by receiver's native application, it should be able to be started by the bookmark button. In such case, both the bookmark list function and registration transmission information list function should co-exist by the display switching function by receiver's native application.

(5) Receiver units that have receiver's native applications providing the registration transmission information list function should display the registration transmission information list function application and exclusive process on display of BML in order to avoid competition with the registration transmission information list function by contents.

5.16.5.3 Reservation transmission function by receiver features

The calling function of registration information by receiver's native applications can have the reservation calling function, which makes calls automatically within the specified range of date/time by using information of "connectRequest", "connectServiceStartTime", "connectSerivceEndTime" fields in registration transmission information.

Performance of reservation calling depends on a model of receiver unitss, however, the

calling should be performed within the time range specified by “connectServiceStartTime” and “connectServiceEndTime” by the date specified by “connectRequest”. If the call can be made close to the time that was specified by “connectRequest” field, then it is better.

Regarding the process when the sending of superimpose fails, or the number of retry times and time accuracy depend on models.

5.16.5.4 Guidelines for sending messages by receiver’s native application

Guidelines for receiver unit performance in case of sending superimpose of receiver’s native application as instructed by the viewer is as follows.

(1) Judgement of whether or not sending of superimpose is possible

- Check the “ExpireDate” and “bmType” and check whether or not they are valid registration transmission information. In case they are valid registration transmission information, then proceed on to the next process. In case they are invalid registration transmission information, a message to notify that they cannot be sent because the data has problems is presented to viewers and after deleting invalid registration transmission information, the process should be end.
- Check the “Iteractive” field in the expansion data and if it is 0, then proceed on to (2). If it is 1, then proceed on to the following process.
- Check the range of “ConnectServiceStartTime” and “ConnectServiceEndTime” and check if the absolute time of the receiver unit is within the corresponding range. If it is within the range, then proceed on to (2). If it is out of the range, then a message to prompt calls before the expiration date should be presented and finish the process.

(2) Identification of the superimposeending method

- Check “ConnectType” in the expansion data and identify the sending method. In case of receiver units with equipped modems, proceed on to the next process. If “ConnectType” is 1 or 2 in case of receiver units without modems, then a message to notify that sending cannot be performed to the viewers should be presented and finish the process.
- Check whether or not the identified sending method can be used. If it can be used, then proceed on to the next process. If it cannot be used, then a message to notify that sending cannot be performed now to the viewers should be presented and finish the process.
- In case the “ConnectType” is 0, then proceed on to the process of (3). If the “ConnectType” is 1, then Tel, bProvider, uid/hostNo, Passwd, nameServer1, nameServer2, softCompression, headerCompression/speed, timeout/idleTime fields are read and proceed on to the process of (3) as the connection parameter. If the “ConnectType” is 2, then read Tel, bProvider, uid/hostNo, headerCompression/speed fields and proceed on to the process of (3) as the connection parameter.

(3) Sending of superimpose

- Refer to “denbun” field in the expansion data and read the superimpose data and proceed on to the next process.
- In case the “ConnectType” is 0, then refer to the “dstURI” field of the basic data section, and for the specified server, superimpose data is sent via TCP/IP protocol using receiver unit ISP setup. In such an event, the sending format should be a format in accordance with the transmitTextDataOverIP() function in section 5.14.6.6. Performance in case sending fails or performance in case reading of superimpose data fails depend on models.
- In case the “ConnectType” is 1 , connect to the TCP/IP network by parameters read in (2) by performance equivalent to the connectPPP() function indicated in section 5.14.6.6. However, the handling of values specified in “timeout/idleTime” field depends on a model of receiver units. Regarding the sending format of superimpose, it is the same as when the “ConnectType” is 0. Performance when the sending fails or performance when the reading of superimpose data fails depends on a model of receiver units.
- In case the “ConnectType” is 2, communication is established by parameters read in (3) by performance equivalent to the “connect()” function of basic procedures in section 5.14.6.5. After that, superimpose data is read from the “denbun” field and superimpose data is sent by performance equivalent to the “sendTextData()” function. The handling of return values from the interaction channel center of receiver units does not have to be considered. In other words, receiver units do not have to guarantee the performance equivalent to “receiveTextData()”. Also, the handling of values specified by “timeout/idleTime” field depends on a model of receiver units. Performance in case sending fails or performance in case reading of superimpose data fails depends on a model of receiver units.

6 Operation of extended service (Optional)

Should be in accordance with ARIB STD-B24 Vol. 2 Appendix 3 “Optional guidelines for implementing extended services for fixed receiving system”.

6.1 Display of HTML contents (Optional)

To play HTML contents from BML contents by starting the internet browser, “Extended function like startResidentApp()” or “launchExApp()” is used. (“startResidentApp()” is recommended.)

Internet browsers started by “launchExApp()” shall be in accordance with ARIB STD-B24 Vol. 2 Appendix 1 section 8.5.2., and display of HTML contents transmitted by broadcasting (In other words, HTML documents with URI’s starting with “arib-dc://”) shall be possible. It is implementation dependent if internet browsers started by “startResidentApp()” may support the display of transmitted HTML contents, or not.

details on the method to start internet browsers by “launchExApp()” shall be follow ARIB STD-B24 Vol. 2 Appendix 1 section 8.5.2.

It is implementation dependent if BML browsers might be terminated by starting internet browse,or not

Regarding the presentation of internet browsers, the contents written in Vol. 2 “9.3 Guarantee of uniqueness of broadcasting programs and contents” should be considered.

6.2 Print function

Regarding functions and specifications relating to the print function, theyshould be in compliance with ARIB STD-B24 Vol. 2 “7.6.17 Functions for printing” and Vol. 2 Appendix 1 “Guidelines on functions for printing”.

6.2.1 Expansion API group

Since the print function is an implementation option, before execution of print related functions,, broadcaster should check by “getBrowserSupport()” if related processes in receiver is possible and should call print related functions only when the process is possible.

Print related functions are classified into the following 2 groups.

A) Function group to print with printers:

- getPrinterStatus()
- printFile()
- printTemplate()
- printUri()
- printStaticScreen()

B) Function group to store printing data into memory cards

- saveImageToMemoryCard()
- saveHttpServerImageToMemoryCard()
- saveStaticScreenToMemoryCard()

Either A) or B), or both A) and B) can be supported by receiver. Detection of the function implementation status of A) or B) of receiver units is executed by specifying “APIGroup” to “functionname” of “getBrowserSupport()”. The extended Functions Group specification that specifies “additionalinfo” in case “functionname” is “APIGroup”, refer to section 5.9.6.6.

6.2.2 Print data format

Print data (XHTML-Print) should be in compliance with ARIB STD-B24 Vol. 2 Appendix 1 “Guidelines on functions for printing”. However, still image files (JPEG, PNG) referred from XHTML documents for printing, they may be shared by presentation and print in BML. In some cases, they may be exclusively for printing in other cases. Rules below should be applied in each respective case.

	Image file	Operation
In case of sharing with display and print.	JPEG	Refer to 3.2.1.
	PNG	Refer to 3.2.2 and ARIB STD-B24 Vol. 1 Part 2 Annex B However, when the color type is 3, then the PLTE chunk may not be omitted.
In case of exclusive printing	JPEG	ISO/IEC10918-1 baseline and JFIF(Jpeg File Interchange Format) and Exit The maximum picture element size is 2560x1920 Maximum file size 3MB* Maximum picture element count 5Mpix Sampling 4:2:0 or 4:2:2 Picture element aspect ratio 1:1(dimetric picture element)
	PNG	Compliance with W3C PNG Specification Ver1.0. However, file type should be “0” and interlace should be “0”. Maximum picture element size 2560x1920 Maximum file size 3MB* Maximum picture element count 5Mpix Picture element aspect ratio 1:1(dimetric picture element)

* In case of using printFile(), printTemplate(),the maximum size of the module total is limited to 1MB.

Regarding the size restriction of print data (XHTML-Print)is as follows.

API	Maximum size of XHTML-Print (Including CSS)	Total of XHTML-Print documents and referred still image files(Maximum size of 1 content)
printFile() printTemplate() printStaticScreen()	256KB	1MB(Total module size)
printUri()	256KB	Not specified

6.2.3 Supplementary information regarding each print related API

- The number of modules that can be specified in “printFile()”, ” printTemplate()”

  The maximum number of modules that can be specified by these functions is 8.
- Image file format that can be specified in “saveHttpServerImageToMemoryCard()”

  The file format of images that can be specified with this function is either JPEG or PNG specified in the table above. The extensions of the file names is either “jpg” or “png” respectively. The maximum size of image files is specified in the table above as well.
- Specifications in this volume for document resolution of data broascasting at the time of “printStaticScreen()” or “saveStaticScreenToMemoryCard()” execution

  This function can be used when document resolution of data broadcasting is 960x540. α composite between planes is not required. Also, the video plane is not rendered.
- Prescriptions regarding URI in printUri()

  The URI specified in printUri() shall be a resource in the form of http:// or https:// and less than 256 bytes in size. The URI of the image file referenced from XHTML-Print shall be less than 128 bytes in size.

  Contact the governing organization described in [Section 4] Appendix 13 regarding the root certificate for performing https communication in the service for printing the printable content that have been placed on the server using printUri().
- URI’s specified by “saveHttpServerImageToMemoryCard()”

  For URI’s specified by “saveHttpServerImageToMemoryCard()”, only resources indicated with http:// or https:// can be specified and it should be less than or equal to 256 bytes.
- Regarding the process for the function return values of “printFile”, “printTemplate”, “printStaticScreen”, “printUri”

  When printing is started, the printer might be out of paper,the receiver might not return proper return value indicated out of paper in submission time, because the printer are not impremented paper sensor as an implementation dependent.In this case, the return value of the function shall be 1: success. After that, the printer will notice about the out of paper status during processing period. Print errors, etc. during processing period may not be

reflected by the return value of the function.Because those errors are handled by the receiver or printer.If printing is in pending status due to running out of paper during processing period, the printer that has executed printing API such as printFile may be in the following status.

a) Printing job is pending in the printer and the buffer is full and out of paper.

b) Buffer is not full, but out of paper.

The status of a) is equivalent to both the API return values of -3 and -4. In other words, whether return value will be -3 or -4 in such case implementation dependent in the features of the receiver and printer.

In order to correspond to the above cases, as a function process on the calling side for the return values of -3 and -4, it should be handled in the same way as the “printer busy status”.

6.2.4 Regarding presentation by receiver

Just as in the example indicated below, messages displayed by the receiver and user interface can be displayed over broadcasting screen. The presentation of messagesby the receivers is implementation dependent, but the size of the display area on the screen should be as small as possible upon securing of the convenience of the users. Receivers do not present user interface prompting confirmation for the user operation within the function.

(1) All print related functions are synchronous functions, possibly implementation to connect with printers within function is assumed. In that case it could take much time, therefore, dialog windows to be currently in communication and to abort execution may be displayed. However, the receivers should delete the above mentioned dialog window when function is returned.

(2) While executing the printing function, if tuning, etc. is done by the user, then the receiver may terminate the execution of the function to tune, or the tuning may be pending and a user interface to confirm whether or not the printing process should continue may be displayed.

(3) For events that occur asynchronously in printers (paper jam, out of ink, etc.), from the perspective of user convenience, a message should be displayed.

In order to prevent printing by an incorrect operation of the user, presentation to confirm the print execution is done at the time of executing the print functions (printFile(), printTemplate(), printUri(), printStaticScreen() function. Receivers should provide a user interface that performs setup of printing conditions (changes of paper size, paper type, number of prints, etc.) separately from the behaviour of BML contents.

Also, at the time of executing store functions of print data to memory cards (such as

saveImageToMemoryCard(), saveHttpServerImageToMemoryCard(), saveStaticScreenToMemoryCard()), BML contents should provide messages prompting the insertion of a memory card to users and should correspond to the case file names are duplicated (contents that users can change or input file name by themselves). Directory in storage device is implementation dependent.

In case a data event is updated, even if the receiver displays a message, etc. in the print function, current document shall be terminated and re-presentation shall be performed by browser.

In the following cases, the receiver will delete above messages displayed in the print function (excluding the above mentioned (3)) and the user interface and printing is aborted.

- When an update of a data event has occurred in the ES being presented.
- When the tuning is executed.

If http:///(or https:///) is specified in “saveHttpServerImageToMemoryCard”, the function can take some time from function call to since the receiver will acquire print data via network. If some event occured while the receiver is acquiring print data, the event is put in an interrupting queue, however, it is not executed until function is returned from the function; this should be taken into consideration.

6.3 Activation of the IPTV browser (Optional)

6.3.1 Activation and display of the IPTV browser

IPTV service can be launched by activating IPTV browser[1] on the receiver from BML content, using startExtraBrowser() with its argument browserName set to a character string representing the browser name as shown in Table in Appendix-14 (1). Some constraints may have been set by the IPTV service operator on the value of argument uri which indicates the first content to be presented when the IPTV browser is launched. The value of argument uri shall therefore be specified according to the guidelines set by each IPTV service operator. Whether or not the BML browser will be terminated upon activation of the IPTV browser shall be implementation dependent. However, if the BML browser will not be terminated, care must be taken to ensure that the presentation will not be mixed up. In presenting programs and contents in the IPTV browser, the content described in Vol.2 “9.3 Guarantee of Uniqueness of Broadcasting Programs and Contents” shall be taken into consideration.

6.3.2 Returning to BML browser from IPTV browser

Argument returnURI of startExtraBrowser() is a hint information for returning to the BML browser from IPTV browser. However, if a return destination has been specified to this argument, it shall be desirable to present the startup document of the specified component (if

possible). The period for which the argument information will be stored on the receiver shall be implementation dependent.

6.3.3 Confirmation of the IPTV browser function from BML content

The getBrowserSupport() can be used to confirm whether IPTV browser is implemented on the receiver from BML content, by specifying the character string for when functionname is ResidentApp in Table in Appendix-14 (2) as argument.

If the return value of this function is 1, it indicates that the relevant IPTV browser can be launched using startExtraBrowser() from the BML browser. It shall be kept in mind that getBrowserSupport() will return 0 if the IPTV browser is implemented on the receiver but cannot be activated from BML content.

Similarly, functions of the IPTV browser on the receiver can be confirmed using the getBrowserSupport() function, by specifying the character string for when functionname is ExtraBrowserFunction in Table in Appendix-14 (2) as argument.

The getResidentAppVersion() function can be used instead to confirm whether any IPTV functions provided uniquely by the service operator have been implemented. To confirm whether any IPTV functions provided uniquely by the service operator have been implemented, a character string representing the browser of the relevant service operator as shown in Table in Appendix-14 (3) shall be specified to argument appName.

6.3.4 Return values of the getResidentAppVersion() function

Refer to Appendix-14 (4) for the character strings to be used as item names in the return value Array[4] of this function. The character string equivalent to UserAgent that can be placed in the final field of Array[4] is not defined in this document. However, it shall desirably be the same as the UserAgent string of the IPTV browser.

6.3.5 Positioning of the IPTV activation function startExtraBrowser()

Although the implementation of startExtraBrowser() is optional, it shall be desirable that a receiver implementing an IPTV browser that can be specified to argument browserName implements this function as well.

On a receiver implementing multiple IPTV browsers, which IPTV browser(s) can be launched from this function shall be implementation dependent.

[1] A browser for presenting the IPTV service defined by the IPTV Forum.

Appendix-1 Receiver unit common fixed colors

Receiver unit common fixed colors are indicated in Table:Appendix-1. There are 64 basic colors including a half transparent color and a transparent color.

This was created based on the following policies.

1) First 16 colors are in accordance with the palette colors of 8-bit character codes. 1 color is transparent.
2) The remaining will be allocated equally in the color space.
3) The Alpha values are allocated equally as well.
4) According to the above policy, there will be 129 colors, so R,G,B,Alpha=255,255,170,128 is deleted.
5) Gamma correction is assumed.
   - Allocation level of RGB
     64 colors with 4 values of RGB= 0,85,170,255
   - Allocation level of Alpha
     3 values of Alpha=0,128,255

Values of RGB need to be converted to Y, Cb, Cr at the end, however, they are kept as RGB here for ease of understanding.

Table:Appendix-1 Receiver unit common fixed colors

(R,G,B = 0, 85, 170, 255 Alpha= 0, 128, 255)

Index value	R	G	B	Alpha	Name/Comments
0	0	0	0	255	Black
1	255	0	0	255	Red
2	0	255	0	255	Green
3	255	255	0	255	Yellow
4	0	0	255	255	Blue
5	255	0	255	255	Magenta
6	0	255	255	255	Cyan
7	255	255	255	255	White
8	0	0	0	0	Transparent
9	170	0	0	255	Half brightness Red
10	0	170	0	255	Half brightness Green
11	170	170	0	255	Half brightness Yellow
12	0	0	170	255	Half brightness Blue
13	170	0	170	255	Half brightness magenta
14	0	170	170	255	Half brightness Cyan
15	170	170	170	255	Half brightness White(Gray)
16	0	0	85	255	

Index value	R	G	B	Alpha	Name/Comments
17	0	85	0	255	
18	0	85	85	255	
19	0	85	170	255	
20	0	85	255	255	
21	0	170	85	255	
22	0	170	255	255	
23	0	255	85	255	
24	0	255	170	255	
25	85	0	0	255	
26	85	0	85	255	
27	85	0	170	255	
28	85	0	255	255	
29	85	85	0	255	
30	85	85	85	255	
31	85	85	170	255	
32	85	85	255	255	
33	85	170	0	255	
34	85	170	85	255	
35	85	170	170	255	
36	85	170	255	255	
37	85	255	0	255	
38	85	255	85	255	
39	85	255	170	255	
40	85	255	255	255	
41	170	0	85	255	
42	170	0	255	255	
43	170	85	0	255	
44	170	85	85	255	
45	170	85	170	255	
46	170	85	255	255	
47	170	170	85	255	
48	170	170	255	255	
49	170	255	0	255	
50	170	255	85	255	
51	170	255	170	255	
52	170	255	255	255	
53	255	0	85	255	
54	255	0	170	255	
55	255	85	0	255	
56	255	85	85	255	
57	255	85	170	255	
58	255	85	255	255	
59	255	170	0	255	
60	255	170	85	255	
61	255	170	170	255	
62	255	170	255	255	
63	255	255	85	255	
64	255	255	170	255	
65	0	0	0	128	Black
66	255	0	0	128	Red
67	0	255	0	128	Green

Index value	R	G	B	Alpha	Name/Comments
68	255	255	0	128	Yellow
69	0	0	255	128	Blue
70	255	0	255	128	magenta
71	0	255	255	128	Cyan
72	255	255	255	128	White
73	170	0	0	128	Half brightness Red
74	0	170	0	128	Half brightness Green
75	170	170	0	128	Half brightness Yellow
76	0	0	170	128	Half brightness Blue
77	170	0	170	128	Half brightness magenta
78	0	170	170	128	Half brightness Cyan
79	170	170	170	128	Half brightness White(Gray)
80	0	0	85	128	
81	0	85	0	128	
82	0	85	85	128	
83	0	85	170	128	
84	0	85	255	128	
85	0	170	85	128	
86	0	170	255	128	
87	0	255	85	128	
88	0	255	170	128	
89	85	0	0	128	
90	85	0	85	128	
91	85	0	170	128	
92	85	0	255	128	
93	85	85	0	128	
94	85	85	85	128	
95	85	85	170	128	
96	85	85	255	128	
97	85	170	0	128	
98	85	170	85	128	
99	85	170	170	128	
100	85	170	255	128	
101	85	255	0	128	
102	85	255	85	128	
103	85	255	170	128	
104	85	255	255	128	
105	170	0	85	128	
106	170	0	255	128	
107	170	85	0	128	
108	170	85	85	128	
109	170	85	170	128	
110	170	85	255	128	
111	170	170	85	128	
112	170	170	255	128	
113	170	255	0	128	
114	170	255	85	128	
115	170	255	170	128	
116	170	255	255	128	

Index value	R	G	B	Alpha	Name/Comments
117	255	0	85	128	
118	255	0	170	128	
119	255	85	0	128	
120	255	85	85	128	
121	255	85	170	128	
122	255	85	255	128	
123	255	170	0	128	
124	255	170	85	128	
125	255	170	170	128	
126	255	170	255	128	
127	255	255	85	128	

Appendix-2 Module compression format

In case of module compression, the same compression format as PNG is used based on RFC-1950(ZLIB Compressed Data Format Specification version 3.3). Refer to Table:Appendix-2.

Table:Appendix-2 Detailed operation of the zlib compression format

Field	Operation
Compression Method(4bit)	8 ("deflate") only
Compression Info(4bit)	Less than or equal to 7 (less than or equal to window 32KB)
Flags	
FCHECK(5bit)	(Values specified in RFC-1950)
Preset Dictionary(1bit)	0 (no preset dictionary) only
Compression Level(2bit)	(As desired. It is ignored at the time of decoding)

Appendix-3 Clip function in video plane

Of all the side panels attached at the time of sending video materials in a 4:3 aspect ratio in format of a 16:9 aspect ratio, the main objective of this function is for side panels in either the left side or right side to be in non-display mode (video is shifted to the left side or right side.) Therefore, the only video materials that this function can be applied to are the ones with “O” for the relationship of the following two in the following Table.

- “display_horizontal_size” of “sequence_display_extension”
- “horizontal_size_value” of “sequence_header” and “aspect_ratio_information”

The same function for the y-coordinate direction is not operated.

		display_horizontal_size		
		No specification of value(*1)	The same value as horizontal_size_value(*2)	Different value as horizontal_size_value(*2)
aspect_ratio_information	2(4:3)	x	X	O
	3(16:9)	O	O	x

(*1) In case sequence_display_extension is not used.

(*2) In case the respective values are 540 and 544, then they are regarded as equal.

Also, continuous location changes of video object put a load on the display function of the receivers, therefore, it is not preferred.

This function is applicable only if the scaling ratio is 128/128 and regarding HD video of HD data pixel size or SD video of SD data pixel size and the width, height attribute of object elements referring to TV video, which is the clip target, will match the size of data broadcasting pixel size respectively, in other words, if the data broadcasting pixel size is 960x540, then the corresponding attributes are 960px,540px respectively, or if data broadcasting pixel size is 720x480, then the corresponding attributes are 720px,480px respectively. This function is realized in models where the “div” element is clipped by the video plane frame. Therefore, the functions are specified by the left attribute of ancestor “div” elements of corresponding video objects. Operational restrictions on other clips are in accordance with ARIB STD B-24 Appendix 2 5.1.4. For example, in case the side panel on the left side of the screen is switched to a non-display mode, ”-” value is specified in the left attribute of the “div” element. BML descriptions and an example of display are indicated in the following. In case of using this function in contents, a separate display object should be located in the equivalent area on character shapes or the still image plane for the area where the video object is not located on the video plane.

BML description example

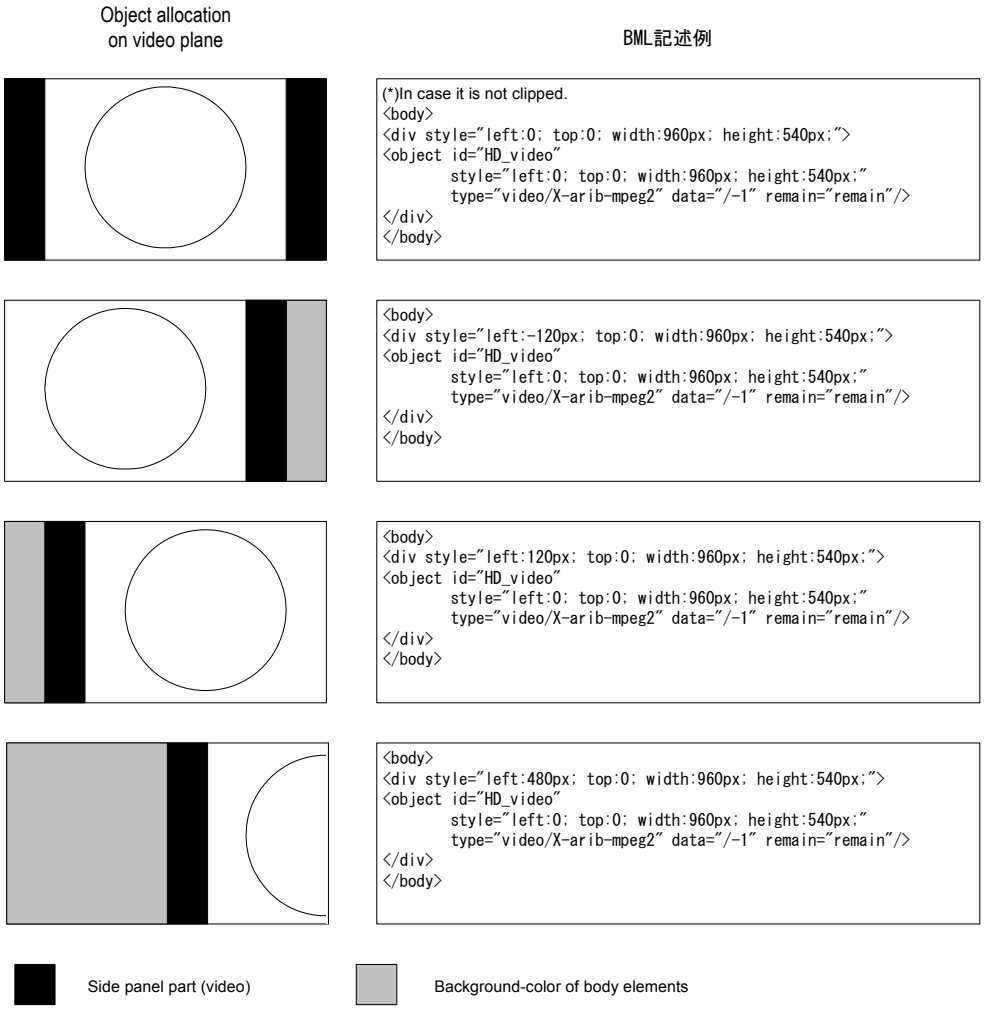


Figure Appendix-3　BML description and display example

Appendix-4 Regarding the scaling of videos

The scaling ratio is mapped to the relationship between the size of object referring to the video in the BML document and the MPEG pixel size of the video referred to by the object. The following indicates the height, width attributesthat are specifiable in object elements and a display example of videos that refer to video in BML documents.

The “display_vertical_size” and “display_horizontal_size” are specified by “sequence_display_extension”, and if those values are different from the “vertical_size_value” specified by the “sequence_header” or values specified by “horizontal_size_value”, then which one the scaling will be based on depends on implementation. Therefore, in object elements that refer to corresponding video, it is assumed that the size of the area where the video is actually displayed may be different. In consideration for this case, produce contents with care to other elements covering up the object element referring to the video.

Appendix 4-1. “Height”, “width” attributes that are specifiable in object elements

(1) In case data broadcasting pixel size is 960x540(16:9).

Number of lines (Vertical size)	Horizontal size		Operation for each signal type				Explanation
	16:9 signal	4:3 signal Excluding HD	HD 1080 lines	SD 480 lines	SIF 240lines	QSIF 120lines	
1080	1920	1440	O	- note)	-	-	HD standard
960	1706	1280	-	O	-	-	Maximum SD
944	1680	1260	O	O	-	-	
810	1440	1080	O	O	-	-	
674	1200	900	O	O	-	-	
540	960	720	O	O	-	-	
480	852	640	-	O	O	-	SD standard Maximum SIF
404	720	540	O	O	O	-	
270	480	360	O	O	O	-	
240	426	320	-	O	O	O	SIF standard Maximum QSIF
120	212	160	-	O	O	O	QSIF standard

O‥Operated -‥Not operated

In case a 16:9 size is specified at the time of a 4:3 signal, then the receiver unit will automatically execute operations such as the insertion of side panels, etc. and the receiver unit needs to display the video part while maintaining the aspect ratio.

note 1) The following values are specified for the width attribute and height attribute of object elements in BML documents.
height: 1/2 of the number of lines(vertical size)
width: 1/2 of the horizontal direction

(2) In case data broadcasting pixel size is 720x480(16:9) or 720x480(4:3)

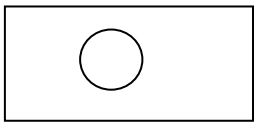
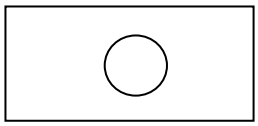
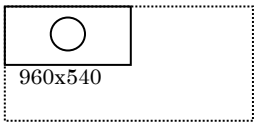
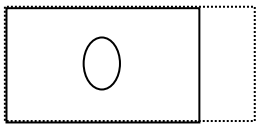
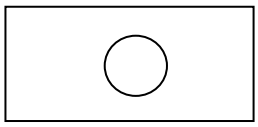
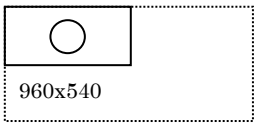
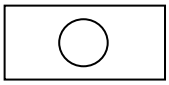
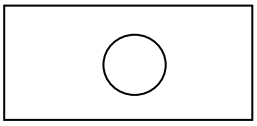
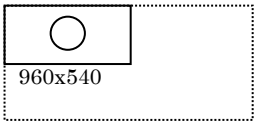
Number of lines (Vertical size)	Horizontal size	Operation by signal types				Explanation
		HD 1080 lines	SD 480 lines	SIF 240 lines	QSIF 120 lines	
480	720	-	O	O	-	SD standard Maximum SIF
420	630	-	O	-	-	
360	540	-	O	O	-	
300	450	-	O	O	-	
240	360	-	O	O	O	SIF standard Maximum QSIF
180	270	-	O	O	O	
150	224	-	-	O	O	
120	180	-	O	O	O	QSIF standard
90	134	-	-	O	O	
74	112	-	-	-	O	
60	90	-	-	O	O	
44	66	-	-	-	O	
30	44	-	-	-	O	

O‥Operated -‥Not operated

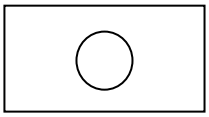
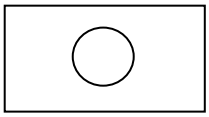
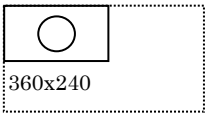
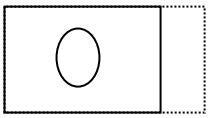
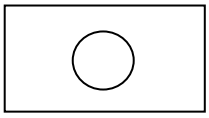
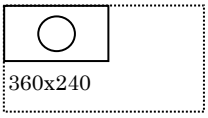
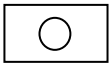
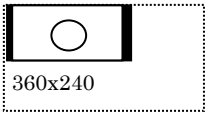
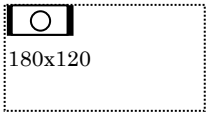
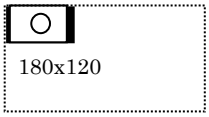
note 1) The following values are specified for the width attribute and height attribute of object elements in BML
height: number of lines (vertical size)
width: horizontal size

Appendix 4-2. Display example of videos

(Example 1) Presentation example by scaling of HD video data

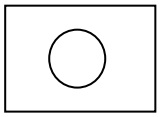
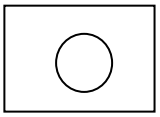
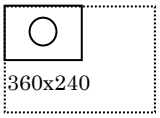
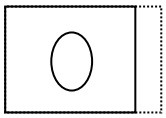
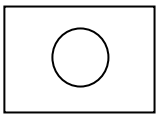
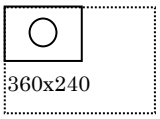
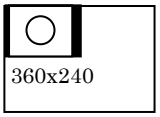
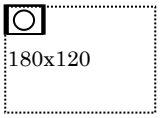
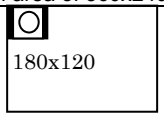
Video data and MPEG pixel size		Presentation on the video plane in case of data broadcasting pixel size 960x540 (16:9) (Scaling ratio =128/128)	Presentation on the video plane in case of data broadcasting pixel size 960x540 (16:9) (Scaling ratio =64/128)
16:9	 1920x1080	 1920x1080	 960x540 1920x1080
	 1440x1080	 1920x1080 Vertical direction is kept as 1080, and horizontal direction is mapped to 1440->1920 to make an aspect ratio of 16:9 and is presented.	 960x540 1920x1080
	 1280x720	 1920x1080 Mapped on 1920x1080(16:9) plane.	 960x540 1920x1080

(Example 2) Presentation example by scaling of SD video data (16:9)

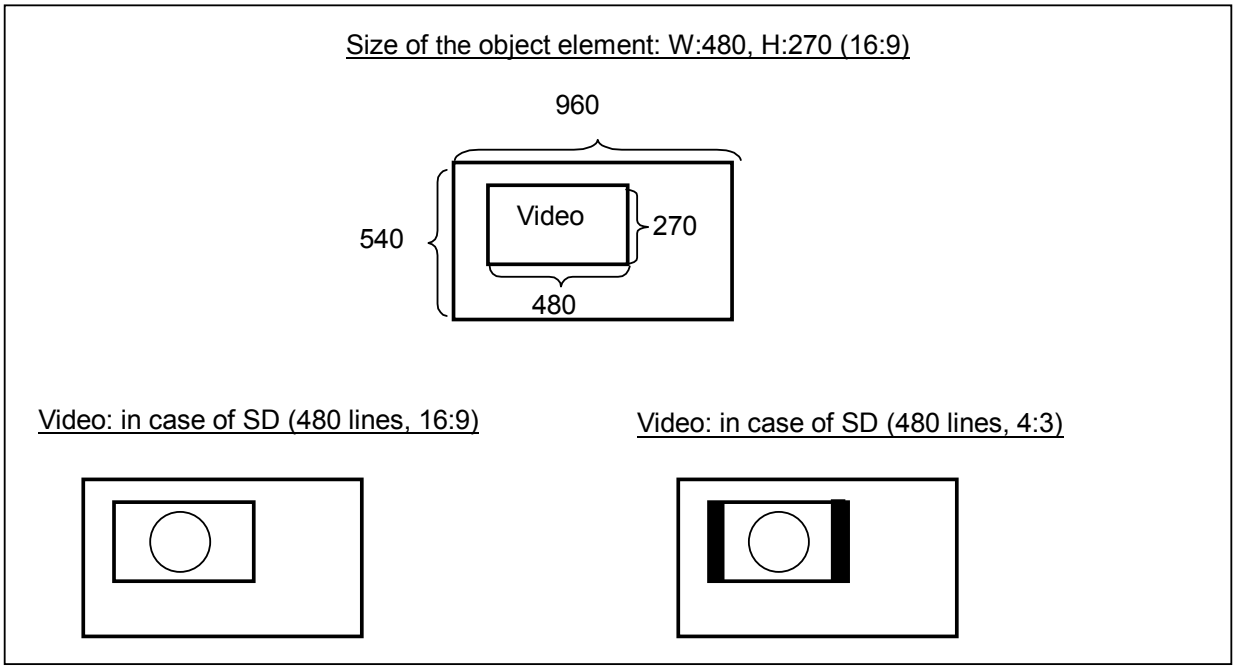
Video data and MPEG pixel size		Presentation on the video plane in case of data broadcasting pixel size720x480 (16:9) (Scaling ratio =128/128)	Presentation on the video plane in case of data broadcasting pixel size720x480 (16:9) (Scaling ratio =64/128)
16:9	 720x480	 720x480	 360x240 720x480
	 544x480 480x480	 720x480 Vertical direction is kept as 480 and the horizontal direction is mapped to {544,480,352}->720 and is presented.	 360x240 720x480
	 352x240	 360x240 720x480 Side panels of 4 picture elements in the left and right respectively are attached and presented in a presentation area of 360x240.	 180x120 720x480
	 176x120	 180x120 720x480 Side panels of 2 picture elements in the left and right respectively are attached and presented in a presentation area of 180x120.	 90x60 720x480

(*) In data broadcasting, the base point of the video display image is the upper left corner.

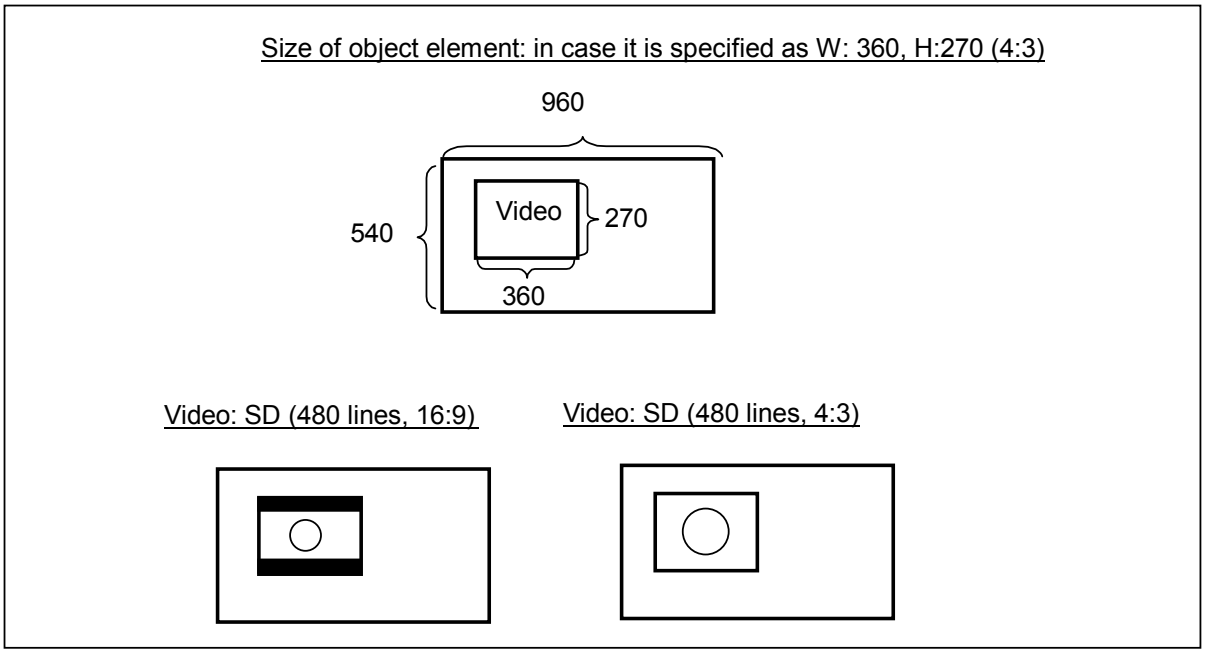
(Example 3) Presentation example by scaling of SD video data(4:3)

Video data and MPEG pixel size		Presentation on the video plane in case of data broadcasting pixel size720x480 (4:3) (Scaling ratio =128/128)	Presentation on the video plane in case of data broadcasting pixel size720x480 (4:3) (Scaling ratio =64/128)
4:3	 720x480	 720x480	 360x240 720x480
	 544x480 480x480	 720x480 Vertical direction is kept as 480 and the horizontal direction is mapped to {544,480,352}->720 and is presented.	 360x240
	 352x240	 360x240 720x480 Side panels of 4 picture elements in the left and right respectively are attached and presented in a presentation area of 360x240.	 180x120 720x480
	 176x120	 180x120 720x480 Side panels of 2 picture elements in the left and right respectively are attached and presented in a presentation area of 180x120.	 90x60 720x480

(Example 4) In case of data broadcasting pixel size 960x540(16:9), when presenting combinations of SD video



(Example 5) In case of data broadcasting pixel size 960x540(16:9), when presenting combinations of SD video (In case of 4:3 size specification)



Appendix-5　DTD for checking the operation area for basic services

Appendix 5-1　DTD

The objective of the following DTD is to check whether or not the BML document is in compliance with the operation area specified in this provision.

```
<!-- ======== Broadcast Markup Language (BML) 3.0 DTD [OPERATABLE] ======== -->
<!ENTITY %        ContentType "CDATA">
<!ENTITY %        Charset "CDATA">
<!ENTITY %        Character "CDATA">
<!ENTITY %        LanguageCode "NMTOKEN">
<!ENTITY %        Number "CDATA">
<!ENTITY %        URI "CDATA">
<!ENTITY %        Script "CDATA">
<!ENTITY %        StyleSheet "CDATA">
<!ENTITY %        Text "CDATA">
<!ENTITY % Events.attrib
"onclick %Script;          #IMPLIED
onkeydown %Script;    #IMPLIED
onkeyup %Script;        #IMPLIED">

<!ATTLIST a
        onfocus %Script; #IMPLIED
        onblur %Script; #IMPLIED
>
<!ATTLIST input
        onfocus %Script; #IMPLIED
        onblur %Script; #IMPLIED
        onchange %Script; #IMPLIED
>
<!ATTLIST body
        onload %Script; #IMPLIED
        onunload %Script; #IMPLIED
>
<!ATTLIST div
        onfocus %Script; #IMPLIED
```

```
        onblur %Script; #IMPLIED
        accesskey %Character; #IMPLIED
>
<!ATTLIST p
        onfocus %Script; #IMPLIED
        onblur %Script; #IMPLIED
        accesskey %Character; #IMPLIED
>
<!ATTLIST object
        onfocus %Script; #IMPLIED
        onblur %Script; #IMPLIED
        accesskey %Character; #IMPLIED
>
<!ATTLIST span
        onfocus %Script; #IMPLIED
        onblur %Script; #IMPLIED
        accesskey %Character; #IMPLIED
>

<!ENTITY % Core.attrib
"id ID #IMPLIED
class CDATA #IMPLIED
style %StyleSheet; #IMPLIED"
>
<!ENTITY % Common.attrib
"%Core.attrib;
%Events.attrib;"
>
<!ENTITY % Inlstruct.class "br | span">
<!ENTITY % Inline.class "%Inlstruct.class;
| a">
<!ENTITY % Inline-noa.class "%Inlstruct.class;">
<!ENTITY % Blkstruct.class "p | div">
<!ENTITY % Block.class "%Blkstruct.class;">
<!ENTITY % Boxed.mix "%Block.class;
| object
| input">
```

```
<!ENTITY % Br.content "EMPTY">
<!ELEMENT br %Br.content;>
<!ATTLIST br
      %Core.attrib;
      %Style.attrib;
>
<!ENTITY % Span.content "( #PCDATA | %Inline.class; )*">
<!ELEMENT span %Span.content;>
<!ATTLIST span
      %Common.attrib;
>
<!ENTITY % Div.content "( %Boxed.mix; )*">
<!ELEMENT div %Div.content;>
<!ATTLIST div
      %Common.attrib;
>
<!ENTITY % P.content "( #PCDATA | %Inline.class; )*">
<!ELEMENT p %P.content;>
<!ATTLIST p
      %Common.attrib;
>
<!ENTITY % Script.content "( #PCDATA )">
<!ELEMENT script %Script.content;>
<!ATTLIST script
      src %URI; #IMPLIED
>
<!ENTITY % Style.content "( #PCDATA )">
<!ELEMENT style %Style.content;>
<!ENTITY % A.content "( #PCDATA | %Inline-noa.class; )*">
<!ELEMENT a %A.content;>
<!ATTLIST a
      %Common.attrib;
      href %URI; #IMPLIED
      accesskey %Character; #IMPLIED
>
<!ENTITY % Object.content "EMPTY">
```

```
<!ELEMENT object %Object.content;>
<!ATTLIST object
        %Common.attrib;
        data %URI; #IMPLIED
        type %ContentType; #IMPLIED
        remain (remain) #IMPLIED
        streamposition %Number; "0"
        streamstatus (stop | play | pause) #IMPLIED
>
<!ENTITY % InputType.class "( text | password )">
<!ENTITY % Input.content "EMPTY">
<!ELEMENT input %Input.content;>
<!ATTLIST input
        %Common.attrib;
        type %InputType.class; "text"
        value CDATA #IMPLIED
        disabled (disabled) #IMPLIED
        readonly (readonly) #IMPLIED
        maxlength %Number; "40"
        accesskey %Character; #IMPLIED
        inputmode (direct | indirect | none) "none"
        charactertype (all|number|alphabet|hankaku|zenkaku|katakana|hiragana) "all"
>
<!ENTITY % Title.content "( #PCDATA )">
<!ELEMENT title %Title.content;>

<!ENTITY % Meta.content "EMPTY">
<!ELEMENT meta %Meta.content;>
<!ATTLIST meta
        name NMTOKEN #IMPLIED
        content CDATA #REQUIRED
>
<!ENTITY % Head.content "( title, meta?, style?, link?, script*, bevent? )">
<!ELEMENT head %Head.content;>
<!ENTITY % Body.content "( div | p )+">
<!ELEMENT body %Body.content;>
<!ATTLIST BODY
        %Core.attrib;
        %Style.attrib;
```

```
        invisible (invisible) #IMPLIED
>
<!ENTITY % Bml.content "( head, body )">
<!ELEMENT bml %Bml.content;>
<!ENTITY % bevent.content "( beitem )+">
<!ELEMENT bevent %bevent.content;>
<!ATTLIST bevent
        id ID #IMPLIED
>
<!ENTITY % BMLEventType "(EventMessageFired|ModuleUpdated|ModuleLocked|TimerFired
|DataEventChanged|CCStatusChanged|MainAudioStreamChanged|NPTReferred
|MediaStopped|DataButtonPressed|IPConnectionTerminated)">
<!ENTITY % BMLTimeMode "(absolute|origAbsolute|NPT)">
<!ENTITY % beitem.content "EMPTY">
<!ELEMENT beitem %beitem.content;>
<!ATTLIST beitem
        id ID #REQUIRED
        type %BMLEventType; #REQUIRED
        onoccur %Script; #REQUIRED
        es_ref %URI; #IMPLIED
        message_id %Number; #IMPLIED
        message_version %Number; #IMPLIED
        message_group_id %Number; "0"
        module_ref %URI; #IMPLIED
        language_tag %Number; #IMPLIED
        time_mode %BMLTimeMode; #IMPLIED
        time_value CDATA #IMPLIED
        object_id CDATA #IMPLIED
        subscribe (subscribe) #IMPLIED
>
<!ENTITY % link.content "EMPTY">
<!ELEMENT link %link.content;>
<!ATTLIST link
        href %URI; #IMPLIED
>
<! End of BML DTD -->
```

Appendix 5-2 Description of the DTD declaration section

Description of the DTD declaration section is in accordance with the following.

```
<?xml version="1.0" encoding="EUC-JP" ?>
<!DOCTYPE bml PUBLIC
        "+//ARIB STD-B24:1999//DTD BML Document//JA"
        "http://www.arib.or.jp/B24/DTD/bml_1_1.dtd">
<?bml bml-version="3.0" ?>
```

Appendix-6 Precautions for Non-volatile memory access

- Regarding the lifespan of Non-volatile memory

Non-volatile memory is assumed to be implemented using the semiconductor memory device called “Flash memory” in general. This device has a limit on how many times it can be written and that is the lifespan of this device. The maximum limit number of times of writing as of year 2000 is about 100,000 times.

If information is to be accumulated along with time of contents presentation, then global variables and Ureg should be used. Regarding temporary data memory for the passing of data between multiple services, if the Greg memory is supported, then the Greg memory should be used. Using Non-volatile memory for temporary memory of data is not recommended.

Appendix-7 Root Certificate Descriptor

Format of Root Certificate Descriptor located in the module information area of DII is as follows.

Root Certificate Descriptor

Data structure	Bit number	Bit string display
root_certificate_descriptor(){		
descriptor_tag	8	uimsbf
descriptor_length	8	uimsbf
root_certificate_type	1	bslbf
reserved	7	bslbf
if (root_certificate_type == 0){		
for (i=0; i<8; i++){		
root_certificate_id	32	uimsbf
root_certificate _version	32	uimsbf
}		
} else {		
for (i=0; i<8; i++){		
reserved	64	bslbf
}		
}		
}		

root_certificate_type:

Root certificate type. If this field is 1, then it means the corresponding root certificate is a broadcaster exclusive certificate. If it is 0, then it means the corresponding root certificate is an general-purpose certificate. (The meanings of broadcaster exclusive certificate and general-purpose certificate are explained later.)

The following fields are located 8 times repeatedly and they correspond to the 8 memory areas of a root certificate.

root_certificate_id:

If the value of this field is “0xFFFFFFFF”, then it means that the root certificate stored in the corresponding root certificate memory area is not included in this module.

If the value of this field is other than “0xFFFFFFFF”, then it means that the root certificate stored in the corresponding root certificate memory area is included in this module. In such case, the value in this field means the ID to identify the root certificate. A unique value is allocated to root certificates operated in domestic digital broadcasting.

root_certificate_version:

If the value of the corresponding “root_certificate_id” is “0xFFFFFFFF”, then value of this field is ”0xFFFFFFFF” at all times. In other cases, this field indicates the version number of the root certificate. (It is not a version number of the certificate format)

Appendix-8 getBookmarkInfo2 function

- getBookmarkInfo2(): Information of bookmarked area specified by filename is obtained.

  Syntax:

  ```
  Array getBookmarkInfo2 (
      input String regison_name
  )
  ```

  Argument:

  region_name : URI that is indicating the entire bookmark area

  Return value:

  Array to store bookmark information: successful

  Array[0]: Number, number of all bookmark areas that are implemented

  Array[1]: Number, number of remaining bookmarks that can be registered.

  Array[2]: String, URI indicating the bookmark areas that can be registered as new.

  null: failure

  Explanation:

  Information relating to the entire bookmark area specified by “region_name” is returned as the return value of array data. The specification method of “region_name” is a URI indicating the entire Non-volatile memory area that is secured as the bookmark area such as “nvram://bookmark(bookmark area)” or “nvram://denbun(registration transmission area)”.

Appendix-9 Operation of argument of “getBrowserStatus()”

The functions that can be inspected with the combinations of “statusname” and “additionalinfo” in case sProvider is "TerrP" are shown in the table below.

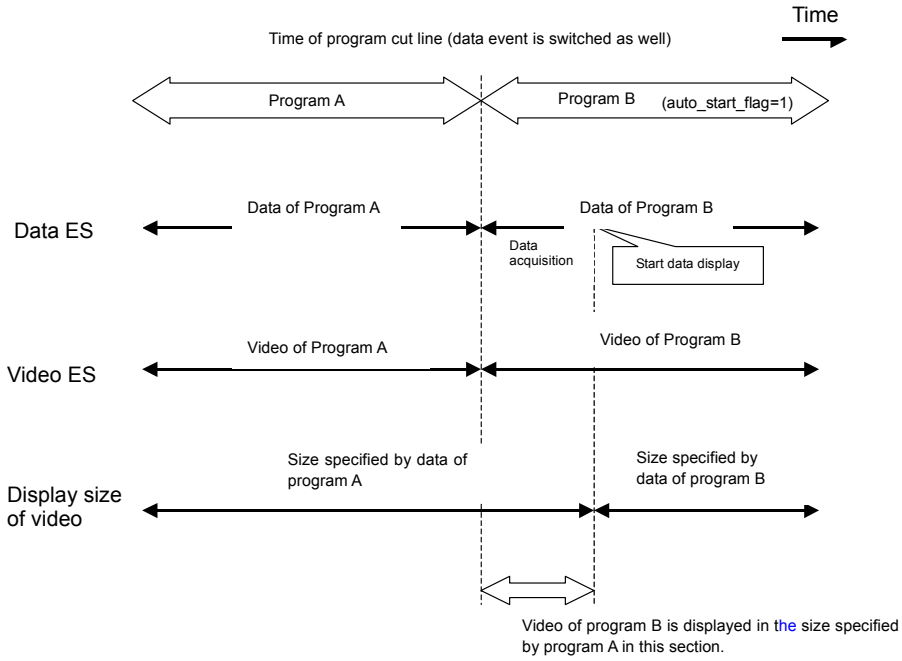
Character strings that can be specified as arguments of “getBrowserStatus()”

statusname	additionalinfo	Performance of “getBrowserStatus()”
IRDState	One of the following: “Broadcast” ”Link” ”UnLink”	If the browser is in a status specified by “additionalinfo”, then return 1. “Broadcast”:Reception condition of data broadcasting “Link”: Linked status “UnLink”: Unlinked status

Appendix-10 Precautions at the time of switching video pixel size

Precautions at the time of switching video pixel size are explained.

The following figure represents the transmission and presentation of data/video at the cut line of data broadcasting programs. In this figure, a case where data broadcasting program A is finished and the data event is switched and data broadcasting program B (auto_start_flag=1)is started is explained as an example. In order to simplify the explanation, only 1 data ES is used for both program A and B.

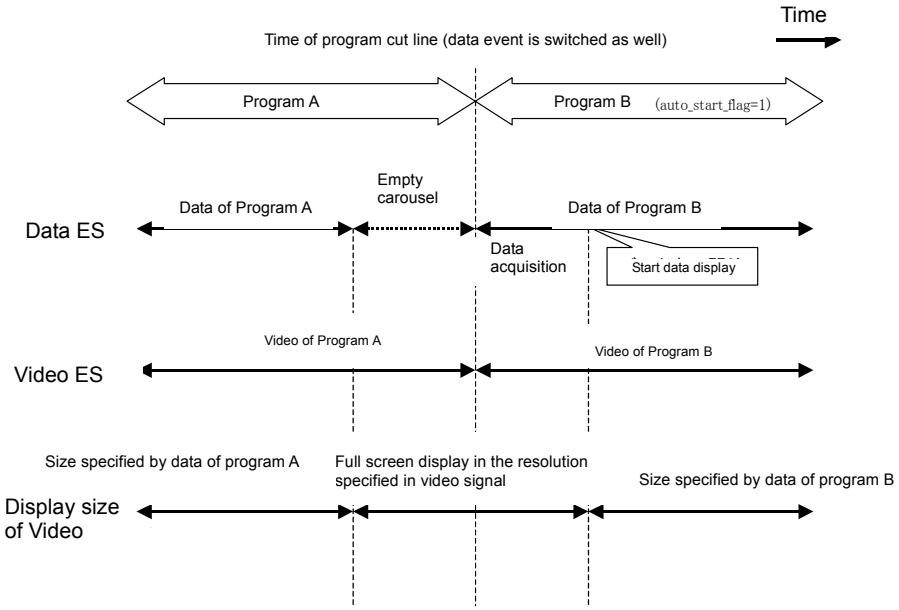


At the time of the program cut line, the display of video of program B is started, however, the display of data of program B is started with a short delay. (This is because acquisition of the startup document takes a certain length of time) During document acquisition, video continues to be displayed in the size specified before acquisition. (Refer to section 5.12.1) Therefore, video of program B is displayed in the size specified by the data broadcasting of program A during the period from the time of program cut line to the start of data display of program B.

In the above mentioned status, some attention may be required not to make scaling of the video incorrect. (Scaling not permitted in Appendix-4. For example, to display SD video of 480 lines in 1920x1080 pixel size, etc.) An example of a sending operation to avoid incorrect scaling is indicated in the following.

(Operation example 1)

It takes a certain amount of time before the video pixel size is switched, data ES is switched to empty carousel. (Figure below) At the reception time of an empty carousel, video is displayed in full screen in the original display pixel size specified to the video signal, therefore, incorrect scaling does not occur.



In case multiple data ES's are included in program A, all data ES's may be switched to empty carousel, or keep return flag of entry ES as 1 and only entry ES may be switched to an empty carousel. The same effect can be obtained by transparent contents (refer to section 1.2.2.) as well instead of empty carousels.

(Operation example 2)

Data broadcasting immediately before the switching of video pixel size, it should be combinable with all types of operated video signals. For example, if there are 2 types of video signals to be operated, which are HD1080 lines and SD480 lines, then

- Pixel size of data broadcasting itself is 960x540

-Video size specified by data broadcasting is 1440x810. (Specification of the object element in the BML document is width=720,height=405)

If data broadcasting such as the above is transmitted, then no matter which one of the 2 types of video signals is combined, incorrect scaling does not occur. (Refer to the table in Appendix-4 "(1) In case data broadcasting pixel size is 960x540(16:9)").

Appendix-11 Operation guidelines of information on data broadcasting non-volatile memory

1. Basic concepts relating to the handling of data broadcasting non-volatile memory

The following concepts are the basics of non-volatile memory used in broadcasting.

- Physical functions and performances are the target of manufacturer warranties.
- Personal information that are written belong to the users.
- Management of information and licensing with users are done by broadcasters.

2. Definition of personal information on non-volatile memory

Personal information on non-volatile memory handled in this guideline is in accordance with the definition of JIS Q 15001 “Requirement of compliance program relating to the protection of personal information”.

> Personal information is information relating to individuals; information that corresponds to individuals can be identified by names, dates of birth, other descriptions, numbers allocated for each individual, symbols and other types of codes, images and audio included in corresponding information etc. (Includes corresponding information that this information alone cannot be used for identification, but can be easily referred to in combination with other information to identify the corresponding individual.)

Also, data such as “points” used for prizes/games/ premium exchanges for free gifts needs to be carefully handled from the perspective of protecting viewers, and it is regarded as personal information handled in this guideline.

3. Specification method of viewers

Viewers should be specified, or consent from viewers is assumed for the execution of presentation, data deletion, restoration, etc. of personal information by data broadcasting contents. As a system to identify viewers, the following are conceivable.

- Specification of individuals by passwords
- Specification of receivers by CAS card ID
- Specification of households by caller ID.

4. Presentation functions guidelines of specifications, etc. for handling personal information by data broadcasting contents

- The contents of the following should be described in “membership specifications”, “FAQ” or “help”, etc.
  (1) Registered information is stored in receivers.
  (2) Personal information stored in receivers is managed by service users (members) themselves, and in case of transferring or discarding the receiver unit, it needs to be deleted by service users (members) themselves.
  (3) Broadcasters of corresponding services manage the registered information and maintain confidentiality and their usages should be clearly explained.
  (4) Broadcasters who use authorization by CAS card ID should clarify the procedure of non-volatile memory data handover (restoration) at the time of exchanging CAS cards, etc.
  (5) At the time of member registration, a disclaimer regarding the handling of points, etc. in case of data deletion should be presented to users and should be agreed to by the users.

- Contents should have a function to present the contents of non-volatile memory data (personal information, etc.) stored in receivers.
  (1) Section for “confirmation of memory contents” should be created on the menu screen for member registration, etc.
  (2) For presentation of non-volatile memory data, identification by password, for example, should be executed.

  All contents of non-volatile memory data (personal information, etc.) should be presented.
  However, for highly confidential information such as passwords and credit card numbers, the actual contents (numbers) will not be presented, and “****” or a “stored” message should be displayed.

5. Guidelines for the non-volatile memory data deletion function by data broadcasting contents

In case of operating services to store personal information in receivers in data broadcasting, each broadcaster (source of program or group) should have the following deletion functions.

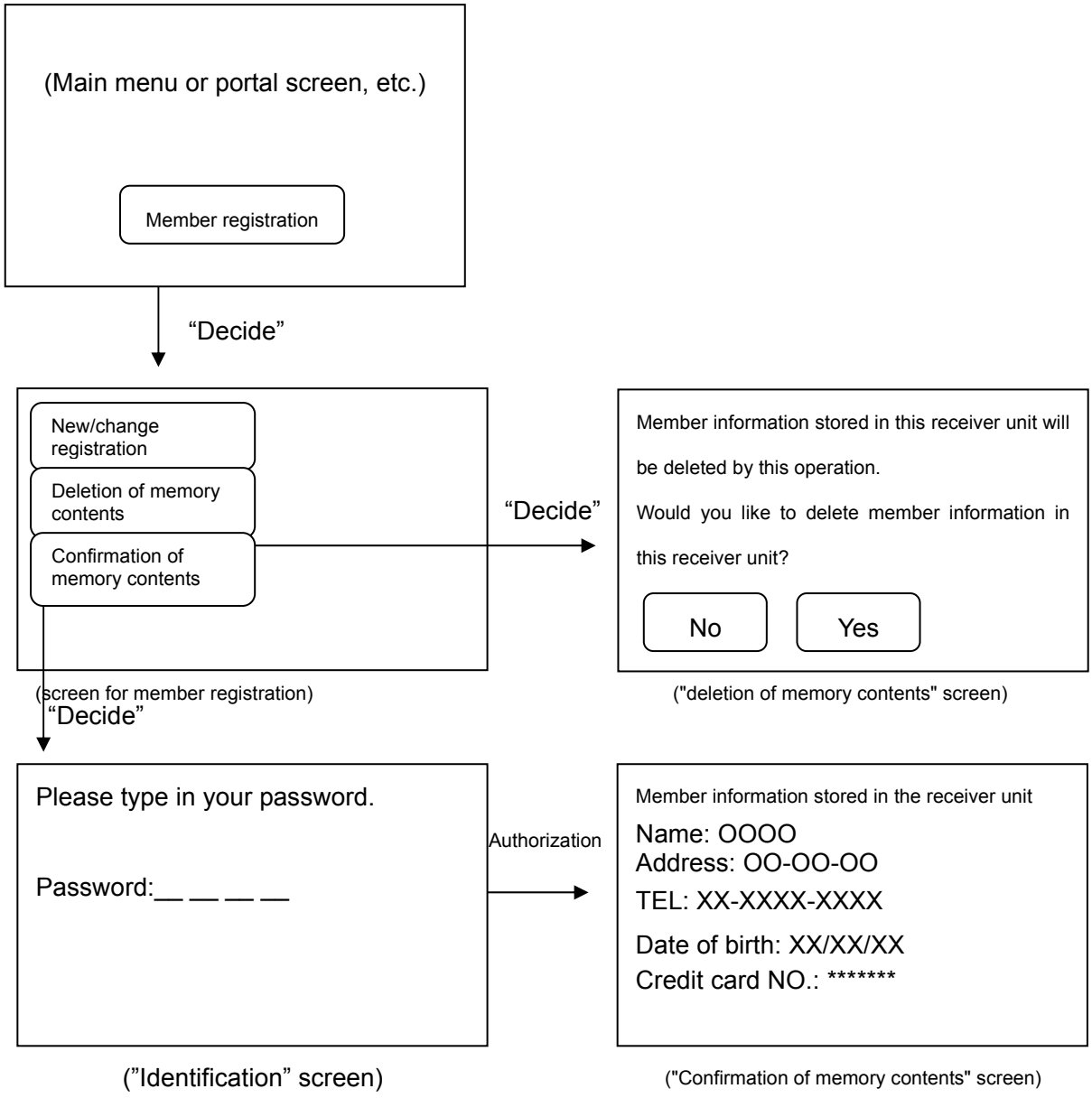
- Deletion target: Parts that the corresponding broadcaster will use exclusively among the following, area of non-volatile memory exclusively for Digital Terrestrial Television Broadcasters, Digital Terrestrial Television Broadcaster exclusive broadcasting communication common area and area exclusively for Digital Terrestrial Television Broadcaster groups.
- User interface: The following points should be implemented to improve operationality for viewers and to prevent wrong operation. (Refer to the operation screen example.)

- “Memory contents deletion” section should be created in the menu screen such as “member registration”.
- For deletion of actual area of non-volatile memory, final confirmation of the viewer’s intention should be executed.
- Considering the possibilities that non-volatile memory is accessed by procedures other than the official procedures of data broadcasting and receivers, 0x00 or 0x20 should be written in all deletion target areas.

Operation to execute only if there is an instruction for withrenderal from viewers without presenting the deletion function contents at all times is possible.

- Display example of specifications for handling personal information
  - Our company will have the responsibility to manage personal information such as membership numbers, addresses, names and credit card numbers.
  - Information of members will not be disclosed to any third parties without consent from our members.
  - Our company will secure the privacy protection of all members.
  - (In case of sending data in an uplink) Among the information entered just now and information that is already registered, information of ***, *** (Example: name, address, TEL, credit card number) will be sent. The use of this information by our company is limited to this service (Example: TV shopping) only.

- Non-volatile memory data deletion operation screen example



Appendix-12 Interaction channel communication based on connection status - Guidelines of return values for TCP/IP related functions

	Setup for each type of priority usage line	Connection status			
		(1) Disconnected status	(4)Automatic connection / Constant connection	(2)Explicit connection	(3)Abort
Return value of "isIPConnected()"	Dial-up	0	1	2	0 *1
	Constant connection	0	1	2	0 *1
Return value of "connectPPP()"	Dial-up	NaN、1 to -8	-100	-100	-100 *3
	Constant connection	NaN, 1 to-8 or -200 *5	NaN、1~-8 or -200 *5	-100	-100 *3
Return value of "connectPPPWithISPParams()"	Dial-up	NaN、1~-8	-100	-100	-100 *3
	Constant connection	-200	-200	-100 or -200 *2	-100 or -200 *2
Return value of "disconnectPPP()"	Dial-up	-1	Return 1 and disconnect the automatic connection.	Return 1 and disconnect the explicit connection and connection parameter is deleted.	Return 1 and connection parameter is deleted.
	Constant connection	-200 or -1 *4	-200 or -1 *4	Return 1 and disconnect "connectPPP" and return to constant connection setup.	Return 1 and delete connection parameter of "connectPPP" and return to constant connection setup.

*1 It is judged by whether the IP connection is established or not at the time of the function query.

*2 It can be interpreted as connected by "connectPPP" even though it is a constant connection receiver or it can also be interpreted as calling "connectPPPWithISPParams" is incorrect because of the constant connection receive r therefore, return values of -100 and -200 are permitted. Which one is returned depends on implementation.

*3 Connection parameters cannot be overwritten in aborted status. PPP connection setup is regarded as already established and returns -100.

*4 An interpretation to return -200 for reason that the connection setup is not dial-up, or another interpretation to return -1 because the call is not established to begin with, therefore, return values of -1 and -200 are permitted. Which one is returned depends on implementation

*5 For receivers that cannot perform dial-up setup, then a return value of -200 is permitted.

Appendix-13 Timing to initialize closed caption

	At the time of updating closed caption management	Closed caption management text Format selection (SWF)	At the reception time closed caption text	At the reception time Closed caption text Data unit	Closed caption text, text screen deletion (CS)	Closed caption text, text , format selection (SWF)
Format (SWF)	Format field of management data	Value specified by corresponding SWF	Not initialized	(*1) Value specified by management data texts	(*1) Value specified by management data texts	Value specified by corresponding SWF
Display location (SDF,SDP)	Table 4-13	(*2) Value specified by management data texts	Not initialized	(*2) Value specified by management data texts	(*2) Value specified by management data texts	(*2) Value specified by management data texts
Character size controls (SSM)	Table 4-15	(*3) Value specified by management data texts	Not initialized	(*3) Value specified by management data texts	(*3) Value specified by management data texts	(*3) Value specified by management data texts
interval between characters/lines (SHS,SVS)	Table 4-15	(*3) Value specified by management data texts	Not initialized	(*3) Value specified by management data texts	(*3) Value specified by management data texts	(*3) Value specified by management data texts
Other status instruction	STD-B24	STD-B24	STD-B24	STD-B24	STD-B24	STD-B24

(*1) Value of format field of management data if not specified by management data text

(*2) Value of Table 4-13 if not specified by management data text

(*3) Value of Table 4-15 if not specified by management data text.

Appendix-14 Arguments and return values for confirming whether the receiver implements the IPTV browser and its functions

(1) Character strings that can be specified as argument "browserName" to startExtraBrowser( )

browserName	Remarks
netTVBrowser	Browser standards represented by the browserName string are defined by the IPTV Forum (*).
IPTVBMLBrowser	

(2) Character strings that can be specified as argument to getBrowserSupport( )

functionname	additionalinfo	Remarks
ResidentApp	netTVBrowser	Browser standards represented by the additionalinfo string are defined by the IPTV Forum (*).
	IPTVBMLBrowser	
ExtraBrowserFunction	netTVVOD	Browser functions represented by the additionalinfo string are defined by the IPTV Forum (*).
	netTVDownload	
	IPTVBMLVOD	
	IPTVBMLDownload	

(3) Character strings that can be specified as argument "appName" to getResidentAppVersion( )

appName	Remarks
netTVBrowser	Browser standards represented by the appName string are defined by the IPTV Forum (*).
IPTVBMLBrowser	

(4) Character strings to be used as item names in the return value Array[4] of getResidentAppVersion( )

appName	Character strings to be used as item names
netTVBrowser	Defined by the IPTV Forum (*).
IPTVBMLBrowser	

(*) Refer to the IPTV Forum specifications (URL http://www.iptvforum.jp/standard/).

[Section 3] Operational specifications related to profile B
(T.B.D.)

<Intentionally blank>

[Section 4] Operational provisions related to C-profile

1 Introduction

C-profile, transmitted by the partial reception layer is configured by the following data.

- Video data, audio data: Low-frame-rate and low-resolution picture (H.264, MPEG-4 AVC), MPEG-2 AAC
- Multimedia data: Data encoding complied with ARIB STD-B24, Vol. 2, Appendix 4
- Closed caption

Video, audio and data are integral as broadcasting content and, as a general rule, a C-profile receiver shall need to be able to present all of them. The functions that should be implemented on a C-profile receiver are described from Chapter 3 to Chapter 8. As an exception, the desired functions of a receiver presenting only video and audio are prescribed in Chapter 9 "Functions with which C-profile basic receiver unit that only receive video and audio".

An exception is when data broadcasting services cannot be completed for some reason, which the communication cannot be used even if a data browser is implemented. There, besides the functions that receiver units should have for reception playback including the multimedia data and the closed caption described in Chapter 3, are descriptions for the functions that the receiver units should have to only receive video and audio in Chapter 9.

2 (Unused number)

3 Functions that basic receiver units that receive video, audio and multimedia data

3.1 Configuration of the receiver units

C-profile basic receiver units that receive video, audio and multimedia data shall be compliant with ARIB STD-B24, Vol. 2, Appendix 4. The following are the specifications for each part of the process of C-profile basic receiver units as hardware configuration point of view, and the specifications for resources within the receiver units by the reference decoders.

It is a matter of product planning of each manufacturer not to implement specified functions, implement the same functions by other means or implement the functions which surpass specifications, and which are not restricted. When the issues occur because of the above, it is assumed broadcasters will not be able to handle the issues.

3.1.1 Hardware configuration

Refer to Figure 3-1 for the hardware configuration of C-profile basic receiver unit.

The digital broadcasting signal input into C-profile basic receiver units are transformed into transport streams by tuners and demodulators. The demodulated transport stream is, by the process of a transport stream decoder, demultiplexed into video, audio and other data and are then output as video stream by video decoding process, and audio stream by audio decoding process. By those processes, C-profile basic receiver unit playback video and audio in a stream format. The receiver transfers data within a data carousel in main memory or non-volatile memory and then process data with CPU. Also, not only the playback process of video and audio but also processes such as audio playback by presenting character and images are possible and, at the same time, by transferring data of main memory into the audio decoder. By using two-way communication lines better interactive performance is expected than analog TV viewing. From the respective viewpoint of the above mentioned hardware process performances, it is necessary to provide the specifications for the following functions.

(1) Receivable data by transport decoders
(2) Playback audio data of the stream system and stored audio data
(3) Playback video data of the stream system
(4) Presentation of video, still images, graphic images, characters, etc.
(5) Two-way communication function using communication lines
(6) Persistent storage data size
(7) Sufficient ROM size (for example, fonts, etc.)

(8) Memory size for decoding obtained data

(9) Guidelines regarding the handling the buttons and the remote controller

(1) TS decoder functions, (2) (3) (4) (7) for presentation functions, (5) for communication functions, (6) (7) (8) memory capacities, and (9) is for remote control functions are especially specified.

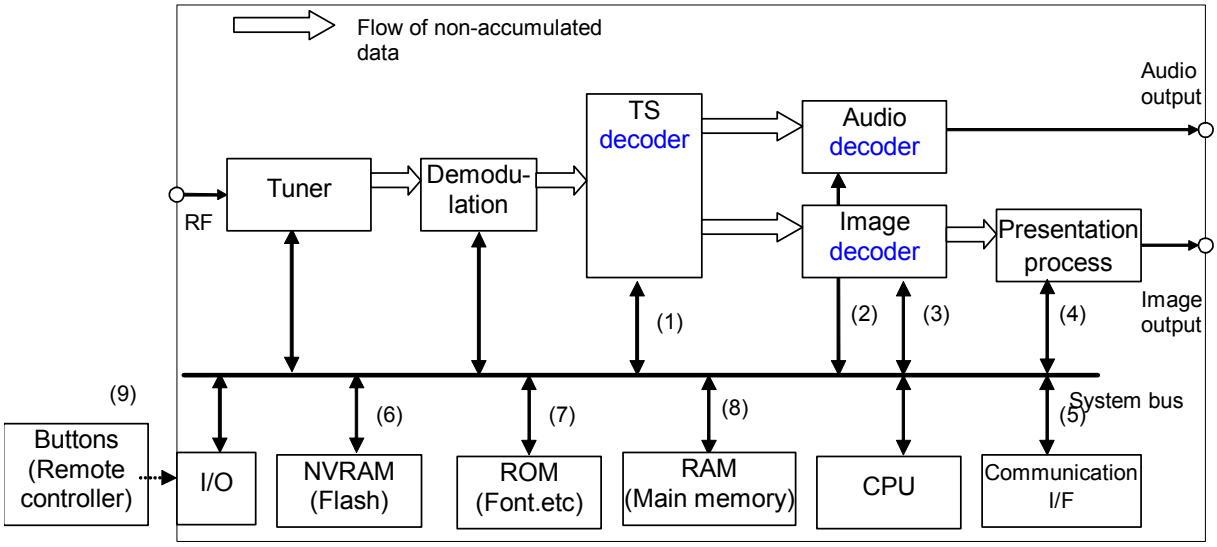


Figure 3-1 Hardware configuration of C-profile basic receiver units

3.1.2 Receiver unit reference model

The resources in the receiver units are classified according to the reference model of the receiver unit, and receivable limits are specified for data broadcasting in C-profile basic receiver units. Refer to Figure 3-2 for the reference model of the receiver unit .

The reference model is specified in compliance with the operation specifications based on the reference decoder models indicated on DAVIC 1.4 Part 9. If they are not specified in on this specifications, they shall be compliant with the definitions of DAVIC.

The received MPEG-2 TS, after filtering in each ES by the PID filter, is stored in the Bn, which is a main buffer, by the TBn, which is a transport buffer for the elementary stream of images and audio transmitted by PES. On the other hand, the elementary stream of multimedia contents transmitted by the data carousel is, after filtering by the PID filter, stored in the Bcontents by the TBn which has already been section filtered.

The received multimedia contents data will be executed by the viewer of the receiver unit. The multimedia engine loads data in the Bcontents following this command and executes

multimedia contents using Bwork as execution memory. At this moment, mono-media contents transmitted by the data carousel are specified from the Bcontents to each decoder, and mono-media contents transmitted by the stream are provided from the Bn to each decoder. The audio mono-media contents are presented by a presentation device of speakers after decoding. However, the visual mono-media contents is allocated in a display area after video, still images, and characters and images have been decoded respectively, and presented by the presentation devices. The presentation of closed caption specified in Chapter 6 is implementation dependent.

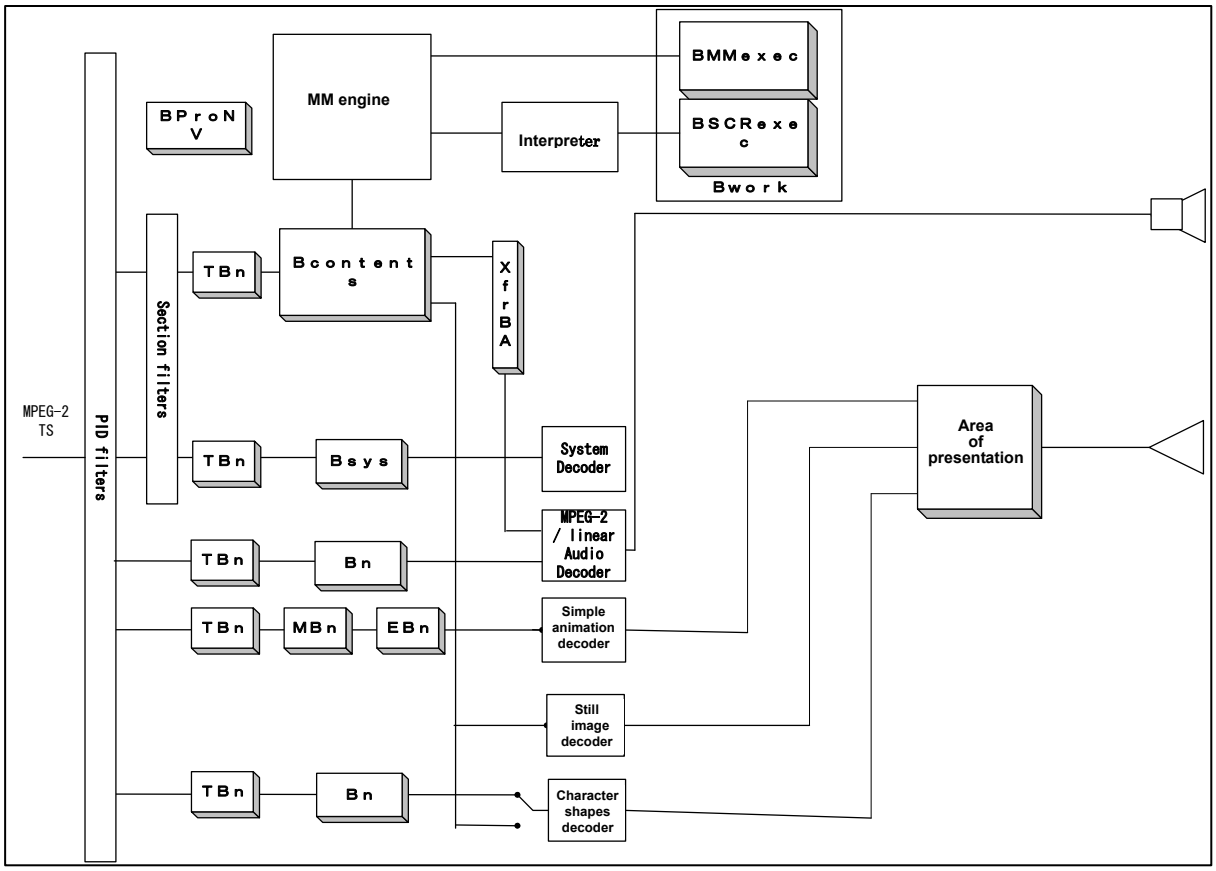


Figure 3-2 C-profile basic receiver unit reference model

TBn	Transport buffers TBn to deliver elementary stream n.
Bn	Main buffer of elementary stream n of decoders.
Bcontents	Buffer Bcontents to store multimedia contents data transmitted by a data carousel. When modules are transmitted as in a compressed format (allocates Compression Type Descriptor of DII), both data before and after compression (compression, decompression) are buffered.
XfrBA	Buffer used to transfer the file format of audio contents into the audio decoder.
BMMexec	Execution memory for multimedia code.

BSCRexec	Execution memory for scripts.
Bwork	Execution memory for multimedia contents in BMMexec and BSCRexec.
BproNV	Non-volatile memory to store information of each viewer and the information of each broadcaster.
Other buffers	Refer to ISO/IEC13818-1 and DAVIC 1.4 Specification Part 9 for details.

3.2 Presentation function

The presentation function of C-profile basic receiver units is modeled on both virtual planes which output the decoded results of each mono-media and a display buffer depend on the display device of the receiver units. The configuration of the display device is implementation dependent. Refer to the guidelines of ARIB STD-B24, Vol. 2, Appendix 4, section 5.1 for the presentation of video and the data in the display area, and Chapter 8 in this specifications.

3.2.1 Presentation technique of receiver units

The receiver unit should present video, closed caption and BML contents according following steps.

1. Decode video stream with a video decoder and output it to a video virtual plane (Y, Cb, Cr 4:2:0 format). The output result shall be (A).
2. Decode closed caption data with a closed caption decoder and output it to a closed caption virtual plane. This output result shall be (B).
3. Decode BML contents with BML browser and output it to a virtual plane for BML browser (width of 240 pixels, height of 480 pixels: each 8 bit format of RGB). This output result shall be (C).
4. Execute a scaling and color space conversion suitable for the display format of each receiver unit in each output from (A), (B) and (C) in order, then output them to the display buffer of the receiver unit. It is not necessary to overlap each virtual plane by alpha blending. Also, it is not mandate to display simultaneously both virtual planes for BML browser and closed caption virtual planes. This indicates that it is not mandate to execute parallel both BML browsers and closed caption decoders. Regarding BML contents, it is permitted to display only a part of them. However, a function for browsing any part of the virtual plane by the scrolling representation by the receiver unit feature has to be offered.

Refer to Figure 3-3.

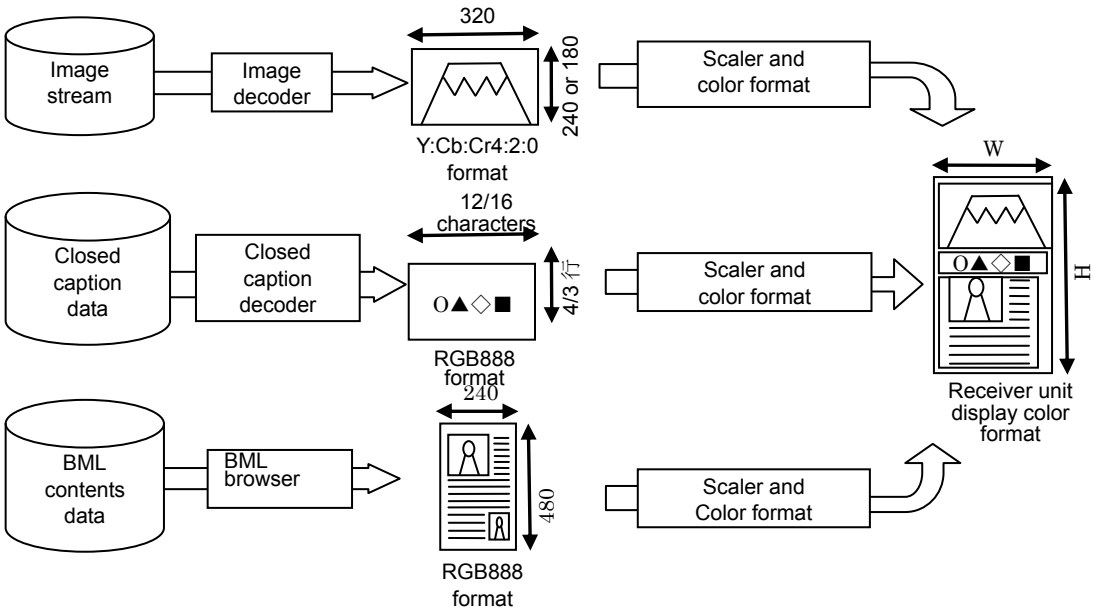


Figure 3-3 Presentation technique of receiver units

3.2.2 Pixel sizes and restrictions of each plane configuring screen

Table 3-1 indicates the specifications for the pixel size of each plane configuring display screen.

Table 3-1 Pixel size of screen planes

Item	Contents of specification	
Video virtual planes	Reso-lution	320 x 180 x 16, YCbCr (4:2:0), 16:9
		320 x 240 x 16, YCbCr (4:2:0), 4:3
BML browser virtual planes	Reso-lution	240 x 480 x 24, RGB each 8bit (*1)
Closed caption virtual planes	Reso-lution	Character writing direction normal size, more than 12 characters x 4 lines, or character writing direction normal size, more than 16 characters x 3 lines. 8 bits of each RGB (*2)

(*1) Regarding the color resolution of pixels; due to there may be receiver units that cannot be expressed by 8-bits resolution of each RGB ,it is recommended to consider using different distinctive colors for different color expression.

(*2) For details refer to section 6.3.

As for restriction matters on these planes, Table 3-2 indicates the specifications related to displayable mono-media codes, display position of mono-media codes, and size etc.

Table 3-2 Presentation of restricted matters in display planes

Item	Contents of specification	
Video virtual planes	Presentable mono-media codes	H.264|MPEG-4 AVC
	Presentation position	Implementation dependent. Not assigned by multimedia contents.
	Size	Same as the pixel configuration of virtual planes.
BML browser virtual planes	Presentable mono-media codes	JPEG, GIF, animation GIF
	Presentation position	Both x-y coordinate, from arbitrary pixels to arbitrary pixels
	Size	Both x-y coordinate, from arbitrary pixel count
	Overlapping	Restrictions on animation GIF and marquee (see section 7.11)
Closed caption virtual planes	Presentable mono-media codes	8-bit character codes for C-profile
	Presentation position	Not assigned by contents.
	Size	Medium size font (Implementation dependent)
Plane switching visual effect	Processing of the plane switching visual effect of presentation is implementarion dependent.	

3.2.3 Visual mono-media coding

The following are the specifications on the encoding method of visual mono-media desired to being presented in C-profile basic receiver unit. Refer to Chapter 5 for the operation details of each encoding method.

Table 3-3 indicates the specifications on the encoding method of visual mono-media desired to be presented in C-profile basic receiver unit.

Table 3-3 Visual mono-media desired to be presented

Encoding method		Contents of specicification	
Video coding	H.264| MPEG-4 AVC	Transmission method	Video PES; stream format identifier = 0x1B
		Video size	QVGA (320x240 (4:3) 320x180 (16:9)) (*1)
		Scaling	Implementation dependent. Not assigned by multimedia contents.

(*1) Order of numerical values is transversals and longitudinal

Table 3-4 indicates the specifications on mono-media encoding method desired to being

presented in BML browsers in C-profile basic receiver units.

Table 3-4 Visual mono-media desired to be presented on BML browser

Encoding method		Contents of specification	
Still image coding	JPEG	Transmission method	Data carousel; stream format identifier = 0x0D
		Image size	Maximum 240x320 (*1)
		Scaling	Implementation dependent. Not assigned by multimedia contents.
Graphic coding	GIF	Transmission method	Data carousel; stream format identifier = 0x0D
		Image size	Maximum 240x320 (*1)
		Scaling	Implementation dependent. Not assigned by multimedia contents.
	animation GIF	Transmission method	Data carousel; stream format identifier = 0x0D
		Image size	Maximum 240x240 (*1) (*2)
		Scaling	Implementation dependent. Not assigned by multimedia contents.
Font coding	Shift JIS	Transmission method	Data carousel; stream format identifier = 0x0D
		Font	Desired to be selected among small, medium, and large.
		Others	For operation details refer to ARIB STD-B24, Vol. 2, Appendix 4.

(*1) Order of numerical values is horizontal and vertical

(*2) Maximum 240 for both x-y coordinates. Refer to section 5.2.2.11 for details.

Table 3-5 indicates the specifications on the encoding method of desired visual mono-media being presented by closed caption in C-profile basic receiver unit. Refer to Chapter 6 for the operation details of each encoding method. However, the adoption of closed caption is implementation dependent.

Table 3-5 Desired visual mono-media to be presented by closed caption

Encoding method		Contents of specification	
Character coding	8-bit character codes for C-profile	Transmission method	Independent PES; stream format identifier = 0x06
		Font	Medium (Implementation dependent)

3.2.4 Audio playback functions

The specifications on each encoding method for the desired audio mono-media being presented in C-profile basic receiver units in shown in Table 3.6. Refer to Chapter 5 for operation details of each encoding method.

Table 3-6 Desired audio mono-media to be presented

Encoding method	Contents of specification	
MPEG-2 AAC	Transmission method	Audio PES; stream format identifier = 0x0F Data carousel; stream format identifier = 0x0D
	Sampling frequency	24kHz, 48kHz
	Maximum file size for continuous playback	128KB
Built-in sound	The encoding method is implementation dependent.	

3.2.5 Fonts

Table 3-7 indicates the specifications of desired fonts to be implemented into receiver units.

Table 3-7 Fonts

Item	Contents of specification
Number of Font face	Implementarion dependent
Character type	Kanji (level 1, 2), Hirakana, Katakana, alphanumeric, symbol, etc. (*1)
Character size controls	Recommended to be selected among small, medium, and large. The number of dots is not specified.
Grayscale font	Implementation dependent (Not assigned by contents)

(*1) For details about on types, refer to ARIB STD-B24, Vol. 2, Appendix 4.

3.3 Buttons (remote controller)

3.3.1 Keys to be used in C-profile reception

Keys expected to be used in reception are classified into following 3 groups.

- Group 1: Key events handling by BML contents
- Group 2: Key events handling by BML browsers
  (Do not handle by BML contents)
- Group 3: To be used for receiver features.

Table 3-8 indicates the type of keys to configure each group, contents production, and the guidelines at the moment of designing receiver units. It is recommended that each key have

its dedicated buttons however, the buttons can share similar functions. In this case, it is assumed that the names of the following key types and names of buttons are different by implementation dependent, and that at least 2 different names, such as one name used in receiver units and another name used by data broadcasting, are used for a single button. Also, multiple buttons shall provide functions for the same key types. The implementation of TVlink button is optional.

The method for key input is assumed to be implemented on the hardware by using dedicated buttons, and on the software by using on a screen keyboard. It is implementation dependent which the receiver unit adopt method, however, this specifications are specified for an implementation method in view of simple user operation that are required in the contents. So it is mandatory implementation that the configuration of receiver units with suitable buttons on a remote controller.

On the other hand, it is permissible implementation on a software that the configuration of receiver units without a suitable buttons on a remote controller, but in this specifications, it is mandatory implementation that the configuraition of the receiver unit with suitable buttons on a remote contoroller, for example, displaying a list of keys on a popup screen which is selectable by a single user operation, should be implemented to allow of directly user operation as possible.

Table 3-8 keys for C-profile

Key types		Implementation		Guidelines
		Mandatory / Optional	Physical Button	
Group 1: Key events handling by BML contents				
	Selection	◎	◎	Such as “Selection” and “Execution”
	Return	◎	◎	Such as “Cancel” and “Return”. Cancel of the user operation. Back space of user input character. (Or batch delete process) (*) BML document can be used for Return purpose however, pay consideration to the existence of the return destination.
	*	◎	◎	Selection (execution) of the user operation
	#	◎	◎	Selection (execution) of the user operation
	0 to 9 (numerical keys)	◎	O	Numerical input. However, the contents are assumed that numerical keys are not allocated to receiver unit specification or user setting, and both options of “Execute by up and down focus movement” and “select directly by numerical keys” have to be necessary.
	TVlink	Δ	Δ	Writing of TVlink
Group 2: Key events bandling by BML browsers (Do not handle by BML contents)				
	↑, ↓(Up and down keys) Jog dial	◎	◎	Movement of scrolling and selection focus.
	← , → (Left and right keys)	Δ	Δ	Movement of focus.

Group 3: To be used for receiver features				
	TV key	◎	O	TV On-Off functions. Selection of active window.
	Volume UP/DOWN	◎	O	Volume control
	TVlink list	◎ (*1)	O (*1)	Call TVlink list function
	Channel UP/DOWN	O	O	Tuning
	EPG key	O	O	EPG On-Off function

◎: mandatory, O: desired, Δ: optional

(*1) These are basic functions in this specifications, however, receiver units not implemented with TVlinks are exceptions.

3.3.2 Key masks

Multimedia contents of C-profile comply with ARIB STD-B24, and can execute key masks. But keys (numerical keys, “*”, “#”) supposed to be used as one-touch tuning cannot be executed by key masks unless it is necessary. Also, depending on the receiver unit, ”*” and ”#” may be assigned to one-touch tuning numbered 10 or higher, so the assign and release of key masks on numerical keys, ”*” and ”#” are desired to be done simultaneously. Also, in C-profile basic receiver units, it is assumed that main functions are not in C-profile basic receiver units but in mobile phones. In this case, in order to provide functions to devices in adaptive ways, it is assumed that the keys being acquired are not by multimedia contents.

3.4 Memory capability of receiver units

In C-profile basic receiver unit refer to section 3.1.2 for the specifications of each memory on receiver unit.

3.4.1 RAM

In the reference model, receiver units implement various memorys. As the use of RAM memory is expected, Bcontents and XfrBA memories are specified. See Table 3-9 for each RAM size. For details refer to section 3.1.2.

Table 3-9 RAM

Item	Contents of specification
Bcontents	512KB
XfrBA	128KB

For the buffer size of closed caption, refer to Chapter 6

3.4.2 NVRAM

BproNV, the mainly non-volatile memory implemented in receiver units of data broadcasting, as indicated in the reference model, stores information on each receiver unit user and specific information of each broadcaster. The area type and capacity usable as BproNV is indicated in Table 3-10. The area of BproNV, apart from the area of mandatory implementation, is capable of the implementation of the dynamically allocation areas where data is stored.

Table 3-10 Type and capacity of BproNV

Type	Capacity
C-profile memory area for the affiliation	288KB (12 affiliations x 24KB)
TVlink area	Over 12.8KB (More than 50 cases)

For details of BproNV, refer to section 7.2. Regarding about the area shown in Table 3-10, for security protection of receiver units, it should be configured in a way that the viewer cannot read or write related provisions, function of receiver units not specified in this specifications, and function of devices connected to receiver units. In order to satisfy the above constraints, a corresponding area type should be allocated within the memory device inside the receiver unit like NVRAM. For details refer to section 3.2.1.

3.5 Communication function

Assumed communication function by two-way communication on data broadcasting services of C-profile is shown in Table 3-11.

Since the communication functions related to mobile phone are dependent on the telecommunication line (i.e. telecommunications carrier), it shall be necessary to confirm them individually with the telecommunications carrier.

Table 3-11 Communication functions

Item	Contents of specification
Non secure communication	Use of communication functions nominating the namespace starting by “http://”. Communication with an http server on the Internet is required.
Secure communication	Use of communication functions nominating the namespace starting by “https://”. Communication with an https server on the Internet is required.

- The receiver shall implement HTTP/1.1 as defined in RFC2616.

- The receiver shall implement "Keep-Alive". Implementation of other headers shall be implementation dependent.
- The receiver shall implement the SSL3.0 and TLS1.0 protocols for encrypted communication.
- Cache-Control and pragma:no-cache are expected to be instructed explicitly due to program configuration. Therefore, if Cache-Control or pragma:no-cache has been specified, it shall be desirable that the receiver interprets it and operates accordingly. Refer to Appendix 11 "1. Cacheing by the receiver function" for the details of the cache function.

  Refer to Appendix 12 for the details of the communication function.

3.6 Character input function

The character input function is specified as a receiver's native application to assist character input to BML contents by user operation. The application supports character input into the input element and the textarea element of the BML contents.

The basic functions to be implemented for the application are mentioned below.

3.6.1 Function specification

The function specified by the application basically implementation dependent. However, the specifications are for character types that can be assigned in BML contents.

3.6.2 User interface

The character input application is permitted to receive exclusive key input and does not on the value of the used-key-list characteristic specified by the BML contents currently being presented.

3.6.3 Character types

Characters that can be entered into the input element and the textarea element refer to ARIB STD-B24, Vol. 1, Part 2, 7.3. However, the Kanji set allocated in Row 90 to Row 94 is excluded. For details refer to section 7.9 in this specification.

3.6.4 Kana Kanji conversion function

The implementation of the function and the function specifications are implementation dependent.

3.7 TVlink

3.7.1 Introduction of TVlink

In C-profile, the load on the functions and implementation required in mobile reception is considered, and an cross-media “bookmark” provided in ARIB STD-B24 and adopted by profile A is not used but TVlink is specified.

3.7.2 What is a TVlink?

This is a function to write the link information to a communication sites where information related to the programs stored in BML contents previously and is operated by the viewer pressing buttons.

The flow of the TVlink service is as follows.

1. There is a communication site where information related to the contents of broadcasting is offered, and when the broadcaster provides this service, the broadcasting contents present a TVlink icon on BML browser following the timing specified previously.
2. The viewer permits the writing of the TVlink by pressing a button when they are interested in the program contents (When the TVlink icon that is stored previously is displayed on the screen). The contents to write the TVlink are wrote to the URI and NVRAM of the communication site that provide the related information.
3. After that the viewer specifies the desired TVlink from the TVlink list screen offered as a receiver features, and browses the communication contents of the site with related information.

The contents for writing the TVlink is assumed in broadcasting as a part of the BML contents associated with A/V program and CM contents.

The destination link to be written as a TVlink is assumed to be the external HTTP server.

3.7.3 Changed points from profile A bookmarks

The changed points from profile A bookmarks of the TVlink are mainly the following three points. See Table 3-12.

(1) "TVlink list" by receiver features transits to the destination link when the viewer selects the desired TVlinks from the displayed list and the list of the TVlink. The TVlink list cannot be displayed by the BML contents, or read the TVlink. Therefore, the implementation of a TVlink list by a receiver features is mandatory.

(2) Transition to the destination link when the viewers select the desired TVlink from the TVlink list, including transition to the type of TVlink that presents communication contents depends on the link conditions. (In profile A, when presenting the communication contents as linked status, implementation is done by a combination of the receiver features and BML contents. In C-profile, it is done only by receiver features.)

(3) Arranging the write items and the extended zone is omitted.

Table 3-12 Comparison between TVlinks and bookmarks of profile A

	Write	Read	List display / transition to destination link	Extension area
"TVlink"	BML contents	Receiver unit function	Receiver unit function	None
"bookmark" of profile A	BML contents / (receiver unit function)	Receiver unit function / BML contents	Receiver unit function (remarks) / BML contents	Exist

Remarks: When the bookmark is bmType=009, it is a combination between tuning by receiver features and transition by BML contents

3.7.4 NVRAM in TVlink

3.7.4.1 Number of TVlink areas

The number of TVlinks is implementation dependent, but writing more than 50 is recommended.

3.7.4.2 Sharing with other medias

The NVRAM area of TVlinks may be shared with bookmark writing service of other media if there are compatible with the specifications of TVlink type, parameter type, data length, and recordable/non-recordable. However, sharing is not possible with bookmark areas of profile A receiver units etc.

3.7.5 Format and details of TVlink

3.7.5.1 Format of TVlink

The parameters used by TVlink and the length of its data, necessity in writing to NVRAM, necessity in displaying the TVlink list function of receiver units are indecated in Table 3-13. The maximum required number of bytes for the NVRAM area necessary for 1 TVlink is assumed to be 256. However, the write format is arbitrary, and the length of the table data (bytes) is a reference because it is not read from BML contents.

Table 3-13 Data configuration of TVlinks

Parameter	Length (bytes)	CproBMtype (*1)					Remarks
		0	1	2	3	4	
title	Max. 40	O/O	O/O	O/O	O/O	O/O	
dstURI	Max. 60	Δ/Δ	O/Δ	O/Δ	O/Δ	O/Δ	(*2)
outline	Max. 130	O/O	O/O	◇/◇	◇/◇	◇/◇	List display not required
CproBMtype	1	O/Δ	O/Δ	O/Δ	O/Δ	O/Δ	
expire	5(*3)	◇/◇	◇/◇	◇/◇	◇/◇	◇/◇	YYYYMMDDHHmm format
network_id	2	Δ/X	O/X	Δ/X	Δ/X	Δ/X	
service_id	2						
affiliation_id	6(*4)						
reserved	10						

(Write/Display) O: Mandatory, ◇: Desired, Δ: implementation dependent, X: Prohibited

(*1) Necessity of support for each CproBMtype is implementation dependent.

(*2) Character used in URI specifications in ARIB STD-B24, Vol. 2, section 9.2.

(*3) Assumed to be written as a numerical value.

(*4) The affiliation_id described in the BIT is 1 byte each and maximum 6.

3.7.5.2 Details of each parameters

- Details of titles (title)
  - The character string displayed in the title column of the TVlink list is written.
  - The title is a maximum of 40 bytes. (The length code is not included). The title when exceeding 40 bytes, the receiver unit behavior is implementation dependent. However, it is recommended that the receiver unit writes 40 bytes, and does not write 41 bytes or more. Moreover, if the 40th byte is the first byte of two-byte character, it is recommended not to write the first byte.
  - It is necessary to write this information into NVRAM.

- Details of URI of destination links (dstURI)
  - The URI of communication contents of destination links is written.
  - The URI of the destination link is maximum of 60 bytes (The length code is not included). The URI of the destination link when exceeding 60 bytes, the receiver unit does not write them, and returns NaN as the return value.
  - The null character string is specified for memo information (CproBMtype=0).
  - It is necessary to write the information into NVRAM.

- Details of TVlink outline (outline)
  - The outline of communication contents of destination links is written.

- If no TVlink outline is given, a null character string is specified.
- The TVlink outline is to be a maximum of 130 bytes (The length code is not included). The TVlink outline when exceeding 130 bytes, the receiver unit behavior is implementation dependent. However, it is recommended that the receiver unit write 130 bytes, and does not write 131 bytes or more. Moreover, if the 130th byte is the first byte of two-byte character, it is recommended not to write the first byte.
- It is necessary to write the TVlink outline into NVRAM, TVlink type in memo information (CproBMtype=0), and the TVlink of C-profile linked contents (CproBMtype=1). Other types are optional. However, it is recomemened to write the TVlink outline for the improvement of the user operations.

- Details of TVlink type (CproBMtype)
  - The TVlink type indicates the TVlink of the destination link contents. The receiver unit can select a browser that can display the communication contents specified by the URI destination link using this information.
  - The response of the TVlink type is shown in Table 3-14.
  - TVlink types are specified by numerical values from 0 to 255.
  - It is necessary to write this information into NVRAM. Display is optional.

- Details of expiration dates (expire)
  - The expiration date shall be written as a final date of the TVlink.
  - The expiration date is always specified by year, month, day and time.
  - The argument of the write function (X_DPA_writeCproBM()) can be omitted. When omitted, the TVlink is always available.
  - Writing this information into NVRAM is optional.

3.7.6 Operation of TVlink types

Table 3-14 indicates the TVlink types.

Table 3-14 TVlink types

	TVlink type
0	Memo information (no destination link)
1	C-profile linked contents (BML contents)
2	C-profile unlinked contents (BML contents)
3	HTML contents
4	Communication contents assumed to use specific networks
5 to 255	Reserved

The operation of TVlink types is as follows.

- Operation of memo information (CproBMtype: 0)
  - This type is used to write text information related to the broadcast program in the memo.
  - URI destination link is not specified in this type. A null character string is specified for the argument of the write function (X_DPA_writeCproBM()). When characters other than null characters are specified, it fails in writing.
  - The writing and reading of the TVlink of memo information (CproBMtype=0) are necessary.

- Operation of C-profile linked contents (CproBMtype: 1)
  - This type is assigned to write link information to C-profile linked contents (BML contents) assumed to be presented by BML browsers.
  - The writing and reading of the TVlinks of C-profile linked contents (CproBMtype=1) are necessary.

- Operation of C-profile unlinked contents (CproBMtype:2)
  - This type is assigned to write link information to C-profile unlinked contents (BML contents) assuming the presentation by the browser for the C-profile contents on the internet.
  - The writing and reading of the TVlinks of C-profile unlinked contents (CproBMtype=2) are optional. It is recommended to be able to write the TVlinks of this type in receiver units which the browser for the C-profile contents on the internet.

- Operation of HTML contents (CproBMtype:3)
  - This type is, for example, used to assign link information to HTML contents that are assumed to be presented in HTML browser implemented in PDA's etc.
  - The writing and reading of the TVlinks of the HTML contents (CproBMtype=3) are optional. It is recommended to be able to write the TVlinks of the types in receiver units which the browser that can display HTML.

- Operation of contents on proprietary network (CproBMtype: 4)
  - This type is assigned to write link information of communication contents that are assumed to be presented in HTML browsers using specific networks implemented in mobile phones, etc.

- The writing and reading of the TVlinks of contents on proprietary network (CproBMtype=4) are optional. It is recommended to be able to write the TVlinks of this type in the receiver units which carrier's proprietary browser assumed to use a specific network with receiver units such as mobile phones, etc. On the other hand, it is recommended not to write the TVlinks of this type in receiver units without carrier's proprietary browser assumed to use a specific network.

3.7.7 Guidelines for the functions for each TVlink type and receiver unit

The following indicates the read operation of each TVlink type of the TVlink list function implemented as receiver features and the behavior of the TVlink write function (X_DPA_writeCproBM()) of each TVlink types.

- In case of memo information (CproBMtype=0)

  (Function behavior when writing)
  - When this type is assigned to the argument, it writes without conditions.
  - Writing fails when a null character string is assigned to the argument (URI of the destination link).
  - When a TVlink area of writing NVRAM cannot be secured, -1 is returned as a return value.

  (Receiver unit behavior when calling)
  - When this type is selected from the TVlink list by the user operation, only the TVlink outline (outline) is displayed, and the transition is not executed.

- In case of C-profile linked contents (CproBMtype=1)

  (Function behavior when writing)
  - In the case where this type is assigned to the argument, when the TVlink is written the receiver unit writes the network_id, service_id and affiliation_id of that broadcaster with individual TVlinks.
  - When the TVlink area of the NVRAM that writes cannot be allocated, -1 is returned as a return value.

  (Receiver unit behavior when calling)
  - When this type is selected from the TVlink list by the user operation, some of following behaviors are executed.

    (Behavior 1)
    - Tuning is by the writed network_id and service_id. The display of the Video display area, when the broadcaster is outside the reception area or when

tuning cannot be executed, is implement dependent. The presentation of the communication contents is executed even if tuning fails.

- The receiver unit obtain the contents specified for URI via communication line, and presented in BML browser as linked status. However, after tuning, and until the communication contents are presented, the broadcasting contents are not presented in BML browser. The behavior of not being able to transit by URI is implementation dependent.
- The access control of NVRAM is used by the wrote network_id and affiliation_id as the TVlink information regardless of the success or failure of tuning.
- The time handled by the Date object of ECMAScript, when tuning fails is recommended to be the TOT obtained by the receiver unit at last or the time revised by other means, or an error may be returned.

(Behavior 2)

- The receiver unit is not executed always reception or tuning and the contents specified in the URI via communication lines are obtained and presented in BML browser in a linked status.
- The access control of NVRAM is used by the wrote network_id and affiliation_id.
- In this case, it is recommended that BML browser display in full screen.
- The time handled by the Date object of ECMAScript is recommended to be a time revised by the TOT obtained by the receiver unit at last or the time revised by other means, or an error may be returned.

- In case of C-profile unlinked contents (CproBMtype=2)

(Function behavior when writing)

- When the receiver unit does not implement the browser for the C-profile contents on the internet, in principle it does not write and -2 is returned as a return value. However, when the presence of the implemented browser for the C-profile contents on the internet cannot be confirmed from BML browser, it does not follow this process.
- When the TVlink area of writing NVRAM cannot be allocated, -1 is returned as a return value.

(Receiver unit behavior when calling)

- When this type is selected from the TVlink list by the user operation, the browser for the C-profile contents on the internet starts and the C-profile unlinked contents

are displayed. The receiver units behavior that cannot start the browser for the C-profile contents on the internet is implementation dependent.

- In case of HTML contents (CproBMtype=3)

  (Function behavior when writing)

  - The receiver units, in principle, do not write when they do not have browsers that can display HTML, and return -2 as a return value. However, when the presence of an implemented HTML browser cannot be confirmed from BML browser, it does not follow this process.
  - When the TVlink area of writing NVRAM cannot be allocated, -1 is returned as a return value.

  (Receiver unit behavior when calling)

  - When this type is selected by the user operation from the TVlink list, the HTML browser starts, and the HTML contents are displayed. The receiver units behavior that cannot start browsers which can display HTML is implementation dependent.

- In case of specific network communication contents (CproBMtype=4)

  (Function behavior when writing)

  - The receiver units, which do not implement the browser that can display contents for a specific network like mobile phones, etc., in principle, do not write and return -2 as a return value. However, when the presence of the implemented browser cannot be confirmed from BML browser, it does not follow the process.
  - When the TVlink area of writing NVRAM cannot be allocated, -1 is returned as a return value.

  (Receiver behavior when calling)

- When this type is selected by the user operation from the TVlink list, a browser that can display contents for a specified network starts, and the communication contents are displayed. The receiver units behavior that cannot start the broadcaster is implementation dependent.

3.7.8 Writing of TVlinks

- The writing into NVRAM of TVlinks is done by the broadcasting contents and C-profile linked contents.

- The TVlink is written without fail based on the permission of the viewer. The contents should not be written automatically without the permission of the viewer.

- When the TVlink is wrote specifying CproBMtype for the option, it is recommended to display the TVlink once the presence of the correspondence of such receiver unit concerned for CproBMtype is examined by contents.
- From the viewpoint of the protection of personal information, the implementation of functions that write or update by receiver features for the TVlink of C-profile linked contents (CproBMtype=1) is not recommended.
- It is recommended not to implement the writing or the update function of TVlinks of other types by receiver features.
- Below, the contents that write the TVlink indicate the receiver behavior when the write function (X_DPA_writeCproBM()) is called.
  - The receiver unit judges whether the receiver unit can display the communication contents according to the TVlink type specified by the argument, and when it is not possible to display, it is recommended not to write it. In this case, the return value of the function returns -2. Moreover, it is recommended to display the reason why the viewer cannot write.
  - When there is no TVlink area available, the following wrote the TVlink can be deleted before writing new TVlinks. In this case, present the title of the TVlink to be deleted to the viewers and obtain user permission.

    (1) The ones that have expired.

    (2) The oldest TVlink.

    1 is returned as the return value of the function when successfully writing the TVlink (after obtaining permission from the viewers). When user permission is not obtained, a failure (NaN) is returned.

    Moreover, in receiver units that do not execute such deletion processes, when no TVlink area available, -1 is returned as a return value.

3.7.9 Guidelines of TVlink list functions

A list of TVlinks is displayed, and it is necessary that the "TVlink list" have a function that can transit the communication contents desired by the viewer for implementation as a receiver features. The function is indicated as follows. Issues not mentioned are implementation dependent.

3.7.9.1 Start and end of the TVlink list by receiver's native application

- Initiation of the TVlink list by the receiver's native application is done by using the buttons on the remote control unit of the receiver unit and the function (X_DPA_startResidentApp()) of the multimedia contents. This function is described later.

- When the TVlink list starts by the receiver's native application during the presentation of multimedia contents, the display of the multimedia contents may be interrupted. However, it is recommended that the presentation of the multimedia contents continues.
- The TVlink list screen display method, for example, is repeatedly displayed in the data area. At this time, it is recommended that the display of video continues.
- Note that key input may not be obtained by BML browser while the TVlink list screen is displayed when authoring contents.
- When the display of multimedia contents is interrupted and the TVlink list is displayed, and a TVlink is selected by the viewers, and the TVlink list ends without being transited, it is recommended that the interrupted multimedia contents be displayed again, and the presentation continues. The key input also returns to the multimedia contents.
- When a TVlink list starts during the display of multimedia contents, BML browser continues the asynchronized operation with the TVlink list while presenting the TVlink list, and the screen presented when the multimedia contents is displayed again after the TVlink list ends, and it depends on BML browser behavior while presenting the TVlink list. BML browser continue asynchronous operations with the TVlink list during the presentation of multimedia contents. Therefore, when the TVlink list starts during the presentation of multimedia contents and the TVlink list finishes to redisplay multimedia contents, the screen displayed depends on BML browser behavior executed during the presention of the TVlink list. For example, when the data event changes while displaying the TVlink list, the display contents of the data broadcasting at the moment that the TVlink list starts, and the display contents when the TVlink list ends are different. Even if the data screen is displayed, when the TVlink list display ends, the data broadcasting might not be displayed.

3.7.9.2 Display of TVlink information

- The TVlink list has the title list display function of the TVlink.
- The display of "URI destination link" is optional.
- "TVlink explanation" cannot necessarily be displayed in the list, and it can only be displayed by the user operating the buttons, etc.
- The display of the "Expiration date" is optional, however, display is recommended.
- The information indicated by "TVlink type" is optional.
- It is recommended that the "Bookmark" wrote by the communication browser other than BML browser should be distinguished clearly in view of the viewer. When the bookmark wrote by the TVlink and a communication browser is displayed by one application, special consideration is necessary. (Especially, TVlinks of C-profile linked contents (CproBMtype=1)).
- In the list display, it is recommended to display, for the understanding of the viewr, that the TVlink type is a memo information (CproBMtype=0) item without a destination link.
- Specification for other displays is implementation dependent.

3.7.9.3 TVlink selection and transition to the destination link

- The viewer can select the TVlinks from the TVlink list and display the communication contents shown in the URI link destination included in the selected TVlink.
- At this time, when necessary, by referring to the TVlink types listed in the TVlink, one can start a browser that can present the communication contents of the specified type.
- When an expired TVlink is selected by the viewer, it has to specify to the viewer if the expiration date has passed, and not to transit to specified communication contents. At this time, it is recommended to delete the TVlink with permission of the viewer.

3.7.9.4 Deletion of TVlink information

- TVlinks are deleted by a TVlink list function (receiver features). As it cannot be deleted from contents, implementation of the deletion function is necessary.
- Deletion of the TVlink is based on the permission of the viewer.
- It is recommended in adequate timing, that expired TVlinks be automatically deleted with permission of the viewer. (*)The timing of automatic deletion is implementation dependent.
- Additionally, the specification of the deletion function in the user interface, etc., is implementation delendent.

* As the writing of expiration dates is optional, it is necessary to note that TVlinks that have expired may not be deleted implementation dependent even if the expiration date is specified at the time of writing. In this case, a TVlink once wrote remains in the receiver unit until the viewer deletes it, so special consideration needs to be taken by broadcasting operators in order not to fail the communication contents at the destination link.

3.8 BML browser

Regarding the functions that BML browser should have, the hardware resources and the software resources are specified in these sections. In the following sections, the specification of transmission, mono-media and multimedia encoding is mentioned. This section, only provides specifications that are not included in the following sections.

3.8.1 Browser specific display

The browser should not display a browser specific display because these contents may hide the intended presentation content. This is a restriction to avoid display loss of the contents by the allocation of the browser’s behavior button for the URI display frame done in general and by a browser of the computer and the browser’s behavior. However, this restriction doesn't

intend to get rid of channel banners and permission etc, when executing the browser. Note that the regulations of mixed display apply to the display of the browser for the C-profile contents on the internet in operation, as well as to the browser for the C-profile contents on the internet. (Refer to section 8.1.4.)
For BML browser, individual display outside the display frame is not assumed. However, this does not restrict implementing a function by implementation dependent to assist the function required by the browser specified in this specifications or for the intention of assisting it. For example, the display of the scroll bar related to the scrolling representation etc., is assumed for this. On the other hand, the functions not provided in this volume, like of the display of URI contents, should not be displayed.

3.9 Browser for the C-profile contents on the internet

This chapter, in receiver units that receive video, audio and multimedia data specified by this chapter, specifies options that may be implemented in the browser for the C-profile contents on the internet as well as BML browser specified in this section etc.The use of each browser and each relationship is specified follows.

1) BML browser

BML browser is work based on C-profile specifications in this volume. The browser is executed in the data broadcasting reception status and linked status, and are used for presenting the provided contents when broadcasters services by broadcasting business, via broadcasting signals and communication lines.

2) Communication browser

The browsers for the contents on the internet are classified into the following the browser for the C-profile contents on the internet and the carrier's proprietary browser. In this volume, they are positioned as one of the receiver's native applications from the viewpoint of BML browser. Therefore, the start by BML browser as the specifications in Chapter 8, uses the same method (start by X_DPA_startResidentApp()) for starting applications. The browsers are used to present contents located on servers that supply services other than broadcasting. Mixing display with the broadcasting contents is limited.

a) Browser for the C-profile contents on the internet

The browser is the specifications (element, attribute, CSS, DOM, and ECMAScript, etc.) related to the BML of C-profile specicications in this volume. However, it is necessary to reatrict some of the extended functions. Moreover, it is prohibited to refer to and transit the broadcasting contents. (Refer to Appendix 5.)

b) Carrier's proprietary browser

It is assumed that browser is implemented based on the specifications defined by the carriers. In this volume, the specifications only for the area used as receiver's native applications in C-profile receiver unit are specified, and the specifitions for the internal part are not specified.

c) HTML browser

It is assumed that browsers are implemented for viewing HTML contents. In this volume, the specifications only for the area used as receiver's native applications in C-profile basic receiver unit are specified, and the specifications for the internal part are not specified.

The following figure indicates the relationship of each browser.
HTML contents can be presented by HTML browser implemented in the receiver unit besides the browser for the C-profile contents on the internet and the carrier's proprietary browser shown in the figure.

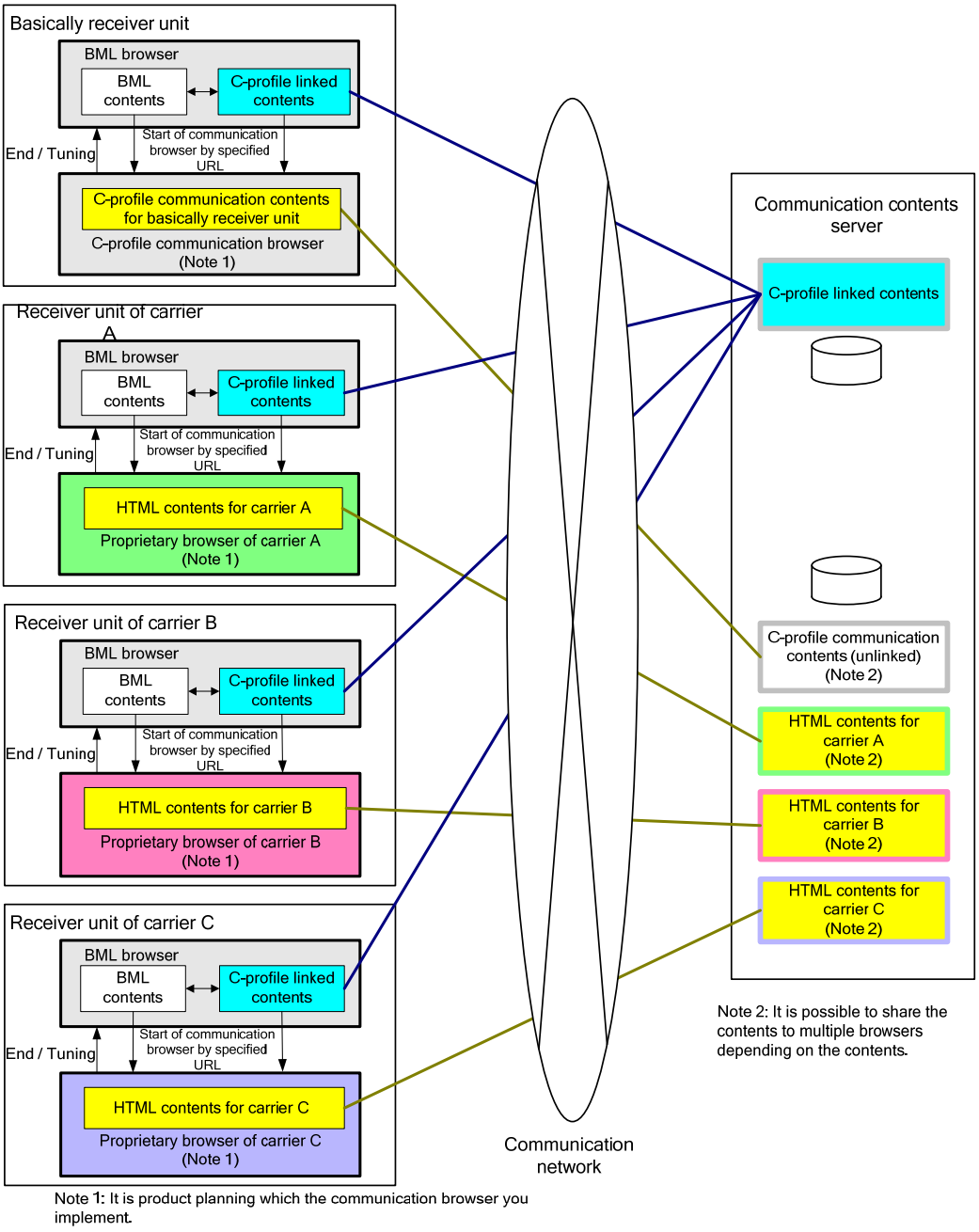


Figure 3-4 Relationship between BML browser and the browser for the contents on the internet

4 Operation of data transfer format

In this chapter, newly defined items, differences or operational limits related to the operation of the transmission method of the multimedia encoding method mainly as an XML base are described based on the provision contents with the described applicable document, as follows. In this chapter, the data broadcasting service indicates the multimedia data ıdcasting service by the multimedia encoding method of the XML base, unless otherwise ıtioned. Items without the following special description are applied as is for the provision contents within the applicable document.

- ARIB STD-B10 "Service information for digital broadcasting system"
- ARIB STD-B24 "Data coding and transmission specification for digital broadcasting"

4.1 PSI/SI

4.1.1 Type of data broadcasting services

4.1.1.1 Channel service type operated by data broadcasting programs

The value of the service_type is 0xC0.

4.1.2 Configuration of the data broadcasting service contents and operation of the components

4.1.2.1 Contents and local contents

The definition of contents and local contents and their relationship to events are shown in Table 4-1 and Figure 4-1　Local contents and events.

Table 4-1　Definition of contents and local contents

	Definition	Operation
Contents	Shows the sets of local contents transmitted in the event period of a certain component. It is identified by the Data Contents Descriptor.	The operation of the contents cannot be specified.
Local contents	Sets of BML documents transmitted in data events with a certain component.	Multiple local contents might be broadcast, alternatively one by one, in process of time, within one component.

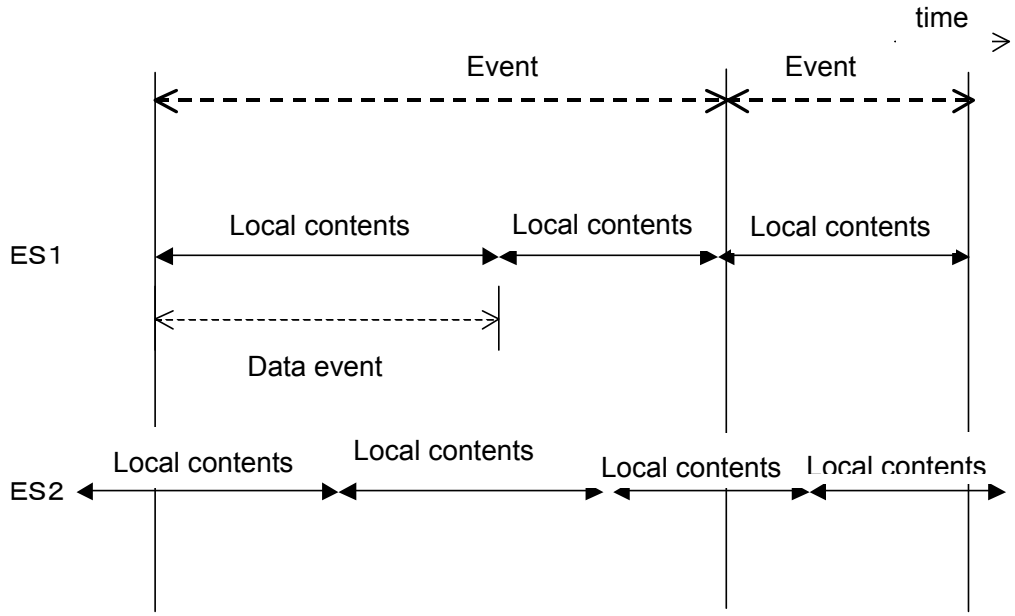


Figure 4-1 Local contents and events

4.1.2.2 Relationship between local contents and ES

A local content is transmitted by one data carousel (ES). The stream identification of the component that transmits the data carousel and the event message is always 0x0D.

4.1.2.3 Operation of component tags

The operation of the component tag value used by data broadcasting services including closed caption follows Vol. 7 section 5.1.1(2).

4.1.2.4 Configuration of the ES transmitted by 1 service

- Basic moving picture (H.264|MPEG-4 AVC): maximum 1ES
- Audio (MPEG-2 AAC): maximum 2ES
- Data carousel for C-profile: maximum 2ES
- Event massage for C-profile: maximum 2ES (do not operate NPT)
- Closed caption: maximum 1ES
- PCR: maximum 1ES

4.1.2.5 Identification of entry components

Default components of data-broadcasting programs are always transmitted by the tag value component of 0x80 in C-profile. Identification of the entry component that transmits a startup document when a channel is selected, is not the entry_point_flag of the Data Component Descriptor and is this tag value. The entry component transmits the entry module (moduleId=0x0000), and the entry module includes one startup document.

4.1.2.6 Identification of event message components

When general event messages are transmitted by the component that transmits the data carousel, they are transmitted by components with a component tag value of 0x80 and 0x8B. When a general event message is transmitted with an ES other than the component transmitting the data carousel, it is always transmitted by components with a tag value 0x89 and 0x8A. 0x89 and 0x8A are the ES for exclusive use general event messages.

4.1.2.7 Detailed specifications for section data transmission

- During the data carousel and the event message transition, multi section transmission shall not be conducted (transmitting two or more sections by 1 TS packet).
- The following standards are established in the transmission bit rate when each section data of the data carousel and the event message is transmitted.
  - Six TS packets or more are not transmitted continuously with the same PID.
  - The total bit rate of the component (4PID or less) received at the same time when one content is received is to be 650kbit/s or less. These include DII, DDB, and event messages.
  - The maximum bit rate of each 1 subtable is also 650kbit/s (2kB±100% for 32 milliseconds) or less.
  - It should be transmitted without exceeding the above-mentioned standards because when transmission exceeds these standards it is assumed that the reception efficiency of the section decreases depending on the receiver and the time required for the acquisition of necessary sections increases.

4.1.2.8 Default maximum bit rate for data broadcasting programs

Follows Vol. 7.

4.1.3 Operation of PMT

4.1.3.1 PMT operation of specific data broadcasting services

- Refer to section 4.1.5.2 for the receiver operation during the update of the PMT.
- The components that allow the ES not to exist although it is registered in the PMT are the following three.
  - Component transmitting data carousels
  - Component transmitting closed caption
  - Component transmitting only event messages
- However, the display indicating the data under reception might keep appearing for sometime due to the fact that receivers try to acquire data when the ES of the data carousel doesn't exist even though it is registered in the PMT. Empty carousels should be

kept sending into components of at least 0x80 even if there is no sending data on the transmission side in order to avoid this status. Refer to Appendix 8.

4.1.3.2 Operation of the Data Component Descriptor in the PMT

The Data Component Descriptor is allocated in the following components.

- Component transmitting closed caption
- Refer to Chapter 6 for detailed operation of the Data Component Descriptor allocated in the closed caption component.
- Component transmitting data carousels
- The Data Component Descriptor is not allocated in other components.
- The operation of the Data Component Descriptor is shown in Table 4-2.

Table 4-2 Operation of Data Component Descriptor

Flag	Operation
data_component_id	0x000D
Contents of additional_data_component_info(additional_arib_bxml_info())	
transmission_format	Shall be 00 (Data carousel transmission method and event message transmission method).
entry_point_flag	• Component component_tag=0x80 shall be 1. (When the data broadcasting program is selected, the component of component_tag=0x80 transmits the module including the document initially starts.) The rest of the components shall be 0. • When the receiver selects stations, acquires and presents a startup document of the data carousel transmitted by the component of component_tag=0x80.
auto_start_flag	Always auto_start_flag=1.
document_pixel size	Always 1111.
use_xml	0 (XML that uses the application dependence tag is not transmitted.).
default_version_flag	Always 0.
independent_flag	Always 1.
style_for_tv_flag	No operation. Value is fixed in 0.
bml_major_version, bml_minor_version	Operated as specified. If the bml_major_version is 12 and when these fields are allocated, the mobile basic receiver judges reception as possible. bml_minor_version always specifies 0. For receiver behavior concerning the version, refer to Appendix 9.
ondemand_retrieval_flag	Always 1.
file_storable_flag	Always 0.
Operation of additional_arib_carousel_info()	
data_event_id	Not operating in PMT. Value is fixed at 0xF(1111).
event_section_flag	Always 1.

4.1.3.3 Target Resion Descriptor

Not operated.

4.1.4 Operation of the L-EIT Data Contents Descriptor

Not operated.

4.1.5 Related receiver operation

Prior condition of data transmission operation specifications

- The data_event_id allocated in the Data Component Descriptor of the PMT data is assumed as fixed (0xF), and is not used.
- The component of the component_tag=0x80 is an entry component, and the data carousel transmitted by the entry component is called an entry carousel.
- Entry component (0x80) of C-profile may not exist. However, the PID value doesn't change while it exists. The receiver behavior when PID value changes is specified in section 4.1.5.2.

4.1.5.1 Receiver operation at the beginning of data broadcasting

1. Playback is executed a stream of basic moving picture (H.264|MPEG-4 AVC) if the component of the component_tag=0x81 is included in the PMT 2nd loop. Moreover, Playback is executed as an audio stream of MPEG-2 AAC (sampling frequency = 24KHz) if the component of the component_tag=0x83 or 0x84 is included. Similarly, Playback is executed out as an audio stream of MPEG-2 AAC (sampling frequency = 48KHz) if the component of the component_tag=0x85 or 0x86 is included.
2. Identifies the entry component (component of the component_tag=0x80) among the components placed in the PMT 2nd loop.
3. If the data_componet_id of the Data Component Descriptor is 0x000D, and the entry component (0x80) exists in the PMT 2nd loop, it is considered as a data broadcasting program by the multimedia encoding method (C profile) of XML base, then proceed to following process. The data broadcasting service doesn't begin if the data encoding method doesn't correspond to the receiver.
4. Based on the version number of the BML specified by the Data Component Descriptor of the entry component, the presentation of data broadcasting is judged. The data broadcasting service is not presented when reception is judged as impossible.
5. The BML-engine begins, and acquisition and presentation of the startup document of the entry component starts after initialization of the BContents and Ureg.
6. However, when the entry component is an empty carousel after the BML-engine starts, the switching of data event of the entry component is monitored, and when a startup

module appears due to data event switching, then the acquisition and presentation of the startup document is done. (Refer to section 4.2.3 for empty carousels.)

4.1.5.2 Receiver behavior during update of the PMT

Receiver behavior (for the receiver that receives C-profile) when the PMT is updated during data broadcasting program reception.

- When the component of C-profile disappears during viewing
  - ➢ Initiates the presentation of a startup document of the C-profile entry carousel by destroying the document under presentation.

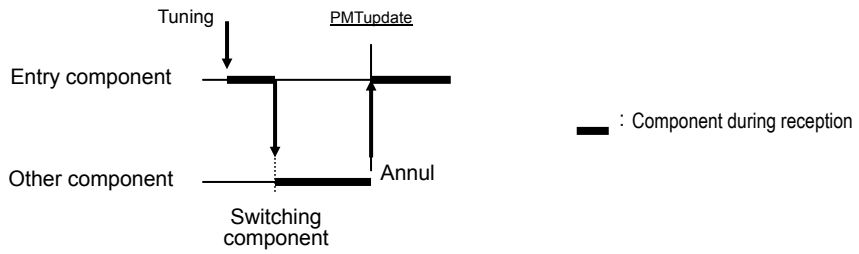


Figure 4-2 Operation when components disappear when the PMT is updated during reception

- When the entry component of C-profile (tag value = 0x80) disappears

  Since the component in the data broadcasting program disappears, the BML-engine ends. However, appendix 8 makes some recommendations on operation for when the link contents are presented.
- When the entry component of C-profile (tag value = 0x80) appears

  When the BML-engine has not started, the BML-engine starts, and behavior are done at the same time as tuning.
- When the PID of the component transmitting data carousel (including empty carousels) is changed during reception

  The receiver judge if the contents of the data program are changed and a process the re-tuning corresponding to the service displayed below is carried out.
  - ▪ The document being presented is destroyed.
  - ▪ All acquired resources are deleted from the BContents memory.
  - ▪ Re-acquisition and presentation of the startup document of the entry component.

4.1.5.3 Specifications on the partial transport stream output-input

When the receiver outputs the data component (component tag value 0x80 to 0x8F) to the partial transport stream, the output follows the descriptors shown in Table 4-3. (Follows the specifications Vol. 2)

Table 4-3 Descriptor output in the partial TS

Descriptor name	Description table	Allocation standard
Data Component Descriptor	PMT 2nd loop	Required
Extended Broadcaster Descriptor	SIT 2nd loop	Required
Broadcast ID Descriptor	SIT 2nd loop	Required

- When the partial transport stream is input and multimedia contents are playbacked, operations that are used when the function required for SI information not included in the partial transport stream and the function to tune implementation dependant. For example, the following broadcasting enhancing function is given as an example.
  - Tuning (X_DPA_tuneWithRF) and reservation (epgReserve)

4.2 Data carousel transmission operation

4.2.1 Data carousel transmission operation

- The maximum number of modules transmitted with one data carousel is 64 for C-profile.
- The module configuration of the data carousel may change according to the event under process and the time (the module increases and decreases). In that case, the DII version is updated.
- The transmission frequency will vary depending on the module that configures the data carousel.

4.2.1.1 Introduction of data event and local contents

- The concept of a data event that is not time related directly to the event is introduced to enable the switching of contents items at anytime, regardless of in a program or between programs, and switches the contents presentation by each data event unit. The contents transmitted in one data event are called local contents.
- The data event is identified with the data_event_id of DII.

4.2.2 Operation of data events

4.2.2.1 Operation of data events

- The data_event_id is updated when local contents change. That means the data_event_id is different before and after local contents switching. (Does not necessarily increase by 1.) (Figure 4-3)
- Data_event_id is not necessarily updated before and after the stop when the ES stops (the component description disappears from the PMT). Start of the ES implies the beginning new local contents. It is not necessary to memorize the data_event_id before the ES stops. (Figure 4-3)

- The data_event_id is managed and updated individually in each component. The data_event_id is operated by values other than 15(0xF).

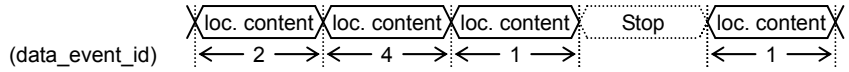


Figure 4-3 Local contents and data_event_id

4.2.2.2 Initiation and termination of local contents

- When the data_event_id of DII is updated, it is considered as a switching of local contents, and the document under presentation is destroyed and a startup document of new local contents is acquired and presented. (Refer to section 4.2.2.3.) The initialization and termination of local contents are ocurred, as a rule, synchronized with the update of the data_event_id of DII in the carousel during reception.

4.2.2.3 Receiver basic behavior during data broadcasting program presentation

- The receiver receiving C-profile will observe constantly DII of all the data components transmitted.
- Receiver processing when the data event of components during reception switches
  The BML-engine sends a "DataEventChanged" message to the document being presented. Afterwards, the document currently being presented is destroyed and the startup document included in the moduleID=0000 module of the component being presented is acquired and presented. When it is an empty carousel after switching the data event, the DII update of the relevant component is monitored after the document being presented isdestroyed, and a startup document is acquired and presented at the moment when it is no longer an empty carousel by switching data events. However, the synchronous/asynchronous event ocurred after "DataEventChanged" is annulled excluding the unload event. Moreover, when the launchDocument()/reloadActiveDocument()/ X_DPA_tuneWithRF()/quitDocument() function is used in the "DataEventChanged" event handler, a function is assigned without the acquisition of a startup document.

4.2.2.4 Operation when DII cannot be obtained for long time

Receivers do not process time-outs for contents presented by PMT and DII reception, etc., when the reception conditions are not stable for certain moments, even in the condition of broadcasting reception or linked condition. However, this is not to eliminate the warning display function, in order for users to choose whether or not to maintain broadcasting reception or tuning. Here, when the contents presentation ends, it is necessary to end all

contents that originate in broadcasting. For example, data broadcasting only ends but video and audio broadcasting continues.
The operation of contents might be destroyed because the receiver cannot acquire the event message sent during the period when the stream cannot be acquired. The contents configuration shall be considered, in order not to destroy operation of contents by temporarily breaking the stream during the transmission period of event messages.

4.2.3 Operation of empty carousels

- Data carousels that do not include a DDB are called empty carousels when the numberOfModules field is only DII of 0.
- An empty carousel is used to switch data transmission and termination of the component without changing the description of the ES in the PMT when certain components are necessary at specific periods of an event. Refer to Appendix 8.
- When switching a data carousel that is not empty and an empty carousel, the data_event_id is switched.
- When an empty carousel is transmitted, the minimum transmission interval specification of DII (section 4.2.4) is applied.
- A newer version of DII may be ocurred while transmitting an empty carousel. Moreover, an update of the data event might be ocurred, too.
- When an empty carousel is detected by tuning and the switching of data events, it is not considered an error and the update of DII is monitored, and when a startup document appears the presentation of a startup document starts.
- When the carousel containing the document on reception switches to an empty carousel during multimedia contents presentation, an event (Identify by a "DataEventChanged" event and status=1) that shows "Change into an empty carousel" to the BML document occurs. This operation is the same whether this is an entry component or not.
- When an empty carousel is transmitted by the component that transmits the assigned BML document during the document change specifying the BML document of the transition destination, like launchDocument(), it is processed as an error equal to when the reference module is not included in the usual carousel.

4.2.4 Operation of the DownloadInfoIndication (DII) message

- When presenting multimedia contents, the reception of DII of all carousels is necessary.
- Refer to 4.2.1.1 for the relationship of DII and local contents/data event.

- For restrictions on the performance of the receiver, the minimum transmission interval of DII of each component (excluded the moment of the switching if carousel) is to be 300 milliseconds.
- DSMCC_section transmitting DII is operated as the standard.
- Module information stored in the DII message should store the moduleId's in ascending order (however, moduleId's are not necessarily continuous).
- The operation of userNetworkMessage() is shown in Table 4-4.

Table 4-4 Operation of DII:userNetworkMessage()

Field	Operation	Remarks
dsmccMessageHeader()		
protocolDiscriminator	Operated as specified.	0x11
dsmccType	Operated as specified.	0x03
messageId	Operated as specified.	0x1002
transaction_id	Operated as specified. Incremented by one on Transaction Number (lower 30bits of transaction_id) as a rule for the following cases. When the value is changed, then the receiver unit should judge that the contents of the DII have been changed, this is not limited to when it is incremented by 1 • When the data event switches • When at least 1 carousel configuration module is updated • When the number of module configuring carousels changes (include before and after numberOfModules=0.)	
dsmccAdaptionHeader()	Not operated.	
downloadId	Operated as specified. It is updated in the switch timing of the data event. bit31-28 data_event_id bit27-0 all 1 Operate for identification and switching of the local contents, and response to event message and local contents.	In order to avoid the mis-reception of the event message on local contents adjoined timewise or to switch the data event, the data_event_id is operated.
blockSize	Operated by a fixed value.	4066
windowSize	Operated as specified.	0
ackPeriod	Operated as specified.	0
tcDownloadWindow	Operated as specified.	0
tcDownloadScenario	Operated. The cycle of a longest module of the transmission cycle, among carousel configuration modules, is described.	The time-out time setting based on this value implementation dependantl.

Field	Operation	Remarks
compatibilityDescriptor()	Follows the standard of when operation is done without contents.	compatibilityDescriptorLength=2 descriptorCount=0
numberOfModules	Maximum number of modules transmitted with one data carousel is 64. Moreover, to indicate that it is an empty carousel, numberOfModules=0 is used. Refer to 4.2.3 for empty carousels.	
moduleId, moduleVersion	Operated as specified.. However, it is not guaranteed that +1 moduleVersion increases continuously.	
moduleSize	The maximum value of the module size is 256KB. Refer to 4.2.5 for details.	
Module information area	The descriptor described later is allocated.	
Private data area	Not operated.	
Descriptor stored in the module information area		
Type Descriptor	When mapping 1 resource to 1 module directly, allocation is necessary. It is not necessary for modules to store resources by entity format.	
Name Descriptor	Not operated.	
Info Descriptor	Not operated.	
Module_Link Descriptor	Not operated.	
CRC Descriptor	Not operated.	
Estimated Download Time Descriptor	It can be operated. The maximum transmission cycle of the corresponding module is specified when operating.	Response is optional.
Expire Descriptor	Not operated.	
ActivationTime Descriptor	Not operated.	
CompressionType Descriptor	Not operated.	
Control Descriptor	Not operated.	
RootCertificate Descriptor	Not operated.	

4.2.5 Operation of DownloadDataBlock (DDB)

- DSMCC_section transmitting the DDB message is operated according to standards.
- The maximum size of the module transmitted by the DDB message is 256KB. In detail, the number of DSMCC_section transmitting DDB's is to be 64 as a maximum. (maximum module size is 4066 x 64 = 260224 bytes)
- The module may be compressed by zlib format and transmitted. In this case, the CompressionType Descriptor is allocated in the module information area of DII corresponding to the pertinent module, and the compression type is to be 0. For details of the compression format refer to Appendix 2.
- The sum total module size before and after compression should not exceed the above-mentioned maximum module size when the module is compressed and sent.
- The operation of DDB (downloadDataMessage()) is shown in Table 4-5.

Table 4-5 Operation of DDB: downloadDataMessage()

Field	Operation	Remarks
dsmccDownloadDataHeader()		
protocolDiscriminator	Operated as specified.	0x11
dsmccType	Operated as specified.	0x03
messageId	Operated as specified.	0x1003
downloadId	Operated as specified.	The same value as the downloadId of DII is stored.
adaptationLength	dsmccAdaptationHeader() is not operated in case of multimedia.	0
moduleId	No special specifications for the moduleId value.	
moduleVersion	Operated.	It is not guaranteed that +1 value is given when updated.
blockNumber	Operated as specified..	Value obtained from moduleSize/blockSize.

4.3 Operation of event messages

Only general event messages are operated. Refer to section 4.5 for the optional "auxiliary information function using general event message".

4.3.1 Purpose for operating event messages

- Event messages (hereinafter general event message), including the General Event Descriptor, are operated to transmit data that accompanies events at the same time events are asynchronously caused in multimedia contents.

4.3.2 Transmission of general event messages

- The general event message is transmitted by components with component tag values 0x80, 0x89, 0x8A, and 0x8B. It transmits by components with a component tag value = 0x89 and 0x8A when general event messages only are transmitted. The event_section_flag allocated in the Data Component Descriptor of the PMT is not operated. It is always one.
- The last_section_number of the DSMCC_section transmitting the event message is always 0. That is, the sub-table that transmits one event message is always transmitted by one section.
- The private_data_byte is operated.
- The maximum number of General Event Descriptors that can be allocated in 1 DSMCC_section is 8.
- When setting to acquire the arbitrary message_id by specifyingthe message_id=255 or omitting message_id, it is not set at the same time as the (0~254) assignment that specifies message_id. Moreover, it is assumed that the message version is omitted without fail in this case (Or, 255 is specified).
- When the message_id is specified, the maximum number of general event messages that can subscribe at the same time is 8.
- Multiple sub-tables with the same contents may be transmitted in order to prevent the receiver from failing to receive the general event message. Neither the transmission interval nor the transmission frequency are specified in this case.
- It should be considered that the receiver can fail to receive the general event message, and to secure 200ms or more at update intervals of the DSMCC_section that transmits the general event message in the same ES. This indicates the recommended value at the intervals from sending the first DSMCC_section of an arbitrary version to sending DSMCC_section of the following updated version, and is not the one that provides the sending interval of the DSMCC_section for a different adjacent version during update.
- Only 0x00 (immediate firing) is operated as a time mode to specify the firing time.
- It is assumed that there are cases when the receiver cannot acquire the event message, due to the temporary deterioration of the receiving status. It is necessary to consider that the receiver may not be able to acquire the event message when authoring contents.

4.3.2.1 Operation of general event message data_event_id, event_msg_group_id

The value of the event_msg_group_id of the general event message uses 0 or 1. The receiver behaves as follows.

- Only when the value of the data_event_id of the event message is the same as the data_event_id of local contents during presentation ,the general event message that is

the event_msg_group_id=0 processed as an effective event message. When the value of the data_event_id is different, the event message is ignored.

- General event messages with an event_msg_group_id=1 are processed as effective event messages regardless of the value of the data_event_id.

The above specification applies to both when the general event message is transmitted by the same component as local contents that use the general event message of among four components shown in section 4.3.2, or the general event message is transmitted by other components.

Event_msg_group_id=0 should be specified usually. This is because, for example, there is a possibility of being mis-received, when event_msg_group_id=1 is specified by adjacent local contents and when the event message is transmitted close to same time as the switching of the local contents as in Figure 4-4.

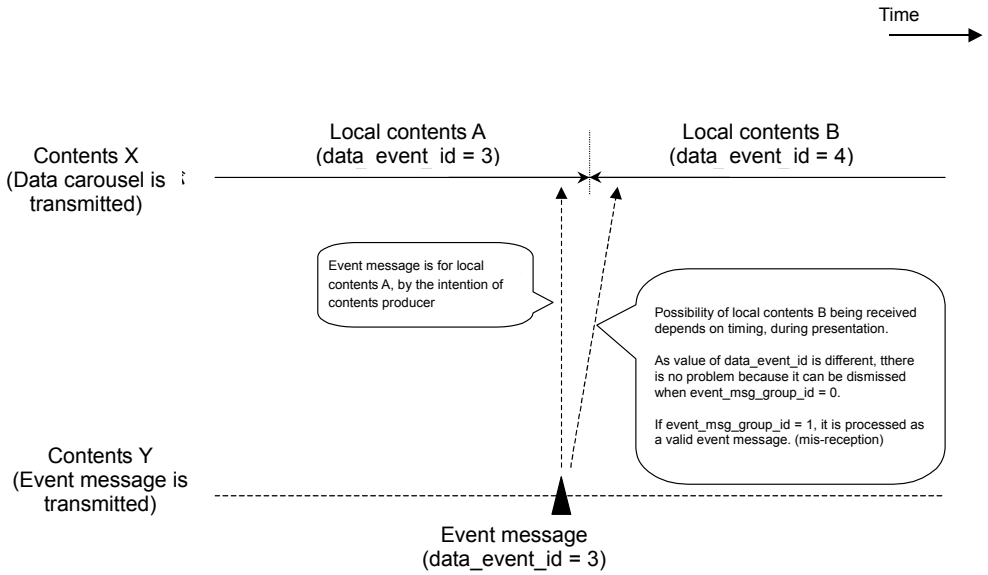


Figure 4-4 Mis-reception of general event messages

For example,The data carousel uses the one sent from the transmission station in network programs etc. and the general event message uses it as if it is transmitted by a reception station, the specification of event_msg_group_id=1 is used when it is difficult to match the value of data_event_id. (Figure 4-5).

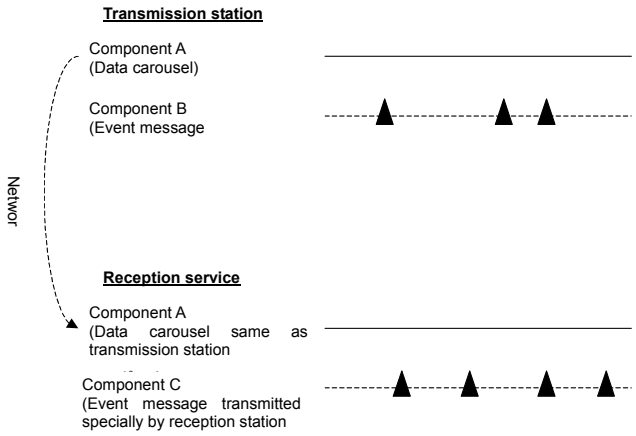


Figure 4-5 When the data_event_id values are not unified

4.3.2.2 Regarding component specifications of general event messages from the BML document

Among the four components shown in section 4.3.2, the component_tag of the general event message that can be observed by 1 BML document, at the same time, is a maximum of 2 types.

4.3.2.3 Event message processing in receivers

(1) General event messages

- When events specified "EventMessageFired" as the type attribute in the BML document, receiver filter the DSMCC_section that transmits the general event message is carried out to comply with either of the following conditions.
  (Condition 1)
  -- component_tag is specified by the es_ref attribute
  -- event_msg_group_id=0x000
  -- data_event_id of local contents during presentation
  (Condition 2)
  -- component_tag is specified by the es_ref attribute
  -- event_msg_group_id=0x001
- For the DSMCC_section to transmit event messages, the components that can filter at the same time is a maximum of two.

- When es_ref is omitted, it is interpreted as the component of local contents currently being presented.
- When the first acquisition of the DSMCC_section or a version-up of the DSMCC_section is detected, an event is carried out according to regulations in the multimedia encoding method based on the message_id/message_version specified by BML document and the the event_msg_id of the General Event Descriptor included in the relevant DSMCC_section
- About the operated time mode
  Only 0x00 (immediate firing) is used as the time mode to specify the firing time.
- The receiver carries out the event in the multimedia contents as quickly as possible after receiving the event message. The target time is 100 milliseconds.
- When multiple general event messages are received, they are fired in the order of reception. The timing of processing is not specified.
- When general event messages that specified immediate multiple firing specification is transmitted by the same DSMCC_section is acquired, the firing is in the order the descriptors located in the DSMCC_section.
- The valid period of the subscribe for the general event message is the period in which the document that specifies subscribe is presented.

4.3.3 Operation of DSMCC_section()

Table 4-6 Operation of DSMCC_section() transmitting event messages

Field	Operation	Remarks
table_id	Operated as specified..	0x3D
section_syntax_indicator	Operated as specified..	1
private_indicator	Operated as specified..	0
dsmcc_section_length	Operated as specified..	
data_event_id	General event message: When event_msg_group_id=0x000, the value is the same as the data_event_id of responding local contents. When event_msg_group_id=0x001, the value is 0x0.	
event_msg_group_id	General event message: 0x000 or 0x001.	
version_number	As a rule, it is incremented by +1 at same subtable identified by the combination of table_id,data_event_id and event_msg_group_id, but continuity is not guaranteed.	
current_next_indicator	Always 1.	

4.3.4 Operation of the General Event Descriptor

Table 4-7 Operation of the General Event Descriptor

Field	Operation	Remarks
event_msg_group_id	Same value as event_msg_group_id of DSMCC_section. (0x000 or 0x001)	
time_mode	Always 0x00 (firing event as soon as reception occurs).	
event_msg_type	Always 1.	
event_msg_id (message_id, message_version)	Operated as specified..	
private_data_byte	Operated as specified..	Max. 244 bites.

4.4 Receiver performance

4.4.1 Filtering resources used during data broadcasting reception

Filtering resources necessary for obtaining a carousel

Required transmission operations

- DII of all carousels is monitored.
- The lock by the lockModuleOnMemoryEx() is enabled for the module of the component that transmits all carousels. (Refer to section 8.1.15.5).
- It is enabled the monitoring of the ModuleUpdated for the module of the component that transmits all carousels.

Table 4-8 Filtering resources required for acquiring the data carousel

Object ES	PID filter number	Section filter number	
		DII	DDB
ES of tag value =0x80	1	1	1
ES of tag value =0x8B	1	1	1

Filtering resources necessary for general event messages

Required transmission operations

The general event message in C-profile is transmitted by components with component tag values of 0x80, 0x89, 0x8A, and 0x8B. When only the general event message is transmitted, transmit them by components with component tag values of 0x89 and 0x8A.

Moreover, two section filters are necessary, respectively, because the value of the event_msg_group_id can be sent by 0 or by 1.

Table 4-9 Filtering resources required for acquiring the general event message

Object ES	PID filter number	Section filter number
ES of general event message tag value =0x89	1	2
ES of general event message tag value=0x8A	1	2

4.4.2 Desirable operation of the receiver in relation to the acquisition of C-profile modules

Improve presentation response by pre cache, obtaining unconditionally modules transmitted in C-profile receivers. In C-profile receivers., the presentation response can be improved by unconditional acquisition, and the pre cache of the module transmitted on a maximum extent. Moreover, the response of the presentation can be improved by acquisition of the multiple ES modules at one time, and the pre cache even if multiple ES transmits the contents.

4.4.3 Receiver operation concerning updating the version

Refer to Appendix 9 for the receiver behavior receiving the major_version excluding 12.

4.5 Auxiliary information function using general event message (Optional)

The following shows the auxiliary information function using general event message. This function handles the information in private_data_byte as "auxiliary information using general event message", using a specific character string in event_msg_id and private_data_byte (hereafter referred to as "specified character string"). Implementation of this function shall be optional.

The image of the operation using the relevant auxiliary information shall be used for purposes such as encouraging the viewer to view data broadcasting when it is not being displayed and providing auxiliary information to draw viewer's attention to a certain program.

4.5.1 Operation of event_msg_id and general event message transmission

event_msg_id (message_id and message_ version) shown below shall be used exclusively for the "auxiliary information function using general event message".

Table 4-10 Specification of ID, etc. to be transmitted

Target ES	message_id	message_version	event_msg_group_id
0x80,0x8B,0x89,0x8A	200	0～255	1

- In the case when a general event message in which event_msg_id (message_id and message_version) shown in Table 4-10 has been specified has been transmitted by multiple ES, only the latest auxiliary information shall be presented (Refer to section 4.5.5 for the operation of the receiver.).
- General event messages shall be transmitted using a component with component tag values of 0x80, 0x89, 0x8A, and 0x8B. When transmitting a general event message only, it shall be transmitted using a component with component tag values of 0x89 and 0x8A. event_section_flag, which will be allocated to the Data Component Descriptor of PMT, shall not be operated. It shall always be set to 1.
- The “last_section_number” of the “DSMCC_section” that transmits event messages is always 0. In other words, the subtable that transmits one event message is always transmitted by 1 section.
- “private_data_byte” is operated.
- The maximum number of General Event Descriptors that can be located in 1 DSMCC_section is 8.
- In order to avoid missed messages, a subtable with the same contents may be sent multiple times in general event messages. In this case, transmission intervals and the number of transmissions are not specified.
- In consideration of messages missed by receiver, it is recommended that the update interval of the DSMCC_section, which transmits general event messages within the same ES, be secured for more than 200ms. This indicates the recommended value of the interval between sending of the first DSMCC_section of any version to the DSMCC_section of next updated version. This is not to define the sending interval for the DSMCC_section of different versions that are adjacent to each other at the time of update.
- Only 0x00 (immediate firing) shall be operated as the time mode for specifying the time of firing.
- It is conceivable that the receiver may fail to acquire the event message due to temporary disruption of the reception status. During content development, it shall be necessary to sufficiently consider the possibility of the receiver failing to acquire the event message.
- Regarding general event messages with the event_msg_group_id = 1, regardless of the data_event_id value, they are processed as a valid event message.
- Refer to section 4.3.3 for the operation of DSMCC_section. However, event_msg_group_id shall only be operated with 1
- Refer to section 4.3.4 for the operation of the General Event Descriptor.
- event_msg_id in Table 4-10 shall not be used as a normal general event message.

4.5.2 Transmission of the specified character string in private_data_byte

Table 4-11 below shows the specified formats for private_data_byte.

Table 4-11 Specified formats

Byte position	No. of bytes	Field name	Specified character string	Other
0 -9	10	HEADER	”DPA-EMSUBI”: Indicates that it is an "auxiliary information function using event message".	
11-15	5	SC1 (specified code 1)	”DMARK”: Displays a mark.	Duplicate specification of the same specified character string shall be prohibited. However, this excludes "NONSC".
			”STEXT” : Displays the character string of the STEXT item.	
			”FUNC1” : (Reserved for future expansion)	
			”FUNC2” : (Reserved for future expansion)	
			”FUNC3” : (Reserved for future expansion)	
			”NONSC” : Indicates "No specification".	
			”CLEAR” : Clears the display.(Only SC1 can be specified)	
17-18	2	SC1T	Describes the length of time (period) in second, for which SC1 will be presented. It shall be described in single-byte alphanumeric characters. The range of values that can be specified shall be "00" and "05" - "99". Refer to Table 4-13 for the operation of the receiver.	"00" shall only be specified when "NONSC" and "CLEAR" are specified in SC1-SC3.
20-24	5	SC2	Same as for SC1.	Same as for SC1.
26-27	2	SC2T	Describes the length of time (period) in second, for which SC2 will be presented.Others shall be the same as for SC1T.	Same as for SC1T.
29-33	5	SC3	Same as for SC1.	Same as for SC1.
35-36	2	SC3T	Describes the length of time (period) in second, for which SC3 will be presented.Others shall be the same as for SC1T.	Same as for SC1T.
38	1	LOCATION1	Display position: Indicates top or bottom ”T” : Align to top of the screen ”B” : Align to bottom of the screen	It shall always be specified.
40	1	LOCATION2	Display position: Indicates left or right ”L” : Align to left of the screen ”R” : Align to right of the screen	It shall always be specified.
42-	Max. 60	STEXT	A character string of up to 60 bytes (30 double-width characters) shall be specified. If "STEXT" has been specified in SC1-SC3, the character string in this field must always be specified. The character code shall follow the Shift-JIS standard defined in 5.4.2 in [Section 4] of this volume. However, the use of a delimiter ("|" (0x7C)) shall not be permitted.	Empty string shall be specified if there is no specification of "STEXT" in SC1-SC3.
	3	TRAILER	"END" must always be specified at the end.	

*1 "|" (0x7C) shall be used for delimiting fields.

Example: DPA-EMSUBI|DMARK|05|STEXT|15|NONSC|00|T|R|In data broadcasting, ...|END
Example: DPA-EMSUBI|DMARK|10|NONSC|00|NONSC|00|T|R||END (Mark only)
Example: DPA-EMSUBI|CLEAR|00|NONSC|00|NONSC|00|T|R||END (Clear)

*2 Combinations of SC1-SC3 that will be operable for the time being are as follows.

Table 4-12 Operable combinations

	SC1	SC2	SC3
Only DMARK is specified	DMARK	NONSC	NONSC
DMARK and STEXT are specified	DMARK	STEXT	NONSC
Clearing of information is specified	CLEAR	NONSC	NONSC

*3 "NONSC" shall always be specified if there is no specification in SC1-SC3.
If "CLEAR" has been specified in SC1, "NONSC" shall always be specified in SC2 and SC3.

* 4 If "NONSC" and "CLEAR" have been specified in SC1-SC3, "00" shall be specified in SC1T-SC3T.

Refer to section 4.5.5 for the operation of the receiver.

4.5.3 Operation of content

event_msg_id of the target ES as shown in Table 4-10 shall be used exclusively for the "auxiliary information function using general event message". The content, time and position of the presentation shall be specified to private_data_byte according to the format specified in Table 4-11.

Moreover, the relevant event_msg_id will not be used as a normal general event message.

4.5.4 Conditions for incorporation on the receiver

A receiver incorporating this function shall observe the following incorporation conditions.

- The receiver must incorporate a data broadcasting browser. This function shall not be implemented if the receiver does not incorporate a data broadcasting browser.
- If PID filtering resources for general event messages will be implemented for this function, the number of PID filtering resources shall be 4 excluding the 2 resources shown in Table 4-9.

4.5.5 Conditions for presentation on the receiver

A receiver incorporating this function shall observe the following presentation conditions.

O In case of data broadcasting not being presented

- A receiver not presenting data broadcasting shall present information content using this function. It shall be presented in the video area.

- Refer to section 4.5.5.2 for the presentation relationship with subtitles and system messages of the receiver.

○ In case of data broadcasting being presented

- If the receiver presents data broadcasting, the presentation of information content using this function shall be a matter of product planning. If information content will be presented using this function, the above conditions for when the receiver is not presenting data broadcasting shall be observed.

4.5.5.1 Presentation time

The presentation time shall be specified using SC1T-SC3T. The presentation shall be terminated when the specified amount of time has elapsed. If the receiver does not support the SC1T-SC3T specifications ("05" - "99" seconds), it must terminate the presentation in less than 15 seconds.

4.5.5.2 Presentation

- The position of the presentation will be specified by the specified character string.
  Refer to Figure 4-6 in section 4.5.5 for the presentation positions of the mark and the character string.
  If the receiver does not support specification of the presentation positions of the mark and the character string, the information shall always be presented at the "top right" of the screen.
- The method of presentation shall be a matter of product planning. While it is recommended that the entire character string be presented by scrolling, it shall also be permissible to split the character string into multiple parts and present them separately.
- All character strings specified in "STEXT" shall be presented. No character string shall be displayed if it is not possible.
- The priority order of the presentation by this function shall be the lowest compared to the presentation of subtitles, etc. (including system messages of the receiver).

  For example, the presentation by this function shall not be performed if viewability will be lowered by the overlapping of the subtitles, etc. with the presentation by this function. (It shall also be possible to adopt a means for improving the visibility of the subtitle display, etc., such as showing a frame around the subtitle.)
- A function for turning this function on and off shall not be incorporated.

4.5.5.3 Presentation update

The presentation shall be updated if message_version has been changed. The update timing shall be dependent on the receiver.

In the event when the same message_version is specified in the same ES after displaying the "auxiliary information using general event message", the information shall not be displayed again.

In the case when a general event message in which event_msg_id shown in Table 4-10 has been specified has been transmitted by multiple ES, only the latest auxiliary information shall be presented regardless of which ES the message was transmitted by (i.e. the latest write has priority). Auxiliary information specified by 2 or more ES must not be displayed simultaneously.

4.5.5.4 Deletion timing

If "CLEAR" has been specified in SC1, all presentations of "auxiliary information using general event message" shall be deleted.

The presentation shall be removed when the time period specified in specified character strings SC1T-SC3T have elapsed. If the receiver does not support the SC1T-SC3T specifications ("05" - "99" seconds), it must terminate the presentation in less than 15 seconds.

4.5.5.5 Character color, background color, character size and character type

Character color, background color, character size and character type shall be a matter of product planning. If the number of characters exceeds the limit specified in Table 4-13, the receiver shall ignore the entire general event message as invalid format.

4.5.5.6 Receiver operation in response to each specified character string

The following shows the receiver operation in response to each specified character string.

Table 4-13 Receiver operation in response to each specified character string

Field	Specified character string	Receiver operation	Support by the receiver incorporating this function
HEADER	”DPA-EMSUBI”	The receiver shall operate as "auxiliary information function using general event message".	Mandatory
SC1～SC3	”DMARK”	The receiver shall display a mark representing data broadcasting. Refer to *6 for the position of the presentation. Contact the governing organization described in Appendix 13 regarding the recommended mark.	Mandatory
	”STEXT”	The receiver shall display the character string.	Optional
	”FUNC1”～”FUNC3”	These shall be reserved for future expansion. The receiver shall ignore the specified strings and not generate any errors.	Optional
	”NONSC”	The receiver shall not display any character string.	Mandatory
	”CLEAR”	The receiver shall clear the information that was specified in SC1-SC3 and displayed on the receiver.	Mandatory
SC1T～SC3T	”05”～”99”	The receiver shall terminate the presentation when the specified amount of time has elapsed. If the receiver does not support the SC1T-SC3T specifications ("05" - "99" seconds), it must terminate the presentation in less than 15 seconds.	Mandatory
	”00”	If "00" is specified in SC1T-SC3T, "NONSC" and "CLEAR" will be specified in SC1-SC3.	Mandatory
LOCATION1 LOCATION2	”T”、”B”、”L”、”R”	The receiver shall present the information at the top left, top right, bottom left or bottom right of the screen as specified. If the receiver does not support specification of the presentation position, the information shall always be presented at the "top right" of the screen.	Mandatory
TRAILER	”END”	The "END" character string. The receiver shall interpret this as the end of the instruction.	Mandatory

*1 Fields shall be delimited using "|" (0x7C).

Example: DPA-EMSUBI|DMARK|05|STEXT|15|NONSC|00|T|R|In data broadcasting, ...|END

Example: DPA-EMSUBI|DMARK|10|NONSC|00|NONSC|00|T|R||END (Mark only)

Example: DPA-EMSUBI|CLEAR|00|NONSC|00|NONSC|00|T|R||END (Clear)

*2 The receiver shall operate as follows when an unsupported or undefined character string has been specified in the SC1-SC3 fields.

- The receiver shall ignore the unsupported/undefined SC1-SC3 fields and their corresponding SC1T-SC3T (and STEXT) fields, without generating any error.

  Example : DPA-EMSUBI|DMARK|05|STEXT|15|NONSC|00|T|R|In data broadcasting, ...|END

  - If the relevant receiver does not support STEXT specified in SC2, it shall ignore SC2, SC2T and STEXT fields and, as a result, only present DMARK that has been specified in SC1.

  Example: DPA-EMSUBI|DMARK|15|SOUND|10|NONSC|00|T|R||END

  - Since SOUND is not defined, the receiver shall ignore SC2 and SC2T fields and, as a result, only present DMARK that has been specified in SC1.

*3 Combinations of SC1-SC3 that will be operated for the time being are as shown in Table 4-12.

*4 As a rule, the receiver shall operate as described in *2 if any combination other than those shown in Table 4-12 has been specified. However, the receiver may also be implemented such that it will ignore the relevant general event message.

*5 If multiple specifications have been made using SC1-SC3 and the receiver supports multiple specifications, they shall be displayed by the receiver simultaneously. If simultaneous display is not possible, the receiver shall display one of the specifications based on the following order of priority. The order of presentation priority shall be as follows (from left to right): SC1 > SC2 > SC3.

*6 The presentation shall be positioned according to LOCATION1 and LOCATION2. For example, the presentation will be aligned to top and left edges of the screen if "T" and "L" are specified. The following shows the presentation image examples based on the specification of marks and character strings.

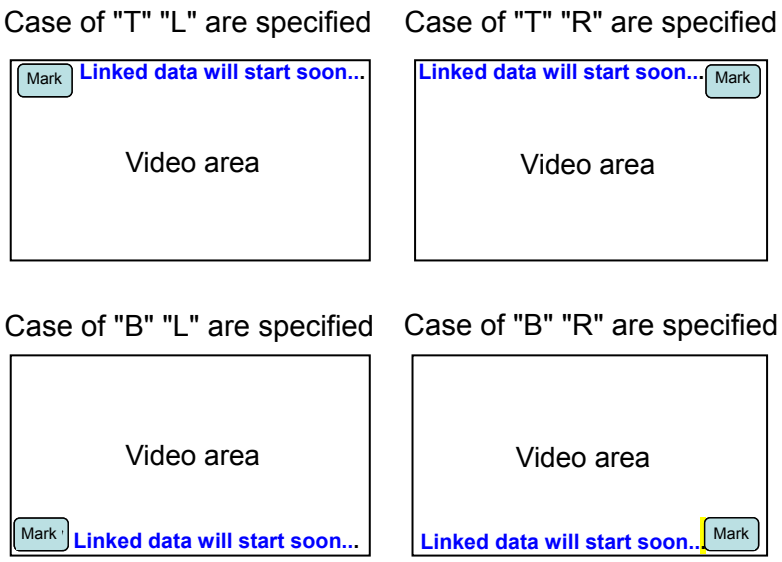


Figure 4-6 Presentation image based on the specification of presentation positions

*7 The receiver shall ignore a general event message if it does not conform to the specified format, such as when (1) the position and/or character code of the delimiter does not conform to the guidelines, (2) a character string other than the specified character strings has been specified in each field (excluding SC1-SC3 fields), or, (3) the size of each field differs from the prescribed number of bytes.

5 Operation of mono-media coding

5.1 Image coding

5.1.1 H.264|MPEG-4 AVC

- Follows the specifications of ITU-T Rec. H.264/ISO/IEC 14496-10:2003.

However, for MPEG2 Systems, please refer to ITU-T Rec. H.222.0|ISO/IEC 13818-1:2000/AMD3(FDAM3). Moreover, the AVC Video Descriptor and AVC timing and HRD Descriptor are not operated.

5.1.1.1 Coding parameters

- For coding parameter restrictions, follow ARIB STD B24 Vol. 1, Part 2, 4.4
- Profile-level is Baseline-level 1.2.

5.1.1.2 Picture format

- Picture format is QVGA (4:3) (screen size : 320x240), QVGA (16:9) (screen size : 320x180).

However, restrictions on PES packets are as below.

- PTS_DTS_flags of the PES packet header is always ‘10’.
- IDR AU is always the first AU of the PES packet.
- PES packet is configured by n pieces of AU (n is an integer 1 or larger).
- The PTS difference of two consecutive PES packets is within 0.7 seconds.

5.1.1.3 Restrictions in the bit stream

(1) Transmission cycle of the IDR access unit

- IDR access units (abbreviated below as AU) are inserted into the bit stream at intervals of every two seconds, normally, to shorten the required time of reproduction. However, they should be inserted in intervals not exceeding five seconds even when extending the IDR-AU interval in order to maintain image quality.
- Each IDR-AU is an elementary stream access point described in ISO/IEC 13818-1:2000 FDAM3.

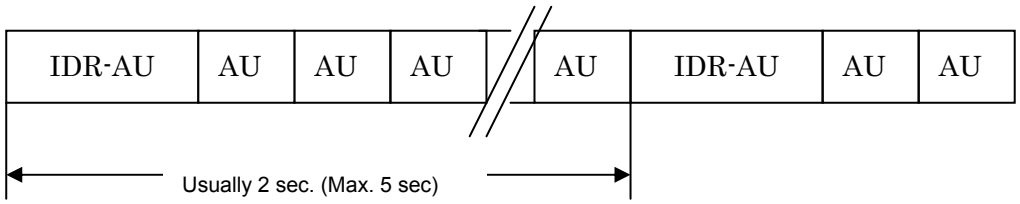


Figure 5-1 IDR-AU intervals

(2) AU configuration

- The number and order of NAL units to configure the IDR AU and non-IDR AU are as follows. However, NAL units other than the following are not operated.

Table 5-1 AU configuration

Type and order of NAL Unit	Quantity	
	IDR AU	non-IDR AU
access unit delimiter	1	1
sequence parameter set (SPS)	1	0
picture parameter set (PPS)	1	0 or 1
supplemental enhancement information (SEI)	0 or 1 (Note 1)	0 or 1(Note 1)
coded slice of an IDR picture	1 and over	0
coded slice of a non-IDR picture	0	1 and over
filler data	0 or 1	0 or 1
end of sequence	0 or 1	0 or 1

(Note1)Insertion condition of SEI NAL unit is Table 5-10.

(3) Operation restrictions on syntax

Restrictions of the flag in the syntax are shown in the tables below. However, the ID of the SPS (Sequence Parameter Set) and PPS (Picture Parameter Set) can be operated by a fixed value regardless of changes in the contents described in the parameters.

Table 5-2 Access unit delimiter restrictions

Flag denomination	Operation	Remarks
primary_pic_type	0 or 1	IDR picture is 0, non IDR picture is 1

Table 5-3 SPS (Sequence Parameter Set) restrictions

Flag denomination	Operation	Remarks
profile_idc	66	Baseline profile
constraint_set0_flag	1	FMO, ASO, RS are not used.
constraint_set1_flag	1	
constraint_set2_flag	1	
level_idc	12	Level 1.2
seq_parameter_set_id	0 - 31	Configure during operation
log2_max_frame_num_minus4	0 - 12	Value not circulating among IDR is configured during operation
pic_order_cnt_type	2	POC type is 2 only
num_ref_frames	1 or 2 or 3	Reference frame number is a maximum of 3
gaps_in_frame_num_value_allowed_flag	0	Missing frames compensation process is not carried out
pic_width_in_mbs_minus1	19	320 / 16 – 1 =19
pic_height_in_map_units_minus1	14 or 11	When 320x180, it is 11 When 320x240, it is 14
frame_mbs_only_flag	1	Only frame MB
direct_8x8_inference_flag	1	No meaning when Baseline
frame_cropping_flag	0 or 1	When 320 x 180, it is 1 When 320 x 240, it is 0
frame_crop_left_offset	0	When it is 320 x 180
frame_crop_right_offset	0	When it is 320 x 180
frame_crop_top_offset	0	When it is 320 x 180
frame_crop_bottom_offset	6	When 320 x 180, it is a 12 line portion crop
vui_parameters_present_flag	1	VUI is mandatory

Table 5-4 VUI parameters restrictions

Flag denomination	Operation	Remarks
aspect_ratio_info_present_flag	0	Aspect ratio 1:1 is specified on operation (default value is unspecified)
overscan_info_present_flag	0	No operation

video_signal_type_present_flag	0	Under specifications of STD-B24, value of color_primaries, transfer_characteristics, matrix_coefficients are interpreted as 1 (Rec.ITU-R BT 709) at the decoder side
chroma_loc_info_present_flag	0	Uses default values
timing_info_present_flag	1	
num_units_in_tick	1001 * N	N≥1, Max. 15fps (Note 1)
time_scale	24000 or 30000	Max. 15fps (Note 1)
fixed_frame_rate_flag	0 or 1	
nal_hrd_parameters_present_flag	0 or 1	
vcl_hrd_parameters_present_flag	0 or 1	
low_delay_hrd_flag	0	Low delay mode is prohibited
pic_struct_present_flag	0	
bitstream_restriction_flag	0 or 1	(Note 2)
motion_vectors_over_pic_boundaries_flag	0 or 1	
max_bytes_per_pic_denom	0 - 16	
max_bits_per_mb_denom	0 - 16	
log2_max_mv_length_horizontal	0 - 9	Horizontal direction ±128
log2_max_mv_length_vertical	0 - 9	
num_reorder_frames	0	Prohibition of reordering
max_dec_frame_buffering	1 - 3	Reference frame number is a maximum of 3

(Note 1) Specifications on the picture display interval are mentioned below

- When the fixed_frame_rate_flag is 0, the interval of decoding and displaying the neighbouring picture shoud be any value more than 1001/15000, and the multiples of num_units_in_tick / time_scale for num_units_in_tick, time_scale specified by VUI.

  (Operation example 1)

  When time_scale=30000, num_units_in_tick=1001,
  num_units_in_tick/time_scale=1001/30000=1/29.97, in other words, the unit value of cpb_removal_delay becomes 1/29.97 seconds. Moreover, since there are restrictions on 1001/15000 or more, the difference of the cpb_removal_delay (interval of adjacent pictures) is two or more.

  (Operation example 2)

  When time_scale=24000,num_units_in_tick=1001, the unit of cpb_removal_delay becomes 1 / 23.9 (sec.). Also, since there are restrictions on 1001/15000 or more, the difference of cpb_removal_delay becomes 2 or more.
- When the fixed_frame_rate_flag is 1, the decoding and display interval of the adjacent picture is 1001/15000 or more and 2 * num_units_in_tick / time_scale for num_units_in_tick, time_scale is specified by VUI.

  (Operation example 1)

When time_scale=30000,num_units_in_tick=1001, 2 * num_units_in_tick / time_scale = 2 * 1001 / 30000 = 2 / 29.97, in other words the decoding and display interval of the adjacent picture is 2/29.97 (sec.).

(Operation example 2)

When time_scale=24000,num_units_in_tick=1001, the interval of the adjacent picture is 2/23.9 (sec.).

(Note 2)The search area of the movement vector is limited in the horizontal direction to ±128. Moreover, num_reorder_frames=0 is to prohibit reordering.

Table 5-5 HRD parameters restrictions

Flag denomination	Operation	Remarks
cpb_cnt_minus1	0	CPB type is 1
bit_rate_scale	0 - 15	Configure during operation
cpb_size_scale	0 - 15	Configure during operation
bit_rate_value_minus1	(bit_rate_value_minus1 + 1) * $2^{(6 + bit_rate_scale)}$ <= 384000 or 460800	Configure based on the bit rate used for real operation
cpb_size_value_minus1	(cpb_size_value_minus1 + 1) * $2^{(4 + cpb_size_scale)}$ <= 1000000 or 1200000	Configure based on the CPB size used for real operation
cbr_flag	0 or 1	
initial_cpb_removal_delay_length_minus1	0 - 31	Configure during operation
cpb_removal_delay_length_minus1	0 - 31	Configure during operation
dpb_output_delay_length_minus1	0 - 31	Configure during operation
time_offset_length	0	

Table 5-6 PPS (Picture Parameter Set) restrictions

Flag denomination	Operation	Remarks
pic_parameter_set_id	0 - 255	Configure during operation
seq_parameter_set_id	0 - 31	Assign reference id of SPS
entropy_coding_mode_flag	0	Only CAVLC
pic_order_present_flag	0	Only Type 2
num_slice_groups_minus1	0	FMO prohibited
num_ref_idx_l0_active_minus1	0 or 1 or 2	Reference frame is 1 to 3 immediately before.
num_ref_idx_l1_active_minus1	0	No B frame
weighted_pred_flag	0	WP prohibited
weighted_bipred_idc	0	No B frame
pic_init_qp_minus26 /* relative to 26 */	-26 - 25	Configure during operation
pic_init_qs_minus26 /* relative to 26 */	0	Not used
chroma_qp_index_offset	-12 - 12	Configure during operation
deblocking_filter_control_present_flag	0 or 1	Configure during operation, no limitations
constrained_intra_pred_flag	0	Do not restrict on intra forecast
redundant_pic_cnt_present_flag	0	RS prohibited

- Restrictions in the SEI (Supplemental Enhancement Information) are shown in the table below. Buffering period, Picture timing, Pan-scan, and Filler payload SEI message can only be inserted in the SEI. (Note 1)

Table 5-7 Buffering period SEI message (Note 2)

Flag denomination	Operation	Remarks
seq_parameter_set_id	0 - 31	Assign reference id of SPS
initial_cpb_removal_delay	initial_cpb_removal_delay + initial_cpb_removal_delay_offset ≤ 135000(recommendation value)	Configure during operation when NalHrdBpPresentFlag and VclHrdBpPresentFlag are ”1” each
initial_cpb_removal_delay_offset		Configure during operation when NalHrdBpPresentFlag and VclHrdBpPresentFlag are "1" each

Table 5-8 Picture timing SEI message (Note 3)

Flag denomination	Operation	Remarks
cpb_removal_delay	0 - 150	Configure during operation when CpbDpbDelaysPresentFlag is "1"
dpb_output_delay	0	Configure during operation when CpbDpbDelaysPresentFlag is "1"

Table 5-9 Pan-scan rectangle SEI message (Note 4)

Flag denomination	Operation	Remarks
pan_scan_rect_id	0	
pan_scan_rect_cancel_flag	0 or 1	
pan_scan_cnt_minus1	0	
pan_scan_rect_left_offset	0 or 640	Refer to section 5.1.1.6
pan_scan_rect_right_offset	0 or -640	Refer to section 5.1.1.6
pan_scan_rect_top_offset	480 or 0	Refer to section 5.1.1.6
pan_scan_rect_bottom_offset	-480 or 0	Refer to section 5.1.1.6
pan_scan_rect_repetition_period	1	

(Note 1)The existence and the order of each AU of SEI message are as follows.

- IDR-AU

  Buffering period SEI message, Picture timing SEI message, Pan-scan rectangle SEI message, and Filler payload SEI message can be inserted in the relevant AU, and inserted if necessary. The insertion order is as follows.

  1 Buffering period SEI message

  2 Picture timing SEI message

  3 Pan-scan rectangle SEI message

  4 Filler payload SEI message

- non IDR-AU

  Only the Picture timing SEI message, Filler payload SEI message can be inserted in the relevant AU, and inserted if necessary. The insertion order is as follows.

  1 Picture timing SEI message

  2 Filler payload SEI message

  In addition, in either of IDR-AU and non IDR-AU, existence or not of the Filler payload SEI is not specified.

(Note 2) (Note 3) Restrictions regarding the insertion of the Buffering period SEI message and Picture timing SEI message are as below.

Table 5-10 Restrictions on the Buffering period SEI message and Picture timing SEI message insertion

AU configuration in 1PES	fixed_frame_ rate_flag	Type of AU	Buffering period SEI message	Picture timing SEI message
1PES=1AU fixed operation	0	IDR	Δ	Δ
		Non IDR	×	Δ
	1	IDR	×	×
		Non IDR	×	×
1PES=1AU other than fixed operation	0	IDR	O	O
		Non IDR	×	O
	1	IDR	×	×
		Non IDR	×	×

O: Input is mandatory Δ: Input is optional ×: Input prohibited

- When the Buffering period SEI message and Picture timing SEI message are inserted, at least nal_hrd_parameters_present_flag or vcl_parameters_present_flag is "1".
- When neither the Buffering period SEI message nor the Picture timing SEI message are inserted, both the nal_hrd_parameters_present_flag and vcl_parameters_present_flag are "0".
- When fixed_frame_rate_flag = 1, the framerate can be obtained in time_scale / num_units_in_tick / 2. (Example: when time_scale = 30000, num_units_in_tick = 1001, the framerate is 29.97 / 2 (frames/sec)

(Note 4) Restriction on inserting the Pan-scan rectangle SEI message
- When operating PanScan, insert it in IDR-AU, without fail.

Table 5-11 Slice header restrictions

Flag denomination	Operation	Remarks
first_mb_in_slice	0 - 299 or 0 - 239	Configure during operation
slice_type	0, 5 or 2, 7	7 in IDR slice, and 0, 2, 5, 7 in non IDR slice
pic_parameter_set_id	0 - 255	Assign reference id of PPS
frame_num	0 ~ MaxFrameNum	Do not circulate between IDR
idr_pic_id	0 - 65535	Configure during operation
num_ref_idx_active_override_flag	0 or 1	Configure during operation
num_ref_idx_l0_active_minus1	0 or 1 or 2	Reference frame is 1 to 3 immediately before
slice_qp_delta	-51 - 51	Configure during operation
disable_deblocking_filter_idc	0 or 1	Modes that do not filter on the slice boundary are prohibited
slice_alpha_c0_offset_div2	-6 - 6	Configure during operation
slice_beta_offset_div2	-6 - 6	Configure during operation

Table 5-12 Reference picture list reordering restrictions

Flag denomination	Operation	Remarks
ref_pic_list_reordering_flag_l0	0	Reordering of reference frame is prohibited

Table 5-13 Decoded reference picture marking restrictions

Flag denomination	Operation	Remarks
no_output_of_prior_pics_flag	0	Do not clear DPB when IDR is input (Due to PTS=DTS, no meaning)
long_term_reference_flag	0	Do not use longtime memory
adaptive_ref_pic_marking_mode_flag	0	

5.1.1.4 Other restrictions

- The stream input in the CPB is recommended to be set to be decoded within 1.5 seconds.

5.1.1.5 Identification of 16:9 pictures

- Broadcasters may transmit images of 16:9 though the image format is only QVGA. They are 16:9 pictures when pic_height_in_map_units_minus1 of the H.264 stream SPS is 11.
  - As a rule, the pic_height_in_map_units_minus1 is not modified for a program, they should be operated at semi fixed status by broadcasters, however, during the period of simultaneous broadcasting with analog spectrum, the aspect program of 4:3 remains, therefore there is a possibility that it can be changed for a program. It can be changed in each CM unit.

5.1.1.6 Operation of PanScan in the picture display area

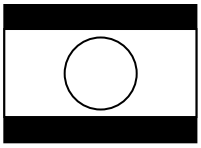
It is possible to not display black borders (picture frame) depending on the picture angle of the receiver unit by configuring the following PanScan parameters when delivering by a different aspect ratio for actual picture source, like side panels and letterboxes.

(1) When displaying a part of an original picture source (320×180) in a 16:9 full screen picture display area, and when the delivery picture format is QVGA 4:3(320×240) and the original image source is 16:9 (letterbox).

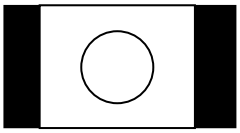
(2) When displaying a part of original picture source (240×180) in 4:3 full screen picture display area, and when the delivery picture format is QVGA 16:9 (320×180) and the original picture source is 4:3 (side panel).

It is necessary to send the center of the delivery image format and the center of the image source corresponding in both horizontal and vertical directions.

The value of each parameter is indicated as follows, when the above-mentioned operation is carried out.



Reference figure (1)



Reference figure (2)

Table 5-14 Restrictions of each parameter during PanScan operation

Definition part	Syntax	Reference figure (1)	Reference figure (2)
Sequence Parameter Set	pic_width_in_mbs_minus1	19	19
	pic_height_in_map_units_minus1	14	11
Pan-scan rectangle SEI message	pan_scan_rect_left_offset	0	640
	pan_scan_rect_right_offset	0	-640
	pan_scan_rect_top_offset	480	0
	pan_scan_rect_bottom_offset	-480	0

PanScan can be turned on-off by the unit of coded video sequence, and when the above PanScan operation is not carried out, the encoding of the Pan-scan rectangle SEI message shall not be done. (When a PanScan operation is done, the Pan-scan rectangle SEI is always included in IDR-AU.)

(*) The receiver unit response to the PanScan is optional.

5.2 Still picture and bitmap figure coding

5.2.1 JPEG

- In conformity with the BaseLine method of ISO/IEC 10918-1(ITU-T T.81).
- For the colorimetryof JPEG, ARIB STD B24 Vol. 1, Part 1, 7.2 "colorimetry" is applied.

5.2.1.1 Coding parameters

- Order of coding: interleaved system
- Loss permitted base line system
- Sampling factor: It is YCBCR = 4:2:0. However, it shall not fail when receiving the 4:2:2 formats. JPEG of the 4:2:0 format indicates the value of SOF0 marker information (H1,V1), (H2,V2), (H3,V3) is each (2,2), (1,1), (1,1), and at the 4:2:2 format indicates, similarly, the value of (H1,V1), (H2,V2), (H3,V3) corresponding to (2,1), (1,1), (1,1) respectively.

5.2.1.2 Other restrictions

- Do not operate progressive mode.

5.2.1.3 Operation markers and marker segments

Markers and marker segments operated by JPEG are shown in Table 5-15.

Table 5-15 Markers and marker segments operated by JPEG

Marker	Observation	Receiver process
SOI	Initiation of image	General process
DQT	Definition of quantization table	General process
DRI	Definition of re-start interval	General process(Note 1)
SOFn	Initiation of frame Only SOF0 (FFC0) is the subject of encoding.	General process
DHT	Definition of Huffman	General process
SOS	Initiation of scan	General process
RSTm	Termination of re-start interval	General process(Note 1)
EOI	Termination of image	General process
COM	Comment	General process
APPn	Used for application	Ignore
DNL	Assignation of special size	Ignore

(Note 1) Response to the error of DRI and RSTm is implementation dependent.

In order for the above SOI – DNL only to appear in the base line system, markers other than the above are treated as errors.

5.2.2 GIF

The graphics file format in GIF uses specifications based on “GRAPHICS INTERCHANGE FORMAT Version 89a" specified by Compuserve Inc. America.

5.2.2.1 Usable blocks

In this chapter, the blocks that can be used are specified, and when the value of each field is limited, the limitations are shown together.

"O" in the following table shows operations as specified.

5.2.2.2 Header

1 will always exist at the start.

Field	Operation	Remarks
Signature	O	Character string and fixed value of “GIF”
Version	89a	File format is Version89a

5.2.2.3 Trailer

1 will always exist at the end.

Field	Operation	Remarks
GIF Trailer	O	Fixed value 0x3B

5.2.2.4 Logical Screen Descriptor

1 will always exist after the Header block.

Field	Operation	Remarks
Logical Screen Width	O	Width of logical screen
Logical Screen Height	O	Height of logical screen
Global Color Table Flag	O	When this bit stands, the Global Color Table block exists right after this block.
Color Pixel size	O	
Sort Flag	O	
Size of Global Color Table	O	Global Color Table size
Background Color Index	O	
Pixel Aspect Ratio	O	

5.2.2.5 Global Color Table

This block is operated according to standards, and the existence is controlled by the Global Color Table Flag of the Logical Screen Descriptor. When it exists, only one exists after the block of Logical Screen Descriptor. Moreover, the Size of Global Color Table of the same block specifies the size.

5.2.2.6 Image Descriptor

Multiple descriptors are possible. Image data follows after this block.

Field	Operation	Remarks
Image Separator	O	Fixed value 0x2C
Image Left Position	O	
Image Top Position	O	
Image Width	0-240	Width of image
Image Height	0-320	Height of image
Local Color Table Flag	O	When this bit stands, the Global Color Table block exists right after this block.
Interlace Flag	O	1 when display interlace, and 0 if not
Sort Flag	O	
Size of Local Color Table	O	Local Color Table size

5.2.2.7 Local Color Table

This block is operated according to standards, and the existence is controlled by the Local Color Table Flag of the Image Descriptor. When it exists, only one exists after the Image Descriptor block. Moreover, the Size of Local Color Table of the same block specifies the size.

5.2.2.8 Image Data

Multiple existences are possible. If there is Image Descriptors or Local Color Tables, this block always exists after it.

Syntax is operated according to standards, Multiple sub-blocks to store the coloration scheme image data compressed in LZW format exist after the 1byte LZW Minimum Code Size field which indicate the size of LZW minimum code used for recovering data.

5.2.2.9 Graphic Control Extension

The operation of this block is arbitrary, but it shall be operated whenever a transparent color and delay time are specified. In that case, it exists only one in front of the Image Descriptor.

Field	Operation	Remarks
Extension Introducer	O	Fixed value 0x21
Graphic Control Extension Label	O	Fixed value 0xF9
Block Size	O	Fixed value 0x04
Disposal Method	O	
User Input Flag	0	0: Do not accept user input at the moment of data display
Transparent Color Flag	O	1 for transparent processing, and 0 when this process is not done. (animation GIF is fixed at 0)
Delay Time	20-500	200 - 5000ms, specified in100ms units
Transparent Color Index	O	Color of transparent color is Global Color Table, or assign by Index from the Local Color Table
Block Terminator	O	Fixed value 0

5.2.2.10 Comment Extension

Operation is arbitrarily and can be multiple. The syntax is operated according to standards.

5.2.2.11 animation GIF operation restrictions

- Total data size

File size is not specified. However, the total size (regardless of display or not)of each 1 document after decoding shall not surpass 76800 for the total of width*height*frame number for all objects.

- Display size

  Size of 1 object is vertical <=240, horizontal <=240

- Number of stored images

  Maximum of 16

- Update cycle specified values

  Minimum: 200 (millisecond)

  Maximum: 5000 (millisecond)

  Specified unit: 100 (millisecond)

- Repetition

  Infinite. But, the actual repetition number is implementation dependent.

- Number of objects for each screen

  Maximum 4. But, each object shall not overlap.

- Rewritable display size for each unit time

  Shall not exceed 38.4KB in 1 second.

- Others

  - Do not change the stored GIF object size and position. In other words it consists of the following.

    LogicalScreenWidth = ImageWidth = width characteristic of object element.

    LogicalScreenHeight = ImageHeight = height characteristic of object element.

    ImageLeftPosition = 0

    ImageTopPosition = 0

  - Even if delays occur in the update of animation GIF that should be generated at the same time as other rendering execution etc., by C-profile basic receivers, do not cut out the GIF images and display the GIF image in order.
  - Transparent processing is not used in animation GIF.

5.3 Audio coding

5.3.1 MPEG-2 AAC

Follows Vol. 7, section 4.2.

5.3.1.1 Coding parameters

Table 5-16 Coding parameters of MPEG-2 AAC

Sampling frequency	Bit length
24kHz (half-rate) 48kHz	16 bit 16 bit

- Quality indicator (quality_indicator) during half-rate sampling is operated at mode 3.
- Multichannel stereo is not operated.

5.3.1.2 Transmission of MPEG-2 AAC

- Audio encoded with MPEG-2 AAC is transmitted by audio PES (stream format identification 0x0F) and the data carousel (stream format identification 0x0D).

5.3.1.3 Limitations of data carousel transmission

- File size is 128KB or less.
- When start orstop controls are received, playing the audio shall start or stop.
- The operation of SBR by data carousel transmission shall be an optional, and the encoding sampling rate is half-rate (24kHz) during SBR operation.

5.3.1.4 Data format of MPEG-2 AAC files

- MPEG-2 AAC Elementary Stream format
- The audio frame composed of audio data that corresponds to the ADTS header as shown in Figure 5-2 is assumed to be one unit, and the format is composed of a single unitor multiple units units. (Because one audio frame becomes a unit of 1024 samples in PCM, it is about 21.3 milliseconds in 48kHz sampling.)

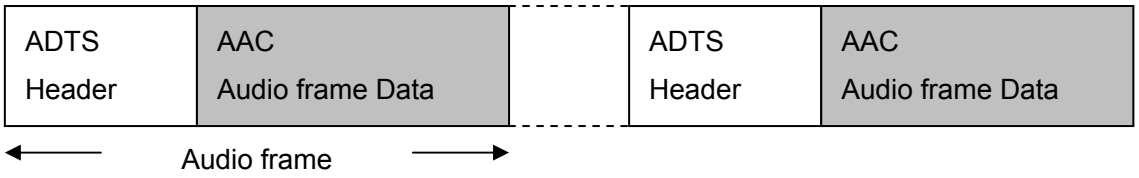


Figure 5-2 Data format of MPEG-2 AAC files

5.3.2 AIFF-C

AIFF-C is not operated.

5.3.3 Built-in sound

The encoding method for the built-in sound of receivers are implementation dependent. The allocation of a built-in sound sources is shown in Table 5-17.

Table 5-17 Allocation of receiver unit built-in sound

0: button operation sound A	1: button operation sound B	2: button operation sound C	3: button operation sound D
4: button operation sound E	5: button operation sound F	6: button operation sound G	7: button operation sound H
8: button operation sound I	9: button operation sound J	10: button operation sound K	11: button operation sound L
12: Alert sound A	13: Alert sound B	14: Alert sound C	15: Alert sound D

The numbers in the table show the sound_id when they are specified from multimedia codes.

5.3.4 MIDI

MIDI is not operated.

5.3.5 Audiosynthesis of receiver units

Simultaneous playback of multiple sounds is optional, but when simultaneous playback is not possible, follow Table 5-18. However, MPEG-2 AAC stream and built-in sound should be output in combination.

Table 5-18 Priority order during multiple audio playback presentation

	MPEG-2 AAC File (storage)	Built-in sound (4)
MPEG-2 AAC Stream (main-line)	Storage priority destination (2)(3)	Built-in sound priority destination (1)
MPEG-2 AAC File (storage)	Afterward priority destination	Built-in sound priority destination
Built-in sound (4)	Built-in sound priority destination	Afterward priority destination

(1) When the MPEG-2 AAC stream is interrupted by built-in sound playback, the interruption time should be minimized in designing the receiver.When mixing both sounds, a sound volume should be 1:1.

(2) When the MPEG-2 AAC file is instructed to playback during MPEG-2 AAC stream playback, the MPEG-2 AAC stream playback stops and priority is given to playback of the MPEG-2 AAC file. However, in this case, the playback of the main line image stream may be stopped.

(3) MPEG-2 AAC stream and main line image stream playback is done after the MPEG-2

AAC file playback terminates however, it is necessary to create contents by noting there is a possibility switching may take some time.

(4) Operations for the timing of audio (for example, button sounds such as mobile phones) and built-in sound overlaps, is implementation dependent. Moreover, the tone quality of a built-in sound is not especially specified. Therefore, the output audio should correspond to the receiver performance.

5.4 Character codes

5.4.1 8-bit character codes for C-profile

Refer to Chapter 6 of this vol..

5.4.2 Shift JIS

Refer to Chapter 3 of this vol..

6 Operation of closed caption coding (option)

6.1 Definition and range of service

In C-profile, the following closed caption services are provided .

Closed caption :Closed caption service (for example, translated character closed caption, etc.) synchronized with main video, audio and data.

This volume complies with [Section 2] "Operational provisions related to profile A", Chapter 4 "Operation of caption and superimpose encoding".

6.2 Organization and transmission operation

6.2.1 Restrictions on organization and transmission

(1) Transmission method

It transmits by the independent PES transmission method (stream format identification 0x06).

(2) Organization

Transmitted by independent ES. Moreover, it is transmitted by the PMT which is the same as the main service, at the same time, and the delivery of closed caption data within the same program or before the program starts is not done.

(3) The Number of ES

Closed caption that can be transmitted simultaneously is 1 ES.

(4) Transmission of multiple languages

The number of languages that are transmitted simultaneously is maximum of two or less per 1ES, and the language identification is done by the closed caption management data and the data group identification of the data group in the ES. However, the display of closed caption for the second language is not necessary in the receiver.

(5) Bitmap data

Bitmap data is not operated.

(6) Display modes that can be used

This is a fixed operation only for the "the choice of caption and displayduring reception, and the choice of caption and displayduring the playback of recording". It is a same when multiple languages are transmitted.

(7) Operation of built-in sound and additional sound

Neither the sound with built-in the receiver nor additional sound is operated.

(8) Existence confirmation of closed caption

The closed caption management data and the closed caption data show that the closed caption exists in the stream. Display of closed caption text is carried out after the closed caption management data or closed caption data is received but because the operation of the data group is fixed (Refer to 6.2.3), the display of the closed caption text after the closed caption data is received is possible even if the closed caption management data is not received.

Standard delivery frequency 1 time/10 sec. (permissible deviation of ±5 sec.)

However, the delivery of the following closed caption management data can be omitted when the closed caption text data is delivered, within the above-mentioned minimum delivery frequency (5 sec.), however, in this case, both the closed caption text data and closed caption management data should meet requirements of the above-mentioned standard delivery frequency. Moreover, the delivery of the closed caption management data may be interrupted by CM's etc.

6.2.2 PES transmission method used in closed caption

The synchronous type PES transmission method is applied and the timing synchronization is maintained by the PTS. Table 6-1 shows the parameters set in the PES packet.

- Configuration parameters : Refer to Table 6-1
- Maximum number of ES's transmitted to the same layer simultaneously : 1ES
- Maximum number of languages per ES : 2 languages
- PES configuration unit : 1 data group
- PES maximum size : 640bytes (However, 3 TS packets is the upper limit)
- PES packet minimum delivery interval : 1000 milliseconds (However, CS: for PES only for the screen erasure does not follow this.)
- Maximum ES rate : 3Kbit/s
- Reception buffer : 1280bytes or more (for both 1language and 2 languages)

Apart from this, 256bytes is needed for the DRCS. Refer to 6.6.

Table 6-1 Configuration parameters of PES packets in closed caption

Field	Operation
Stream_id	0xBD(private_stream_1)
PES_packet_length	Number of bytes in a PES packet follow later. *1
data_identifier	0x80
private_stream_id	0xFF
PES_data_packet_header_length	Shows the length of PES_data_private_data_byte. Normally 0x00 is input. *2
PES_data_private_data_byte	This field can be skipped. *2
Synchronized_PES_data_byte	Stores data of the closed caption data group.

*1 Input 0 into this value, and operations not provided in the PES packet length are prohibited.

*2 When operating PES_data_private_data_byte, specify the correct length of PES_data_private_data_byte in the PES_data_packet_header_length, without fail.

For PES packet delivery, the following restrictions are established.

- Delivery order of PES packets and the time order of PTS should not be interchanged.
- In the ‘n’th order of the PES packet at PTS time, the total information volume of PES packets starting delivering in ‘n’th order should not exceed the reception buffer capacity (1280bytes).
- Complete sending PES packets before Td, and from the PTS time. Here, Td is the time from reception completion to presentation completion, almost 0.5 seconds as a rough estimate.
- Regarding the interval of PES packet delivery of the closed caption text data group, the interval of the PTS time of delivery order ‘n’th PES packet and PTS time of ‘n-1’th PES packet should be bigger than the Td of nth data.

  Transmission of synchronized with Video are possible on the transmission side when the image of the total delay T meets the following requirements.

  T > LX 8 / R + Td

  Here, L is the maximum PES packet length and R is the ES bit-rate.
- The movement of receiver exceeding the reception buffer is implementation dependant. The movement of receiver exceeding the reception buffer for the DRCS is implementation dependant, however, it is recommended to have movements described in 6.6.

6.2.3 Operation of data groups

Data group transmission shall be operated as shown in Table 6-2. The “data_group_version” will not be operated.

When class A is received, the receiver shall judge the number of languages to be 1 and process only the 0x01 closed caption text of class A (body text and DRCS). When class B is received, the receiver shall judge the number of languages to be 2 and, if the user selects

the closed caption of the first language, process only the 0x21 closed caption text of class B (body text and DRCS). If the user selects the closed caption of the second language, the receiver shall process only the 0x22 closed caption text of class B (body text and DRCS).

Table 6-2 Data group parameters

Field	Operation
data_group_id	• If the number of closed caption languages is 1, the closed caption management data and the closed caption text shall be operated with 0x00 and 0x01 of class A, respectively. 0x20, 0x21 and 0x22 of class B will not be operated. • If the number of languages is 2, the closed caption management data shall be operated with 0x20 of class B and the first and second closed captions shall be operated with 0x21 and 0x22 of class B, respectively. 0x00 and 0x01 of class A will not be operated.
data_group_version	Not operated.
data_group_link_number	0x00
last_data_group_link_number	0x00
data_group_size	Operated as defined. However, 1 PES packet should not exceed 640bytes.
data_group_data_byte	Data group data (closed caption management data, closed caption text data) is stored.
CRC_16	Operate error checking by CRC16. On detecting errors, receiver delete aforesaid data group.

6.2.4 Operation of closed caption management data

Data units are not allocated in closed caption management data. The movement of receiver that do not receive closed caption management data is implementation dependant. Table 6-3 shows the parameters that can be specified for the closed caption management data used in closed caption.

Table 6-3 Closed caption management data parameters

Field	Operation
TMD	‘00’ (free)
num_languages	1 - 2
language_tag	0 - 1
DMF	‘1010’ (the choice of caption and display on reception and the choice of caption and display on recording playback)
ISO_639_language_code	Used language code (“jpn” fixed)
Format	Not operated ‘1111’
TCS	‘00’ (8-bit character codes)
rollup_mode	‘00’ (non-rollup)
data_unit_loop_length	‘0’
data_unit	Do not allocate data units.

6.2.5 Operation of closed caption text data

In the same closed caption text data, multiple data units of the same or different data unit parameters can be allocated. When multiple data units exist in the same closed caption text data, they are processed in order of appearance of the data units.

Configuration parameters for closed caption text data is shown in Table 6-4.

Table 6-4 Parameters of closed caption text data

Field	Operation
TMD	’00’ (free)
STM	Not operated.
data_unit_loop_length	Operated as defined. However, 1 PES packet should not exceed 640bytes.
data_unit	Data unit (text, 1 byte DRCS) is stored.

6.2.6 Operation of data units

Parameters that can be configured in data units are shown in Table 6-5.

Table 6-5 Data unit parameters

Field	Operation
unit_separator	0x1F as defined.
data_unit_parameter	0x20 (text) 0x30 (1 byte DRCS)
data_unit_size	Operates as defined. However, 1 PES packet should not exceed 640bytes.
data_unit_data_byte	Stores data unit data.

6.2.7 Operation of PSI/SI

6.2.7.1 Operation of component tags

Component tag value of the closed caption ES is 0x87.

6.2.7.2 Operation of the PMT

Updating the PMT is recommended to add and delete ES information at the moment of initiation and termination of closed caption. However, an operation to enable the description of ES information at all times is needed.

6.2.7.3 Stream format identification

The stream_type of the closed caption ES is 0x06 (independent PES_packet).

6.2.7.4 Descriptor operation

Descriptor operation of the PMT and EIT for closed caption is shown in Table 6-6.

Table 6-6 Descriptor operation of the PMT and EIT

Descriptor	PMT	EIT
Stream Identifier Descriptor	Mandatory	-
Data Component Descriptor	Mandatory	-
Data Contents Descriptor	-	Not operated

In the Short Event Descriptor of EIT, it is recommended to descript character information that can inform users that this program comes with a closed caption function.

6.2.7.5 Data Component Descriptor

The data_component_id of the Data Component Descriptor is 0x0012. And Table 6-7 shows the parameters to configure the additional information identification.

Table 6-7 Configuration parameters of additional information identification for the Data Component Descriptor

Field	Operation
DMF	’1010’
Timing	Closed caption: ’01’ (Program synchronization)

6.2.7.6 Target Region Descriptor

The Target Region Descriptor is not operated.

6.2.7.7 Data Contents Descriptor

The Data Contents Descriptor is not operated.

6.3 Display format of closed caption

6.3.1 Display format

The broadcasting station considers two kinds of rectangular areas, a lateral direction of a normal size of 12 characters or more with 4 lines (in case a portrait display device is used on a vertical screen) and a lateral direction of a normal size of 16 characters or more with three lines (in case of a portrait display device is placed horizontally and used on a horizontal screen), as display formats to present closed caption in the receiver , and to produce the closed caption text in consideration of displaying all characters of the closed caption text data, even if the receiver provides a display area with the above-mentioned features.

The receiver should have one of above-mentioned display areas, and displays with a linefeed in the place of the control code of the active position line feed of the closed caption text, and displays with line feed for the character writing direction edge in the display area of the receiver . Only 1 linefeed, not 2, is processed when the line feed of the operation position linefeed control code and the display area character writing direction edge overlaps.

The display area size of the receiver depends on the implementation dependant of the receiver , but when receiving the closed caption text produced according to the above-mentioned assumption, all characters should be able to be displayed.

Overlapping of the closed caption text and the image is implementation dependant. Moreover, it is recommended, in implementation dependant, to display the closed caption text for receiver that do not meet the display format of the above-mentioned assumption appropriately.

The image chart in Figure 6-1 shows the conversion and production of closed caption from profile A to C-profile.

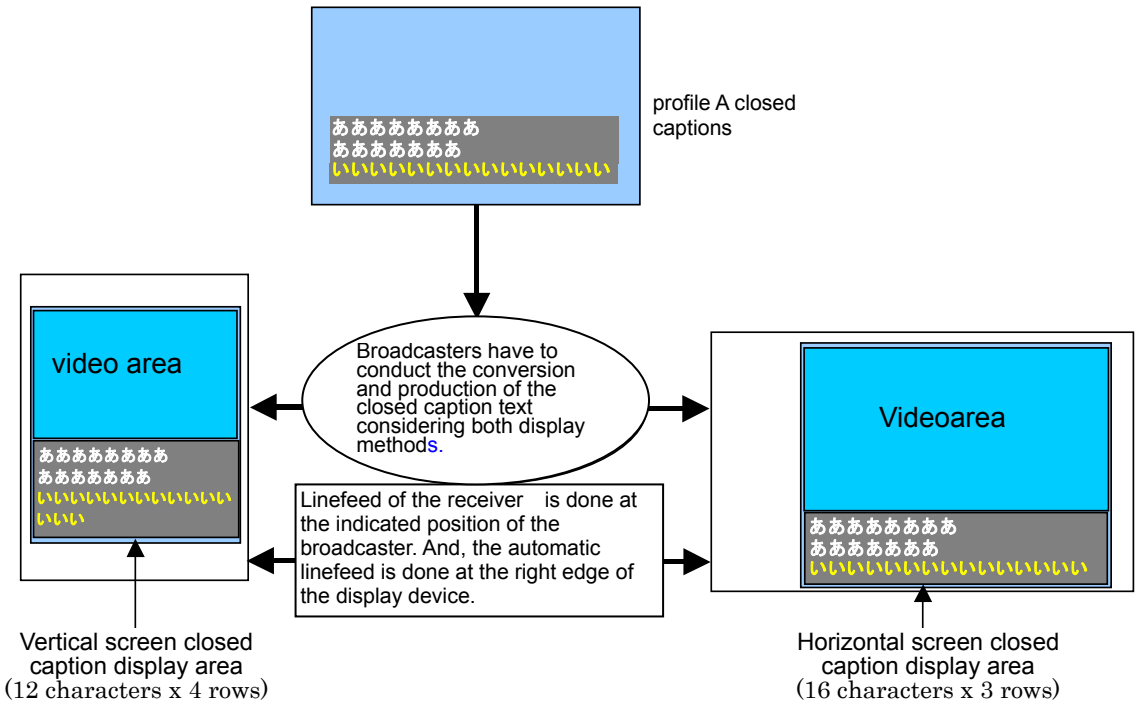


Figure 6-1 Conversion and production image of C-profile closed caption

6.4 Characters used in closed caption

6.4.1 Character entity

The character encoding method used in closed caption uses 8-bit character codes for C-profile with a subset of 8-bit character codes for C-profile. In 8-bit character code for C-profile, the GL code area is fixed to the DRCS character set (one byte code) provisioned by ARIB STD-B24, the GR code area is fixed to the Kanji character set (two byte code, Row 1 to Row 94) provisioned by ARIB STD-B24, and the character set table is not changed. Moreover, (3) below not used as a character code is used as a control code.

(1) Kanji character set range

First byte A1 - FEh

Second byte A1 - FEh

(2) DRCS set range

21 - 7Eh

(3) Control code range

00 - 1Fh, 20h, 7Fh, 80 - 9Fh

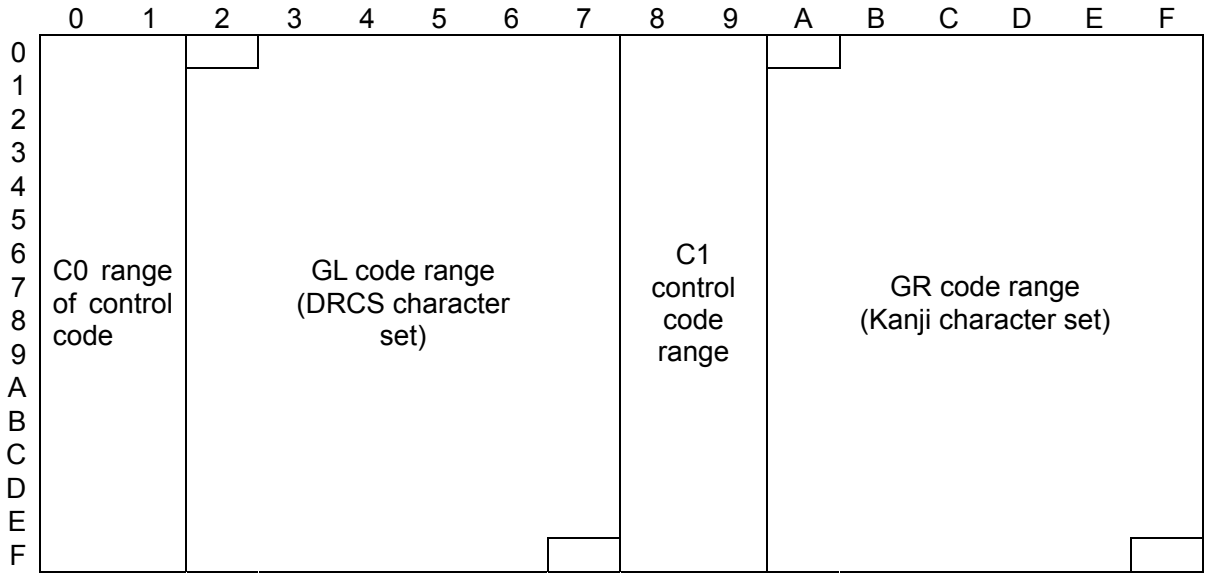


Figure 6-2 Table of 8-bit character codes range for C-profile

6.4.2 Character fonts

This is implementation dependant of the receiver .

6.4.3 Character size

Character size displayed in closed caption is assumed as medium font. Only standard and medium character size are operated during transmission.

Normal size: medium font equivalent

Medium size: Only the size of the horizontal direction is half the size of characters from the standard.

Medium size is not operated except for characters specified by operation of medium size characters shown in Table 6-8. Moreover, the voiced sound mark specified as medium may be converted into medium size double characters in the receiver .

Table 6-8 shows the restrictions of character display.

Table 6-8 Range of code set used to specify the display size format and characters

Character	2 byte code Row (Cell)	Standard	Medium
Symbol	1, 2	O	O*1 *2
Alphanumeric	3	O	O
Hirakana	4	O	O*2
Katakana	5	O	O*2
Greek alphabet	6	O	
Russian character	7	O	
Ruled line	8	O	
Kanji	16 - 84	O	
Additional symbols	90 (1 - 6) (8 - 11) (16 - 17) (20 - 40) (64 - 65)		
	91 (1 - 49)		
	92 (1 - 4) (5 - 12) (13 - 15) (16 - 25) (26 - 31) (32 - 41) (42 - 47) (48 - 52) (53 - 54) (55 - 91)	O O O O O O O O O	
	93 (1 - 45) (48 - 91)	O O	
	94 (1 - 93)	O	
DRCS	Normal size pixel configuration Medium size pixel configuration	O	
Special code (space, delete)		O	O

*1 Exclude Row 2, Cell 94.

*2 This may be displayed by the normal size in the receiver .

6.4.4 Display partition

As this is a letter space and line space without a specified operation, the definition of display partition is not done.

6.4.5 Non-spacing characters

Non-spacing characters are not used.

6.5 Control code used in closed caption

6.5.1 Control code

The control code used in the closed caption is compliant with ARIB STD-B24, Vol. 1, Part 2, 7.1.2. However, the operation limitations in Table 6-9, Table 6-10, Table 6-11 are established.

Table 6-9 C0 control range

C0 Control code	Control function	Yes or no, of use	Control item and supplement
NUL	Space	O	
BEL	BEL	X	
APB	Active position backward	X	
APF	Active position forward	X	
APD	Operation line advance	X	
APU	Operation line retreat	X	
APR	Active position return	O	
PAPF	Specified operation position advance	X	
APS	Active position specification	X	
CS	Clear screen	O	
CAN	Cancel	X	
ESC	Escape	X	
LS1	Locking-shift 1	X	
LS0	Locking-shift 0	X	
SS2	Single-shift 2	X	
SS3	Single-shift 3	X	
RS	Data header identification code	X	
US	Data unit identification code	O	Used for identification of data units, but cannot be used at 8-bit character codes for C-profile.

O: possible use Δ: possible use with limitations X: cannot be used

Table 6-10 C1 control range

C1 Control code	Control function	Yes or no, of use	Control item and supplement
BKF (CFLA0)	Foreground color Black	O	Specify Black to foreground (index value =0)
RDF (CFLA1)	Foreground color Red	O	Specify Red to foreground (index value =1)
GRF (CFLA2)	Foreground color Green	O	Specify Green to foreground (index value =2)
YLF (CFLA3)	Foreground color Yellow	O	Specify Yellow to foreground (index value =3)
BLF (CFLA4)	Foreground color Blue	O	Specify Blue to foreground (index value =4)
MGF (CFLA5)	Foreground color Magenta	O	Specify Magenta to foreground (index value =5)
CNF (CFLA6)	Foreground color Cyan	O	Specify Cyan to foreground (index value =6)
WHF (CFLA7)	Foreground color White	O	Specify White to foreground (index value =7)
COL	Color specification	X	
POL	Pattern polarity	X	
SSZ	Small size	X	
MSZ	Medium size	Δ	Operation complies with Table6-8.
NSZ	Normal size	Δ	Operation complies with Table6-8.
SZX	Specification size	X	
FLC	Flashing control	Δ	Flashing speed is so that users can recognize the flashing of characters. Operation of flashing, refer to 6.5.2.
CDC	Conceal control	X	
WMM	Writing mode modification	X	
TIME	Time control	Δ	Only usable for process waiting.
MACRO	Macro specification	X	
RPC	Character repeat	X	
STL	Start underline and mosaic separation	X	
SPL	Stop underline and mosaic separation	X	
HLC	Enclose control	X	
CSI	Control sequence introducer	X	

O: possible use Δ: possible use with limitation X: cannot be used

6.5.1.1 Operation of color specifications

The color specification is only operated for the foreground color, and specified from 8 colors of index0 to index7 of the CLUT common fixed colors allocated in CFLA0 to CFLA7 of C1 control code in Table 6-10. The background color, the half foreground colors and backgrounds neutral color are not specified.

It is recommended that the broadcaster operate the color specification in the same line within 3 times.

Refer to Appendix 1 of [Section 2] related to profile A for the CLUT common fixed colors.

6.5.2 Operation of flashing

Flashing of 8-bit character codes character string for C-profile should flash at a speed which the user can recognize the character flashing. Timing in which the flashing is started is the time when the character of flashing specification is shown. Moreover, flashing is assumed to be only a positive-phase-sequence. When the closed caption initialization operation defined in 6.7 is performed and the deletion of the display screen is indicated by the CS control code, the flashing of an aforesaid character is terminated.

6.5.3 Extension control code

Operation of the extension control code is shown in Table 6-11.

Table 6-11 Extension control code (CSI)

Character	Control function	Yes or no, of use	Control item and supplement
SWF	Format selection	X	
CCC	Synthesis control	X	
RCS	Luster color control	X	
ACPS	Operation position coordinates specification	X	
SDF	Display configuration dot specification	X	
SDP	Display position specification	X	
SSM	Character configuration dot specification	X	

Character	Control function	Yes or no, of use	Control item and supplement
PLD	Partial line down	X	
PLU	Partial line up	X	
SHS	Character interval specification	X	
SVS	Line interval specification	X	
GSM	Character transformation	X	
GAA	Coloring division	X	
SRC	Luster specification	X	
TCC	Switch control	X	
CFS	Character font setting	X	
ORN	Character decorations specification	X	
MDF	Character format at specification	X	
PRA	Built-in sound playback	X	
XCS	External character set alternative code string definition	X	
SCR	Scrolling specification	X	

O: possible use Δ: possible use with limitation X: cannot be used

6.6 Operation of the DRCS

Calling of the DRCS uses only one arbitrary set from DRCS-1 to DRCS-15 in the one byte DRCS sets in ARIB STD-B24, Vol. 1, Part 2, 7.1.1.5, and the two byte DRCS set DRCS-0 is not used. Moreover, the encoding of the DRCS pattern operates only the pattern transmission, and operation by geometric is not done. Moreover, the fontId (font identification) should specify 0. When specified numbers except 0, the receiver considers it as 0. Buffers ensured by the receiver for DRCS is 256 bytes in DRCS for closed caption. The maximum number of DRCS used, at the same time, in closed caption should be 5. When 6 or more are delivered, it is mandatory that the first 5 be kept in the buffer of the receiver , and reading continues repeatedly. The handling of the 6th depends on the receiver . The patternData should send two tone data of the design frame based on the dot

configuration (horizontal and vertical of 16×18) in ARIB STD-B37 "Structure and operation of closed caption data conveyed by ancillary data packets", Supplement "Operational guidelines for closed captions".

Table 6-12 Active parameters of DRCS figure coding

Field	Operation
NumberOfCode	Operated as defined in ARIB STD-B37
CharacterCode	Operated as defined in ARIB STD-B37
NumberOfFont	Operated as defined in ARIB STD-B37
FontId	Specification of only 0 is possible. (0, even if specified other than 0)
Mode	Specification of only 0000 is possible.
Depth	Specification of only 0 is possible.
Width	Operated as defined in ARIB STD-B37
Height	Operated as defined in ARIB STD-B37
patternData	Operated as defined in ARIB STD-B37

6.7 Operation of the initialization operation

The receiver shall perform the closed caption initialization operation shown in 6.7.1 to 6.7.5.

Table 6-13 Initialization status

Item	Target	Initial status	
Display screen	Display image	Character delete status	
Definition data	DRCS	Data clear	
Operation instruction	Character coding	Time control operation	End status
		Operation position	Left top corner of display area
Status instruction	Character coding	Character size controls	1X1 (standard)
		Foreground color	Maximum white brightness (CMLA7)
		Background color	Transparent (CMLA8)
		Flashing control	Flashing end (FLC04/15)

6.7.1 Initialization by switching classes in the data group

The receiver shall initialize all items prescribed in Table 6-13 when it receives closed caption text data or closed caption management data of a different data group class than those currently being processed for presentation.

6.7.2 Initialization by closed caption text

The receiver shall initialize all items prescribed in Table 6-13 when it receives closed caption text data of the same data group class and language as those currently being processed for presentation.

6.7.3 Initialization by main text data unit

The receiver shall initialize the items in "Status instruction" in Table 6-13 when the received closed caption text data of the same data group class and language as those currently being processed for presentation which contains the body text data unit, immediately before processing the body text data unit for presentation on the receiver.

6.7.4 Initialization by character control codes

The receiver shall initialize the items under "Display screen", "Operation instruction" and "Status instruction" in Table 6-13 immediately before executing the clear screen (CS) process.

6.7.5 Initialization by receiver operation

The receiver shall initialize all items prescribed in Table 6-13 when a station selection operation or switching of the closed caption language has been performed.

6.8 Mono-media used in closed caption

6.8.1 Operation of geometric

Geometric is not operated.

6.8.2 Operation of bitmap data

Bitmap data is not used.

6.8.3 Operation of built-in sound

Built-in sound is not operated.

6.8.4 Additional sound

Additional sound is not used.

6.9 Ideal receiver unit operation

It is preferable that the receiver unit judges the presence of closed caption data reception by the presence of the closed caption management data. The presence of the closed caption management data or PMT data is set as standard, when displaying symbols that inform

viewers of closed caption reception, and it is based on the presence of the title management data or the PMT data.

6.9.1 Initialization and termination of closed caption display

The receiver unit controls initialization and termination of the closed caption function by operation of the user. The control of initialization and termination of closed caption function tied with other data like the title management data or PMT data, etc. is a matter of product planning.

7 Operation of multimedia encoding

7.1 Introduction

The operation of the multimedia encoding follows,

- "Appendix 1, Operational Guidelines" and
- "Appendix 4, Operational Guidelines for Implementing Services for Portable Receiving System",
  in ARIB STD-B24, Vol. 2 " XML-based Multimedia Coding Scheme ".

However if there is contradiction, the specification in this volume have a priority to above normative references. The normative reference that is not operated in the specification in this volume is not applicable.

When the data broadcasting reception function is implemented in receivers that receive C-profile data broadcasting contents, it is assumed that the communication function is implemented at the same time. Because, in portable reception, the bandwidth that can be used by data broadcasting is narrow, it is assumed that contents are produced on the assumption of a tie-up with communication. Therefore, when such contents are presented in receivers that do not have the communication function, only incomplete functions and information may be provided to the users.

However, if receiver unit designers are aware of the above assumption, they may design receivers without the communication function.

7.2 Operation of NVRAM in Digital Terrestrial Television C-profile broadcasting

Digital Terrestrial Television C-profile area for the affiliation (hereafter referred to as C-profile area for the affiliation) and TVlink area shown in Table 7-1 shall be set up for allocation as NVRAM to be used for storing permanent information in the C-profile digital terrestrial television broadcasting.

Table 7-1 NVRAM used by Digital Terrestrial Television C-profile broadcasting

Class	Purpose	Capacity of NVRAM
Digital Terrestrial Television C-profile area for the affiliation	Area used commonly by operators belonging to the same affiliation area	• 24KB per affiliation (64 byte fixed blockx 384) • 8KB out of24KB is for inner affiliation common area, and the remaining 16KB is divided by 8 and used as the individual operator area. • Affiliation number: 12 affiliations
TVlink area	Area used for TVlink service	• Maximum of 256 bytes variable block • Writable block number :50 or more

C-profile area for the affiliation and TVlink area shall ideally be allocated statically on the receiver; however, implementation in which they will not be allocated statically shall also be acceptable. In other words, the non-volatile physical memory area can be shared with other applications on the receiver.

In the case of a receiver on which the above areas have not been allocated statically, the writing operation to the area(s) from a BML document may fail if the relevant memory area is in use by another application. On such a receiver, relevant area is assumed to be allocated at the time of the first writing operation to each affiliation area. And, the unit (capacity) for the C-profile area for the affiliation shall ideally be allocated in the unit of 1 affiliation (24KB).

7.2.1 Writing frequency of the NVRAM area

NVRAM implemented in the receiver uses a device that has an upper limitation in the writing frequency. These devices may breakdown, and results in shortening the life of the receiver when the writing frequency is exceeded the limit. Therefore, it is recommended to consider not generating excessive writing frequency to NVRAM. This is explained in appendix 1.

7.2.2 Allocation of NVRAM

7.2.2.1 Allocation of the C-profile area for the affiliation

- The maximum number of affiliations to which a C-profile area for the affiliation can be allocated shall be 12. In addition to the allocation to affiliations whose affiliation identification (affiliation_id) defined in Vol.7 have the following 8 values, allocation to affiliation_id=08-0B shall also be possible as reserve areas, which shall not be operated for the time being.

  Value of affiliation identification where the C-profile area for the affiliation is currently allocated:

  00, 01, 02, 03, 04, 05, 06, 07

- The reserve affiliation _id will be allocated by the governing organization described in Appendix 13.
- An initial value of the reserved C-profile area for the affiliation is a null character string.
- When the C-profile area for the affiliation has not been allocated yet, the readPersistentArray function returns null.
- Although the C-profile area for the affiliation has already been allocated, the readPersistentArray function shall return Array with the Array[0] set to null in the event when the access to the C-profile area for the affiliation was rejected by user operation or receiver setting (*).
- When writing is instructed for the first time in receivers with an implementation to reserve the memory area, it is allowed that reserving the area may be impossible due to a lack of memory.

  (*) For example, it refers to cases such as when the user was prompted by the receiver to either grant or reject access to the C-profile area for the affiliation and chose to reject the access; or when the user has configured the receiver to reject access to the C-profile area for the affiliation.

7.2.2.2 Allocation of individual operator areas in the C-profile area for the affiliation

- An affiliation-common area (8KB), which can be freely accessed by stations within the affiliation and 8 areas, which can be used exclusively by individual broadcasters shall be set in the C-profile area for the affiliation. Each operator shall be allowed to use an area of up to 2KB.
- It is necessary for receivers to acquire and release the area administrating the individual operator area by an original network unit (2KB, 32 block unit).
- In the case of a C-profile receiver, which is assumed for mobile reception, receivable broadcasting stations will change depending on the receiving location. Therefore, it shall not be possible to associate the individual operator areas in the C-profile area for the affiliation with broadcasting stations beforehand, nor shall it be possible to specify the maximum number of operators that will perform writing to the areas.
- The receiver shall allocate an individual operator area within the C-profile area for the affiliation to the relevant broadcaster in the order that each station's content wrote to the area by specifying an original network ID. When a writing operation exceeding the number of areas available on the receiver (i.e. 8) has been instructed, the writing operation shall be performed after deleting unnecessary areas by user instruction. The writing will fail if sufficient capacity could not be allocated.

7.2.3 Deletion function of the C-profile area for the affiliation

- It is necessary for receivers to have a delete function for each affiliation area and, for each individual operator area set in the affiliated area, both by instruction of the user.
- Regarding the individual operator areas, it is necessary that the user can delete by one area unit.

The guidelines of the deletion management in the C-profile area for the affiliation of the receiver are shown as follows.

- It is recommended that deletion function is able to be executed (without stopping the execution of contents) when a write function is calling with specifying original network ID in the case of no space for the individual operator area, and also is implemented as the deletion function from the menu prepared by the receiver.

- Specifically, when the writing function is called with specifying a new original network ID by an affiliated operator with all 8 individual operator areas already is used, the receiver unit shows the candidate area to delete to the user through a display window, etc., and executes the deletion of each of area according to the user's selection. When the user doesn't order the deletion, writing fails. When presenting candidates of deletion, it is recommended to present the broadcasting station name. It is also recommended to record the presented broadcasting station name with SI information (TS name transmitted by the TS Information Descriptor of NIT) when writing is done specifying original network ID at the beginning (when securing the area).
- An initial value of the individual operator area when the deletion is executed is a null character string.
- The deletion function is executed from the menu prepared by the receiver, and the deletion of each affiliated area is executed by instruction of the user. It is recommended to present an affiliated name when presenting the candidate of the deletion of each affiliated area. The affiliation name presented is assumed to be stored in the receiver beforehand. At this time, by presenting a dialog box or another way to the viewers, it is recommended to display that serious problems may occur to a certain kind of the broadcasted services by the deleting.

- Other additional deletion functions are a matter of product planning.

- It shall be desirable to set up the function for deleting the C-profile affiliation area exclusively for the broadcasting application. If the deletion function will be shared with other applications, the viewer shall be given sufficient warning to prevent accidental erasure of information.

7.2.4 Identification of the C-profile area for the affiliation

When reading and writing information from the Multimedia service to the C-profile area for the affiliation, readPersistentArray()/ writePersistentArray() shall be executed treating each fixed length block as 1 file. Reading and writing of information from the Multimedia service to C-profile area for the affiliation shall be performed in units of fixed length blocks. The following URI shall be used for identifying the fixed length blocks.

nvram://<affiliation _id>;<original_network_id>;Cprogroup/<block number>
<affiliation _id>: Affiliation ID. Described by 2 hexadecimal digits
(0 have should be placed on the left end to make it to 2 digits when the value does not reach 2 digits)
<original_network_id>: Original network ID. Described by 4 hexadecimal digits.
<Block number>: 0 to 31

<affiliation_id>,<original_network_id> cannot be omitted

Access to this area shall fail when the affiliation ID is specified with an ID that does not belong to the broadcaster broadcasting contents currently being played. For the affiliate ID, See Vol. 4, Chapter 9.

"0000" to "0003" can be specified for the area of original_network_id, besides the original network ID of the broadcaster currently broadcasting. When these values are specified, they are treated as a common area in the affiliation area, and they permit access from all stations of same affiliation_id. Access control by original network ID is not done.
Access to this area fails when an original network ID other than the broadcaster broadcastings contents currently playing in the area of original_network_id is specified.

7.3 Operation of keys from the Multimedia service

7.3.1 Values handled for used-key-list characteristics

The table below shows the possible combination of the <key-group> values and the keys that can be specified in the CSS property “used-key-list" to control the exclusive access of keys by a BML browser or the tuning function. The remote control keys shown in Table 7-3 are virtual names not physical keys. A key to allocate physically in the key code shown in Table 7-3 is a matter of product planning.

However, the receiver unit should offer viewers the keys virtually (software key or physical key), because contents are produced assuming the key codes shown in Table 7-3 can be sent.

Table 7-2 Values of key-groups

<key-group>	Meaning
basic	Enter key, back key
numeric-tuning	Numerical key (from 0 to 9) (Note 1)
special-1	*, # (Note 1) (Note 2)
special-2	TVlink key

(Note 1) Note that the key-group is specified for CSS, there can be no way to tune.

(Note 2) The key-group should be used in conjunction with numeric-tuning key-group.

7.3.2 Correspondence of keys, key codes and access keys

Table 7-3 shows the mapping of the characters specified as keys, key codes, and access keys that can be used by the Multimedia service.

Table 7-3 Correspondence table of keys, key codes and access keys

Key	Key code	Access key character
0 to 9	5-14	N/A
“Enter”	18	N/A
“Back”	19	‘X’
TVlink key	100	N/A
*	101	N/A
#	102	N/A

7.4 Operation of the BML version

The BML version in the BML document is major_version=12 and minor_version=0. This PI is described following the XML declaration and the DOCTYPE declaration. Refer to appendix 9 for the receiver behavier when the update of major_version and minor_version.

<?bml bml-version="12.0" ?>

7.5 Operation of character coding schemes

See ARIB STD-B24, Vol. 2, Appendix 4, “4.1. Character Coding Schemes”.

7.6 Operation range of media type and monomedia

It follows ARIB STD-B24, Vol. 2, Appendix 4, "4.2. Media Types and Operational Range of Monomedia ".

7.7 Operation of the BML elements

This section is specified for the operation of BML elements and attributes. The following modifications and restrictions are added although ARIB STD-B24 Vol. 2, Appendix 4, are referred.For the area not described here, ARIB STD-B24 is applied .

7.7.1 Declaration of XML and DOCTYPE

The XML declaration and DOCTYPE declaration in this operation are as follows.

- XML declaration
  <?xml version="1.0" encoding="Shift_JIS"?>
- DOCTYPE declaration
  <!DOCTYPE html PUBLIC "-//ARIB//DTD XHTML BML 12.0//JA"
  "http://www.arib.or.jp/B24/DTD/bml_12_0.dtd">

7.7.2 Operation of the BML elements

Operation in the basic service of elements used by BML are shown. The treatment of the element in "-" is implementation dependent. See appendix 3 for DTD of the operation.

Table 7-4 Operation of elements

Module		BML (operation)	Element
Core	Structure (required)	O	body
		O	head
		O	html
		O	title
	Text (required)	-	abbr
		-	acronym
		-	address
		-	blockquote
		O	br
		-	cite
		-	code
		-	dfn
		O	div
		-	em
		-	h1 - h6
		-	kbd
		O	p
		O	pre
		-	q
		-	samp
		O	span
		-	strong
		-	var
	Hypertext (required)	O	a
	List (required)	-	dl
		-	dt
		-	dd
		-	ol
		-	ul
		-	li
Applet (deprecated)		-	applet
		-	param
Text Extension	Presentation	-	b
		-	big
		-	hr
		-	i
		-	small
		-	sub
		-	sup
		-	tt
	Edit	-	del
		-	ins
	Interaction channel	-	bdo
Form	Basic Forms	-	form
		-	input
		-	label
		-	select
		-	option
		-	textarea
	Forms	O	form
		O	input
		-	select
		-	option
		O	textarea

Module		BML (operation)	Element
		-	button
		-	fieldset
		-	label
		-	legend
		-	optgroup
Table	Basic Tables	-	caption
		-	table
		-	td
		-	th
		-	tr
	Tables	-	caption
		-	table
		-	td
		-	th
		-	tr
		-	col
		-	colgroup
		-	tbody
		-	thead
		-	tfoot
Image		O	img
CS Image Map		-	*a&*
		-	area
		-	*img&*
		-	map
		-	*object&*
SS Image Map		-	*img&*
Object		O	object
		-	param
Frames		-	frameset
		-	frame
		-	noframes
Target		-	*a&*
		-	*area&*
		-	*base&*
		-	*link&*
		-	*form&*
Iframe		-	iframe
Intrinsic Events		O	*a&*
		-	*area&*
		O	*form&*
		O	*body&*
		-	*label&*
		O	*input&*
		-	*select&*
		O	*textarea&*
		-	*button&*
Metainformation		O	meta
Scripting		-	noscript
		O	script
Stylesheet		-	style
Style Attribute		O	
Link		O	link
Base		-	base
Name Identification (deprecated)		-	*a&*
			applet&
			form&

Module		BML (operation)	Element
			frame&
			iframe&
			img&
			map&
Legacy (deprecated)		-	basefont
		-	center
		-	font
		-	s
		-	strike
		-	u
		-	*body&*
		-	*br&*
		-	*caption&*
		-	*div&*
		-	*h1-h6&*
		-	*ht&*
		-	*img&*
		-	*input&*
		-	*legend&*
		-	*li&*
		-	*ol&*
		-	*p&*
		-	*pre&*
		-	*script&*
		-	*table&*
		-	*tr&*
		-	*th&*
		-	*td&*
		-	*ul&*
BML	BML	-	*bml:bml*
		-	*bml:bevent*
		-	*bml:beitem*
		-	*body&*
		-	*div&*
		-	*p&*
		-	*a&*
		-	*bdo&*
		-	*object&*
	Basic BML	-	*bml:bevent*
		-	*bml:beitem*
		-	*body&*
		-	*div&*
		-	*p&*
		-	*span&*
		-	*object&*
	Basic Mobile BML	O	*bml:bevent*
		O	*bml:beitem*
		O	*object&*

7.7.3 Attributes

Table 7-5 shows the operation of the attributes specified as operated in section 7.7.2 (div p br pre span a form input textarea img object meta title script link body head html bml:bevent bml:beitem).

Table 7-5 Operations for attributes of elements

Legend:

In “Operation” column, "O" means the attribute can be shownin the BML document, "Δ" means the value is fixed in the receiver units, then should not be shown in the BML document.

Element	Attribute	Operation	Restriction on operation
Common Attributes			
Core Attributes			
	id	O	Character string, maximum of 128 bytes.
	class	O	
	title	-	
I18N Attributes			
	xml:lang	-	Not operated.
Events Attributes			%Events.attrib; only operates a, input, textarea, object.
	onclick	O	
	ondbclick	-	
	onmousedown	-	
	onmouseup	-	
	onmouseover	-	
	onmousemove	-	
	onmouseout	-	
	onkeypress	-	
	onkeydown	O	
	onkeyup	O	
Style Attributes			
	style	O	
Core Modules			
Structure Module			
body	%Core.attrib;	O	Composed of %id.attrib + %class.attrib;. Definition of configuring elements are compliant with XHTML.
	%Style.attrib;	O	
head	%I18n.attrib;	-	
	profile	-	
html	xmlns	Δ	Operated with a fixed value of xmlns="http://www.w3.org/1999/xhtml" Operated with a fixed value of xmlns:bml= “http://www.arib.or.jp/bml”
	%I18n.attrib;	-	
	version	-	
title	%I18n.attrib;	-	
Text Module			
blockquote	%Common.attrib;	-	
	%Style.attrib;	-	
	cite	-	
br	%Core.attrib;	O	
	%Style.attrib;	O	
div	%Core.attrib;	O	
	%Style.attrib;	O	
h1-h6	%Common.attrib;	-	
	%Style.attrib;	-	
p	%Core.attrib;	O	

Element	Attribute	Operation	Restriction on operation
	%Style.attrib;	O	
pre	%Core.attrib;	O	Note 4)
	%Style.attrib;	O	
	xml:space	Δ	Fixed as "preserve"
span	%Core.attrib;	O	
	%Style.attrib;	O	
Text Extention Module			
hr	%Common.attrib;	-	
	%Style.attrib;	-	
Hypertext Module			
a	%Common.attrib;	O	
	%Style.attrib;	O	
	(%Events.attrib;)	O	element a operates %Events.attrib;.
	accesskey	O	
	charset	Δ	Fixed as "Shift_JIS".
	href	O	
	hreflang	-	
	rel	-	
	rev	-	
	tabindex	-	
	type	-	
List Module			
dl	%Common.attrib;	-	
	%Style.attrib;	-	
dt	%Common.attrib;	-	
	%Style.attrib;	-	
dd	%Common.attrib;	-	
	%Style.attrib;	-	
ol	%Common.attrib;	-	
	%Style.attrib;	-	
ul	%Common.attrib;	-	
	%Style.attrib;	-	
li	%Common.attrib;	-	
	%Style.attrib;	-	
Forms Module			
form	%Core.attrib;	O	
	%Style.attrib;	O	
	accept	-	
	accept-charset	-	
	action	O	
	method	O	
	enctype	Δ	Fixed as "application/x-www-form-urlencoded".
input	%Common.attrib;	O	
	%Style.attrib;	O	
	(%Events.attrib;);	O	Input element operates %Events.attrib;.
	accept	-	
	accesskey	O	
	alt	-	
	checked	-	
	disabled	-	

Element	Attribute	Operation	Restriction on operation
	maxlength	O	1 to 40. Note 1)
	name	O	
	readonly	O	
	size	-	
	src	-	
	tabindex	-	
	type	O	Use one of "text", "password", "submit".
	value	O	
label	%Common.attrib;	-	
	%Style.attrib;	-	
	for	-	
	accesskey	-	
select	%Common.attrib;	-	
	%Style.attrib;	-	
	disabled	-	
	name	-	
	size	-	
	multiple	-	
	tabindex	-	
option	%Common.attrib;	-	
	%Style.attrib;	-	
	disabled	-	
	label	-	
	selected	-	
	value	-	
textarea Note 3)	%Common.attrib;	O	Note 1) Note4)
	%Style.attrib;	O	
	(%Events.attrib;)	O	Textarea element operates %Events.attrib;.
	accesskey	O	
	cols	-	Input area is specified by CSS(width,height).
	disabled	-	
	name	O	
	readonly	O	
	rows	-	Input area is specified by CSS(width,height).
	tabindex	-	
	xml:space	Δ	Fixed as "preserve"
button	%Common.attrib;	-	
	%Style.attrib;	-	
	accesskey	-	
	disabled	-	
	name	-	
	tabindex	-	
	type	-	
	value	-	
fieldset	%Common.attrib;	-	
	%Style.attrib;	-	
legend	%Common.attrib;	-	
	%Style.attrib;	-	
	accesskey	-	
optgroup	%Common.attrib;	-	
	%Style.attrib;	-	
	disabled	-	
	label	-	

Element	Attribute	Operation	Restriction on operation
Image Module			
img	%Core.attrib;	O	
	%Style.attrib;	O	
	src	O	
	alt	O	The display is receiver dependent. optional to be presented.
	longdesc	-	
	height	-	
	width	-	
Object Module			
object	%Common.attrib;	O	
	%Style.attrib;	O	
	(%Events.attrib;)	O	Object element operates %Events.attrib;.
	archive	-	
	classid	-	
	codebase	-	
	codetype	-	
	data	O	
	declare	-	
	height	-	
	name	-	
	standby	-	
	tabindex	-	
	type	O	
	width	-	
Intrinsic Events Module			
a&	onblur	-	
	onfocus	-	
area&	onblur	-	
	onfocus	-	
frameset&	onload	-	
	onunload	-	
form&	onreset	-	
	onsubmit	O	
body&	onload	O	
	onunload	O	Note 2)
label&	onblur	-	
	onfocus	-	
input&	onfocus	-	
	onblur	-	
	onselect	-	
	onchange	O	
select &	onblur	-	
	onfocus	-	
	onchange	-	
textarea&	onfocus	-	
	onblur	-	
	onselect	-	
	onchange	-	
Metainformation Module			
meta	%I18n.attrib;	-	
	http-equiv	-	
	name	O	
	content	O	

Element	Attribute	Operation	Restriction on operation
	scheme	-	
Scripting Module			
script	id	-	
	charset	Δ	Fixed as "Shift_JIS".
	type	Δ	Fixed as "text/X-arib-ecmascript; charset=" Shift_JIS"".
	src	O	The externally referenced script must be one and be completed in syntax
	defer	-	
	xml:space	Δ	Fixed as "preserve".
Link Module			
link	%Core.attrib;	-	
	%Style.attrib;	-	
	charset	Δ	Fixed as "Shift_JIS".
	href	O	
	hreflang	-	
	type	Δ	Fixed as "text/css".
	rel	Δ	Fixed to "stylesheet".
	rev	-	
	media	Δ	Fixed as "tv".
BML module			
bml:bevent	id	O	Character string, maximum of 128 bytes.
bml:beitem	id	O	Character string, maximum of 128 bytes.
	type	O	One of following: EventMessageFired, ModuleUpdated, ModuleLocked TimerFired, DataEventChanged, MediaStopped. MainAudioStreamChanged is not operated.
	onoccur	O	
	es_ref	O	
	message_group_id	O	0 or 1
	message_id	O	
	message_version	O	
	module_ref	O	
	time_mode	O	One of following: "absolute", "origAbsolute"
	time_value	O	
	object_id	O	Only ID's of the object element that has an attribute type of "audio/X-arib-mpeg2-aac", and referes to the data transmitted in a data carousel
	subscribe	O	
object&	bml:streamstatus	O	
	bml:onfocus	O	
	bml:onblur	O	
	bml:accesskey	O	

Note 1) Characters exceeding maxlength (textarea is always 240 characters in two-byte code character) in the input and textarea are trancated. the excess part is not displayed.
One tab (0x09) or one white space character (0x20) is counted as one character. Line-feed character (0x0D0A) is counted as 1 or 2 characters however, which to use is implementation dependent.

Note 2) The expanded functions for broadcasting that can be used in the onunload event handler are only readPersistentArray(), writePersistentArray(), unlockModuleOnMemoryEx(), and unlockAllModulesOnMemory().To move quickly to the target document, it is preferable that the procedure be limited to the processes ending in a short time, like the setting to Ureg and the simple

condition evaluation. etc.

Note 3) The generation of the DOM tree is the same as the operation of pre element specified in ARIB STD B24, Vol. 2, Appendix 4 5.3.2, following XHTML1.0 3.2 User Agent Conformance (http://www.w3.org/TR/2000/REC-xhtml1-20000126/#uaconf), shall be keep all the control codes except the control code of beginning and trailing (space, linefeed and tab).

Note 4) During the generation of DOM node, the tab (0x09) is kept, but the display to the screen follows CSS and only one tab character is replaced with one character of space character (0x20). In addition, in case of the textarea element, the displayed character string shall be displayed as folded at the right edge of the display area.

7.7.3.1 Restrictions on the order of elements in the head element

In the head element, the title element, meta element, link element, script element, and bml: bevent element shall appear in this order. The title element appears once, meta element, link element, and bml:bevent element appears none or once. The script element without specifying the src attribute appears none or once, and the script element with the specification of the src attribute appears none to twice.

7.7.3.2 Operation of bml:beitem element

When the attribute value is dynamically changed by DOM, first, set the subscribe attribute to false, then change the value, and after that, set the subscribe attribute to true.

7.7.3.3 Expansion of object elements

In this operation, the bml:onfocus/bml:onblur attribute are added to the object element in order that the focus can be obtained directly by the object element.

7.7.4 Entity

See ARIB STD-B24, Vol. 2, Appendix 4, “4.4.3. Entity”.

7.7.5 Operation of the BML element extended module (event)

- The maximum number of ModuleUpdated events that can have the subscribe attribute set to subscribe at the same time is 8.
- The maximum number of TimerFired events that can have the subscribe attribute set to subscribe, at the same time, is 2. It is assumed that it fires immediately when the time value of absolute playback time/ time at reception specified by TimerFired has already passed, at the moment of BML document interpretation.
- ModuleUpdated of any data carousel component can be observed regardless of which component is being presented by the contents .
- Behavier when two or more event handlers are associated with the same event is a matter of product planning. For instance, the following descriptions correspond to this.

  - Multiple bml:beitem elements which have the type attribute of "DateEventChanged" of the type attribute are described.
  - Multiple bml:beitem elements which have the type attribute of "ModuleUpdated" are described, and specify the same module in those module_ref attributes.
- The DataEventChanged event doesn't occur even if the data event is updated in an ES not being presented.
- When the ModuleUpdated event is subscribed for modules transmitted in ES's not being presented, a ModuleUpdated event is generated when the data event is updated in the ES that transmits the module. In this case, as a status value of the ModuleUpdated event, the following new values are operated.

Table 7-6 Value and meaning of the ModuleUpdated event status

Status value	Meaning
4	The data event was updated in the ES from which the module was transmitted. The module is not delivered before a data event update, and the module has been transmitted after a data event is updated.
5	The data event was updated in the ES from which the module was transmitted. The module has been delivered before a data event update, and the module has not been transmitted after a data event update.
6	The data event was updated in the ES from which the module was transmitted. The module has been delivered in both, before a data event update and after a data event update.

7.8 Operation of CSS

Table 7-7 shows the operation of CSS2 properties.

Table 7-7 Operation of CSS properties

Property	Operation
Selector	
*	O
E	O
E F	-
E > F	-
E:first-child	-
E:link	-
E:visited	-
E:focus	-
E:active	-
E:hover	-
E:lang(c)	-
E + F	-
E[foo]	-
E[foo="warning"]	-
E[lang="en"]	-
E:first-line	-
E:first-letter	-
E:before	-
E:after	-
div.warning	-
E.myclass	O
#myid	O
Assigning property values, Cascading, and Inheritance	
@import	-
!important	-
Other setting rules	
@charset	-
@font-face	-
@color-profile	-
Type of medium	
@media	O
Box model	
margin-top	-
margin-right	-
margin-bottom	-
margin-left	-
margin	O
padding-top	O
padding-right	O
padding-bottom	O
padding-left	O
padding	-
border-top-width	-
border-right-width	-
border-bottom-width	-
border-left-width	-
border-width	O
border-top-color	O
border-right-color	O

Property	Operation
border-bottom-color	O
border-left-color	O
border-color	-
border-top-style	-
border-right-style	-
border-bottom-style	-
border-left-style	-
border-style	O
border-top	-
border-right	-
border-bottom	-
border-left	-
border	-
Visual formatting model	
position	O
left	O
top	O
width	O
height	O
z-index	O
line-height	O
vertical-align	-
display	O
bottom	-
right	-
float	-
clear	-
direction	-
unicode-bidi	-
min-width	-
max-width	-
min-height	-
max-height	-
Other visual effect	
visibility	O
overflow	O
clip	-
Generated content, automatic numbering, and lists	
content	-
quotes	-
counter-reset	-
counter-increment	-
marker-offset	-
list-style-type	-
list-style-image	-
list-style-position	-
list-style	-
Paged media	
"@page"	-
size	-

Property	Operation
marks	-
page-break-before	-
page-break-after	-
page-break-inside	-
page	-
orphans	-
widows	-
Colors and backgrounds	
color	O
background	-
background-color	O
background-image	O
background-repeat	O
background-position	-
background-attachment	-
Font	
font-family	O
font-style	-
font-size	O
font-variant	-
font-weight	O
font	-
font-stretch	-
font-size-adjust	-
Text	
text-indent	-
text-align	O
text-decoration	-
text-shadow	-
letter-spacing	-
word-spacing	-
text-transform	-
white-space	O
Pseudo-class, pseudo-element	
:link	-
:visited	-
:active	-
:hover	-
:focus	-
:lang	-
:first-child	-
:first-line	-
:first-letter	-
:before	-
:after	-
Tables	
caption-side	-
border-collapse	-
border-spacing	-
table-layout	-
empty-cells	-
speak-header	-
User interface	
outline-color	-
outline-width	-
outline-style	-

Property	Operation
outline	-
cursor	-
Aural style sheets	
volume	-
speak	-
pause-before	-
pause-after	-
pause	-
cue-before	-
cue-after	-
cue	-
play-during	-
azimuth	-
elevation	-
speech-rate	-
voice-family	-
pitch	-
pitch-range	-
stress	-
richness	-
speak-punctuation	-
peak-numeral	-
Extended property	
clut	-
color-index	-
background-color-index	-
border-color-index	-
border-top-color-index	-
border-right-color-index	-
border-bottom-color-index	-
border-left-color-index	-
outline-color-index	-
pixel size	O
display-aspect-ratio	-
grayscale-color-index	-
nav-index	-
nav-up	-
nav-down	-
nav-left	-
nav-right	-
used-key-list	O
-wap-marquee-style	O
-wap-marquee-loop	O
-wap-marquee-dir	O
-wap-marquee-speed	O
-wap-accesskey	-
-wap-input-format	O
-wap-input-required	-

The ones specified as fixed values are defined as the most important rule (!important) in the default style sheet. The most important rule (!important) always overrides a normal rule. The most important rule (!important) is not operated in the BML document , then all are normal rules, the defined value is obtained as a fixed value.

7.8.1 Element applied each properties

Table 7-8 shows the combination of operated properties and applied BML elements

Table 7-8 Element that apply the CSS properties

	body	div	p	pre	br	span,a	form	input,textarea	object	img
Box model										
margin	-	Δ	Δ	Δ	-	-	Δ	Δ	Δ	Δ
padding-top	-	Δ	O	O	-	-	Δ	O	Δ	Δ
padding-right	-	Δ	O	O	-	-	Δ	O	Δ	Δ
padding-bottom	-	Δ	O	O	-	-	Δ	O	Δ	Δ
padding-left	-	Δ	O	O	-	-	Δ	O	Δ	Δ
border-width	-	O	O	O	-	-	O	O	Δ	Δ
border-top-color	-	O	O	O	-	-	O	O	-	-
border-right-color	-	O	O	O	-	-	O	O	-	-
border-left-color	-	O	O	O	-	-	O	O	-	-
border-bottom-color	-	O	O	O	-	-	O	O	-	-
border-style	-	O	O	O	-	-	O	O	Δ	Δ
Visual model										
position	-	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ
left	-	O	O	O	-	-	O	O	O	O
top	-	O	O	O	-	-	O	O	O	O
width	-	O	O	O	-	-	O	O	O	O
height	-	O	O	O	-	-	O	O	O	O
z-index	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ	Δ
line-height	-	-	O	O	Δ	Δ	-	O	-	-
display	Δ	Δ	O	Δ	Δ	Δ	Δ	Δ	Δ	Δ
Other visual effects										
visibility	Δ	O	O	O	-	Δ	O	O	O	O
overflow	-	Δ	Δ	Δ	-	-	Δ	Δ	Δ	Δ
Background										
color	-	-	O	O	-	O	-	O	-	-
background-color	O	O	O	O	-	O	O	O	Δ	Δ
background-image	O	-	-	-	-	-	-	-	-	-
background-repeat	Δ	-	-	-	-	-	-	-	-	-
Fonts										
font-family	-	-	Δ	Δ	-	Δ	-	Δ	-	-
font-size	-	-	O	O	-	O	-	O	-	-
font-weight	-	-	O	O	-	O	-	O	-	-
Text										

	body	div	p	pre	br	span,a	form	input,textarea	object	img
text-align	-	-	O	-	-	-	-	O	-	-
white-space	-	-	Δ	Δ	-	-	-	Δ	-	-
Extended properties										
pixel size	Δ	-	-	-	-	-	-	-	-	-
used-key-list	O	-	-	-	-	-	-	-	-	-
-wap-marquee-style	-	-	O	-	-	-	-	-	-	-
-wap-marquee-loop	-	-	O	-	-	-	-	-	-	-
-wap-marquee-dir	-	-	Δ	-	-	-	-	-	-	-
-wap-marquee-speed	-	-	O	-	-	-	-	-	-	-
-wap-input-format	-	-	-	-	-	-	-	O	-	-

7.8.2 Selector operation

The selector is used only in the contents of the default style sheet and in the CSS data transmitted by an independent resource, and only the ones shown below are operated.

- All elements Example: * { color: white }
- Type selector Example: div { color: white }
- Class selector Example: div.class { color: white }
  or .class { color: white }
- ID selector Example: #chap1 { color: white }

However, the batch assignment by combining 2 or more selectors cannot be done, except for usage other than the ones mentioned above. In addition, "," cannot be used to specify two or more selectors. Moreover, those selectors are applied only to the elements indicated as operated in Table 7-8.

7.8.3 Media type

The media type operated is only tv. Specifying the media type is made by @media rule in style sheet. Moreover, the receiver unit should ignore any specification of the properties within the @media rule that has any other type than tv.

7.8.4 Box model

The operation of the properties concerning the box model are shown below.

- margin

  This property is operated by a fixed value as provided in the following.

  Fixed value: 0

- padding-top, padding-right, padding-bottom, padding-left
  Values: <length> (integer value in the unit of px, or 0.)
  default : 0
  Inheritance : no
  For div element, object element, form element, and img element, operated as a fixed value 0.

- border-width
  This property is operated as below.
  Values: <length> (integer value in the unit of px, or 0.)
  default : 0
  Inheritance : no.
  All 4 sides are affected. For the object element and img element, operated as a fixed value 0.

- border-top-color, border-right-color, border-left-color, border-bottom-color
  These property are operated as below.
  Values: <color>|transparent | inherit
  default : transparent
  Inheritance : no
  The borders are rendered in the order of left,right,top,bottom. Each vertex is rendered again in rendering order of the borders. When these properties are not specified, these values are default (“transparent”), not the value of “color” property of the element.they are interpreted that it was not a color characteristic value of the foresaid element but the initial value (transparent). These properties use the same color unit as color property.
- border-style
  This property is operated as below.
  Values: dotted | dashed | solid | none
  default : none
  Inheritance : no
  All 4 sides are affected. The display form of dotted, dashed, and solid conforms to the specification of the geometric in ARIB STD-B24, Vol. 1, Part 2, Chapter 8, 8.1. If not specified by the geometric, the form depends on the receiver unit.
  Note, for img and object elements, the value is fixed as none in the operation.

7.8.5 Visual formatting model

Operation of the visual formatting model is shown as follows.

- display

  This property has a different fixed value for each element as defined in the default style sheet.

  div, body, input, textarea, img, object, form, and pre are fixed to block.

  br, span, and a are fixed to inline.

  Exceptionally, for p element, the value can be block | - wap-marquee. The default is block. And the value is not inherited.

  Note 1)

  meta, title, script, head, html, link, bml:bevent, and bml:beitem are fixed to none.

  Note 2) The operation of p element for which -wap-marquee is specified is as follows.

  - It only contains PCDATA, that should have no linefeed code (i.e. one line string).
  - The maximum length of the above-mentioned PCDATA is 128 bytes. The excessive characters are trancated to display.
  - The string is clipped according to the value of overflow (“hidden”), when the string is larger than the area to display specified with width and height Exceptionally, in the p element with –wap-marquee specified, clipping is not done, so the whole of the string is displayed as a line (not folded).

- position

  This property has a different fixed value for each element as defined in the default style sheet. p, div, textarea, input, img, object, pre, and form are fixed to absolute.

  br, span, and a are fixed to static.

  It is not applied to elements other than the above.

- left, top, width, height

  These properties are operated as below.

  Values: <length> (integer value in the unit of px, or 0.)

  default : 0

  Inheritance : no

- z-index

  This property is operated as a fixed value defained as follows.

  Fixed value : auto

- line-height

  This property is operated as below..

  Values: normal | <length> (integer value in the unit of px, or 0.)| inherit

  default : normal

  Inheritance : no

  When normal is specified, the value is 1.0 times the font-size. But, for br, span, and a elements, the value is fixed as inherit in the operation. And the property can have the value “inherit” only for these elements.

7.8.6 Other visual effects

The operation of the characteristics concerning other visual effects is shown as follows.

- visibility

  This property is fixed as below in the operation.

  Values: visible | hidden | inherit

  default : inherit

  Inheritance : no

  When the visibility is set to “hidden” in the parent element, and “visible” in the child element, only the parent is hidden. the value is fixed as “inherit” for span and a elements in the operation. Also, the value is fixed as “visible” for body element.

- overflow

  This property is fixed as below in the operation.

  Fixed value : hidden

7.8.7 Colors and backgrounds

Color and background are defined as below in the operation.

- color

  This property is specified as below in the operation.

  Values: <color> | inherit

  Default : black

  Inheritance : yes

  The color unit follows ARIB STD-B24 Vol. 2, Appendix 1, 4.4.1. There, only the six-digit notation of #rgb can be specified, so the notation of rgb(0.0%-100.0%,…) is prohibited. To specify by keywords, the aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, white, and yellow at the very least can be available

- background-color

  This property is specified as below in the operation.

  Values: <color> | transparent | inherit

  default : transparent

  Inheritance : no

  However, the values of object and img elements are fixed as transparent. Refer to “color” property for the specification of the color unit.

- background-image

  This property is specified as below in the operation.

  Values: <uri> | none

  default : none

  Inheritance : no

  The encoding of the image specified by the <uri> is jpeg, and the upper left corner of the body element is the rendering origin.

- background-repeat

  This property is fixed as below in the operation.

  Fixed value : repeat

7.8.8 Fonts

Table 7-9 defines the specification of the fonts in BML documents. One-byte character is displayed in the 1/2 width of the font size. Assignment of font type in Table 7-9 for the fonts of the receiver unit depend on the receiver unit. When a font of Gothic system is installed in the receiver unit, it is preferable to use aforesaid font.

Table 7-9 Used font

Font type	Font size
Maru gothic style	small, medium, large

The operation of the font properties is specified as below.

- font-family

  The value of the property is fixed as below in the operation.

  Fixed value : "Maru gothic"

  However, actual results of rendering depends on the receiver.

- font-size

  This property is specified as below.

  Values: small | medium | large | inherit

  default : medium

  Inheritance : yes

  It should be noted when authoring, that there are terminals with same size of “small” and “medium”.

- font-weight

  This property is specified as below.

  Values: normal | bold | inherit

  default : normal

  Inheritance : yes

  The rendering result of “bold” depends on the receiver unit.

7.8.9 Text

The properties of “Text” is specified in the operation as below.

- text-align

  This property is specified as below in the operation.

  Values: left | right | center

  default : left

  Inheritance :yes

- white-space

  This property is fixed in the operation as below.

  except for pre and textarea elements, that value is fixed as pre.

  Fixed value : normal (pre, for pre and textarea element)

7.8.10 Default style sheets

The default style sheet is defined in appendix 4. This style sheet is assumed to be loaded virtually in every receiver unit and it does not need be transmitted.

7.8.11 length specification

The length must be specified with an integer value in the unit of pixel, e.g. "100px", etc. or “0” for 0px. For "0", DOM always returns “0px”.

7.9 extended property specification

The extended properties are specified as below in the operation.

- used-key-list

  This property is specified as below.operated as provisioned in the following. Refer to 7.3.1, for the valid key-group's in the operation.

  default: <key-group>+ | none

  default : basic

  Inheritance : no

- resolution

  This property is fixed as below in the operation..

  Fixed value : 240x480

  Inheritance : no

- -wap-marquee-style

  This property is specified as below in the operation..

  Values: scroll | slide

  Applicable element: p element

  default : scroll

  Inheritance : no

  The animation specified by this property is repeated in "-wap-marquee-loop" times.

- -wap-marquee-loop

  This property is specified as below in the operation..

  Values: <integer>|infinite

  Applicable element: p element

  default : 1

  Inheritance : no

  The maximum value that can be specified is 16 for the finite times to loop.

  For the infinite times to loop, specify "infinite". The receiver can be implemented with the "infinite" as an finite times to loop. Moreover, in "0", the display stays as if it has looped in the specified times..

- -wap-marquee-dir

  This property is fixed as below in the operation..

  Fixed value : rtl

  This property specifies the direction of the scrolling. The only rtl direction can be specified.

  rtl: Scrolling from the right side to the left side

- -wap-marquee-speed

  This property is specified as below in the operation.
  Values: slow | normal | fast
  Applicable element: p element
  default : normal
  Inheritance : no
  This property specifies the scrolling speed. The actual speed for each value varies on the respective receiver unit.

- -wap-input-format

  This property is specified as below in the operation.
  Values: <format>
  Applicable element: input element and textarea element
  default : "*M"
  Inheritance : no

  - Each value of <format> is as below.

    A : Any uppercase alphabetical characters (symbols and punctuations included, numbers excluded)

    a : Any lowercase alphabetical characters (symboles and punctuation included, numbers excluded)

    N : Any numeric characters, only

    n : Numeric characters (symbols and punctuation included)

    X : Any uppercase alphabetical or numeric characters(symbols and punctuation included)

    x : Lowercase alphabetical or numeric characters (symbols and punctuation included)

    M : Any characters (set to uppercase characters as the initial value)

    m : Any characters (set to lowercase characters as the initial value)

  - With the above values, a number of the input character can be limited like "MMM" and "3M", etc. When limits are not set, "*" should be prepended to the value described above, e.g. "* M". However, it cannot be specified by a combination.
  - Refer to the ARIB STD-B24, Vol. 1, Part 2, Chapter 7, 7.3 "Shift-JIS character codes" for the characters that can be specified. The codes allocated from Row 90 to Row 94 of ARIB-STD-B5 Kanji set, tab (09) and delete character are excluded.
  - Criteria of classifying a character into a numeric, alphabetical, or symbolic character is implementation dependent.
  - Receiver shall process appropriately on input of linefeed character (0x0D0A), because the “value” attribute of input elements shall not include linefeed characters although the linefeed character is not within the scope of control for the property.

7.10 Operations for procedural descriptions

7.10.1 Operations DOM

7.10.1.1 DOM Core Fundamental interfaces

Table 7-10 specifies the DOM Core Fundamental interfaces in the operation. In this operation, the mechanism that dynamically changes the tree structure of the document shall not be operated.

Table 7-10 DOM Core Fundamental interfaces

Interface	Operation
Fundamental interfaces	
DOMException	-
DOMImplementation	O
DocumentFragment	-
Document	O
Node	O
NodeList	-
NamedNodeMap	-
CharacterData	O
Attr	-
Element	O
Text	O
Comment	-
Extended interfaces	
CDATASection	O
DocumentType	-
Notation	-
Entity	-
EntityReference	-
ProcessingInstruction	-

The attributes of DOM Core Fundamental interfaces is specified in Table 7-11 in the operation.

Table 7-11 attributes and methods (DOM Core interfaces)

Interface	Attribute, method	Operation	Restrictions
DOMImplementation			
	hasFeature()	O	
Document			
	doctype	-	
	implementation	O	R
	documentElement	O	R
	createElement()	-	
	createDocumentFragment()	-	
	createTextNode()	-	
	createComment()	-	
	createCDATASection()	-	
	createProcessingInstruction()	-	

Interface	Attribute, method	Operation	Restrictions
	createAttribute()	-	
	createEntityReference()	-	
	getElementsByTagName()	-	
Node			
	nodeName	-	
	nodeValue	-	
	nodeType	-	
	parentNode	O	R
	childNodes	-	
	firstChild	O	R
	lastChild	O	R
	previousSibling	O	R
	nextSibling	O	R
	attrbutes	-	
	ownerDocument	-	
	insertBefore()	-	
	replaceChild()	-	
	removeChild()	-	
	appendChild()	-	
	hasChildNodes()	-	
	cloneNode()	-	
CharacterData			
	data	O	RW(Note)
	length	O	R(Note)
	substringData()	-	
	appendData()	-	
	insertData()	-	
	deleteData()	-	
	replaceData()	-	
Element			
	tagName	O	R
	getAttribute()	-	
	setAttribute()	-	
	removeAttribute()	-	
	getAttributeNode()	-	
	setAttributeNode()	-	
	removeAttributeNode()	-	
	getElementsByTagName()	-	
	normalize()	-	
Text			
	splitText()	-	
CDATASection			

(Note) The access to the child nodes of “script” or “textarea” elements should not be operated. only the child node of p, span, a, and pre can be written in the operation. One tab (0x09) or one white space (0x20) is counted as one character. Linefeed character (0x0D0A) is counted as 1 or 2 characters, although which to take depends on the receiver.

7.10.1.2 DOM HTML interfaces

- Operation of the interfaces applied to events

The interfaces applicable to events, belonged to the DOM HTML interfaces, is specified in the table below. Moreover, the values obtained by reading the “type” or “target” attribute in the BMLEvent interface is specifed in the latter table below according to the situation in which the access to the BMLEvent interface is done.

Table 7-12 attributes and methods (BML interfaces)

Interface name	Attribute or method name	Operation	Restriction
BMLEvent	type	O	R
	target	O	R
BMLIntrinsicEvent	keyCode	O	R
BMLBeventEvent:	status	O	R
	privateData	O	R
	esRef	O(Note 1)	R
	messageId	O	R
	messageVersion	O	R
	messageGroupId	O	R
	moduleRef	O(Note 2)	R
	object	O	R

(Note 1) The format of the value obtained by reading this attribute is /< component_tag > as the notation specified in ARIB STD-B24 Vol. 2, 9.2.

(Note 2) The format of the value obtained by reading this attribute is /<component_tag>/<module_id as the notation specified in ARIB STD-B24 Vol. 2, 9.2.

The response of the “type” attribute in the BMLEvent to the respective event is specifed in Table 7-13.

Table 7-13 Correspondence of the type attribute of the event and BMLEvent

Event	Value of type
Key specified in used-key-list is pushed down.	"keydown" (Note 3)
Key specified in used-key-list is released.	"keyup" (Note 3)
An Element is selected by pushing the “Enter” key or an access key.	"click" (Note 3)
in the focus	"focus"
out of the focus	"blur"
a document is loaded.	"load"
The advanced notice of unloading the document	"unload"
A change of the “value” attribute of an element is detected when the focus of the element blurs.	"change"
”submit” button of the form element is pushed or the “submit” method of the form element is called.	"submit"
Event message is received.	"EventMessageFired"

Event	Value of type
Module update is detected.	"ModuleUpdated"
Module is being locked.	"ModuleLocked"
Timer configured in beitem is fired.	"TimerFired"
presentation by monomedia decoder is terminated. (Note 2)	"MediaStopped"
Detected update of data_event_id.	"DataEventChanged"
The execution of a global code has been started or the function specified by executing setInterval() was invoked.	Undefined (Note 1)

Note 1) In this case, the target attribute is null.

Note 2) The monomedia that caused this event are audio/X-arib-mpeg2-aac.

Note 3) Whether the occurrence of the key event by the “Enter” key in the input or textarea element is receiver unit dependent. (Excludes input elements with a type attribute of "submit”.)

Table 7-14 execution of event handlers, and value of type attributes and target attributes

When the values are read			Value of type attribute	Value of target attribute	Occurs on receiving the broadcast?	Occurs on link status
onkeydown event handler being executed			"keydown"	Element specifying the handler	O	O
onkeyup event handler being executed			"keyup"	Element specifying the handler	O	O
onclick event handler being executed			"click"	Element specifying the handler	O	O
onsubmit event handler being executed			“submit”	Element specifying the handler	O	O
onload event handler being executed			“load”	Element specifying the handler	O	O
onunload event handler being executed			“unload”	Element specifying the handler	O	O
onchange event handler being executed			“change”	Element specifying the handler	O	O
onfocus event handler being executed			“focus”	Element specifying the handler	O	O
onblur event handler being executed			“blur”	Element specifying the handler	O	O
Onoccur event handler being executed	Type of the event	EventMessageFired	"EventMessageFired"	Element specifying the handler	O	O
		ModuleUpdated	"ModuleUpdated"		O	O
		ModuleLocked	"ModuleLocked"		O	X
		TimerFired	"TimerFired"		O	O
		DataEventChanged	"DataEventChanged"		O	X
		MediaStopped	"MediaStopped"		O	O
global code being executed			undefined	null	O	O
The procedural description specified by the call of setInterval() of the extended function for broadcasting being executed			undefined	null	O	O

- Operation of interfaces applied to the document node

  The interfaces applicable to the document node is specified in the table below, belonging to the DOM HTML interfaces or DOM BML interfaces.

Table 7-15 the interfaces applied to the document node

Interface name	Attribute or method name	Operation	Restriction
HTMLDocument	title	-	
	referrer	-	
	domain	-	
	URL	-	
	body	-	
	images	-	
	applets	-	
	links	-	
	forms	-	
	anchors	-	
	cookie	-	
	open()	-	
	close()	-	
	write()	-	
	writeln()	-	
	getElementById()	O	
	getElementsByName()	-	
BMLDocument	currentFocus	O	R
	currentEvent	O	R (Note)

(Note) The behavior when the value obtained by reading this attribute is assigned to a global variable is a receiver unit dependent.

- Operation of the interfaces applied to the element node

The operation of the interfaces applicable to the element node is specified in the table, belonging to the DOM HTML interfaces.

Table 7-16 the interfaces applied to the element node

Element by the node	Interface applied	Attribute or method	Operation	Restriction
html	HTMLElement	id	-	
		className	-	
		title	-	
		lang	-	
		dir	-	
	HTMLHtmlElement	version	-	
bml:bevent	HTMLElement	id	O	R
		className	-	
		title	-	
		lang	-	
		dir	-	
	BMLBeventElement			

Element by the node	Interface applied	Attribute or method	Operation	Restriction
bml:beitem	HTMLElement	id	O	R
		className	-	
		title	-	
		lang	-	
		dir	-	
	BMLBeitemElement	type	O	R
		esRef	O	RW
		messageGroupId	O	R
		messageId	O	RW
		messageVersion	O	RW
		moduleRef	O	RW
		timeMode	O	R
		timeValue	O	RW
		objectId	O	R
		subscribe	O	RW
body	HTMLElement	id	O	R
		className	O	R
		title	-	
		lang	-	
		dir	-	
	HTMLBodyElement			
	BMLBodyElement	invisible	-	
		style	-	
		normalStyle	O	RW (Note 2)
		focusStyle	-	
		activeStyle	-	
head	HTMLElement	id	-	
		className	-	
		title	-	
		lang	-	
		dir	-	
	HTMLHeadElement	profile	-	
title	HTMLElement	id	-	
		className	-	
		title	-	
		lang	-	
		dir	-	
	HTMLTitleElement	text	O	R

Element by the node	Interface applied	Attribute or method	Operation	Restriction
div	HTMLElement	id	O	R
		className	O	R
		title	-	
		lang	-	
		dir	-	
	HTMLDivElement			
	BMLDivElement	style	-	
		normalStyle	O	RW (Note 2)
		focusStyle	-	
		activeStyle	-	
		accessKey	-	
		focus()	-	
		blur()	-	
p	HTMLElement	id	O	R
		className	O	R
		title	-	
		lang	-	
		dir	-	
	HTMLParagraphElement			
	BMLParagraphElement	style	-	
		normalStyle	O	RW (Note 2)
		focusStyle	-	
		activeStyle	-	
		accessKey	-	
		focus()	-	
		blur()	-	
pre	HTMLElement	id	O	R
		className	O	R
		title	-	
		lang	-	
		dir	-	
	HTMLPreElement			
	BMLPreElement	style	-	
		normalStyle	O	RW (Note 2)
		focusStyle	-	
		activeStyle	-	

Element by the node	Interface applied	Attribute or method	Operation	Restriction
br	HTMLElement	id	O	R
		className	O	R
		title	-	
		lang	-	
		dir	-	
	HTMLBRElement			
	BMLBRElement	style	-	
		normalStyle	O	RW (Note 2)
		focusStyle	-	
		activeStyle	-	
span	HTMLElement	id	O	R
		className	O	R
		title	-	
		lang	-	
		dir	-	
	BMLSpanElement	style	-	
		normalStyle	O	RW (Note 2)
		focusStyle	-	
		activeStyle	-	
		accessKey	-	
		focus()	-	
		blur()	-	

Element by the node	Interface applied	Attribute or method	Operation	Restriction
a	HTMLElement	id	O	R
		className	O	R
		title	-	
		lang	-	
		dir	-	
	HTMLAnchorElement	accessKey	O	R
		charset	-	
		coords	-	
		href	O	RW
		hreflang	-	
		name	-	
		rel	-	
		rev	-	
		shape	-	
		tabIndex	-	
		target	-	
		type	-	
		blur()	-	
		focus()	-	
	BMLAnchorElement	style	-	
		normalStyle	O	RW (Note 2)
		focusStyle	-	
		activeStyle	-	
		effect	-	
form	HTMLElement	id	O	R
		className	O	R
		title	-	
		lang	-	
		dir	-	

Element by the node	Interface applied	Attribute or method	Operation	Restriction
	HTMLFormElement	elements	-	
		length	-	
		name	-	
		acceptCharset	-	
		action	O	RW
		enctype	-	
		method	O	R
		submit()	O	
		reset()	-	
	BMLFormElement	style	-	
		normalStyle	O	RW (Note 2)
		focusStyle	-	
		activeStyle	-	
		accept	-	
input	HTMLElement	id	O	R
		className	O	R
		title	-	
		lang	-	
		dir	-	
	HTMLInputElement	defaultValue	O	R
		defaultChecked	-	
		form	O	R
		accept	-	
		accessKey	O	R
		alt	-	
		checked	-	
		disabled	-	
		maxLength	O	R
		name	O	R
		readOnly	O	RW
		size	-	
		src	-	
		tabIndex	-	
		type	O	R
		useMap	-	
		value	O	RW
		blur()	-	
		focus()	-	
		select()	-	
		click()	-	
	BMLInputElement	style	-	
		normalStyle	O	RW
		focusStyle	-	
		activeStyle	-	

Element by the node	Interface applied	Attribute or method	Operation	Restriction
textarea	HTMLElement	id	O	R
		className	O	R
		title	-	
		lang	-	
		dir	-	
	HTMLTextAreaElement	defaultValue	O	R
		form	O	R
		accessKey	O	R
		cols	-	
		disabled	-	
		name	O	R
		readOnly	O	RW
		rows	-	
		tabIndex	-	
		value	O	RW
		blur()	-	
		focus()	-	
		select()	-	
	BMLTextAreaElement	style	-	
		normalStyle	O	RW
		focusStyle	-	
		activeStyle	-	
img	HTMLElement	id	O	R
		className	O	R
		title	-	
		lang	-	
		dir	-	
	HTMLImageElement	lowSrc	-	
		name	-	
		alt	O	R
		height	-	
		isMap	-	
		longDesc	-	
		src	O	RW (Note)
		useMap	-	
		width	-	
	BMLImageElement	style	-	
		normalStyle	O	RW (Note 2)
		focusStyle	-	
		activeStyle	-	
object	HTMLElement	id	O	R
		className	O	R
		title	-	
		lang	-	
		dir	-	

Element by the node	Interface applied	Attribute or method	Operation	Restriction
	HTMLObjectElement	form	-	
		code	-	
		archive	-	
		codeBase	-	
		codeType	-	
		data	O	RW(Note)
		declare	-	
		height	-	
		name	-	
		standby	-	
		tabIndex	-	
		type	O	R
		useMap	-	
		width	-	
	BMLObjectElement	style	-	
		normalStyle	O	RW (Note 2)
		focusStyle	-	
		activeStyle	-	
		classId	-	
		accessKey	O	R
		remain	-	
		streamPosition	-	
		streamStatus	O	RW
		streamLooping	-	
		streamSpeedNumerator	-	
		streamSpeedDenominator	-	
		streamLevel	-	
		setSpeed()	-	
		movePosition()	-	
		hasAssociatedIndex()	-	
		assignToLocalEvnet()	-	
		assignToNodePlayMode()	-	
		getMainAudioStream()	-	
		setMainAudioStream()	-	
		focus()	-	
		blur()	-	

Element by the node	Interface applied	Attribute or method	Operation	Restriction
meta	HTMLElement	id	-	
		className	-	
		title	-	
		lang	-	
		dir	-	
	HTMLMetaElement	content	O	R
		httpEquiv	-	
		name	O	R
		scheme	-	
script	HTMLElement	id	-	
		className	-	
		title	-	
		lang	-	
		dir	-	
	HTMLScriptElement	text	-	
		htmlFor	-	
		event	-	
		charset	-	
		defer	-	
		src	-	
		type	-	
link	HTMLElement	id	-	
		className	-	
		title	-	
		lang	-	
		dir	-	
	HTMLLinkElement	charset	-	
		href	-	
		hreflang	-	
		type	-	
		rel	-	
		rev	-	
		target	-	
		media	-	

(Note) For the resulting display of the img/object element in which the updated area displayed by the new resource doesn't match with the origianl area displayed by the original resource when the resource to be presented is updated, see section 8.1.8.5. In the img element, the media type of the resources to be presented must be same as before when update.

(Note 2) These attributes are accessed through the BMLCSS2Properties interfaces. When these attributes are read directly, the object is returned. Moreover, it is prohibited to write directly.

7.10.1.3 BMLCSS2Properties interface

The operation of BMLCSS2Properties interface used in Table 7-17 is shown.

Table 7-17 Operation of BMLCSS2Properties interface

Attribute	Operation	Remarks
Box model		
marginTop	-	
marginRight	-	
marginBottom	-	
marginLeft	-	
margin	-	
paddingTop	O	R
paddingRight	O	R
paddingBottom	O	R
paddingLeft	O	R
padding	-	
borderTopWidth	-	
borderRightWidth	-	
borderBottomWidth	-	
borderLeftWidth	-	
borderWidth	O	R
borderTopColor	O	RW
borderRightColor	O	RW
borderBottomColor	O	RW
borderLeftColor	O	RW
borderColor	-	
borderTopStyle	-	
borderRightStyle	-	
borderBottomStyle	-	
borderLeftStyle	-	
borderStyle	O	R
borderTop	-	
borderRight	-	
borderBottom	-	
borderLeft	-	
border	-	
Reception disturbance model		
position	-	
left	O	R
top	O	R
width	O	R
height	O	R
zIndex	-	
lineHeight	-	
verticalAlign	-	
display	-	
bottom	-	
right	-	
cssFloat	-	
clear	-	
direction	-	
unicodeBidi	-	
maxHeight	-	
minHeight	-	
maxWidth	-	
minWidth	-	
Other visual effects		
visibility	O	RW
overflow	-	
clip	-	
Generation contents/ automatic number appearance/list		
content	-	
quotes	-	
counterReset	-	
counterIncrement	-	
markerOffset	-	
listStyleType	-	
listStyleImage	-	
listStylePosition	-	
listStyle	-	
Paged media		
size	-	
marks	-	
pageBreakBefore	-	
pageBreakAfter	-	
pageBreakInside	-	
page	-	
orphans	-	
widows	-	
Background		
background	-	
backgroundColor	O	RW
backgroundImage	-	
backgroundRepeat	-	
backgroundPosition	-	
backgroundAttachment	-	
Fonts		
color	O	RW
fontFamily	-	
fontStyle	-	
fontSize	-	
fontVariant	-	
fontWeight	O	RW
font	-	
fontStretch	-	
fontSizeAdjust	-	
Text		
textIndent	-	
textAlign	O	R
textDecoration	-	
textShadow	-	

Attribute	Operation	Remarks
letterSpacing	-	
wordSpacing	-	
textTransform	-	
whiteSpace	-	
Table related		
captionSide	-	
borderCollapse	-	
borderSpacing	-	
tableLayout	-	
emptyCells	-	
speakHeader	-	
User interface		
outlineColor	-	
outlineStyle	-	
outlineWidth	-	
outline	-	
cursor	-	
Voice style sheet		
volume	-	
speak	-	
pauseBefore	-	
pauseAfter	-	
pause	-	
cueBefore	-	
cueAfter	-	
cue	-	
playDuring	-	
azimuth	-	
elevation	-	
speechRate	-	
voiceFamily	-	
pitch	-	
pitchRange	-	
stress	-	
richness	-	
speakPunctuation	-	
speakNumeral	-	
BML extention		
borderColorIndex	-	
borderTopColorIndex	-	
borderRightColorIndex	-	
borderrLeftColorIndex	-	
borderBottomColorIndex	-	
backgroundColorIndex	-	
colorIndex	-	
grayscaleColorIndex	-	
outlineColorIndex	-	
clut	-	
pixel size	-	
displayAspectRatio	-	
navIndex	-	
navUp	-	
navDown	-	
navLeft	-	
navRight	-	
usedKeyList	O	RW
WapMarqueeStyle	O	R
WapMarqueeLoop	O	R
WapMarqueeDir	-	
WapMarqueeSpeed	O	R
WapAccesskey	-	
WapInputFormat	O	R(Note 1)
WapInputRequired	-	

(Note 1) When this characteristic is read, returns the character string as defined.

The value of the DOM attribute operated as mentioned above follows the operation of the CSS2 properties.

7.10.2 Operation area of built-in objects

Operation of ECMAScript built-in objects is shown in Table 7-18.

Table 7-18 Operation of ECMAScript embedded objects

Embedded object	Method and property	Operation	Remarks
(global)			
	NaN	O	
	Infinity	-	Note 1)
	eval(x)	-	
	parseInt(string, radix)	O	Note 7)
	parseFloat(string)	-	Note 1)
	escape(string)	-	Note 2)
	unescape(string)	-	Note 2)
	isNaN(number)	O	
	isFinite(number)	-	Note 1)
Object		All O	
	prototype	O	
	Object([value])	O	
	new Object([value])	O	
Object.prototype		All O	
	constructor	O	
	toString()	O	
	valueOf()	O	
Function			
	prototype	O	
	length	O	
	Function(p1, p2, . . . , pn, body)	-	
	new Function(p1, p2, . . . , pn, body)	-	
Function.prototype		All O	
	constructor	O	
	toString()	O	Note 3)
Array		All O	
	prototype	O	
	length	O	
	Array(item0, item1, . . .)	O	
	new Array(item0, item1, . . .)	O	
	new Array([len])	O	
Array.prototype		All O	
	constructor	O	
	toString()	O	
	join([separator])	O	
	reverse()	O	
	sort([comparefn])	O	
String		All O	
	prototype	O	

Embedded object	Method and property	Operation	Remarks
	length	O	
	String([value])	O	
	new String([value])	O	
	String.fromCharCode(char0[, char1, . . .])	O	
String.prototype		All O	
	constructor	O	
	toString()	O	
	valueOf()	O	
	charAt(pos)	O	
	charCodeAt(pos)	O	
	indexOf(searchString, position)	O	
	lastIndexOf(searchString, position)	O	
	split(separator)	O	
	substring(start [,end])	O	
	toLowerCase()	O	
	toUpperCase()	O	
Boolean		All O	
	prototype	O	
	Boolean([value])	O	
	new Boolean([value])	O	
Boolean.prototype		All O	
	constructor	O	
	toString()	O	
	valueOf()	O	
Number			
	prototype	O	
	MAX_VALUE	O	
	MIN_VALUE	O	
	NaN	O	
	NEGATIVE_INFINITY	-	Note 1)
	POSITIVE_INFINITY	-	Note 1)
	Number([value])	O	
	new Number([value])	O	
Number.prototype		All O	
	constructor	O	
	toString([radix])	O	
	valueOf()	O	
Math		All -	
	E	-	
	LN10	-	
	LN2	-	
	LOG2E	-	
	LOG10E	-	
	PI	-	
	SQRT1_2	-	
	SQRT2	-	
	abs(x)	-	
	acos(x)	-	
	asin(x)	-	
	atan(x)	-	
	atan2(y, x)	-	
	cos(x)	-	
	exp(x)	-	

Embedded object	Method and property	Operation	Remarks
	floor(x)	-	
	log(x)	-	
	max(x, y)	-	
	min(x, y)	-	
	pow(x, y)	-	
	random()	-	
	round(x)	-	
	sin(x)	-	
	sqrt(x)	-	
	tan(x)	-	
Date			
	prototype	O	
	Date([year , month [, date [, hours [, minutes [, seconds [, ms]]]]]])	O	
	new Date([year , month [, date [, hours [, minutes [, seconds [, ms]]]]]])	O	
	Date(value)	-	Note 4)
	new Date(value)	-	Note 4)
	Date.parse(string)	-	Note 4)
	Date.UTC([year [, month [, date [, hours [, minutes [, seconds [, ms]]]]]]])	-	Note 4)
Date.prototype			
	contructor	O	
	toString()	O	Note 3)
	valueOf()	-	Note 4)
	getTime()	-	Note 4)
	getYear()	-	Note 5)
	getFullYear()	O	
	getUTCFullYear()	O	
	getMonth()	O	
	getUTCMonth()	O	
	getDate()	O	
	getUTCDate()	O	
	getDay()	O	
	getUTCDay()	O	
	getHours()	O	
	getUTCHours()	O	
	getMinutes()	O	
	getUTCMinutes()	O	
	getSeconds()	O	
	getUTCSeconds()	O	
	getMilliseconds()	O	
	getUTCMilliseconds()	O	
	getTimezoneOffset()	O	
	setTime(time)	-	Note 4)
	setMilliseconds(ms)	O	Note 6)
	setUTCMilliseconds(ms)	O	Note 6)
	setSeconds(sec [, ms])	O	Note 6)
	setUTCSeconds(sec [, ms])	O	Note 6)
	setMinutes(min [, sec [, ms]])	O	Note 6)
	setUTCMinutes(min [, sec [, ms]])	O	Note 6)
	setHours(hour [, min [, sec [, ms]]])	O	Note 6)
	setUTCHours(hour [, min [, sec [, ms]]])	O	Note 6)
	setDate(date)	O	Note 6)

Embedded object	Method and property	Operation	Remarks
	setMonth(mon [, date])	O	Note 6)
	setUTCMonth(mon [, date])	O	Note 6)
	setFullYear(year [, mon [, date]])	O	Note 6)
	setUTCFullYear(year [, mon [, date]])	O	Note 6)
	setYear(year)	-	Note 5)
	toLocaleString()	O	Note 3)
	toUTCString()	O	Note 3)
	toGMTString()	-	Note 5)

Note 1) Prohibit to use for Float related objects.
Note 2) Prohibit to use for Unicode related objects.
Note 3) Operation is specified in ARIB STD-B24, Vol. 2, Appendix 2.
Note 4) Prohibit to use for Number related objects.
Note 5) Prohibit to use, as this is for the interchangeability with an old source code.
Note 6) Operated under restricted specifications when it is Number related objects.
Note 7) radix of parseInt() is 8, 10, 16 (same as when 0 is specified as 10).

7.10.3 Operation area of extension objects for broadcasting

Table 7-19 shows the operation of extension objects for broadcasting ECMAScript. Operation of the defined extension object for broadcasting not described in Table 7-19 is assumed to be "-".

Table 7-19 Operation of extension objects for broadcasting

Object	Method and property	Operation	Remarks
CSVTable			
	prototype	-	
	new CSVTable()	-	
CSVTable.prototype			
	constructor	-	
	close()	-	
	toString()	-	
	toNumber()	-	
	toArray()	-	
	search()	-	
BinaryTable			
	prototype	O	
	new Binarytable()	O	
BinaryTable.prototype			
	constructor	O	
	close()	O	
	toString()	O	
	toNumber()	O	
	toArray()	O	
	search()	O	

7.10.4 Operation area of Navigator pseudo-objects

Navigator pseudo-objects are not operated.

7.10.5 Operation area of browser pseudo-objects

Operated as shown in Table 7-20.

The meaning of the "Operation" column is as follows.

"O" A basic function in these provisions.

"O (*1)" An optional feature in these provisions. Therefore, when these functions are used by contents, the aforementioned function is called only when processing is possible by inspecting the processing function of the aforementioned function in the receiver unit by the getBrowserSupport() function.

"O (*2)" Installation of these functions is not necessary only when receiver units that can not use the telecommunication function from a BML browser and when the function is called, returns a failure for the return value, although it is a basic function in these provisions, as a rule.

"O (*3)" It is necessary in receiver units with a function to playback the partial TS. When these functions are used by contents, the aforementioned function is called only when processing is possible by inspecting the processing function of aforementioned function in the receiver unit by the getBrowserSupport() function.

"-" Not a basic function nor an optional function in these provisions. An error will occur in the receiver unit when the aforementioned function is called.

Table 7-20 Operation area of browser pseudo-objects

	Function	Operation	Observation
Ureg related function			
	Ureg[]	O	
Greg related function			
	Greg[]	O	Refer to 8.1.15.2.
EPG functions			
	epgGetEventStartTime()	O	
	epgGetEventDuration()	O	
	epgTune()	-	
	epgTuneToComponent()	-	
	epgTuneToDocument()	-	
	epgIsReserved()	-	
	epgReserve()	O(*1)	
	epgCancelReservation()	O(*1)	
	epgRecIsReserved()	-	
	epgRecReserve()	-	
	epgRecCancelReservation()	-	
Event group index functions			
	grpIsReserved()	-	

	Function	Operation	Observation
	grpReserve()	-	
	grpCancelReservation()	-	
	grpRecIsReserved()	-	
	grpRecReserve()	-	
	grpRecCancelReservation()	-	
	grpGetNodeEventList()	-	
	grpGetERTNodeName()	-	
	grpGetERTNodeDescription()	-	
	epgXTune()	-	
Series reservation functions			
	seriesIsReserved()	-	
	seriesReserve()	-	
	seriesCancelReservation()	-	
	seriesRecIsReserved()	-	
	seriesRecReserve()	-	
	seriesRecCancelReservation()	-	
Non-volatile memory functionss			
	readPersistentString()	-	
	readPersistentNumber()	-	
	readPersistentArray()	O	
	writePersistentString()	-	
	writePersistentNumber()	-	
	writePersistentArray()	O	
	copyPersistent()	-	
	getPersistentInfoList()	-	
	deletePersistent()	-	
	getFreeSpace()	-	
Functions for controlling access-controlled non-volatile memory areas			
	isSupportedPersistentType()	-	
	setAccessInfoOfPersistentArray()	-	
	checkAccessInfoOfPersistentArray()	-	
	writePersistentArrayWithAccessCheck()	-	
	readPersistentArrayWithAccessCheck()	-	
Interaction channel communication			
Interaction channel communication--delay calls			
	registerTransmission()	-	
	registerTransmissionStatus()	-	
	getTransmissionStatus()	-	
	setDelayedTransmissionDataOverBasic()	-	
Interaction channel communication--BASIC procedure			
	connect()	-	
	disconnect()	-	
	sendBinaryData()	-	
	receiveBinaryData()	-	
	sendTextData()	-	
	receiveTextData()	-	
Interaction channel communication--TCP/IP			
	setISPParams()	-	
	getISPParams()	-	
	connectPPP()	-	
	connectPPPWithISPParams()	-	
	disconnectPPP()	-	
	getConnectionType()	O(*2)	
	isIPConnected()	O(*2)	
	saveHttpServerFileAs()	-	
	saveHttpServerFile()	-	
	sendHttpServerFileAs()	-	

	Function	Operation	Observation
	saveFtpServerFileAs()	-	
	saveFtpServerFile()	-	
	sendFtpServerFileAs()	-	
	sendTextMail()	-	
	sendMIMEMail()	-	
	transmitTextDataOverIP()	O(*2)	
	setDelayedTransmissionData()	-	
	getTransmissionStatus()	-	
	getTransmissionResult()	-	
	setCacheResourceoverIP()	-	
Interaction channel function--Acquisition function in common state of delay calls to BASIC system procedure and IP connection			
	getDelayedTransmissionStatus()	-	
	getDelayedTransmissionResult()	-	
Interaction channel function--Function to acquire line connection state			
	getPrefixNumber()	-	
Interaction channel function--large amount of call receipt service			
	vote()	-	
Interaction channel function--Encrypted communication using CAS			
	startCASEncryption()	-	
	transmitWithCASEncryption()	-	
	endCASEncryption()	-	
Interaction channel function--Communication by secret key cryptosystem without using CAS			
	setEncryptionKey()	-	
	beginEncryption()	-	
	endEncryption()	-	
Operation control function			
	reloadActiveDocument()	O	
	getNPT()	-	
	getProgramRelativeTime()	O	
	isBeingBroadcast()	-	
	lockExecution()	-	
	unlockExecution()	-	
	lockModuleOnMemory()	-	
	unlockModuleOnMemory()	-	
	setCachePriority()	-	
	getTuningLinkageSource()	-	
	getTuningLinkageType()	-	
	getLinkSourceServiceStr()	-	
	getLinkSourceEventStr()	-	
	getIRDID()	-	
	getBrowserVersion()	O	
	getProgramID()	O	
	getActiveDocument()	O	
	lockScreen()	O	
	unlockScreen()	O	
	getBrowserSupport()	O	
	launchDocument()	O	
	launchDocumentRestricted()	-	
	quitDocument()	O	
	launchExApp()	-	
	getFreeContentsMemory()	-	
	isSupportedMedia()	-	

	Function	Operation	Observation
	detectComponent()	O	
	lockModuleOnMemoryEx()	O	
	unlockModuleOnMemoryEx()	O	
	unlockAllModulesOnMemory()	O	
	getLockedModuleInfo()	O	
	getBrowserStatus()	O	
	getResidentAppVersion()	O	
	isRootCertificateExisting()	-	
	getRootCertificateInfo()	-	
	startResidentApp()	-	
	getDataDisplayAreaSize()	-	
	setFullDataDisplayArea()	O(*1)	
Receiver unit audio control			
	playRomSound()	O	
Timer function			
	sleep()	-	
	setTimeout()	-	
	setInterval()	O	
	clearTimer()	O	
	pauseTimer()	-	
	resumeTimer()	-	
	setCurrentDateMode()	O(*3)	
External character function			
	loadDRCS()	-	
	unloadDRCS()	-	
External device control function			
	enumPeripherals()	-	
	passXMLDocToPeripheral()	-	
Other function			
	random()	O	
	subDate()	O	
	addDate()	O	
	formatNumber()	O	
Caption display control function			
	setCCStreamReference()	-	
	getCCStreamReference()	-	
	setCCDisplayStatus()	-	
	getCCDisplayStatus()	-	
	getCCLanguageStatus()	-	
Directory control function			
	saveDir()	-	
	saveDirAs()	-	
	createDir()	-	
	getParentDirName()	-	
	getDirNames()	-	
	isDirExisting()	-	
File control function			
	saveFile()	-	
	saveFileAs()	-	
	getFileNames()	-	
	isFileExisting()	-	
File input-output function			
	writeArray()	-	
	readArray()	-	
Enquiry function			
	getDirInfo()	-	
	getFileInfo()	-	
	getCarouselInfo()	-	
	getModuleInfo()	-	

	Function	Operation	Observation
	getContentSource()	-	
	getStorageInfo()	-	
Data carousel accumulation function			
	saveCarouselAs()	-	
	saveCarousel()	-	
	saveModuleAs()	-	
	saveModule()	O(*1)	Detailed specifications in this volume are T.B.D
	saveResourceAs()	-	
	saveResource()	O(*1)	Detailed specifications in this volume are T.B.D
Bookmark control function			
	writeBookmarkArray()	-	
	readBookmarkArray()	-	
	deleteBookmark()	-	
	lockBookmark()	-	
	unlockBookmark()	-	
	getBookmarkInfo()	-	
	getBookmarkInfo2()	-	
	startResidentBookmarkList()	-	
Printing related function API – Print basic function			
	getPrinterStatus()	-	
	printFile()	-	
	printTemplate()	-	
	printUri()	-	
	printStaticScreen()	-	
Printing related function API – Memory card related			
	saveImageToMemoryCard()	O(*1)	
	saveHttpServerImageToMemoryCard()	O (*1)	
	saveStaticScreenToMemoryCard()	-	
Digital terrestrial broadcasting inherent function			
	X_DPA_mailTo()	O (*1)	
	X_DPA_startResidentApp()	O	
	X_DPA_phoneTo()	O (*1)	
	X_DPA_getRcvCond ()	O (*1)	
	X_DPA_getCurPos ()	O (*1)	
	X_DPA_saveExAppFile ()	O (*1)	Detailed specifications in this volume are T.B.D
	X_DPA_startExAV()	O (*1)	
	X_DPA_stopExAV()	O (*1)	
	X_DPA_tuneWithRF()	O (*1)	
	X_DPA_writeSchInfo()	O (*1)	
	X_DPA_getComBrowserUA ()	O (*2)	
	X_DPA_writeAddressBookInfo()	O (*1)	
	X_DPA_launchDocWithLink()	O (*2)	
	X_DPA_chkAVtype()	O (*1)	
	X_DPA_getIRDID()	O	
	X_DPA_writeCproBM()	O (*2)	

7.10.5.1 Operational rule of each extended function

- Operational rule of lockScreen()

  When using lockScreen(), take into consideration that unlockScreen() is done in contents afterwards.

- Operational rule of unlockScreen()

  When using lockScreen(), take into consideration that unlockScreen() is done in contents afterwards.

- Operational rule of getActiveDocument()

  When the function is called for the BML document being changed by the URI specified by fragment, returns URI as a return value excluding the fragment specification.

- Operational rule of reloadActiveDocument()

  Does not repeat the call of this function by contents automatically when connection to communication is not possible. Re-reading is carried out assuming that the URI excluding fragment specification are specified when the aforementioned function is called for the BML document that has changed by the fragment specification.

- Operational rule of setInterval()

  It is preferable to avoid the usage when the receiver has to bear a remarkably heavy load for processing when this function is used. Moreover, the number of timers that can be set at the same time is a maximum of 4.

- Operational rule of X_DPA_launchDocWithLink()

  This function can be used only in a linked state. Moreover, using this function can change the base URI directory. Therefore, the broadcaster should operate with maximum attention when describing the URI of the link contents transition destination so that the link state is not improperly set.

  Basically, the transition within the link contents uses the href attribute of launchDocumet() and a element, and when there is a requirement to change the base URI directory, use of this function is preferable.

7.10.5.2 Operation guidelines of function that generate communication

Assume the description of contents from which a corresponding function is executed by user operation, when using function that generate communication.

7.10.6 Extended functions provided by digital terrestrial broadcasting (1)

- getResidentAppVersion() : Receiver's native application recognition function

Refer to " [Section 2] 5.9.6 Extended functions that have been added in the digital terrestrial broadcasting" for the definition of the function. In Array[1] however, returns the character string of 20 characters or less that uses byte range between 21 to 7F of one byte character of the Shift JIS code set (refer to ARIB STD-B24 Vol. 2, Appendix 4, 4.1.1) provided arbitrarily by each manufacturer. Moreover, refer to Table 7-21 for the value that can be specified for argument appName. However, ComBrowser is not specified for appName.

7.10.7 Extended function provided by digital terrestrial broadcasting (2)

- X_DPA_mailTo(): Text mail transmission by mail applications of receivers.

Syntax:

```
Number  X_DPA_mailTo(
        input String subject,
        input String body,
        input String toAddress
)
```

Argument:

`subject`	Title of send mail
`body`	Text of send mail
`toAddress`	Destination address of send mail

Return value:

`1:`	Success
`NaN:`	Failure

Explanation:

Delivers the address and text to the e-mail application that the receiver possesses as a receiver features, and a mail is sent by e-mail application.

The difference to sendTextMail() is text is delivered to the e-mail application without directly establishing a session by the SMTP protocol. The one that actually transmits the mail is by the e-mail application function (For instance, the transmission button is selected), and the data broadcasting contents is not involved in the transmission timing.

It is preferable that the data broadcasting contents continue while the e-mail application is

in operation, and returns to the BML browser presentation state when the e-mail application ends.

It fails when the contents of the parameters, like the format etc., of the mail address are improper.

The maximum size of each argument is as follows.

body: 500bytes

subject: 30bytes

toAddress: 50bytes

For the contents of the parameters specified that exceed the maximum size, although there is a possibility of rounding down the part that exceeds the maximum size, it is preferable to not fail as a function. Meanwhile, the transmission of mail exceeding the maximum size can be done depending on the e-mail application capability.

- X_DPA_startResidentApp(): Receiver's native application initialization function

Syntax:

```
Number X_DPA_startResidentApp(
       input String appName,
       input Number showAV,
       input String returnURI
       (, input String Ex_info)+
)
```

Argument:

`appName` Receiver's native application for initialization

`showAV` Flag that determines whether to continue the TV image audio playback currently being presented even after receiver's native application starts. However, it is not an error, but is considered as 1, when the values other than the following are specified.

1: Playback continuation of the TV (video and audio)is permitted

0: Playback continuation of the TV (video and audio) is prohibited

`returnURI` It is URI of the component presented first, when the receiver's native application started by this function ends and the BML browser re-starts. In particular, when not specified, it is a null character string. It is not necessary for the receiver unit to work according to the specification of this argument, as this argument is only hint information for the receiver unit.

`Ex_info` Character string that shows supplementary information related to the initialization of receiver's native applications.

Return value:

`1:` Success

`NaN:` Failure

Explanation:

The receiver's native application specified with appName is initialized. When the receiver's native application specified with appName doesn't exist, NaN is returned.

It is assumed that contents switch by acquiring the type of communication browser installed in the receiver unit using X_DPA_getComBrowserUA() when a communication browser is started, and by changing to the appropriate URI. The operation follows the specifications in section 7.10.8, for whether to end the BML browser by starting the receiver unit built-in application specified with appName function.

The combination of assignable values for the arguments appName, showAV, and Ex_info are as follows. However, appName, showAV and Ex_info other than shown below shall require registration and they shall be managed by the governing organization described in Appendix 13.

Table 7-21 Value addressable in appName /showAV/Ex_info

appName	showAV	Ex_info
ComBrowser	Operated as defined	The fourth argument: URI character string handed over to a communication browser The fifth argument: Specification of an ideal display browser (hint information) Note 1) A browser that can display contents based on this specification in this volume: 0 A carrier specification browser : 1 HTML browser: 2 The sixth argument: A browser is displayed on all screens: 1 Note 2)
BookmarkList	1 only	None

Note 1) This describes the ideal browser to be used to display the communication contents of the transition destination.

Note 2) As a rule, the display of a data broadcasting browser is ended, when one is specified. At this time, the display size of ComBrowser is implementatiion dependent. However, it complies with the restriction of section 8.1.4.2 when leaving the display of a BML browser according to the condition of non-display of ComBrowser on all screens and indicated in section 8.1.4.2. When values other than the provisioned are specified, the operation is the

same as in the case where 1 is specified. Moreover, when omitted, it is an error.

- X_DPA_phoneTo (): Make calls by the specified number. " mobile phone native features"

Syntax:

```
Number X_DPA_phoneTo(
       input String phone_number
)
```

Argument:

`phone_number` Phone number

Return value:

`1:` Success

`NaN:` Failure

Explanation:

Makes calls to the specified telephone number. It is preferable to continue the presentation of data even after this function is called.

The character string that can be specified for the argument is the following 1 byte character.

"0" to "9"
"#" "*"
"(" "-" ")"
"P"

Among these, three characters of "(","-",")" are used for viewability of the notation, and should be ignored without considering it an error.

When the character string including characters other than the above-mentioned are passed, dialing should not operate, and it is an error.

When "P" is specified, it is preferable to stop dialing during a fixed time of the receiver unit (pose).

When the pose is not done, "P" is ignored (posed time 0 seconds).

Handling of dialing "#" and "*", pose time of "P", restoring method to the dial operation after pose, and etc. is an implementation dependent.

Example:

When specified "(03)1234-5678P910" in the argument

Dial first "0312345678", then dial "910" after the fixed time passes.

- X_DPA_getRcvCond (): Acquires the communication state in the telecommunication lines

Syntax:

```
Number X_DPA_getRcvCond ()
```

Argument:

None

Return value:

0:	No service
1:	Low reception state
2:	Medium reception state
3:	Excellent reception state
NaN	Receiver errors other than the mentioned above

Explanation:

The communication status in the telecommunication line is acquired. When 0 is returned, the communication status of no service is shown. Moreover, take it into consideration that there are receivers that don't display any service even in the communication restricted status, and this function shall return 0 in the state of the call restriction. Which status is allocated into each return value, when the division of the return value and the communication status in the receiver are different is an implementation dependent.

- X_DPA_getCurPos (): Obtain receiver geographical position.

Syntax:

```
Array X_DPA_getCurPos(
      [,input String posInfo]
)
```

Argument:

`posInfo:` Character string that indicates positioning method of location information and return value form

`"GPS":` `GPS` measurement first

`"CB" :` Base station, (cell base) positioning priority

Return value:

Array [0]:

"1": In case of returning by the latitude and longitude of the world positioning system (decimal number code)

"2": In case of returning by other geodetic information

Array [1]: Character string 1 showing present location (longitude, etc.)

Array [2]: Character string 2 showing present location (latitude, etc.)

Array [3]: Character string 1 showing supplementary information (positioning system and coordinate system, etc.)

Array [4]: Character string 2 showing supplementary information (error information on positioning result)

null: Failure

Explanation:

The geographical location information of the receiver is returned by using information, etc., obtained at the base station established by GPS (Global Positioning System) information using GPS satellites and mobile phones based on the location information acquiring function of receivers. It is preferable that the return value of this function returns the latitude and longitude in decimal number notation based on the world geographic coordinate system.

The positioning method is specified for argument (posInfo). When “GPS” is specified for the argument, it gives priority to information of the positioning method of GPS measurement and related to it, and returns it as a return value. When “GPS” is specified for the argument, and for receivers that do not correspond to GPS measurement, the return value of other positioning methods is returned. Similarly, it returns return values in priority to each cell base measurement information by the reception base station such as mobile phones, in case “CB” is specified and returns return values of other positioning methods that it doesn't correspond to. The positioning method when the argument is omitted is a matter of product planning and returns it by some positioning method of which the receiver unit corresponds to.

Null is returned without returning the array when the position information cannot be returned during execution of the function. When the return value returns the latitude and longitude in decimal number notation based on the world geographic coordinate system, "1" is returned to Array [0]. "2" is returned for other geographic coordinate systems. Operation (the location information acquired immediately before is returned) in which the location information cannot be acquired during execution of this function is an implementation dependent.

It is preferable that Array [1] and Array [2] return the longitude and the latitude in decimal number notation based on the world geographic coordinate system, respectively. The

latitude-longitude is expressed by a one-byte character row the part of integer values in which 10^7 is multiplied by the real number of each grade. For the south latitude and the west longitude add "-" (1 byte minus) at the beginning of the character string. The format of the return value is as follows.

Array [1]: Longitude (degree)

Return value The east longitude is expressed by the character string that shows the ten digit integer.

The west longitude is expressed by the character string that shows the 11 digit integer adding "-" at the beginning.

Example of return value: "1397597250" (For east longitude 139 degrees 45 minutes 35.01 seconds)

Longitude notation range "-1800000000" (180° west) to "1800000000 (180° east)

Array [2]: Latitude (degree)

Return value A character string that shows a nine digit integer to express the North latitude.

The south latitude is expressed by a character string that shows a ten digits integer adding "-" at the beginning.

Example of return value: "356641694" (For north latitude 35 degrees 39 minutes 51.01 seconds)

Latitude notation range "-900000000" (90° south) to "900000000" (90° north)

When it is not possible to return a value by the above-mentioned return value, but by the longitude and the latitude of sexagesimal number notation, the format of the return value is as follows.

Array [1]: Longitude (ddd degrees mm minutes ss.ss seconds)

Return value East longitude "dddmmssss", west longitude "-dddmmssss"

East longitude is expressed by the character string that shows the 9-digit integer.

The west longitude is expressed by the character string that shows the 10-digit integer adding "-" at the beginning.

Example of return value : "139453501" (For east longitude 139 degrees 45 minutes 35.01 seconds)

Longitude notation range "-1800000000" (180° west) to "1800000000" (180° east)

Array [2]: Latitude (dd degrees mm minutes ss.ss seconds)

Return value: North latitude "ddmmssss", south latitude "-ddmmssss"
North longitude is expressed by a character string that shows an 8-digit integer.
The south latitude is expressed by a character string that shows a 9-digit integer adding "-" at the beginning.
Example of return value : "35395101" (For north latitude 35 degrees 39 minutes 51.01seconds)

Latitude notation range: "-90000000" (90° south) to "90000000" (90° north)

It is preferable in Array [3] to return the geographic coordinate system denomination and the coordinate system denomination of the format shown in the following.

"wgs84_dgree": For World Geodetic System 1984 (WGS84,ITRF) and decimal notation coordinate system

"tokyo_dgree": For Tokyo Geodetic Datum and the decimal notation coordinate system

"wgs84_dms": For World Geodetic System 1984 (WGS84,ITRF) and sexagesimal notation coordinate system

"tokyo _dms": For Tokyo Geodetic Datum and sexagesimal notation coordinate system

The return values in case of other geographic coordinate systems, coordinate systems, and the measuring methods are a matter of product planning.

It is preferable to return the value indicated in the following as an error of the positioning result in Array [4].

null: Error unclear

"1": 0m ≤ error of positioning result < 50m

"2": 50m ≤ error of positioning result < 300m

"3": 300m ≤ error of positioning result < 1km

"4": 1km ≤ error of positioning result

"5": Cell base (positioning by base station of mobile phone)

- X_DPA_saveExAppFile(): An external application is stored in the destination according to the receiver setting. (T.B.D.)

  Syntax:

```
Number X_DPA_saveExAppFile(input String src_path
```

```
                    [,input String type
                     ,input String title
                    [,input Boolean executable_flag]] )
```

Argument:

src_path	URI that specifies acquired external application
type	MIME type of contents
title	Title name used to display list of contents
executable_flag	Executable flag

Return value:

1:	Success
NaN:	Failure

Explanation:

It stores the external application (communication contents of another encoding method) specified with the src_path in the area of the receiver unit set beforehand. The external application is executed at once after saving when the executable_flag is set to true. The execution operation will not begin when it is not set to true.

After execution ends (When the executable_flag is true), a startup document of the channel entry component that calls this function is presented. The operation of the contents continues when an external application is loaded and saved in case of the executable_flag is false.

- X_ DPA _startExAV (): The audio and video from the communication contents is displayed.

Syntax:

```
Number X_ DPA_startExAV(
       input String avURI
)
```

Argument:

avURI	URI showing video and audio

Return value:

1:	Success
NaN:	Failure

Explanation:

The audio and video stream by way of the communication is loaded and presented by the URI specified from the document that is presented. It is preferable to maintain the operation of the data contents during presentation.

The return value is not returned when the BML browser ends when the AV playback application, etc. is started. This function can be used by the broadcasting contents and linked contents.

- X_ DPA _stopExAV (): The audio and video from the communication contents is discontinued.

Syntax:

```
Number X_DPA_stopExAV()
```

Argument:

None

Return value:

1:	Success
NaN:	Failure

Explanation:

The presentation of the audio and video stream by way of communication presented according to the function (X_DPA_startExAV()) is stopped. When the audio and video stream by way of communication ends, it is preferable to return display to the broadcasting video and audio.

The receiver that ends a BML browser when the AV playback application, etc. starts when this function is executed, only returns 1 and nothing will happen.

- X_DPA_tuneWithRF(): A channel of mobile reception is selected by specifying the reception frequency by the physical channel number.

Syntax:

```
Number X_DPA_tuneWithRF(
       input Number freqChannelNo
       , input String ServiceID
       [, input Number Mode
       , input Number GuardInterval]
)
```

Argument:

freqChannelNo	Physical channel number
ServiceID	Service ID
Mode	OFDM mode
GuardInterval	Guard interval (guard interval ratio)

Return value:

NaN

Explanation:

Function, which selects a channel of portable reception, specifies a reception frequency, by a physical channel number. This function, mainly transmits information of the actual position to the communication destination server, is assumed to be used for channel selection after loading the frequency information of the station that can be received at the actual position from server. As an example, during the viewing of a certain program, it is assumed services can continue viewing the actual program by acquiring station information (ratio of physical channel number, mode, and guard interval) that broadcast the same program and the station that can receive from actual position, when the reception becomes unstable by the state of electrical waves, by changing the reception station with this function. It is an important function for services of mobile units in which reception during movement is assumed, and it is preferable to install this function.

When this function is called, the receiver unit does the channel selection display operation by physical channel number and service ID specified in the argument.

In the specification of physical channels only, receivers should always select channels by the service ID specified in the argument and not by the physical channel number, in order to avoid the incorrect selection of a broadcasting wave of a different broadcaster which uses the same physical channel, and to correctly select when multiple services exist within one physical channel.

The channel selection in display operation can be done at high speed by OFDM mode and the guard interval specified in the argument according to the receiver. An original system may be used for the scanning method of the mode and guard interval, and it is not necessary for receiver units to refer to this argument during channel selection operations.

Moreover, when an initial scan has already been done to the broadcasting service of physical channel specified in this function, selection can be done only by the service ID omitting the physical channel selection operation. Although it is an implementation dependent, when an initial channel is scanned, the physical channel where the scanning leakage occurs when the channel is selected by calling this function may be allocated in the remote control key.

The script following this function is not continued. Moreover, when the execution of this function fails, the continuance of the script execution that follows afterwards is not guaranteed.

Do not describe this function in a global code. The operation when described is an implementation dependent.

The range of each argument is as follows.

freqChannelNo: 1 to 62 (1 to 12 VHF band selection is not indispensable.)

ServiceID : "0000" to "FFFF"

Service ID is specified in a hexadecimal character string. However, the character (string) that shows that it is a hexadecimal character string such as the "h" symbol at the end and "0x" at the beginning of the character string is not given, and to become a fixed length character string with a length of four digits, 0 is the necessary number given at the start of the character string.

Mode : 1 - 3

GuardInterval : 0 - 3

0: Guard interval ratio 1/4
1: Guard interval ratio 1/8
2: Guard interval ratio 1/16
3: Guard interval ratio 1/32

- X_DPA_writeSchInfo(): Information is written in the schedule notebook of the receiver features.

Syntax:

```
Number X_DPA_writeSchInfo (
       input Date   date,
       input String title,
       input String text,
       input Boolean sound_flag
)
```

Argument:

`date`	Date of schedule written
`title`	Title of schedule
`text`	Contents of schedule
`sound_flag`	Alarm on/off. The alarm sound format and kind is an implementation dependent.

Return value:

1:	Success
NaN:	Failure

Explanation:

The schedule management application, etc., of the receiver starts, and the information specified for the argument is handed over. The contents of the schedule might not be preserved for the character number limitation in each item of the schedule notebook in the receiver features that was exceeded. The title of the schedule is saved as the title of the schedule notebook items, though which column to allocate it is an implementation dependent. The response to the alarm sound and tone setting is an implementation dependent although the setting in which the alarm sounds at a specified time and date is possible when sound_flag is true. Installation of the function does not need to be registered within the

schedule notebook prepared in the receiver itself which users write normally, although the function is specified assuming that the program is notified, and can be a separate installation.

Maximum value of each argument is as below

`title`: 40bytes

`text`: 80bytes

Operation on the schedule management application initiation and a method of information storage are implementation dependent.

- X_ DPA _getComBrowserUA (): Information to identify if a communication contents browser is acquired.

Syntax:

```
Array X_DPA_getComBrowserUA ()
```

Argument:

None

Return value:

Array [0][0]: Character string that shows the manufacturer ID (1)

Array [0][1]: Character string of User-Agent Header of a browser (1)

Array [1][0]: Character string that shows the manufacturer ID (2)

Array [1][1]: Character string of User-Agent Header of a browser (2)

* * *

Explanation:

Information to identify a communication browser installed in the receiver is acquired. When multiple numbers of browsers are installed in the receiver, all browser information is returned to the array in two dimensions. Returns the values which show the manufacturer ID administered in ARIB in Array [n][0]. The return ID of the carriers as a manufacturer ID when an aforesaid communication browser connects to a proprietary network of the carrier such as mobile phones. Contents use this function to identify the carriers. Character string of hexadecimal number notation is returned in Array [n][0], however, 0 is applied at the beginning of character strings as a mandatory number padding to make it 2 digits, although without giving a character (string) that shows it is a hexadecimal number string such as “0x” at the beginning and “h” at the end.

A character string of 127 characters or less is returned by one byte characters in Array [n][1].

The User-Agent Header character string in carriers browsers for the contents on the internet returns character strings based on the specifications of each carrier.

The User-Agent Header character string in Browser for the C-profile contents on the internets returns character strings provided by 8.3.14.

- X_ DPA _writeAddressBookInfo (): Information is written in the address book of the receiver features.

  Syntax:

```
Number X_DPA_writeAddressBookInfo (
        input String name,
        input String kana,
        input String tel,
        input String mail
)
```

  Argument:

`name`	Registered name
`kana`	Hiragana printed to name (2 byte katakana)
`tel`	Registered telephone number
`mail`	Registered mail address

  Return value:

1:	Success
NaN:	Failure

  Explanation:

  The address book and the telephone book management function, etc. in the receiverstart, and information is handed over to the address book that manages information in arguments (name, kana, tel, mail) as a single-unit. Coexistence with address books that already exist as a receiver features is not necessary.

  It is an implementation dependent for whether an item that does not correspond is registered when the argument information on the function is mixed together in an existing address book.

- X_DPA_launchDocWithLink (): Base URI directory is switched, and the BML document is presented while linked.

  Syntax:

```
Number X_DPA_launchDocWithLink (
        input String documentName
)
```

  Argument:

`documentName`	Character string that specifies BML document

Return value:

1:	Success
NaN:	Failure

Explanation:

The BML document specified by the documentName is opened and presented on the screen. It is specified only when it is changed to the BML document offered by a server from the BML document offered by a server in a linked status. The host name or the directory name specified by this function is set to the base URI directory (Refer to section 8.3.7 for the definition of base URI directory). The base URI directory before it changes is invalid. The base URI directory is assumed to be set the same as when changing while linked from the broadcasting contents in the a element and launchDocument() function.

The API whose operation is permitted in linked status, and the information shared by NVRAM becomes possible in the BML document of the changed destination with this function.

- The script following X_DPA_launchDocWithLink() is not continued.
- Neither load nor the unload event occur when X_DPA_launchDocWithLink() is executed in a global code.
- The continuance of the script execution that continues behind, when X_DPA_launchDocWithLink() fails, is not guaranteed.
- Receiver operation fails, and presents the broadcasting contents according to section 8.3.11.4. when X_DPA_launchDocWithLink() is used in the broadcasting reception status.

- X_DPA_chkAVtype (): The type of audio and video from the communication contents is confirmed.

Syntax:

```
Number X_DPA_chkAVtype (
       input String MIME_Type
       [,input String mediaName]
)
```

Argument:

`MIME_Type`	MIME type of image audio
`mediaName`	Service name of image audio

Return value:

1:	Success
NaN:	Failure

Explanation:

When there is an audio and video stream presentation function by way of communication, success is returned for the contents type specified by MIME_Type and mediaName. The MIME type is specified for MIME_Type. The service name of the image voice that cannot be specified by MIME_Type alone can be specified for the mediaName. For example, even when the MIME_Type is “video/3gpp”, a detailed specification might be different according to the service, and a mediaName specification may be necessary.

- X_DPA_getIRDID (): The identifier to specify the receiver or the viewer is acquired.

Syntax:

```
String X_DPA_getIRDID (
       input Number type
)
```

Argument:

type	Type of identifier that demands acquisition

Return value:

Unique identifier:	Success
null:	Failure

Explanation:

Whether it succeeded in acquiring the receiver unit specified by type, the user unique identifier or identifier is returned as a return value. The format of the identifier is an implementation dependent. When the unique identifier specified for the argument cannot be returned, null is returned as the return value.

The following are specified for argument (type).

1: Receiver unique identifier

Identifier of each receiver that specifies the manufacturer’s serial number, terminal number, etc. Return values are some ID (character string) by which the manufacturer or each carrier can uniquely identify the receiver.

2: Televiewer-unique identifier

Identifier which specifies the viewer individually like user ID and telephone number, etc. ID (character string), which uniquely identifies the user of user.

3: Receiver unique identifier or televiewer-unique identifier

Either a receiver unique identifier or a televiewer-unique identifier is returned.

When a receiver unique identifier can be returned, it is given priority.

- X_DPA_writeCproBM (): Registration of TVlink.

  Syntax:

```
Number X_DPA_writeCproBM(
       input String title,
       input String dstURI,
       input String outline,
       input Number CproBMtype
       [,input Date expire]
)
```

  Argument:

`title`	Title of TVlink
`dstURI`	URI link destination of TVlink
`outline`	Explanation of TVlink
`CproBMtype`	Type of TVlink
`expire`	Expiration date of TVlink

  Return value:

1:	Success
-1:	The NVRAM area of a data size necessary for registration cannot be reserved.
-2:	As it does not support the contents presentation in the specified format of argument CproBMtype, registration has not been done.
NaN:	Other failure

  Explanation:

  It is a function to register information on the TVlink. Since loading is done only by the receiver's native application, the format in the receiver is an implementation dependent.

  Argument title is a character string displayed in the title column of the TVlink list. The title is 40 bytes or less.

  Argument dstURI is the URI of the communication contents in the destination link. URI in the destination link is 60 bytes or less.

  Argument outline is an explanation of the communication contents in the destination link. The TVlink outline is 130 bytes or less. A null character string is specified when there is no TVlink outline on what should be registered.

  Argument CproBMtype shows the receiver operation when registered information is used. For example, the receiver unit can select a browser that can display the communication

contents specified with destination link URI by using this information. The TVlink type is specified with a numerical value from 0 to 255. The TVlink type that the receiver can present can be confirmed by using the (getBrowserSupport()) function. It is preferable to confirm the possibility of presentation of the TVlink type specified before registration by this function, if necessary.

Argument expire specifies the valid final date of TVlink information in year, month, day, hour and minute. When the expiration date is omitted, the TVlink is assumed to be always effective.

<Specification in this volume of receiver behaviour>

Receiver behaviour is an implementation dependent when an argument title is specified exceeding 40 bytes. However, it is preferable that the receiver can register 40 bytes when it has exceeded 40 bytes, and does not store the 41st byte or more. Moreover, it is preferable not store the first byte of a two byte character when this is the 40th byte.

When argument dstURI has exceeded 60 bytes, the receiver unit assumes the execution of this function is a failure, and returns the return value NaN.

The receiver behaviour is an implementation dependent when the argument outline specified exceeds 130 bytes. However, it is preferable that 130 bytes are recorded, and the receiver does not record the 131st byte or more when it has exceeded 130 bytes. Moreover, it is preferable not to store the first byte of a two byte character when this is the 130th byte. When the TVlink type is memo (CproBMtype=0) or C-profile linked contents (CproBMtype=1), the registration of the TVlink outline to the receiver is necessary.

It is preferable to record the TVlink outline to improve user operation though the registration is optional for other types.

Argument CproBMtype does not need to be registered in the receiver unit, which the operation of the TVlink list application can be concluded without registration, of this information.

Registration of the information given by an argument expire is an implementation dependent.

Receiver with a user interface function that selects and eliminates TVlinks already recorded and maintained in an area when the NVRAM data size area necessary for a registration can not be reserved during this function execution, not executes deletion as a result of the user using this function, and when it fails to register, it returns NaN as the return value.

7.10.8 Execution operation of the extended function

It is assumed that the processes of X_DPA_startResidentApp(), X_DPA_mailTo(), X_DPA_phoneTo(), X_DPA_startExAV(), X_DPA_writeSchInfo(), X_DPA_writeAddressBookInfo () are asynchronously processed among the extended functions specified by 7.10.7. It is preferable that BML browsers continue the execution of script after the extended functions are executed. However, it is preferable to present the default component again after ending the function execution that is called by the extended function when it is impossible to continue.

7.11 Restrictions in BML document descriptions

7.11.1 Restrictions of the script element

Script elements in which the src attribute is not specified may appear 0 or 1 time, and to which the src attribute is specified may appear between 0 to 2 times. When the content of the script element is described, all of the content is enclosed with “<![CDATA[“ and “]]>” and it is one CDATA section.

The argument is not written in the function used as an event handler. That is, setting of the event handler is followed by the syntax which becomes attr = "func();" (Where attr is an event attribute name and func is a function name).

Operation of automatic semicolon insertion is not mandatory. (ECMA-262 p.20)

7.11.2 Positioning and restrictions

This section provides operation related to positionings for the allocation of elements in the body element.

It shall add a style sheet to specify the position in the BML document. Elements included in the document should be presented properly to users, although there is no uniqueness among various platforms in presenting each element.

The constraints defined below are only applicable to elements that will occupy the 2-dimensional space when placed within the body element, and by no means they prohibit other elements from being included in the body element.

Moreover, in the sentence on the restriction conditions, it is simply recorded as, "The position is specified" for the size and position of the box specified by absolute positioning. The method of specifying the position is the relative coordinates from the parent element (equivalent to the containing block of CSS2) is as follows:

`left`: Distance (number of pixels) from the left end of the `box` of the parent element;

`top`: Distance (number of pixels) from the top of the `box` of the parent element;

`width`: Width (number of pixels);

`height`: Height (number of pixels);

`visibility`: `"visible"` or `"hidden"` or `"inherit"`;

As the positioning is operated by a fixed value (i.e. the specified value is “absolute”), Therefore it is not specified in the style part.

Furthermore, the following terms are used in in the sentence below.

boxed element: a) elements (input, textarea) that have two dimensional expansion, b) element (img, object) that have intrinsic dimension. c) div element, d) form element, e) p element, f) pre element is generically called ‘boxed element’.

normal flow element: Elements (br span a) other than boxed elements are called ‘normal flow elements’ in elements that can be used in operation.

Note) html head body meta script link title bml:bevent bml:beitem elements do not belong to either.

Restriction 1:	A position of boxed element must be always specified.
Restriction 2:	A position of normal flow element must not be specified.
Restriction 3:	The child element of body element must be either div, form, p, or pre elements.
Restriction 4:	Only boxed elements can be involved in the div element and form element. A nest positioning becomes possible by having div element nested.
Restriction 5:	The p element and pre element can have only the element of normal flow and the sequence of the text as the child element.
Restriction 6:	Nesting of the span elements and a element must not be done.

The condition provided by DTD is overwritten by the above-mentioned restriction condition.

(Example)

```
<p style="...box specification...">
      abcde<span style="...">fghij</span>
</p>
```

In CSS2, although absolute positioning and normal flow can be specified in each element, the basic idea of the above restriction conditions is to completely separate the arrangement of the elements of normal flow that are not position specified and elements specified by absolute positioning.

7.11.3 Vertical writing

The specification of the direction of writing by the BML document is not operated. Vertical writing is not operated.

7.11.4 Restrictions of element positioning

- A visibility property shall not set in visible at the same time as multiple p elements that specify -wap-marquee properties.
- For the p element, that -wap-marquee property is specified and the object element, which refers to animation GIF, both visibility properties shall not be set in visible simultaneously.
- Object elements with which the animation GIF figure is associated can present up to four at the same time within the range of the restriction in section 5.2.2.11.

Elements may be allocated in the back of the rendering area of the elements, and the background-image of the body element that includes the element can be specified. However, object elements associated with visible animation GIF figures and p elements for which marquee is specified, cannot be placed in the background. In other words, there must be no overlapping area among animation GIF figures with visibility set in visible and p element for which marquee is specified.

When other elements are overlapped in front of the object element with which the animation GIF figure is associated and the p element for which marquee is specified, it is not possible to display both at the same time. That is, visibility should be “hidden” in either the elements or all elements placed in front of them.

Table 7-22 Restriction of visibility when elements are allocated on top of each other

Element			Rear					
			animation GIF		marquee		Other elements	
		visibility	visible	hidden	visible	hidden	visible	hidden
Front	animation GIF	visible	-	O	-	O	O	O
		hidden	O	O	O	O	O	O
	marquee	visible	-	O	-	O	O	O
		hidden	O	O	O	O	O	O
	Other elements	visible	-	O	-	O	O	O
		hidden	O	O	O	O	O	O

O *** Operated - *** Not operated

7.11.5 Operation related to object elements

Note) The presentation behaviour of individual monomedia presented by object elements is specified in section 7.12.3.

7.11.5.1 Operation related to the presentation operation of animation GIF

This section specifies for the interpretation of attributes when animation GIF is specified for object elements and operation related to screen presentation. For the operation of the presentation behaviour of animation GIF related to streamstatus, See 7.12.6.2.

- The data attribute can be changed only when animation GIF image has not been played.
- Whether to stop the rewriting of the presentation frame of animation GIF during which a screen update is prohibited by execution of lockScreen() or to continue playback is implementation dependent.

7.11.6 Operation related to nested elements

Neither the span element nor the a element become nested. In fact, other elements are not included in the span element and a element. However, as an exception the br element can be included.

7.11.7 Operation related to the p element

The interpretation when -wap-marquee property is specified in the p element and the operation related to screen presentation is provided below.

- The contents of p element can be rewritten only when the visibility is hidden.
- Whether to stop the modification of a display position in p element contents during which a screen update is prohibited by execution of lockScreen() or to continue playback is implementation dependent.
- When visibility changes from hidden in visible, playback is from the initialization status.
- The child element of p element applied is only one PCDATA node. Neither span, a, nor br elements are applicable. The maximum length of the PCDATA is 128 bytes. When exceeding this, the character displayed is rounded down.
- Do not set the visibility property of multiple p elements for which the wap-marquee property is specified to visible at the same time.
- For the p element,that the -wap-marquee property is specified and the object element, which refers to animation GIF, both visibility properties shall not be set in visible simultaneously.
- Transparency cannot be specified for the background color of p elements for which this property is specified.

7.12 Operational guidelines related to presentation

7.12.1 Operational guidelines of object presentation

The presentation order of objects is the appearance order of the elements. In other word, the element that appears first in the BML document is presented at the inner part (position far from viewer) of the screen. The relationship between the structure and presentation of DOM objects is shown in Figure 7-1. DOM objects generate a tree structure in the order of element presentation. The presentation is done in order of 1, 2, 4, 5, 7, 8, 9, 3, and 6 when the structure of object 1 to object 9 becomes as shown in Figure 7-1.

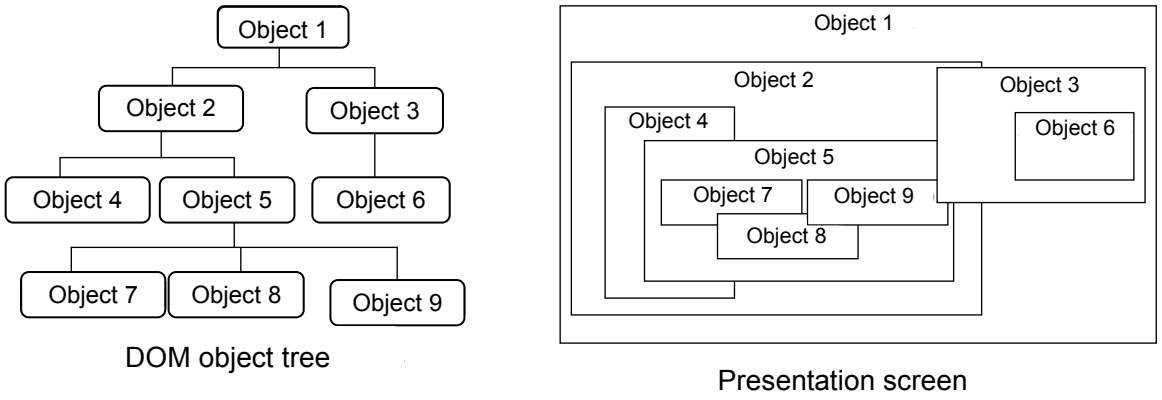


Figure 7-1 Structure and presentation of DOM objects

7.12.2 Presentation plane

Rendering of the objects described in 7.12.1 is done in a virtual plane for a BML browser shown in section 3.2.1. The virtual plane has a logical Cartesian coordinate system where the upper left corner is the starting point (0,0), and the coordinates are in units of integers. A size of 1 pixel of each vertical and horizontal corresponds to 1px in the BML contents. Therefore, the horizontal width of this virtual plane becomes 240 pixels and the vertical width becomes 480 pixels.

Moreover, each pixel has color information of RGB in each 8 bits. However, the bit depth of color information is implementation-dependent. For this detail, see section 3.2.2.

7.12.3 Operation of monomedia presentation

The following shows operation of monomedia presentation using an object elements and img elements.

7.12.3.1 Positioning by object elements and img elements

Operations of the top, left, width, and height in the CSS2 properties applied to object elements and img elements are shown below.

■ top property, left property

The value that the top property and the left property may take according to type attribute (media type of the monomania specified as the src attribute in case of the img attribute) of object elements is shown in Table 7-23.

Table 7-23 The value available for top/left properties

type attribute (Media type)	
image/jpeg	Arbitrary in both x and y coordinates
image/gif (case of GIF)	Arbitrary in both x and y coordinates
image/gif (case of animation GIF)	Arbitrary in both x and y coordinates

■ width property, height property

The formula for computation of the width characteristic and height characteristic is shown in Table 7-24. The screen presentation when the value, in which the width property and height property of object elements are not filled in with the relationships of this table, is implementation-dependent.

Table 7-24 width/height property

type attribute (Media type)	width	height
image/jpeg	W	H
image/gif (case of GIF)	W	H
image/gif (case of animation GIF)	W	H

Note) For W and H, refer to Table 7-25

■ Vertical and horizontal pixel numbers

W (Image horizontal pixel number), H (Image vertical pixel number) will take the values of Table 7-25.

Table 7-25 Values applicable to W and H

type attribute (Media type)	W	H
image/jpeg	16 - 240	16 - 320
image/gif (case of GIF)	2- 240	2- 320
image/gif (case of animation GIF)	2- 240	2- 240

7.12.4 Guidelineson clipping

Clipping can be done to the child element by the parent element because it is an operation done according to a fixed value in which the overflow property is specified as hidden in these guidelines. When the content overflows a rectangular area of a specified element (object 1), clipping is done as in the left of Figure 7-2. In addition, when clipping the child element (object 3) by the parent element (object 2), as shown this figure on the right, it becomes non-display excluding the rectangular area specified by the parent element.

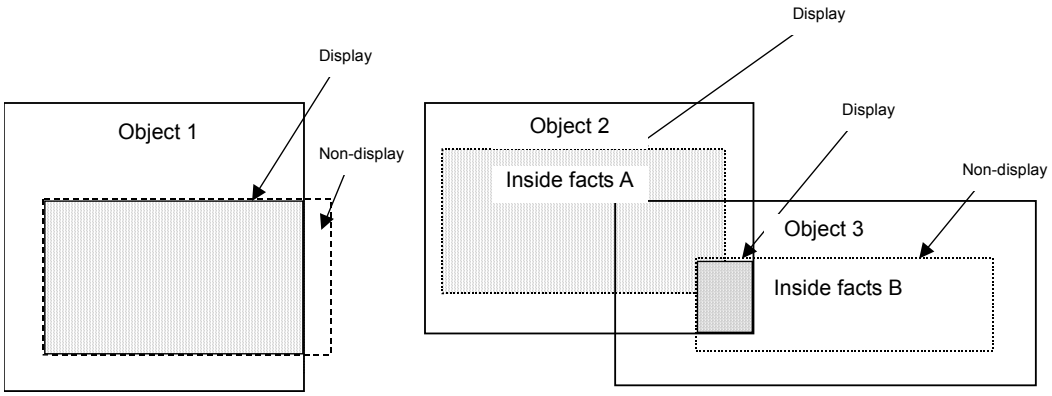


Figure 7-2 Example of a clipping behaviour

When jpeg images, GIF images, and animation GIF images are referenced by object elements or img elements, the clipping by the parent div elements is undefined. It should not be described in the contents specifying that the monomedia image specified by the object/img element overflows the div element of parents. Moreover, when the “-wap-marquee” property is specified in the p element, the clipping by the parent div element is undefined.

These rules are similar for clipping the second generation or higher and should not described the ancestor elements to perform clipping in the contents.

7.12.5 Range of values where CSS2 properties is applicable

The value range that is applicable “<length>” as the value, among CSS2 properties, is shown in Table 7-26 .

Table 7-26 Range of values where CSS2 properties can be taken

Property	Range of values
top	-480px to 480px
left	-240px to 240px
width	0 to 240px
height	0 to 480px
border-width	0 to 240px
padding-top,bottom	0 to 480px
padding-right,left	0 to 240px
line-height	normal or 30px - 420px

7.12.6 Provision of box models

The margin property is fixed to 0. Therefore, this model is operated only by the borders and padding shown in Figure 7-3. Moreover, the arbitrary boxed elements shown by 7.11.2 shall satisfy the following, when (x0,y0) is the relative position of the border area upper left corner of this element to the body element upper left corner and when “w” and “h” are width and height respectively before clipping taking into account the contents area, the padding area, and the border area of this element.

x0 ≥ -240, y0 ≥ -480, x0 + w ≤ 480, y0 + h ≤ 960, 0≤w≤240, 0≤h≤480

where , w = width + padding-right + padding-left + border-width * 2 and h = height + padding-top + padding-bottom + border-width * 2

When the value of CSS2 properties that do not satisfy the above-mentioned conditions, the presentation is implementation- dependent.

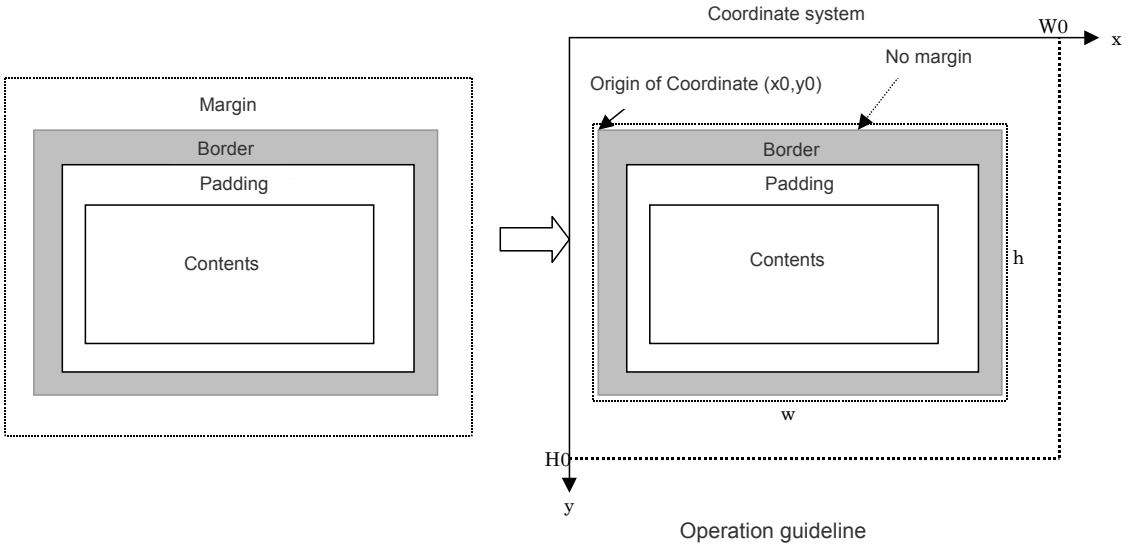


Figure 7-3 Presentation of the box model

7.12.6.1 Attribute operation related to stream presentation behaviour

Table 7-27 shows the operation of the streamstatus attribute of object elements

Table 7-27 Attribute operation related to stream presentation

Type attribute	Streamstatus attribute
audio/X-arib-mpeg2-aac	play,stop Note 2) Note 4)
image/jpeg	play Note 1)
image/gif	play,stop,pause Note 3)

Note 1) The initial value of the streamstatus attribute of the object element that refers to media of the type attribute which only “play” is taken, shall be “play”.

Note 2) The initial value of the streamstatus attribute of the object element that refers to media of the type attribute which “stop” can be taken, shall be “stop”. The MediaStopped event doesn't occur in the initial status. Moreover, the dynamic changes of the type attribute and the dynamic changes of the scheme by changes in the data attribute in audio cannot be done in the object element.

Note 3)The initial value of the streamstatus attribute shall be “play”. Moreover, the assignment of the streamstatus attribute and the change by script cannot be done when it is not animation GIF. In addition, the switching of GIF and animation GIF cannot be done by a change in the data attribute.

Note 4)When the playback ends, the streamstatus attribute becomes stop.

7.12.6.2 Presentation behaviour of animation GIF and operation of the streamstatus attribute

The interpretation of the attribute when animation GIF is specified for the object element and operation of the screen presentation is explained.

- The operation of animation GIF according to the initial value of the streamstatus attribute is shown in Table 7-28.

Table 7-28 Operation of initial value of the streamstatus attribute

Initial value of Streamstatus attribute	Behaviour
play	Start playback from the first frame.
stop	The frame shall not be displayed. The playback frame position is reset at the beginning.
pause	Display the first frame.

- The presentation status of animation GIF when the streamstatus attribute is changed by using DOM API is shown in Table 7-29.

Table 7-29 Access to the streamstatus attribute

Value of original streamstatus attribute	Set to play	Set to stop	Set to pause
play	-	• Stop playback and do not show the frame.	• Stop playback and display any one frame. Which frame to display is implementation dependent.
stop	• Start playback from first frame.	-	• First frame shall be displayed.
pause	• Start playback. Which frame to display first is implementation dependent.	• Any Frame shall not display.	-

8 Guidelines for browser's behaviors

8.1 Guidelines for presentation

The guidelines for the presentation function of C-profile receivers are shown below.

8.1.1 Principle of presentation in C-profile receivers

8.1.1.1 Principle of simultaneous presentation of video, audio and multimedia data

In C-profile receivers, video, audio and data are unified as broadcasting contents, and in principle shall not be presented separately. However, the following cases are exceptions, provided that "8.1.3 Operation during the startup of the TV reception function and tuning" is followed accordingly.

- If audio is presented by itself by user operation
- If BML browser is displayed in full screen by user operation
- If it applies to "8.1.8.2 Exceptions in multimedia data display"

8.1.2 Presentation of video and audio

8.1.2.1 Presentation and control of broadcasting video and broadcasting audio

Presentation and control of broadcasting video and broadcasting audio shall be conducted by receiver features. No control of presentation including scaling from multimedia shall be conducted.

8.1.2.2 Display position and display size of video

The display position and display size to display video in the receiver are Implementation dependent. The receiver does not necessarily need to display the video transmitted in QVGA in the same pixel size. Specifically, display is permissible with a reduced video pixel size and frame rate if the processing performance is insufficient. Also, the display may be enlarged in receivers with a pixel size level of QVGA or higher.

8.1.2.3 Optimizing video display size

With receivers whose display device pixel size is not 320 dots in the horizontal direction, it is recommended scaling the video depending on the receiver features according to the pixel size of the display device.

8.1.2.4 Display of QVGA 16:9 video

When displaying QVGA (16:9) video, the area within the display device that the video possesses is 60 dots smaller in the vertical direction in an imaginary plane compared to the

area possessed by QVGA (4:3) video. It is recommended setting 60 dots of area space as the visible area for data broadcasting rather than leaving it empty.

8.1.2.5 Changing the video display size

Receivers may switch from several display sizes to display video.

8.1.3 Operation during the startup of the TV reception function and tuning

8.1.3.1 Principle of display status when the TV reception function startups and during tuning

When the TV reception function starts up and immediately after tuning using the procedure below, only video, audio, and data from broadcasting is presented on the receiver's display device and no other contents outside of broadcasting or applications shall be displayed (such as e-mail screen, etc.). Contents and applications that were displayed before tuning shall not be displayed. Also, this principle may be skipped only when users operate tuning related buttons during operation of the TV reception function.

- When the function (X_DPA_tuneWithRF()) for tuning is called from data broadcasting contents or C-profile linked contents.
- When channels are selected using receiver features other than BML browser. (Tuning by communication browser's "a" element URL expansion, etc., tuning by API's of java applications, tuning using API's to call the tuning function implemented in receivers, etc.)
- Tuning from a TVlink list

8.1.3.2 Principle of BML browser key acquisition

During operation of the TV reception function, BML browsers shall always be able to acquire the enter key and back key in principle. In the same manner, focus transfer and scrolling shall be useable by receiver features. However, if other applications are simultaneously displayed and the application contains the focus, this may not apply. Focus transfer and scrolling implementation procedures (keys to use, etc.) are implementation dependent, but it is preferable to use the up/down keys.

8.1.4 Prohibition of mixed display and simultaneous display

8.1.4.1 Princilple of mixed display prohibition

It is prohibited to possess mixed display functions in receiver. This "mixed display" means

displaying contents provided by multiple different suppliers which may cause viewers to misunderstand that those contents are provided by same supplier such as cases where certain contents suppliers intentionally display contents provided by different suppliers at the same time by getting one set of contents to be related to other contents or cases where one set of contents affects the other display contents, etc. The function, which enables such display, is called a mixed display function. However, with C-profile linked contents in a link state, 2 contents of TV and communication can be regarded as being supplied by the same supplier, and thus is not a mixed display.

8.1.4.2 Simultaneous display of broadcasting screens and screens other than broadcasting (browsers for the contents on the internet, applications, etc.)

- When starting up contents not supplied by the broadcasting company while presenting broadcasting contents, it is recommended to turn off the broadcasting screen and present the contents concerned on a full screen in order to avoid mixed display.
- While simultaneous display of broadcasting screens and screens other than broadcasting is not encouraged, when conducting simultaneous display according to implementation dependent, leave BML browser on as shown in 8.1.8.1 and display the following at the minimum. Also, same manner of handling is required for cases where caption display is conducted in place of BML browser by user operation, etc. Even if these measures are taken, keep in mind that conducting simultaneous display has a risk of becoming mixed display, and thus consideration in implementation dependent shall be taken.
  (i) When starting up contents other than broadcasting during TV presentation, in principle, notify the viewers by with a dialog box, etc. that contents unrelated to broadcasting will be displayed.
  (ii) Presenting options for the viewers to choose when starting contents other than those broadcast during TV presentation is recommended, unless the choice between simultaneously display or switching is specified by API(X_DPA_startResidentApp()).
  (iii) Always display so that it is clear to everyone that contents unrelated to broadcasting are being displayed during simultaneous display.

8.1.4.3 Various operations during simultaneous display

- Even when simultaneous display of BML browsers and browsers for the contents on the internet is conducted, C-profile linked contents shall be displayed on BML browser.
- When an e-mail screen is started when receiving e-mail and is simultaneously displayed, a simultaneous display function is recommended to limit e-mail from addresses stored in the address books of terminals, even while being displayed under the above restrictions.

8.1.4.4 Operations of broadcasting content during simultaneous display

- Even during simultaneous display, BML browsers shall continue to operate and be able to acquire each event from broadcasting.
- Even when browsers other than broadcasting and applications are simultaneously displayed, BML browser shall be able to acquire key entries according to operation of the viewers.

8.1.5 Prohibition of simultaneous display while displaying a part of broadcasting content (video audio only, audio only or BML browsers)

8.1.5.1 Principle of prohibition of simultaneous display while displaying a part of the broadcasting contents

- As a rule, when only a part of the broadcasting content (video or BML browser) is being presented on the screen, the partial content shall be displayed on its own and must not be displayed simultaneously with any content and/or application provided by other operators.
- To start presentation of contents or applications supplied by another company during presentation of a part of the broadcasting contents, the broadcasting screen shall be turned off and switching to a full screen shall be conducted, excluding the exceptions given in the following clause. To continue to receive broadcasting and to present audio in the background is implementation dependent.

8.1.5.2 Exceptions to simultaneous display prohibition when presenting only a part of the broadcasting contents

The following screens may be simultaneously displayed as exceptions even when presenting only a part of broadcasting content.

- Screens below in e-mail application
    - e-mail composition screen
    - Transmission related screen
    - e-mail reception notification excluding subject line
    - e-mail sender, subject line and content of e-mail from senders whose address entry is in the address book of the receiver
- Screen which is displayed by a receiver specific application which was installed in the receiver at the time of purchase and does not conduct communication. (Including when the receiver specific application is updated using communication in order for debugging, etc.)

- To present streamed videos acquired via communication using the X_DPA_startExAV() function according to the specifications in “8.1.7.6 Display of videos acquired from communication”
- Message, etc. provided in 8.1.6

8.1.6 Simultaneous display of messages presented by receiver systems

Messages displayed by receiver systems for the convenience of viewers or to seek consent from viewers may be simultaneously displayed. (Same for cases where a part of the broadcasting contents is presented)

- Program table and information using SI information acquired from broadcasting
- Receiver’s system status display
- Receiver’s set up screen
- Operation support information for users
- Incoming telephone call notification and caller number notification (for compound terminals with telephones)
- Supplementary information for transmission such as displaying the entered phone number, etc. when users make phone calls (for compound terminals with telephones)
- Display of user interfaces to display contents for permission and to obtain the approval of users when the permission of users is required to for communication connection
- TVlink list screen
- Alert of operations which may cause inconvenience to the user

8.1.7 Specifications regarding communication contents presentation

8.1.7.1 Contents presented in BML browsers

BML browsers are browsers implemented based on these specifications, and present data broadcasting contents and C-profile linked contents

8.1.7.2 Activation of communication content other than C-profile linked content

If unlinked communication content will be activated from data broadcasting or C-profile linked content, the content shall be presented by activating another receiver-specific communication browser using X_DPA_startResidentApp(). Specification of a receiver specific communication browser is implementation dependent, and the choice of the browser to implement in the receiver is free. However, fulfilling the Browser for the C-profile contents on the internet specifications provided in chapter 7 is recommended.

8.1.7.3 Specifications to display receiver specific browsers for the contents on the internet

To startup receiver specific browsers for the contents on the internet using X_DPA_startResidentApp(), display strictly according to the specifications of mixed display prohibition provided in 8.1.4. Also, if full screen view is specified by the argument of this function, the receiver shall turn off the broadcasting screen. (Follow the receiver setting if full screen view is not specified by the argument of this function)

Even if non-linked communication contents are displayed by using receiver specific browsers for the contents on the internet by receiver features and not the function, strictly follow the principle of mixed display prohibition. From the view point of mixed display prohibition, turning off the BML browser and video, and switching full screen to communication browser is recommended (generic term for Browser for the C-profile contents on the carriers' proprietary browsers).

8.1.7.4 Common use of BML browsers and Browser for the C-profile contents on the internets

BML browsers and receiver specific browsers for the contents on the internet are logical concepts and they do not limit the implementation procedures of receivers. For example, it is possible to implement a browser which has BML browser functions and Browser for the C-profile contents on the internet functions and use them by logically switching the operations. However, in that case, the principle of mixed display prohibition shall strictly be followed to present non-linked communication contents and to conduct display.

Moreover, whether or not to implement a C-profile communication browser to a receiver already incorporating a browser other than a C-profile communication browser as its specific communication browser shall be implementation dependent.

8.1.7.5 Browser to present with TVlink

When presenting communication contents based on the TVlink information (URI) recorded by broadcasting contents, the contents shall not be presented in a link state due to the principle of mixed display prohibition, regardless of the channel that recorded TVlink and the channel currently being shown. Therefore, presenting communication contents from TVlink information, for instance, on receiver specific communication browser is assumed. (Excluding CproBMtype=1 TVlink)

8.1.7.6 Display of videos acquired from communication (optional)

To display streamed videos acquired via communication by X_DPA_startExAV(), as a general rule, stop to present broadcasting video and audio, and display this movie in a broadcasting video display area. However, it is recommended to continue with presentation by data broadcasting.

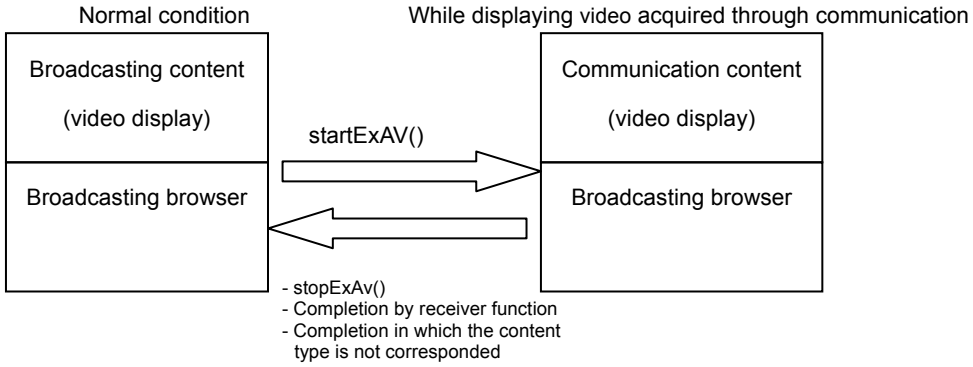
In this case, the receiver shall consider that the viewer can identify that the video was acquired via communication.

Completion of communication videos may be conducted by X_DPA_stopExAV() and by using the completion function of communication video in receiver features if it is implemented. When the communication video ends, returning the display to broadcasting video is recommended.

The transition in which data broadcasting presentation continues when displaying video that the receiver acquired via communication is shown in A of ”Figure 8-1 Display of video acquired through communication”. Data broadcasting continues to be shown while displaying streamed video acquired through communication.

If it is impossible to continue presenting data broadcasting while displaying video that receiver acquired through communication, BML browser can be stopped once as shown in B of ”Figure 8-1 Display of video acquired through communication”. However, it is recommended to quickly re-open the broadcasting contents once the display of video acquired through communication stops.

A. If data broadcasting is continued while displaying video acquired through communication



B. If data broadcasting presentation cannot continue while displaying video acquired through communication

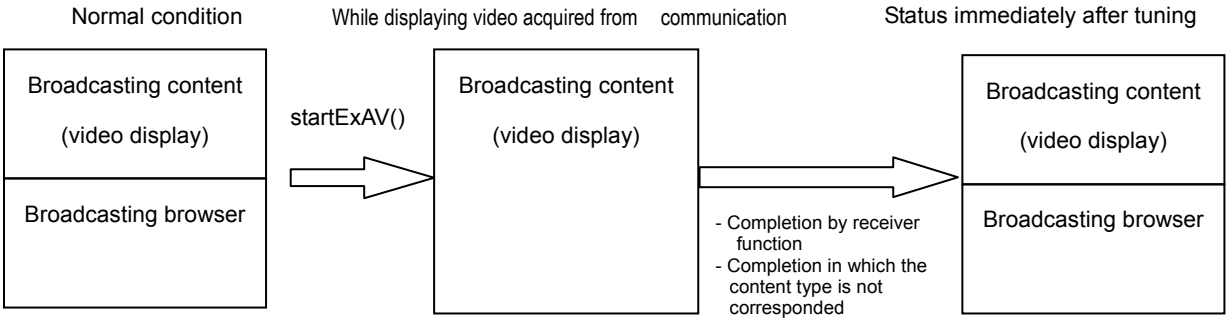


Figure 8-1 Display of video acquired through communication

8.1.7.7 Permission during communication generation and personal information dispatch

Permission from the viewer shall be acquired for BML browser to communicate. Permission acquisition is supposed to be conducted using receiver features, and the timing to acquire the permission is implementation dependent. However, some content may frequently access a single server within short duration, so it is necessary to implement keeping in mind that the service may break if permission is issued every time communication is generated. (Especially if transmitTextDataOverIP() function is called) As a specific implementation procedure, it is ideal to acquire permission at once when communication is generated after tuning, and not display the permission screen until another tuning or TV function completion is conducted.

Also, when the contents transmit private information, the broadcasting company (on the contents side) shall acquire user permission without fail independent from whether the receiver features has acquired permission or not.

8.1.8 Presentation of multimedia data

8.1.8.1 Multimedia data display area

In the multimedia data display area (visible area for data broadcasting), it is recommended to be able to display contents of 125 dots in the Y axis direction over an imaginary plane (6 lines in normal size characters). Leave enough area space to display contents of at least 50 dots in the Y axis direction over the imaginary plane (2 lines in normal size characters) when the data area is temporarily reduced for reasons such as simultaneous display, etc.

8.1.8.2 Exceptions in multimedia data display

As exceptions to the principle of simultaneous display of video, audio and multimedia data, it does not matter if the display of BML browser is not carried out even while video is being displayed is limited to the following cases:

- Cases where displaying video on a full screen, etc., when the visible area for data broadcasting specified in the previous clause cannot be placed in the screen after placing the video area in the display device.
- When the video is displayed in a direction which is not the original display direction with receivers that possess a function to rotate video 90 degrees from its original display direction in order to display video larger by using vertical display devices in a horizontal direction. (*)
- If a data carousel ES does not exist in the PMT, there is no need to take BML browser display area. However, if a data ES does exist in the PMT, a BML browser display area needs to be kept, independent from the fact a data service is conducted or not. If the data service is not temporarily conducted, an empty carousel shall be played in principle.
- For receivers which cannot secure other display areas for caption other than the video display area and visible area for data broadcasting etc., caption may be displayed in the visible area for data broadcasting.

* However, it is recommended to be able to display as much data broadcasting as possible even in such cases. Especially with terminals whose original display direction is in the vertical direction, it is preferable to display data broadcasting in the area below the video when video in a 16:9 aspect is displayed.

8.1.8.3 Optimizing presentation size

It is recommended to display the size of multimedia data (characters and images) on display by scaling to appropriate size depending on the pixel size of the display device, size, presentation size of the video, regardless of the pixel size of mono-media to transmit.

Contents shall be rendered on an imaginary plane of 240 pixels in the horizontal direction and 480 pixels in the vertical directions according to the next clause “Assumptions in contents creation”. Either scale the rendering results in the imaginary plane following receiver specific presentation procedures, keeping the vertical/horizontal ratio, or possess a function to review arbitrary parts of the imaginary plane by using the scrolling representation.

8.1.8.4 Changing the size of BML browser display area

The purpose of the function (setFullDataDisplayArea()), that enlarges BML browser display area, is to (temporarily) increase the number of displayed characters in a vertical direction according to the contents. Receivers may choose BML browser display area enlarged by these functions on their own, unit by unit, depending on the pixel size of display device. It is implementation dependent whether or not to change the video display area to BML browser display area by implementing these functions.

8.1.8.5 Image size possible for display

GIF, animation GIF, JPEG images of 240 dots in the horizontal direction and 320 dots in the vertical direction (240 dots for animation GIF) or smaller shall be able to display. They may not be displayed depending on the receiver if an image exceeding this restriction is used. Also, receivers may display images by scrolling depending on the vertical size of the display area even if the image size is within the restrictions.

If a still image resource size is within the specified size of “width” and “height” of the “img/object” element, it shall be displayed in the same size. In other cases, perform the clipping process or the reduction process keeping aspect ratio to display.

If display by the clipping process is performed, place the top left corner of the still image resource in the top left corner of the “img/object” element.

Displays within the “img/object” element range but outside of still image resource rendering range shall be processed as “transparent”. Also, display accordingly for cases where replacement with a different size still image by DOM is generated. Also, the display of animation GIF whose size is different from the “object” element size shall be implementation dependent.

8.1.8.6 Character fonts and possible number of characters for display

For “medium” sized characters, the font shall be of a dot number which enables 12 characters or more to be displayed in the horizontal direction. For example, it shall be a 20 or less dot font for receivers with display devices of 240 dots in the horizontal direction.

For “large” sized characters, the font shall be of the dot number which enables 8 characters or more to be displayed in the horizontal direction. For example, it shall be a 30 or less dot font for receivers with display device of 240 dots in the horizontal direction.

For “small” sized characters, the font shall be a dot number which enables 15 characters or more to be displayed in the horizontal direction. For example, it shall be a 16 or less dot font for receivers with display device of 240 dots in the horizontal direction. Also, “small” sized character fonts may not be able to be implemented due to the display device pixel size, and implementation may not be possible in such case.

8.1.8.7 Position of CSS and contents

Contents shall be positioned in the imaginary plane specified by the CSS. For receivers that cannot display as specified by the CSS due to the pixel size of the display device or aspect, the receiver shall optimize the display.

8.1.9 Guidelines for contents creation

Below are the guidelines to be followed when broadcasters, etc. create contents etc.

8.1.9.1 Pixel size assumed in contents creation

Pixel size for the imaginary plane for BML browsers assumed by contents creators shall be 240 pixels in the horizontal direction and 480 pixels in the vertical direction.

8.1.9.2 Character size controls and layout to be assumed in contents

For “medium” size characters, a 20 dot font is assumed, and a maximum of 12 characters are placed in the horizontal direction. For “large” size characters, a 30 dot font is assumed, and a maximum of 8 characters are placed in the horizontal direction, and for “small” size characters a 16 dot font is assumed and a maximum of 15 characters are placed in the horizontal direction.

If a “small” size font is specified, some receivers may display them in “medium”. Therefore, contents shall be considered to avoid collapse of presentation in such cases.

Table 8-1 Character size controls assumed in contents

Size	Dots	Number of displayed characters in the horizontal direction
large	30	8 characters maximum
medium	20	12 characters maximum
small	16	15 characters maximum

8.1.9.3 Size of images used

The maximum size for GIF, animation GIF, JPEG images is 240 dots in the horizontal direction and 320 dots in the vertical direction (240 dots for animation GIF). However, contents shall be created that assume display with scrolling in the vertical direction.

If the still image resource size is smaller than the “width” and “height” specification of ”img/object” element and concerned element’s “visibility” property is also “visible”, it is recommended not to layer elements that match all of the following conditions in the “width” and “height” property specified range of the concerned element. (Strongly suggested)

1) It is not “body” element.

2) It appears before the “img/object” element concerned in the document.

3) The “visibility” property value is “visible”.

4) It possesses display contents.

For example, the “div” element whose “border-style” property value is none and “background-color” property value is “transparent” is an element which does not posses display contents.

When these elements are layered and the “data” attribution of the “img/object” element concerned is re-written as a different size mono-media by DOM API, it is possible it will take some time before display depending on the receiver. Therefore, this point shall be kept in mind when authoring contents.

8.1.9.4 CSS

Always perform positioning specification by the CSS. Display is not performed accordingly if the CSS is not written.

8.1.9.5 Acquiring and releasing “*”, “#” and number keys

Once a document acquires a number key, “*” or “#” key by the “used-key-list”, one-touch tuning is disabled until it is released explicitly or the document transits. Therefore, the use of such keys shall be kept to a minimum. Also, if these keys are acquired due to necessity, release them as soon as entry is completed.

8.1.9.6 Notes on communication charges generated and information transmission

The permission for BML browsers to generate communication is assumed to be performed by the receiver, and thus it is unnecessary to reacquire permission on the contents side.

Also, when contents transmit private information, independent from whether the receiver features acquires permission or not, user permission shall always be acquired by the broadcaster (on the contents side).

Information acquired by getProgramID() and getActiveDocument() shall not be transmitted in connection to information that can identify the individual.

8.1.9.7 Permission when acquiring position information

The permission when acquiring position information is assumed to be performed by the receiver depending on necessity, and thus it is unnecessary to reacquire permission on the contents side. However, follow 8.1.9.6 when transmitting the position information acquired to interactive servers.

8.1.9.8 Permission when acquiring receiver-unique identifiers or televiewer-unique identifiers

The permission when acquiring receiver-unique identifiers or televiewer-unique identifiers is assumed to be performed by the receiver depending on necessity, and thus it is unnecessary to reacquire permission on the contents side. However, follow 8.1.9.6 when transmitting the receiver-unique identifier or the televiewer-unique identifier acquired to servers.

8.1.10 Display position for caption

It is recommended to display caption in a position that does not layer the video display area or data broadcasting area in display devices. However, for receiver that cannot secure display areas outside of the video display area and the data broadcasting area, caption may be displayed in the data broadcasting area by user operation. Also, if the display of caption in the video display area does not hinder the viewing the video, caption may be displayed over the video display area.

8.1.11 Guidelines for DOM operation

Refer to ARIB STD-B24 Vol. 2 Appendix 4 “5.3. Guidelines on behaviors of DOM”.
However, ignore the comments and do not generate the comment node.

8.1.12 Guidelines for external characters operation

External characters are not operated in C-profile.

8.1.13 Guidelines for ECMA script implementation

Refer below. In BML documents in which multiple script elements are recorded, the following restrictions are applied for the status where all scripts (scripts recorded in the resources specified by the src attribute of script elements, as well as scripts recorded inside script elements which do not possess the src attribute) are read.

- ARIB STD-B24 Vol. 2 Appendix 4 “5.4.1. Operation of script working environment”
- ARIB STD-B24 Vol. 2 Appendix 4 “5.4.2. Data type”, ”5.4.3. Effects on basic objects caused by data type restrictions” and “5.4.4. Operation rule of implementation-dependent portion”

8.1.14 Guidelines for operation of extended objects for broadcasting

8.1.14.1 Operational specifications for BinaryTable objects

- BinaryTable object size shall be 128 KB maximum.
- Number of lines shall be 2048 maximum.
- Maximum record length shall be 512 Bytes.
- Maximum field number shall be 32.
- Maximum comparison condition number shall be 4.

8.1.15 Guidelines for browser pseudo-object operations

Operations of browser pseudo-object are provided below. Follow,

- “Appendix 1 Operational guidelines”
- “Appendix 2 Operation guidelines for implementing basic services”

in ARIB STD-B24 Vol. 2 “XML-based multimedia coding scheme” for those not specifically recorded.

8.1.15.1 Operations of Ureg

Ureg values shall be valid within contents groups. Ureg values are initialized at browser startup and contents groups change (at tuning including media change).

Independent from ARIB STD-B24 Vol. 2 Appendix 2 “5.6.1. Operation of Ureg”, receivers shall not write the contents group identifier in Ureg[0] (in which the service_id is changed to hexadecimal characters in a "0xXXXX" format).

Ureg values shall be initialized at browser startup and contents group changes (at tuning including media change) by receiver.

The first value read after initialization is empty string.

8.1.15.2 Operations of Greg

Greg implementation is in principle necessary but implementation of the following manner may be conducted for receivers which do not receive other broadcasting media (media including data broadcasting) but only receive C-profile as exceptions:

- Writing value in Greg: Does not memorize the value. However, it does not generate an error.
- Reading Greg value: Returns empty string.

Follow ARIB STD-B24 Vol. 2 “7.6.16 Greg pseudo object properties”. Greg values are initialized at the startup of the broadcasting receiving function, and the value is always maintained while the broadcasting receiving function is valid. The first value to be read after initialization is empty string.

While broadcasting reception function is valid, the Greg value will be maintained even while presenting non-linked contents that are not supported by Greg on browsers other than BML browsers. In the same manner, it is ideal to maintain the Greg value while presenting media that does not support Greg.

If the Greg value cannot be maintained while presenting other media, etc. for some reason, receivers shall initialize the Greg value when presenting media that supports Greg for the first time after that.

8.1.15.3 Operation of EPG functions

In order to correspond to EPG functions, L-EIT reception is needed by the receiver. However, functions operated in C-profile are epgGetEventStartTime() and epgGetEventDuration(), so compatibility with 8-bit character codes is not necessary.

(1) epgGetEventStartTime()

Specifiable event by event_ref shall only be the event currently presented, and the start_time value shall be acquired by referring to L-EIT.

(2) epgGetEventDuration()

Specifiable event by event_ref shall only be the event currently presented, and duration value shall be acquired by referring to L-EIT.

(3) epgReserve()

Always specify startTime.

If there is another viewing reservation overlapping with the viewing reservation time by this function, the process shall be implementation dependent.

8.1.15.4 Interaction channel communication – Operation of TCP/IP

Refer to 8.3.6.3.

8.1.15.5 Operation of the operational control function

Follow STD-B24 for operations of functions not especially relevant to this clause.

Refer to 8.3.6.4 for utilization from communication contents.

(1) Operation of lockModuleOnMemoryEx() and unlockModuleOnMemoryEx()

Not only the module of the component currently being presented but also components that are not currently being presented can be locked by lockModuleOnMemoryEx(). This function is operated as follows:

- All ES (0x80 0x8B) modules that transmit Cprofile data carousels, independent from whether it is being presented or not, are always lockable using lockModuleOnMemoryEx(). 0x80 0x8B modules are simultaneously instructed to lock by lockModuleOnMemoryEx() from time to time, and thus it is necessary for receivers to be able to acquire 2ES modules at the same time. Operation to acquire a module at a time from 2 ES is not permitted. However, this excludes cases where the total of the module size acquired at the same time exceeds 256KB.
- Always reserve 256KB of available space in receivers to acquire modules from the carousel, and do not perform the module fixation process if 256KB memory space cannot be secured by implementing lockModuleOnMemoryEx() function.
- Receivers shall always reserve the available capacity specified by the argument which specifies the available capacity of this function. Do not perform the module fixation process if the memory area concerned cannot be secured by implementation. The capacity specified by the argument shall not include the previously mentioned 256KB space. Interpret this as 0B if the argument is omitted.
- If it is judged by the PMT that the module specified by lockModuleOnMemoryEx() does not exist (such as the ES to transmit the concerned module not being included in the PMT, etc.), lockModuleOnMemoryEx() ends with a return value of -3. If the module specified by

lockModuleOnMemoryEx() does not exist but cannot be judged by the PMT (such as an existing ES but not the module, etc.), lockModuleOnMemoryEx() ends with a return value of 1. And as soon as it is judged that the module concerned does not indeed exist, the ModuleLocked event is occured at status=-2.

- Even if the version of the module fixed in the memory is updated, the receiver does not automatically reacquire the module. The processes to detect module update, release the lock, and re-fix module to the memory shall be described in BML document.
- If transition to another document within the same service occurs before the completion of locking after executing lockModuleOnMemoryEx(), the locking operation will continue. In this case, the ModuleLocked event will be occured once this locking operation is complete in principle, but it may not happen depending on the document transition timing. If the data event of the ES, the lock object, is updated before the completion of locking after executing lockModuleOnMemoryEx(),the ModuleLocked event generates at status=-1, and the locking operation is not performed. If the lock object module is updated before the completion of locking after executing lockModuleOnMemoryEx(), the receiver locks the updated module.
- The procedure using lockModuleOnMemory() to update the presentation when updating the module that the object is referring to is described in ARIB STD-B24 Vol. 2 Appendix 1 “6.6.2 Relationship between update and module lock”, but the same effect can be acquired by using lockModuleOnMemoryEx() also.
- The procedure using lockModuleOnMemory() to update the presentation when updating the binary table that the object is referring to is described in ARIB STD-B24 Vol. 2 Appendix 2 “5.5.2.2 Behavior of BinaryTable”, but the same effect can be acquired by using lockModuleOnMemoryEx() also.
- The procedure using lockModuleOnMemory() to share JPEG images and binary tables within documents is described in ARIB STD-B24 Vol. 2 Appendix 2 “5.6.7 Operation of operational control functions”, but the same effect can be acquired by using lockModuleOnMemoryEx() also. It is described in Appendix 2 that sharing of resource documents is only valid within the document group, but it shall be valid within the contents group in this specification.
- In broadcasting reception status, the existence of modules can be confirmed by locking the module which includes the document at the transition destination by previously using lockModuleOnMemoryEx() before the transition to another BML document of another ES.
- If the ES that was transmitting the module locked by lockModuleOnMemoryEx() is deleted, it is implementation dependent on whether or not the module lock continues. It is

recommended to release the lock of such modules from the contents explicitly. Transition to such modules cannot be performed.

- The module locked by lockModuleOnMemoryEx() can be unlocked by unlockModuleOnMemoryEx() independent from whether the module is included in the carousel being transmitted at the moment.
  (Hereafter, refer to 8.3.7 for the meaning of link state)
- In the following cases, the lock by lockModuleOnMemoryEx( ) is unlocked.
  - When it is explicitly released by execution of unlockModuleOnMemoryEx() or unlockAllModulesOnMemory().
  - When the service tuning in presentation is finished.
  - When the data event in presentation is finished (includes deletion of the component in presentation)
  - When the entry component is vanished due to a PMT update
  - When transition to the entry component due to execution of quitDocument()
- In the following cases, the lock by lockModuleOnMemoryEx() continues.
  - When the data event is updated in components not in presentation
  - When the locked module version is updated.
  - When the ES that was transmitting the locked module becomes an empty carousel.
  - When transition to another component in the same service as the document in presentation is performed.
  - When transition to link state.
- lockModuleOnMemoryEx() is fixable (including own components) within the service where the BML document in presentation is transmitted. Therefore, it stays fixed in memory unless the service tuning switches or there is a data event update of the carousel, where the BML document in presentation is transmitted, is extracted.
  Table 8-2 is a summary of conditions to release lockModuleOnMemoryEx.

Table 8-2 Release conditions of lockModuleOnMemoryEx

	Module locked by using lockModuleOnMemoryEx
Data event update in components in presentation	Unlock
Data event update in components not in presentation	No change
Update date of version of locked modules	No change
Tuning operation	Unlock
Transition to another component in the same service as the document in presentation	No change
Transition to entry component to quitDocument() (including cases where quitDocument() was executed during entry	Unlock

component viewing)	
Transition to link state	No change
When another application was started up by using X_DPA_startResidentApp(), and also when BML browser continues to be presented	No change

Operation example 1

A typical operation example of lockModuleOnMemoryEx() is shown in Figure 8-2. Services in this example consist of 2 components of component A and component B. The rhombus symbols over the straight lines which stand for components show the points where data events are updated in the component. The explanation below is according to the figure.

(a) Component A is presented initially. At time t1, component B module is locked by lockModuleOnMemoryEx().

(b) At time t2, it is moves to the component B module locked in (a). At this point, the lock on the module previously locked continues.

(c) At time t3, the component A module is locked by lockModuleOnMemoryEx().

(d) At time t4, it is moved to a module of component A locked in (c). The 2 modules locked previously continue to be locked.

(e) At time t5, data event of component A in presentation is updated, thus releasing all the locks on the modules previously locked by lockModuleOnMemoryEx().

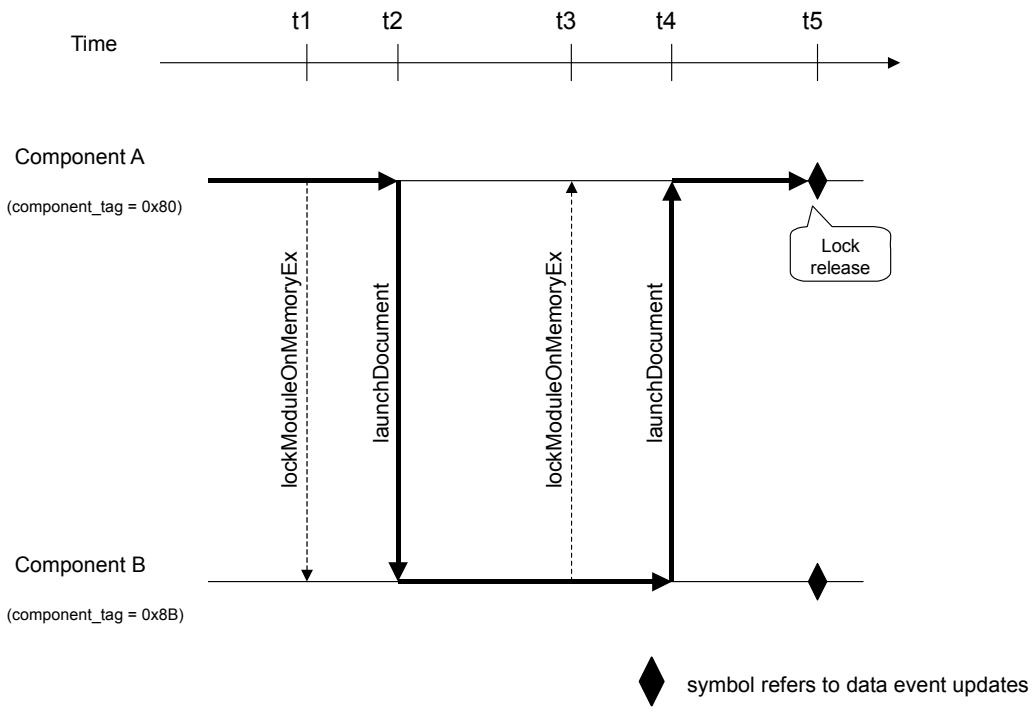


Figure 8-2 Example 1 of lockModuleOnMemoryEx operations

Operation example 2

A complicated operation example of lockModuleOnMemoryEx() is shown in Figure 8-3. As in operation example 1, service consists of 2 components of component A and component B. The explanation below is according to the figure.

(a) Component A is presented initially. At time t1, the presented module is locked by lockModuleOnMemoryEx().

(b) At time t2, the component B module is locked by lockModuleOnMemoryEx().

(c) At time t3, the module locked in (b) is moved to the component B module. At this point, the locks on the 2 modules previously locked continue.

(d) At time t4, the data event of component A is updated, but it does not affect the module already locked.

(e) At time t5, it is moved to previously presented module of component A. A new data event is already transmitted in component A, but the document presented by this transition is the document that was transmitted at the time of the lock in (a).

(f) At time t6, the component B data event is updated, but it does not affect the module already locked.

(g) At time t7, the data event of component A in presentation is updated, thus unlocking all the locks on modules previously locked by lockModuleOnMemoryEx().

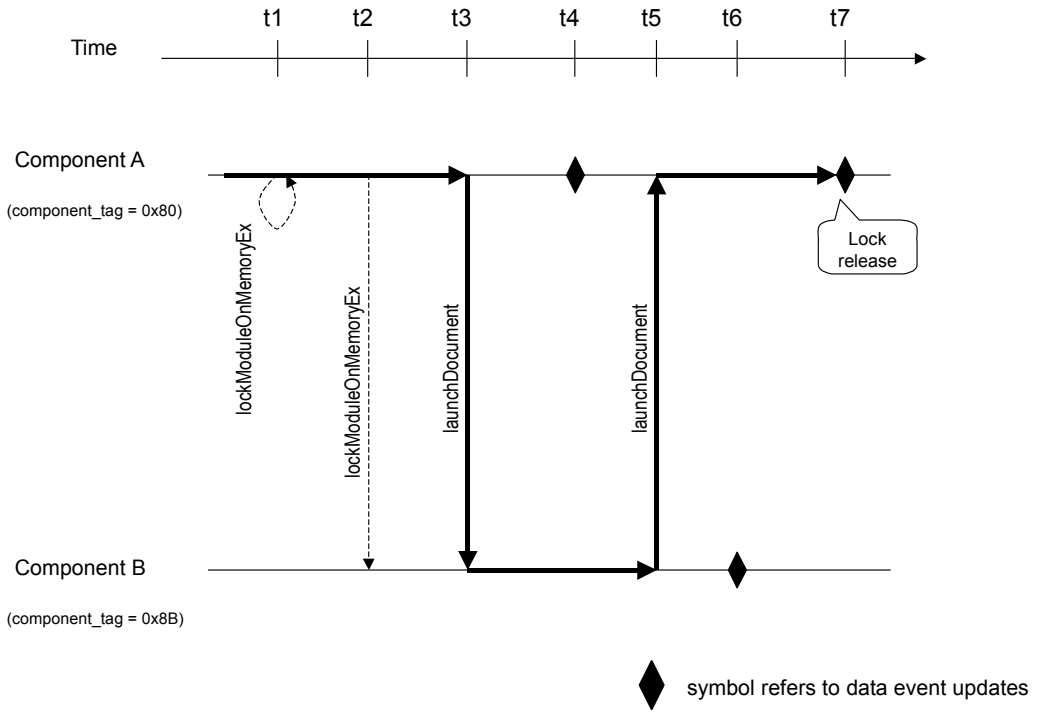


Figure 8-3 Example 2 of lockModuleOnMemoryEx operations

(2) Operation of launchDocument ()

Process of the second argument's transition effect based on the specification of transitionStyle is implementation dependent.

(3) Operation of launchDocumentRestricted()

This is not operated in C-profile.

(4) Operation of quitDocument()

Refer to "8.3.6.4 Operational control functions"

(5) Operation of getLockedModuleInfo()

The module name Array[n][0] (module name) of getLockedModuleInfo function's return value is a String object, and it is always stored in the:

/<component_tag>/<moduleName>

format.

Return value Array[n][1] and Array[n][2] are Number types.

If there is not even a single fixed or fixation-in-progress module, array with 0 length is returned, but receivers that return null are permitted as well.

When authoring contents, pay attention to the fact that one of the return value returns.

(6) Operation of detectComponent()

The component specified as the argument of detectComponent() is limited to the data component included in the service being presented. Follow the specifications in 8.2.3 to use the broadcasting contents, and use abbreviations starting with "/" for URI's that specify components. Follow the specifications in 8.3.10.4 to use in communication contents, and use absolute URI's starting with "arib-dc://-1.-1.-1" for URI's that specify components.

(7) Operation of getProgramRelativeTime()

Returns the starting time (start_time) of programs recorded in L-EIT and current time acquired by receivers per second.

(8) Operation of getActiveDocument()

Return abbreviated URI values (relative path starting with "/").

(9) Operation of getBrowserStatus()

Refer to Appendix 6 for argument operations.

(10) Operation of getResidentAppVersion()

This is necessary for function to acquire a TVlink list application version. In the return values from receivers with TVlink list implementation calling this function, specifying BookmarkList as appName, Array[4] returns empty strings.

ComBrowser is not specified in appName. If the TVlink list function is not implemented in

receivers that do not possess communication functions, the return value after calling this function, specifying BookmarkList as appName, is null.

If the specified appName is not compatible, null is returned as the return value. (Note)

(Note) There is no appName specifications of operations other than BookmarkList in current operations because ComBrowser is not supposed to be specified, but these specifications are performed in case appName is added to specifications in the future.

(11) Operations of startResidentApp()

This is not operated in C-profile

(12) Operations of setFullDataDisplayArea() (option)

When receivers are specified full screen view (mode=1), the maximum area possible for data display (position and size of the area are up to receiver) is given to data display. At this point, it is not necessary for video to be displayed. However, it is recommended for audio to continue.

This function is an option but implementation is recommended.

(13) Operations of getBrowserSupport()

Arguments are operated as follows.

- "DPACpro" is specified in sProvider. Do not specify "ARIB".
- The corresponding relationships between the value of additional info, when the function name is "APIGroup", and the extended function are shown in the table below. Extended functions where the additionalinfo is blank are not operated.

Table 8-3 Value of additional info

	Function	Additional info
Ureg related functions		
	Ureg[]	Misc.Basic
Greg related functions		
	Greg[]	Misc.Basic
EPG functions		
	epgGetEventStartTime()	EPG.Basic
	epgGetEventDuration()	EPG.Basic
	epgTune()	
	epgTuneToComponent()	
	epgTuneToDocument()	
	epgIsReserved()	
	epgReserve()	EPG.Ext
	epgCancelReservation()	EPG.Ext
	epgRecIsReserved()	
	epgRecReserve()	
	epgRecCancelReservation()	
Event group index functions		
	grpIsReserved()	
	grpReserve()	

	Function	Additional info
	grpCancelReservation()	
	grpRecIsReserved()	
	grpRecReserve()	
	grpRecCancelReservation()	
	grpGetNodeEventList()	
	grpGetERTNodeName()	
	grpGetERTNodeDescription()	
	epgXTune()	
Series reservation functions		
	seriesIsReserved()	
	seriesReserve()	
	seriesCancelReservation()	
	seriesRecIsReserved()	
	seriesRecReserve()	
	seriesRecCancelReservation()	
Permanent memory		
	readPersistentString()	
	readPersistentNumber()	
	readPersistentArray()	Persistent.Basic
	writePersistentString()	
	writePersistentNumber()	
	writePersistentArray()	Persistent.Basic
	copyPersistent()	
	getPersistentInfoList()	
	deletePersistent()	
	getFreeSpace()	
Functions for controlling access-controlled non-volatile memory areas		
	isSupportedPersistentType()	
	setAccessInfoOfPersistentArray()	
	checkAccessInfoOfPersistentArray()	
	writePersistentArrayWithAccessCheck()	
	readPersistentArrayWithAccessCheck()	
Interaction channel communication		
Interaction channel communication– Delayed calls		
	registerTransmission()	
	registerTransmissionStatus()	
	getTransmissionStatus()	
	setDelayedTransmissionDataOverBasic()	
Interaction channel communication – BASIC procedures		
	connect()	
	disconnect()	
	sendBinaryData()	
	receiveBinaryData()	
	sendTextData()	
	receiveTextData()	
Interaction channel communication - TCP/IP		
	setISPParams()	
	getISPParams()	
	connectPPP()	
	connectPPPWithISPParams()	
	disconnectPPP()	
	getConnectionType()	Com.IP.GetType
	isIPConnected()	Com.IP
	saveHttpServerFileAs()	
	saveHttpServerFile()	
	sendHttpServerFileAs()	
	saveFtpServerFileAs()	
	saveFtpServerFile()	

	Function	Additional info
	sendFtpServerFileAs()	
	sendTextMail()	
	sendMIMEMail()	
	transmitTextDataOverIP()	Com.IP.Transmit
	setDelayedTransmissionData()	
	getTransmissionStatus()	
	getTransmissionResult()	
	setCacheResourceoverIP()	
Interaction channel communication – Acquisition function in delayed call status common in BASIC procedures and IP connection		
	getDelayedTransmissionStatus()	
	getDelayedTransmissionResult()	
Interaction channel communication – Function to acquire line connection status		
	getPrefixNumber()	
Interaction channel communication – Mass calls reception service		
	vote( )	
Interaction channel communication – Encrypted communication using CAS		
	startCASEncryption()	
	transmitWithCASEncryption()	
	endCASEncryption()	
Interaction channel communication – Communication by encrypted code not using CAS		
	setEncryptionKey()	
	beginEncryption()	
	endEncryption()	
Operational control function		
	reloadActiveDocument()	Ctrl.Basic
	getNPT()	
	getProgramRelativeTime()	Ctrl.Basic
	isBeingBroadcast()	
	lockExecution()	
	unlockExecution()	
	lockModuleOnMemory()	
	unlockModuleOnMemory()	
	setCachePriority()	
	getTuningLinkageSource()	
	getTuningLinkageType()	
	getLinkSourceServiceStr()	
	getLinkSourceEventStr()	
	getIRDID( )	
	getBrowserVersion()	Ctrl.Basic
	getProgramID()	Ctrl.Basic
	getActiveDocument()	Ctrl.Basic
	lockScreen()	Ctrl.Basic
	unlockScreen()	Ctrl.Basic
	getBrowserSupport()	Ctrl.Basic
	launchDocument()	Ctrl.Basic
	launchDocumentRestricted()	
	quitDocument()	Ctrl.Basic
	launchExApp()	
	getFreeContentsMemory()	
	isSupportedMedia()	
	detectComponent()	Ctrl.Basic
	lockModuleOnMemoryEx()	Ctrl.Basic

	Function	Additional info
	unlockModuleOnMemoryEx()	Ctrl.Basic
	unlockAllModulesOnMemory()	Ctrl.Basic
	getLockedModuleInfo()	Ctrl.Basic
	getBrowserStatus()	Ctrl.Basic
	getResidentAppVersion()	Ctrl.RAVersion
	isRootCertificateExisting()	
	getRootCertificateInfo()	
	startResidentApp()	
	getDataDisplayAreaSize()	
	setFullDataDisplayArea()	Ctrl.MobileDisplay
Receiver sound control		
	playRomSound()	RomSound.Basic
Timer functions		
	sleep()	
	setTimeout()	
	setInterval()	Timer.Basic
	clearTimer()	Timer.Basic
	pauseTimer()	
	resumeTimer()	
	setCurrentDateMode()	Timer.DateMode
External character functions		
	loadDRCS()	
	unloadDRCS()	
External device control functions		
	enumPeripherals()	
	passXMLDocToPeripheral()	
Other functions		
	random()	Misc.Basic
	subDate()	Misc.Basic
	addDate()	Misc.Basic
	formatNumber()	Misc.Basic
Caption display control functions		
	setCCStreamReference()	
	getCCStreamReference()	
	setCCDisplayStatus()	
	getCCDisplayStatus()	
	getCCLanguageStatus()	
Directory operation functions		
	saveDir()	
	saveDirAs()	
	createDir( )	
	getParentDirName()	
	getDirNames()	
	isDirExisting()	
File operation functions		
	saveFile()	
	saveFileAs()	
	getFileNames()	
	isFileExisting()	
File input/output functions		
	writeArray()	
	readArray()	
Inquiry functions		
	getDirInfo()	
	getFileInfo()	
	getCarouselInfo()	
	getModuleInfo()	

	Function	Additional info
	getContentSource()	
	getStorageInfo()	
Data carousel storage functions		
	saveCarouselAs()	
	saveCarousel()	
	saveModuleAs()	
	saveModule()	Storage.Ext
	saveResourceAs()	
	saveResource()	Storage.Ext
Bookmark control functions		
	writeBookmarkArray()	
	readBookmarkArray()	
	deleteBookmark()	
	lockBookmark()	
	unlockBookmark()	
	getBookmarkInfo()	
	getBookmarkInfo2()	
	startResidentBookmarkList()	
Print related functions API – Print basic functions		
	getPrinterStatus()	
	printFile()	
	printTemplate()	
	printUri()	
	printStaticScreen()	
Print related function API – Memory card related		
	saveImageToMemoryCard()	Print.MemoryCard1
	saveHttpServerImageToMemoryCard()	Print.MemoryCard2
	saveStaticScreenToMemoryCard()	
Digital terrestrial television broadcasting specific functions		
	X_DPA_mailTo()	Xdpa.mailTo
	X_DPA_startResidentApp()	Xdpa.RAStart
	X_DPA_phoneTo()	Xdpa.phoneTo
	X_DPA_getRcvCond ()	Xdpa.RcvCond
	X_DPA_getCurPos ()	Xdpa.CurPos
	X_DPA_saveExAppFile()	Xdpa.saveExApp
	X_DPA_startExAV()	Xdpa.startExAv
	X_DPA_stopExAV()	Xdpa.stopExAv
	X_DPA_tuneWithRF()	Xdpa.tuneRF
	X_DPA_writeSchInfo()	Xdpa.SchInfo
	X_DPA_getComBrowserUA ()	Xdpa.ComBrowserUA
	X_DPA_writeAddressBookInfo()	Xdpa.AddressBook
	X_DPA_launchDocWithLink()	Xdpa.launchWithL
	X_DPA_chkAVtype()	Xdpa.chkAV
	X_DPA_getIRDID()	Xdpa.getIRDID
	X_DPA_writeCproBM()	Xdpa.CproBM

Below are specifiable value combinations for arguments of getBrowserSupport().

functionname	additionalinfo	Operations of getBrowserSupport()
ResidentApp	"ComBrowser"	If a communication browser is installed in the receiver as a receiver's native application, return 1
	"Bookmark"	If a TV link list function is installed in the receiver as a receiver's native application, return 1
	"JapaneseInput"	If a character input function (refer to 3.6) is installed in the receiver as a receiver's native application, return 1
WriteCproBM	"BMtype02"	If the receiver can write TVlink of CproBMtype=2, return 1
	"BMtype03"	If the receiver can write TVlink of CproBMtype=3, return 1
	"BMtype04"	If the receiver can write TVlink of CproBMtype=4, return 1
OSDPixel size	"240x480"	If the receiver possesses 240 x 480 coordinate area as the imaginary plane for data broadcasting, return 1
BMLversion	"major number. minor number"	If the browser supports playback of BML documents in the version specified by additionalinfo, return 1. If Additionalinfo is omitted, "12.0" is considered to be specified.
MediaDecoder	"scheme name" ,"media type" Refer to ARIB STD-B24 Vol. 2 (1/2) Annex C for scheme name and media type. If the "media type" is "audio/X-arib-mpeg2-aac", specify "sampling frequency" (Hz unit) as the third additionalinfo. (If omitted, "48000" is considered to be specified)	If the browser possesses a decoding function of mono-media that is identified by media type and scheme name, return 1. If the media type is "audio/X-arib-mpeg2-aac", return 1 only when it corresponds to specified sampling frequency.
APIGroup	Extended function group specification Refer to table 8-3 for correspondence to extended function for each broadcaster and extended function group specification.	If it is possible for the browser to implement all of extended function group that was specified by argument's extended function group specification, return 1.
TransmissionProtocol	"application", "HTTP" [, HTTP version]	Return 1 if it possesses a interaction channel communication function using HTTP, whose version is the version specified in [HTTP version], in the priority connection line's connection type used by the browser. It is interpreted as 1.0 if the version number is omitted.
	"application", "TLS" [, TLS version]	Return 1 if it possesses a secure interaction channel communication function by TLS, in the priority connection line's connection type used by the browser. It is interpreted as 1.0 if the version number is omitted.
Storage	"cachesize", size	Return 1 if the browser possesses content memory specified by size. (Note) Size unit shall be 1024 bytes. For example, specify "512" if 512Kbytes.
BookmarkButton		Return 1 if the receiver possesses a TVlink button.
AudioFile	filesize	Return 1 if the audio content in the file type that is the size (in byte unit) specified by filesize is available for playback.

(14) Operation of lockScreen()

When lockScreen() is started up, the presentation screen changes from DOM and the API is stopped until unlockScreen() is started up. However, when the event handler ends by lockScreen() without starting up unlockScreen(), a presentation screen change is performed immediately after it ends.

If multiple values are set within the same attribute or multiple method callings are performed, only the last change shall be valid.

If lockScreen() is called multiple times before unlockScreen() is called, calls after the first one will normally be ignored. Nesting of the lock process is not performed. It is implementation dependent whether or not to stop the rewriting of the presentation frame of animation GIF during the operation of lockScreen().

Presentation and operation of object elements whose type attribute starts with audio are not influenced by the screen lock status.

(15) Operation of unlockScreen()

If unlockScreen() is called without lockScreen() being called, it will normally be ignored. Even when lockScreen() is being called multiple times, the lock can be released by starting up unlockScreen() once.

(16) Operation of setInterval()

The first argument, with the same description as the calling of functions as the event handler, is controlled operationally only in "func();" type. The maximum timer number that can be set simultaneously is 4, and the setting shall be in 100msec units. Depending on the number of interrupting seconds, it is permitted that accuracy corresponds to each receiver.

8.1.15.6 Non-volatile memory functions and operation of the data carousel storage function

(1) Operation of writePersistentArray()

Period, as an omittable argument, does not require operation. Lifetime control of the written data is not performed, and it is always overwritable. Specification of structure is the same as BinaryTable objects, but the following restrictions apply.

Useable field type	Corresponding data type	Operation restriction
B	Boolean	Error if byte aligning does not match
I	Number	Cut off upper bit if it is less than 32 bit. Error if value over 32 bit is set.
U	Number	Cut off upper bit if less than 32 bit. Error if a value over 32 bit is set. Error if a negative value is written.
S	String	Cut off if it exceeds the specified character length. Add space(0 x 20) if it is too short. The variable length string is specified as S:1V.
P	--	

If the actual data and BinaryTable data type are different, utilize the type conversion specifications in ARIB STD-B24 Vol. 2 Appendix 4 5.4.2.2, but return an error if it is not appropriate. Also, if the array data is not sufficient for the field number specified by structure, it is an error. Operations are not guaranteed with arrays of two or more dimensions.

(2) Operation of readPersistentArray()

Operation of structure shall be the same as the writePersistentArray(). Operations are not guaranteed if reading was performed with a structure that is not the structure used in the writePersistentArray().

For operation of the data carousel storage function, T.B.D.

8.1.15.7 Operation of functions related to TVlinks

When recording C-profile contents (CproBMtype=1) as a TVlink, there are cases in which broadcasting is not receivable when calling the concerned link contents. Therefore, consider the following possibilities when authoring contents.

1. Time handled with ECMA Script's Date object is not accurate, or error is returned.
2. Possibility that event EventMessageFired does not occure.
3. Possibility that event ModuleUpdated does not occure.
4. Possibility that the extended function returns an error because broadcasting is not receivable.

8.1.15.8 Operation of special function for terrestrials

(1) Operation of X_DPA_mailTo() (optional)

Delivers the address and contents to the argument and transmits e-mail. The application that the receiver possesses is what actually transmits the e-mail, and transmission operation and permission procedure are implementation dependent.

(2) Operation of X_DPA_phoneTo () (optional)

This calls the telephone number specified by the argument. Follow "8.1.4 Prohibition of mixed display and simultaneous display" when this function is called, and pay extra attention to mixed display when using the TV telephone function. The calling operation and permission procedure are implementation dependent.

(3) Operation of X_DPA_ getRcvCond () (optional)

This acquires communication status in the communication line. The layout procedure and permission procedure, when the return value status of this function is different from the

classification of the radio receive condition within the communication function, are implementation dependent.

(4) Operation of X_DPA_ getCurPos () (optional)

This returns the current location information of the receiver based on the receiver's location information acquisition function, using Global Positioning System geodetic information using GPS satellites and information acquired from base stations established for mobile phones, etc. It is recommended to return the latitude and longitude of decimal number writing based on the global positioning system for the return value of this function. Permission, etc. when calling this function is implementation dependent.

(5) Operation of X_DPA_saveExAppFile() (T.B.D.)

(6) Operation of X_DPA_startExAV () (optional)

To display streamed video acquired via communication using this function, stop the presentation of video and audio of broadcasting, in principle, and display the video via communication in the display area of the broadcasting video. However, it is recommended to continue to present data broadcasting.

Receivers shall consider the viewers so that they can identify that it is video acquired via communication.

The return value is not returned if BML browser stops when starting up the AV playback application, etc. This function is useable in broadcasting contents and contents in link state.

Follow "8.1.7.6 Display of videos acquired from communication" when this function is called.

(7) Operation of X_DPA_stopExAV () (optional)

This stops the presentation of video sound streamed via communication presented by the (X_ DPA _startExAV ()) function. It is recommended to return the display back to broadcasting video and audio when the video sound stream via communication is complete. With receivers that stop BML browser when starting AV playback applications, etc., nothing other than returning NaN is performed if this function is implemented.

(8) Operation of X_DPA_tuneWithRF() (optional)

When the service in current tuning is re-selected by this function, BML browser performs the same operation as tuning. Only broadcasting contents shall be presented on the screen immediately after tuning, according to "8.1.3 Operation during the startup of the TV reception function and tuning". It is recommended to continue presenting the video section without sudden cut-off's, noise generation, etc.

The operation when tuning fails upon the calling of this function is implementation dependent.

(9) Operation of X_DPA_writeSchInfo() (optional)

This starts up the schedule management application within receivers and delivers the specified information to the argument. Some schedule contents, which exceeds the maximum character number restriction in each item of schedule in the receiver features, may not be saved. It is recommended to save the schedule title as the schedule book's item title, but it is implementation dependent to choose which column to allot. An alarm can be set to ring at specified time and date, but the alarm sound correspondence and sound color settings are implementation dependent. The function is defined, supposing to notify programs. However, this function's implementation does not need to be registered among the schedule book which the user normally writes in, and can be implemented on its' own.

The operation upon schedule management application startup and information saving procedure, etc., are implementation dependent.

(10) Operation of X_DPA_writeAddressBookInfo() (optional)

The startup of the address book within receivers and the address book control function, and transmission information such as the name, telephone number, e-mail address, etc. are specified in the function to the address book. Mixed existence with address books that already exist as receiver features is not necessary.

For mixing the existence of the information of concerned functions in existing address books, is implementation dependent for whether or not to register items that do not match.

(11) Operation of X_DPA_startResidentApp ()

This is mandatory for functions to startup TVlink list. Other receiver's native applications startup is optional. Also, for receivers that do not possess the communication function, implementation of the TVlink list function is not necessary. If this function is called, NaN is returned as the return value.

(12) Operation of X_DPA_getComBrowserUA () (optional)

If multiple browsers are installed in a receiver, all browser information returns to 2 dimensional array. To connect to a proprietary network of the carrier such as the communication browser, mobile phone, etc. concerned, the communication company's ID shall be returned as the maker ID. Contents utilize this function in order to identify communication companies.

(13) Operation of X_DPA_launchDocWithLink ()

Unlike the launchDocment() function, the base URI directory is changed if used in link state. This changes the base URI directory to the URI directory specified in the function, and transmits it.

(14) Operation of X_DPA_chkAVtype () (optional)

This is a function that confirms the type of video sound from communication contents. The receiver confirms whether it corresponds to the video sound stream type that was specified as the function, before the (X_DPA_startExAV ()) function implementation.

(15) Operation of X_DPA_getIRDID ()

This is a function to acquire the ID that identifies the receiver and viewer. It can acquire the receiver-unique identifier and the viewer specific identifier by argument specification. The type of identifier, permission for function call, etc., are implementation dependent.

8.1.15.9 Operation of print functions (optional)

Printing function

Refer to and based on ARIB STD-B24 Vol. 2 “7.6.17 Functions for printing” and Vol. 2 Appendix 1 “Guideliines on functions for printing” for functions and specifications regarding printing.

Extended API group

Printing functions are receiver implementation options. Therefore, when printing related functions are utilized in contents, examine whether printing related processing is available in the receiver by getBrowserSupport(), and call the printing related function only when the processing are available.

Printing related functions are divided into the following groups:

A) Function group to print using printers

- getPrinterStatus()
- printUri()

B) Function group to store data for printing in receiver’s internal memory, memory card, etc.

- saveImageToMemoryCard()
- saveHttpServerImageToMemoryCard()
- saveStaticScreenToMemoryCard()

Receivers that support printing functions shall extract each function implementation status by specifying “APIGroup” in the function name of getBrowserSupport().

API in A) is not operated in C-profile.

API in B) is useable even when the printing device is offline. However, saveStaticScreenToMemoryCard() is not operated. The following procedures are options to print the saved data within a receiver or memory card.

- Use the printing function by outputting the printing data saved within a receiver or memory card that was implemented as receiver features to the printer when the printer is on-line.
- Also, in some cases, a memory card can be utilized as the bridge media to deliver data to printing devices.

Table 8-3 provides the extended function group specification to specify as additionalinfo when functionname is "APIGroup".

Printing data format

For still image files (JPEG) for printing, in case they are to be shared by both display and printing in the BML, and there is a case only for printing. For each case, operate as follows.

	Still image file	Operation
Shared by both display and printing	JPEG	Refer to 5.2.1
Printing by itself	JPEG	ISO/IEC10918-1 base line, JFIF (Jpeg File Interchange Format) and Exif Maximum pixel size 640x480 Maximum file size 256KBytes Sampling 4:2:0 or 4:2:2 Pixel aspect ratio 1:1 (square pixels)

- Still image file format specifiable by saveHttpServerImageToMemoryCard()

  The still image file format that can be specified by this function is JPEG, specified in the above table. Each file name extension is “jpg”. The maximum size for the still image files is specified in the above table.

Supplemental items regarding each printing related API

- URI specified by saveHttpServerImageToMemoryCard()

  For the URI, which is specified by saveHttpServerImageToMemoryCard(), only resources that are described with http:// or https:// and are 256 Bytes or less are specifiable.

Presentation by receiver

Even when implementing the store functions (saveImageToMemoryCard() and saveHttpServerImageToMemoryCard()) of printing data on memory card, the BML contents shall present a reminder message for memory card insertion, and processing overlapping file names, etc. The storage directory name within the receiver or memory card is implementation dependent.

Even during the display of messages etc., by the receiver within the printing function, the process from document cancellation to re-presentation will be executed if there is a data event update.

In the following cases, the receiver system shall delete the message/interface and discontinue the process.

- When a data event update is occured in the currently viewed ES.
- When tuning

When http:// (or https://) is specified in saveHttpServerImageToMemoryCard function, the receiver acquires printing data within the function via communication. Therefore, there may be cases where it takes some time from the function call until the return. If an event is generated while the receiver is acquiring the printing data via communication, the event will be added to the interrupting queue. However, it will not be implemented until it is returned from the function, so be sure to consider this point.

8.1.16 Built-in objects

Time handled by the Date object shall be time corrected by the TOT or another procedure.

The Date object shall be able to acquire values down to figures of 0.01 seconds. It is recommended to avoid implementation that returns figures 0.1 seconds and 0.01 seconds in absolute value. (Note) Even though the accuracy of the Date object's absolute value is assumed to be about the accuracy level of TOT, it is recommended that the accuracy in the difference between 2 Date objects is plus/minus 0.1 seconds or less. It is only necessary for editing by TOT, etc. to be performed at the time of tuning.

Time handled by the ECMA Script Date object shall be the time JST (UTC + 9hours) that does not include summer daylight savings time offset in the calculation. If local time with summer daylight savings time is offset, consideration is necessary in order to display the current time etc., and local time shall be acquired by adding the time offset by using addDate(), etc., to the time acquired the Date() function in the contents.

Time handled by the Date object within the communication contents display shall follow the time information which the broadcasting contents were referring to before it changed to communication contents.

(Note) The reason that acquisition of a figure 0.01 seconds is necessary is (like when acquiring the time difference from answering start time and time the button is pushed in a quiz game, for example) because the application to acquire the relative time difference between 2 points which are relatively close is assumed.

8.1.17 Other restrictions

Refer to ARIB STD-B24 Vol. 2 Appendix 4 “5.7. Other restrictions”

8.2 Transmission, reference and name space of content

8.2.1 Scope mapping to the transmission system

Follow ARIB STD-B24 Vol. 2 Appendix4 “6.1. Scope mapping to the transmission system”.
Refer to 8.3.7 for communication contents.

8.2.2 Restrictions when a mono-media is referred across a different media

In C-profile, reference of mono-media, etc. is not performed other than in the following cases.
Refer also to 8.3.9 for communication contents.

- Tuning by functions for tuning
- Event message reference when the reference origin is C-profile linked contents and the reference destination is broadcasting contents
- To subscribe the event in th URI beginning with "arib-dc://-1.-1.-1" in C-profile linked contents (refer to table 7-14)

8.2.3 Operations of name space

Follow ARIB STD-B24 Vol. 2 Appendix 4 “6.3 Namespace”. Refer to 8.3.10 for C-profile linked contents.

The following specifications in this volume are added for broadcasting contents.

- Reference for other services shall only be available in the following extended functions for broadcasting.
  X_DPA_tuneWithRF(), epgReserve() and epgCancelReservation()
- Other than the following cases, refer to the abbreviated form (ARIB STD-B24 Vol. 2 9.2) for the name space description when specifying broadcasting contents within BML documents.
  - Reference to other services
  - Argument of the extended function for broadcasting whose argument is an event specification
  - Reference of an event message transmitted by broadcasting from C-profile linked contents
- Always omit event_id, excluding argument for the extended function for broadcasting whose argument is an event specification
- As described in ARIB STD-B24 Vol. 2 “9.2.5.1 Identification of currently selected broadcasting service on receiver”, currently selected service is considered to be specified if

"arib://-1.-1.-1" was specified as the service name. Here, currently selected service refers to the service currently being received if it is in broadcasting reception status.

- For the “href” attribute of the “a” element and launchDocument(), the BML document that is transmitted with the component which was included in the same service as the BML document being presented or C-profile linked contents placed in a server can be specified. For the “action” attribute for “form” element, C-profile linked contents placed in servers is specifiable. For X_DPA_launchDocWithLink(),C-profile linked contents placed in servers is specifiable.
- Operation of #fragment is available, according to ARIB STD-B24 Vol. 2 Appendix 4 “6.3 Namespace”, but restrictions specified in 8.2.3.4 are established.

8.2.3.1 Restriction for the number of resources which are managed by receiver

- The maximum number of resources (including both broadcasting contents and C-profile linked contents) simultaneously securable in receiver’s content memory is 256. For the restriction, it should keep the total number of resources (which possess specific name spaces) in a single data event period within 256. However, if the resource number can be limited within the limitation above at one time when authoring, the total number of resources in a data event period may be 256 or more.
- If, against the above restriction, fixation of resources that exceed the above number were specified due to lockModuleOnMemoryEx(), etc., the receiver may not perform this.
  Also, resources refer to the following two here.
  - Resource directly mapped in modules
  - Resource stored in modules in HTTP/1.1’s entity format

8.2.3.2 Name space regarding multiple ES module locks

- Regardless of ES being presented, all data carousel modules within the same service can be specified as an argument of lockModuleOnMemoryEx().

8.2.3.3 Name space regarding multiple ES module version watch

- Modules that watch module version updates (module_ref whose type attribute is “bml:beitem” element of ModuleUpdated) is also specifiable for any module in a data carousel of C-Profile within the same service.

8.2.3.4 Reception operation and contents guidelines in #fragment operations

- It is recommended to present documents without reloading if only #fragment is used in the “href” attribute of the “a” element and launchDocument() and transition within the same document is specified.
  (ex: browser.launchDocument("#top", "cut");)

- However, it is recommended to reload in cases like below, even if transition within the same document was specified.
  (ex: browser.launchDocument("startup.bml#top", "cut");))
- Follow the specification and present the document if transition to another document was specified using #fragment by means of specification of X_DPA_launchDocWithLink() and the “href” attribute of the “a” element and launchDocument().
- Presentation that includes elements specified by #fragment if transitioned by URI specifications that use #fragment., the presentation is implementation dependent.

8.2.4 Reference guidelines of contents transmitted by components different from the BML document being presented

- In C-profile, resources which are transmitted by components different from BML document being presented can be referred to. In such occasions, conduct locking by lockModuleOnMemoryEx().
- With resources that will be required immediately after BML document presentation starts, such as the JPEG referred to as the background-image of body, it is necessary to pay attention to make sure locking completely by lockModuleOnMemoryEx() is performed before BML document presentation upon creating contents.
- Specifications below are specially established for reference of ECMAScript and CSS, that are transmitted as independent resources.
  - When referring to an ECMAScript and CSS from the BML document of a component different from the component in which they are transmitted, perform locking of ECMAScript and CSS by lockModuleOnMemoryEx(), and perform transition to the BML document after the completion of locking. Operation when document transition was performed without completing locking is implementation dependent.
  - When referring to the CSS and ECMAScript that are transmitted in components different from BML document being presented, there are cases that CSS and ECMAScript are not referable to from the vanished ES, etc. In that case, failures as above can be avoided by transmitting CSS data and ECMAScript data in a data entry component.
  - Even when referring to CSS and ECMAScript transmitted in the BML document being presented from the BML document, it is recommended to lock ECMAScript and CSS prior to the BML document transition concerned and perform transition after the completion of locking.

8.3 Operation of C-profile communicating content

8.3.1 Guidelines regarding presentation for C-profile communicating contents

Same as 8.1 ”Guidelines for presentation”

8.3.2 Guidelines for operations of external fonts in C-profile communicating contents

Same as 8.1.12 “Guidelines for external characters operation”

8.3.3 Operation of DOM in C-profile communicating contents

Same as 8.1.11 “Guidelines for DOM operation”

8.3.4 Operation of ECMAScript implementation script in C-profile communicating contents

Same as 8.1.13 “Guidelines for ECMA script implementation”

8.3.5 External objects for broadcasting in C-profile communicating contents

Same as 8.1.14 “Guidelines for operation of extended objects for broadcasting”

8.3.6 Operation of browser pseudo-objects in C-profile communicating contents

The behaviourof the extended function for broadcasting which is in browser pseudo-objects in C-profile communicating contents differs depending on the status (data broadcasting reception status and link status) of the receiver. Refer to 8.3.7 for the basic concept of data broadcasting reception status and link status. Actual function operation is specified from Table 8-4 to Table 8-15. The meanings of “O”, “O(*1)”, “O(*2)”, “O(*3)”, “X” and “-” are specified below.

“O”	Basic function in this volume.
“O(*1)”	Optional function in this volume. Therefore, if these functions are used in contents, inspect the availability of this function in receiver by the getBrowserSupport() function. Only when it is available for processing can this function be called.
“O(*2)”	Basic function in this volume in principle. However, implementation of these functions is not necessary only for receivers that cannot utilize the communication function from BML browsers. When those functions is called, the value for the failure is returned as the return value.
“O(*3)”	Necessary for receiver that possess a function to playback the partial TS. If these functions are used in contents, inspect the availability of processing this function in receivers by the getBrowserSupport() function. Only when it is available for processing can this function be called.

"X" Implementation is prohibited in broadcasting status, link status and browser for the C-profile contents on the internets. If the function is called, receiver performs a failure behaviour. Refer to 8.3.11.4 for specifications in this volume of failure behaviours.

"-" Neither basic functions nor optional functions in this volume. If the function is called, an error is ocurred in the receiver.

8.3.6.1 "Ureg" function

Operations of "Ureg" functions in C-profile linked contents are specified in Table 8-4. Refer to 8.1.15.1 and 8.1.15.2 for operations in data broadcasting reception status.

Table 8-4 Behaviours of the Ureg function in C-profile linked contents

	Link status
Ureg[ ]	O
Greg[ ]	O

8.3.6.2 EPG functions

Operations of EPG functions in C-profile linked contents are specified in Table 8-5. Refer to 8.1.15.3 for behaviours in data broadcasting reception status. epgReserve() and epgCancelReservation() are options.

Table 8-5 Behaviour of EPG functions in C-profile links

	Link status
epgGetEventStartTime()	O
epgGetEventDuration()	O
epgReserve()	O(*1)
epgCancelReservation()	O(*1)

8.3.6.3 Interaction channel function -TCP/IP

Operations of TCP/IP are specified in Table 8-6. Details regarding behaviour of each function are described below.

Table 8-6 Behaviours of TCP/IP interaction channel communication in C-profile linked contents

	Link status
getConnectionType()	O(*2)
isIPConnected()	O(*2)
transmitTextDataOverIP()	O(*2)

- getConnectionType() function behaviours

This function is only utilized to acquire hintn in order to estimate communication speed from information on what kind of line connection is made.

This is used for confirmation of preferred line type in ARIB STD-B24 Vol. 2 Attachment 1 Informative Explanation 4. However, provisions of getConnectionType( ) do not grasp all of the line types and thus cases that do not fit in the sequence are in Informative Explanation 4. Therefore, note that it is only used as hint.

Refer to ARIB STD-B24 Vol. 2 Attachment 3: 5.6.5.2 for details on the return value of the function. In environments where the receiver is connected by using a line type not described in Appendix 3, this function returns NaN. This function returns NaN in receiver that do not possess a communication function.

Also, return the fixed value “300” (mobile phones (if type cannot be differentiated)) in mobile phones.

- isIPConnected() function behaviour

Refer to ARIB STD-B24 Vol. 2 Attachment 3: 5.6.5.2 for details on the return value of the function. Using this function, receiver judge if resources on the internet are acquirable and returns the value. This function returns NaN in receiver that do not possess the communication function.

Also, return the fixed value “1” (IP connection is established by automatic connection) in mobile phones.

- transmitTextDataOverIP() function behaviour

Operate in link status.

This is a function that assumes telegram transmission and reception is performed in the communication network. It is mainly utilized from broadcasting contents. The restrictions below are established for the arguments of this function.

- Scheme for URI as argument are only 2 types: "http" and "https".
- The maximum length of the string for argument text is 4096Bytes.
- Character code of text data to transmit shall be a fixed operation of only "Shift_JIS".

  Therefore, specify "Shift_JIS" for the third argument charset.

When receiver transmit text to servers, they transmit to the URI specified by the argument using the POST method. If there is no text to transmit, specify empty string (“”) for the argument.

Receiver specify "application/x-www-form-urlencoded" as Content-Type within the requirement message, and "Denbun" for name attribute. Also, follow the URI encoding provisions specified in RFC2396 (2.4.1 Escaped Encoding) for text encoding to encode. Upon this, Japanese strings should be encoded as Shift-JIS character code. When an empty string (“”) is specified in the text, specify 7 in Content-Length to consider a string of "Denbun=".

Figure 8-4 provides the requirement message upon transmitting text “transmit telegram 20060101” to the server.

```
POST http://localhost/test.cgi HTTP/1.1
…header omitted…
Content-Type: application/x-www-form-urlencoded
Content-Length: 85

Denbun=%74%72%61%6E%73%6D%69%74%20%74%65%6C%65%67%72%61%6D%20
%32%30%30%36%30%31%30%31
```

Figure 8-4 An example of requirement messages which transmit TextDataOverIP( )

Servers having processed the requirements may add text data in addition to the response status if there is any text data to be delivered to the receiver. In that case, specify Content-Type as "text/plain", and Charset as "Shift_JIS".

Figure 8-5 specifies an example of a response message which contains the text data “Text will be returned” in the entity-body.

```
HTTP/1.1 200 OK
…header omitted…
Content-Length: 22
Content-Type: text/plain; Charset=Shift_JIS

Text will be returned.
```

Figure 8-5 An example of respose message that transmitTextDataOverIP( ) receives

Upon this, the maximum entity-body size that receiver receives shall be 4096Bytes. If a receiver receives text that exceeds the limit, text reception process after 4097Bytes is implementation dependent. The letter code of text data shall be a fixed operation of "Shift_JIS" only.

Table 8-7 specifies detailed contents regarding return values.

Table 8-7 Return value of transmitTextDataOverIP() (Array[0])

Return value		Meaning	Detailed Content
Array[0]	1	Success	Argument text was successfully transmitted to the internet resource specified by argument uri and received its response.
	-1	Wrong parameter	Wrong argument given to the function
	-2	Line was disconnected during transmission	Physical line was disconnected during data transmission and reception
	-3	Time-out	Data transmission process and reception process did not complete within certain period of time
	-300	Failure upon automatic connection	An error occurs upon automatic connection and connection to the Internet resource specified by argument uri was not performed. This error value is returned when error generates during physical line connection process, PPP and TCP process.
	-400	Failure upon DNS name conversion	Host name specified by argument uri was not able to convert to IP address
	-500	Failure upon TLS process	Failed in TLS process performed upon HTTPS use
	NaN	Failure by other cause	Condition of restrictions of automatic retransmission was detected.

8.3.6.4 Operational control functions

Table 8-8 specified the operation of function for operational control function in C-profile linked contents. Then, details for each function operation is provided. Refer to 8.1.15.5 for function operations of the data broadcasting reception status.

Table 8-8 Operations of C-profile linked contents operational control function

	Link status
reloadActiveDocument()	O
getProgramRelativeTime()	O
getBrowserVersion()	O
getProgramID()	O
getActiveDocument()	O
lockScreen()	O
unlockScreen()	O
getBrowserSupport()	O
launchDocument()	O
quitDocument()	O
detectComponent()	O
lockModuleOnMemoryEx()	X
unlockModuleOnMemoryEx()	O

	Link status
unlockAllModulesOnMemory()	O
getLockedModuleInfo()	O
getBrowserStatus()	O
getResidentAppVersion()	O(*2)
setFullDataDisplayArea()	O(*1)
saveImageToMemoryCard()	O(*1)
saveHttpServerImageToMemoryCard()	O(*1)

- reloadActiveDocument() function behaviour

  When this function is called, the receiver should re-acquire BML document being presented and mono-media from the server. If already acquired BML documents and mono-media exist within a receiver, the receiver may display without change. However, such cache function shall be implementation dependent. When the receiver is playing C-profile linked contents that cannot specify the resource name, such as when "/" was specified at the end of URI or when ?query was specified, perform to re-acquire the currently presented C-profile linked content by re-transmitting the URI that specified the concerned BML document to the server.

  For example, the operation when C-profile linked contents transitioned by launchDocument("http://localhost/hoge/", "cut") calls the reloadActiveDocument function shall be the same as when launchDocument function is called.

- getProgramRelativeTime() function behaviour

  Behaviour in link status shall be the same as the data broadcasting reception status.

- getBrowserVersion() function behaviour

  Behaviour in link status shall be the same as the data broadcasting reception status.

- getProgramID() function behaviour

  Behaviour in link status shall be the same as data broadcasting reception status.

- getActiveDocument() function behaviour

  This operates in link status. Return value shall be returned in a format that starts with "/" (abs_path specified in RFC1808), omitting scheme and host name.

  When receiver is playing C-profile linked content that cannot specify the resource name, such as when "/" was specified at the end of URI or when ?query was specified, return value excluding scheme and host name of URI that specified the concerned BML document.

  For example, the return value is “/hoge/foo?query" when getActiveDocument function was called by C-profile link content transitioned by launchDocument("http://localhost/hoge/foo?query", "cut").

- lockScreen() function behaviour

  Behaviour in link status shall be the same as the data broadcasting reception status.

- unlockScreen() function behaviour

  Behaviour in link status shall be the same as the data broadcasting reception status.

- getBrowserSupport() function behaviour

  Behaviour in link status shall be the same as the data broadcasting reception status.

- launchDocument() function behaviour

  This is operated in link status. Restrictions below are established in link status.

  - Restriction in link status
    - When transitioning to broadcasting contents, specify the absolute URI that starts with "arib-dc: ". Refer to 8.3.10 for details.
    - If outside of the base URI directory scope, the data broadcasting browser should be in a failed operation and transitioned to broadcasting contents. Refer to 8.3.11.3 for details.

- quitDocument() function behaviour

  Behaviour of the quitDocument() function differ depending on the data broadcasting reception status and link status. Refer to Table 8-9. Refer to 8.3.11.3 for status transition.

Table 8-9 behaviourof quitDocument()

Status	Behaviour of quitDocument()
Data broadcasting transmission/reception status	When quitDocument() is executed, the presentation of currently presented broadcasting contents ends, and it is executed step 3 or later for “Receiver operation at the beginning of data broadcasting” described in 4.1.5.1, and is transitted to the startup document in the entry component. Upon this, all modules locked by lockModuleOnMemoryEx function are released.
Link status	When quitDocument() is executed, the presentation of the currently presented C-profile linked contents ends, and it is executed step 3 or later for “Receiver operation at the beginning of data broadcasting” described in 4.1.5.1, and is transitted to startup document in entry component. Upon this, all modules locked by lockModuleOnMemoryEx function are released. If connection was performed at this time, the receiver may disconnect the connection, following 8.3.12.2. For example, the receiver unit may make a judgment whether to disconnect or not, depending on the status such as communicating with other than BML browser, etc. when quitDocument() is specified.

- detectComponent() function behaviour

  Behaviour in link status is the same as data broadcasting reception status. Refer to 8.1.15.

- lockModuleOnMemoryEx() function behaviour

  This does not operate in link status.

- unlockModuleOnMemoryEx() function behaviour

  Behaviour in link status is the same as data broadcasting reception status. Refer to 8.1.15.
  Use of this function in link status can be thought of as releasing the broadcasting contents module locked status by lockModuleOnMemoryEx()in broadcasting reception after its transition to link status.

- unlockAllModulesOnMemory() function operation

  Behaviour in link status is the same as data broadcasting reception status. Refer to 8.1.15.
  Use of this factor in link status can be thought of as releasing the broadcasting contents module locked by lockModuleOnMemoryEx()in broadcasting reception status after its transition to link status.

- getLockedModuleInfo() function behaviour

  Behaviour in link status is the same as data broadcasting reception status. Refer to 8.1.15.
  Use of this function in link status can be thought of as releasing the broadcasting contents

module locked by lockModuleOnMemoryEx() in broadcasting reception status after its transition to link status.

- getBrowserStatus() function behaviour

  This operates in link status. Behaviour in link status is the same as the data broadcasting reception status.

- getResidentAppVersion () function behaviour

  This operates in link status. Behaviour in link status is the same as the data broadcasting reception status.

- setFullDataDisplayArea () function behaviour

  setFullDataDisplayArea() is an option. This operates in link status. Behaviour in link status is the same as the broadcasting reception status.

- saveImageToMemoryCard() behaviour

  saveImageToMemoryCard() is an option. This operates in link status. Behaviour in link status is the same as the broadcasting reception status. Refer to 8.1.15.9 for details.

- saveHttpServerImageToMemoryCard() behaviour

  saveHttpServerImageToMemoryCard() is an option. This operates in link status. Behaviour in link status is the same as the broadcasting reception status. Refer to 8.1.15.9 for details.

8.3.6.5 Receiver sound control

Table 8-10 specified functions behaviour of receiver sound control in C-profile linked contents.

Table 8-10 Behaviour of receiver sound control in C-profile linked contents

	Link status
playRomSound()	O

This is operated in link status. Behaviour is the same as the data broadcasting reception status. Refer to 8.1.15 for behaviour of the data broadcasting reception status.

Behaviour when the timing of operation sound (ex: button operation sounds on mobile phones) and that of sound by playRomSound() are overlapping, is implementation dependent. Also, the sound quarity ringing by playRomSound() is not provided. Therefore, output of sound according to receiver features is permitted.

8.3.6.6 Timer functions

Table 8-11 specifies the behaviour for timer functions in C-profile linked contents. Refer to 5.12.6 for behaviours in the data broadcasting reception status.

Table 8-11 Behaviour of timer functions in C-profile linked contents

	Link status
setInterval()	O
clearTimer()	O
setCurrentDateMode()	O(*3)

- setInterval() function behaviour

This operates in link status. behaviourin link status is the same as the data broadcasting reception status.

- clearTimer() function behaviour

This operates in link status. behaviourin link status is the same as the data broadcasting reception status.

- setCurrentDateMode() function behaviour

setCurrentDateMode() shall be an option. This operates in link status. behaviourin link status is the same as the data broadcasting reception status.

8.3.6.7 Other functions

Table 8-12 specified function behaviour of other functions in C-profile linked contents. Refer to 8.1.15 for the data broadcasting reception status.

Table 8-12 Behaviourof other functions in C-profile linked contents

	Link status
random()	O
subDate()	O
addDate()	O
formatNumber()	O

"Other functions" operate in link status. Behaviours in link status are the same as the data broadcasting reception status.

8.3.6.8 Data carousel storage functions (T.B.D.)

Table 8-13 specified the behaviour of the data carousel storage functions in C-profile linked contents.

Table 8-13 Operation of data carousel storage functions in C-profile linked contents

	Link status
saveModule()	O (*1)
saveResource()	O (*1)

saveModule() and saveResource() are optional. The “Data carousel storage function” operates in link status. Behaviours in link status are the same as the data broadcasting reception status.

8.3.6.9 Digital terrestrial broadcasting specific functions

Table 8-14 specified behaviour of digital terrestrial broadcasting specific functions in communication contents.

Table 8-14 Behaviours of digital terrestrial broadcasting specific functions

	Link status
X_DPA_mailTo()	O(*1)
X_DPA_startResidentApp()	O
X_DPA_phoneTo()	O(*1)
X_DPA_getRcvCond ()	O(*1)
X_DPA_getCurPos ()	O(*1)
X_DPA_saveExAppFile()	O(*1)
X_DPA_startExAV()	O(*1)
X_DPA_stopExAV()	O(*1)
X_DPA_tuneWithRF()	O(*1)
X_DPA_writeSchInfo()	O(*1)
X_DPA_writeAddressBookInfo()	X
X_DPA_getComBrowserUA ()	O(*2)
X_DPA_launchDocWithLink()	O(*2)
X_DPA_chkAVtype()	O(*1)
X_DPA_getIRDID()	O
X_DPA_writeCproBM()	O(*2)

- Behaviour of X_DPA_mailTo()

  X_DPA_mailTo() is an option. Operation in link status is the same as the data broadcasting reception status. Refer to 8.1.15.

- Behaviour of X_DPA_startResidentApp()

  Operation in link status is the same as the data broadcasting reception status. Refer to 8.1.15.

- Behaviour of X_DPA_phoneTo ()

  X_DPA_phoneTo () is an option. Behaviour in link status is the same as the data broadcasting reception status. Refer to 8.1.15.

- Behaviour of X_DPA_getRcvCond ()

  X_DPA_getRcvCond () is an option. Behaviour in link status is the same as the data broadcasting reception status. Refer to 8.1.15.

- Behaviour of X_DPA_getCurPos ()

  X_DPA_getCurPos () is an option. Behaviour in link status is the same as the data broadcasting reception status. Refer to 8.1.15.

- Behaviour of X_DPA_saveExAppFile()

  X_DPA_saveExAppFile() is an option. Behaviours in link status shall be the same as data broadcasting reception status. Refer to 8.1.15.

- Behaviour of X_DPA_startExAV ()

  X_DPA_startExAV () is an option. Behaviour in link status is the same as the data broadcasting reception status. Refer to 8.1.15.

- Behaviour of X_DPA_stopExAV ()

  X_DPA_stopExAV () is an option. Behaviour in link status is the same as the data broadcasting reception status. Refer to 8.1.15.

- Behaviour of X_DPA_tuneWithRF()

  Behaviour in link status is the same as the data broadcasting reception status. Refer to 8.1.15.

- Behaviour of X_DPA_ writeSchInfo()

  X_DPA_ writeSchInfo() is an option. Behaviour in link status is the same as the data broadcasting reception status. Refer to 8.1.15.

- Behaviour of X_DPA_writeAddressBookInfo()

  This does not operate in link status.

- Behaviour of X_DPA_getComBrowserUA()

  Behaviour in link status is the same as the data broadcasting reception status. Refer to 8.1.15.

- Behaviour of X_DPA_getIRDID()

  Behaviour in link status is the same as the data broadcasting reception status.

- Behaviour of X_DPA_launchDocWithLink()

  This is mandate for receiver units that possess communication functions. However, this function is allowed to use only in link status. If this function would be used in broadcasting reception status, the receiver shall fail and shall present broadcasting contents following 8.3.11.4.

- Behaviour of X_DPA_chkAVtype()

  Behaviour in link status is the same as the data broadcasting reception status.

- Behaviour of X_DPA_writeCproBM()

  Behaviour in link status is the same as the data broadcasting reception status.

8.3.6.10 Non-volatile memory functions

Table 8-15 specified behaviour of non-volatile memory functions in communication contents. Refer to 8.1.15 for operations in data broadcasting reception status.

Table 8-15 Behaviour of non-volatile memory functionss in communication contents

	Link status
readPersisitentArray()	O
wirtePersisitentArray()	O

The NVRAM area that can be used in link status is C-profile area for the affiliation. Refer to 7.2.4. for the name space. If a name space other than that is specified, the receiver will perform a failure behaviour.

8.3.7 Communication contents scope mapping

BML browser has 2 types of status, which are the “broadcasting reception status” and “link status”. These statuses mean condition for behaviour of receiver. From the contents point of view, there are 2 types of contents of “broadcasting contents”, which is content that receiver can refer to and present in broadcasting reception status, and “C-profile linked contents”, which are contents that receiver can acquire in link status. These 2 contents together are called “broadcaster’s contents”. Here, we will explain the “link status” of the above.

- Link status

- A status in which the BML document specified in the URI under the base URI directory is being presented shall be referred to as link status. Refer to "Description of the base URI directory" in this section for the concept of the base URI directory.
- Host name or directory name specified by X_DPA_launchDocWithLink() is set as the URI base directory. Status presenting the BML document described by the URI which is under the URI of this base URI directory shall be link status.
- Refer to 8.3.9 for references.

C-profile linked contents behave as one of the contents of the contents group. Communication contents in this status shall behave as the contents group in which data events do not exist.

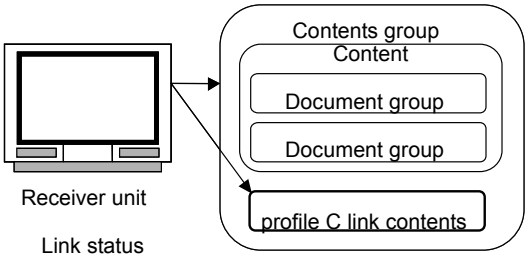


Figure 8-6 Link status

- Restrictions of C-profile linked contents
- In link status, the extended functions for broadcasting that can be utilized are restricted.
- The BML document of broadcasting contents and BML document in C-profile linked contents have different name spaces. Scheme of “arib:” or “arib-dc:” should always be specified in order to specify broadcasting contents from C-profile linked contents. Scheme of "http: " or "https: " should be specified in order to specify C-profile linked contents from broadcasting contents. Be aware of the difference in abbreviation handling as well. Refer to 8.3.10 for operation of name spaces in C-profile linked contents.
- In C-profile linked contents, there is no concept of a startup document, unlike broadcasting contents. However, receiver may specify the URI on a directory level, without specifying resources of C-profile linked contents directly. In that case, C-profile linked contents of specifications in this volume which follow the setting of servers is returned. (ex: index.bml, etc.)
- Because there is no data event in C-profile linked contents, receivable event messages are only “ event_msg_group_id=1 ”. If the concerned id receives “ event_msg_group_id=0 ”, it is ignored. Refer to 4.3 for operations of event_msg_group_id.
- If transitioned to C-profile linked contents located under the URI (base URI directory hereafter) directory specified by the broadcasting contents, the link status should be kept. However, if another URI was specified by use of the “href” attribute of the “a” element, launchDocument() or the “action” attribute of the “form” element, etc., BML browser will fail and will present broadcasting contents following 8.3.11.4. Refer to "Description of the base URI directory" in this section for the concept of the base URI directory.

- After transition from C-profile linked contents to link status by X_DPA_launchDocWithLink(), if URI under the base URI directory which was specified by X_DPA_launchDocWithLink() is specified by following method, the “href” attribute of the “a“element, launchDocument() or

the “action” attribute of the “form” element, etc., the link status is kept. Also, if other than base URI directory was specified, receiver unit will fail and will present broadcasting contents following 8.3.11.4.

- With receivers that possess closed caption presentation functions, closed caption should be able to be presented if closed caption are included in the broadcasting service they belong to, even in link status.

* If the receiver unit receives response in the 300’s and is redirect-specified, the link status is kept as long as the redirect destination is below the URI directory.
* Also, follow separate provisions for communication protocols regarding mobile phones.

- Description of the base URI directory

The base URI directory is utilized as a C-profile link content document group identifier in link status. Base URI directory refers to the specified host name and directory name by: launchDocument(), the “href” attribute of the “a” element and the “action” attribute of the “form” element from broadcasting contents or by X_DPA_launchDocWithLink() from C-profile linked contents. For example, the identifier that refers to the C-profile linked contents base URI directory specified by launchDocument("http://localhost/hoge/index.bml", "cut") is "//localhost/hoge/".

Specifications below are established for judgment of base URI directories.

- Identifier that describes base URI directories that do not include port numbers. For example, base URI directory of C-profile linked contents which is specified by launchDocument("http://localhost:10080/hoge/index.bml", "cut") from data broadcasting BML document is ”//localhost/hoge/”. Therefore, even in C-profile link content transition between different port numbers, link status is kept as long as the base URI directory matches.
- Name space whose URI was encoded should be handled as the same level as name space that is not encoded. For example, the base URI directory for "http://localhost/%7Ehoge/index.bml" and "http://localhost/~hoge/ test.bml" should be handled as matching directories.
- Upper case letters and lower case letters are not differentiated with host names. Upper case letters and lower case letters are differentiated with directory names.
- The base URI directory to be set is one. Also, execution of X_DPA_launchDocWithLink() will be the operation that changes the base URI directory.

Resources directly under the base URI directory and resources stored in directories under are all considered to be within the document group of C-profile linked contents, and the link status is kept. If the URI that does not match the base URI directory was specified, data

broadcasting browser will fail and will present broadcasting contents following 8.3.11.4. Refer to 8.3.12 for status transition details.

- Link status due to X_DPA_launchDocWithLink() transition

  If transitioned using the X_DPA_launchDocWithLink() function from link status, the base URI directory before transition shall be invalid. Set the host name or directory name specified by this function in the URI directory, and the receiver shall be in link status.

8.3.8 Guidelines for C-profile linked contents transmission

The following shows the operational guidelines for the transmission of C-profile linked contents. Refer to section 3.5 for the communication function of the receiver. Details of the communication protocol related to mobile phone shall follow the separately defined prescriptions.

- C-profile linked contents shall be transmitted in accordance with HTTP/1.1 defined in RFC2616.
- If "http:" has been specified in the URI, the receiver and the interactive Web server shall perform communication via HTTP/1.1 using the port specified by the URI.
- If "https:" has been specified in the URI, the interactive Web server shall perform encrypted communication based on HTTP/1.1 after having established a connection via TLS1.0 or SSL3.0 using the port specified by the URI.
- If the port number was not specified in the URI, ports 80 and 443 shall be used for "http:" and "https:" respectively.
- Refer to Appendix 11 for the details of the cache function.
- Servers should consider cases where the following media types are requested.

  text/plain, text/html,
  text/css, text/X-arib-ecmascript,
  image/jpeg, image/gif
  audio/X-arib-mpeg2-aac,
  application/X-arib-btable
  application/octet-stream(T.B.D.)

*In the case of mobile phones,follow other specification for communication protocols..

8.3.9 Resource references between C-profile linked contents and broadcasting contents

8.3.9.1 Resource references from broadcasting contents to C-profile linked contents

For resource references from broadcasting contents to C-profile linked contents, follow ARIB STD-B24 Vol. 2 Appendix 4　6.2 “Guidelines on reference across media types” and do not perform reference. Only BML document transition is possible from broadcasting contents to C-profile linked contents.

8.3.9.2 Reference from C-profile linked contents to broadcasting contents

In cases where referring from C-profile linked contents to event messages transmitted by broadcasting or where specifying resources transmitted by broadcasting upon event’s subscribe, etc., always only use absolute URI’s that start with "arib-dc://-1.-1.-1".

(ex: <bml:beitem id="em" type="EventMessageFired" es_ref="arib-dc://-1.-1.-1/89" …omission…/>)

8.3.10 Name space in C-profile linked contents

8.3.10.1 Restriction for URI

For operations of name space, follow ARIB STD-B24 Vol. 2 Appendix 4 6.3 “Namespace”. However, when presenting BML communication contents acquired from servers, the following restrictions are established for the BML document’s name space.

- Multi Byte characters such as Japanese are not utilized for the URI.
- BML document should be specified for “href” attributes of the “a” element, arguments of launchDocument() and arguments of X_DPA_launchDocWithLink () described in BML documents in broadcasting reception status and link status.

8.3.10.2 Operation of name spaces in servers

In link status, if “/” was specified at the end of the URI upon contents acquisition from servers, servers should return BML documents in C-profile linked contents, according to the settings of server.

(ex: browser.launchDocument("http://localhost/hoge/", "cut");)

Receiver behaviour is implementation dependent,when BML documents were not returned from servers due to each status.

When abbreviation of URI (relative URI) is ued, its scheme is the same as scheme of presenting contents. It is treated as relative URI from directory names located presenting contents. (ex: If the BML document URI is http://localhost/test/tmp/index.bml and "../hoge.bml" was described in the document, its URI will be http://localhost/test/hoge.bml.)

However, when using X_DPA_launchDocWithLink(), use an absolute URI for its argument.

8.3.10.3 Operation of name spaces upon transition from broacasting contents to C-profile linked contents

Reference from broadcasting contents to C-profile linked contents shall not be performed, according to specifications in 8.3.9.1. When transitioning from broadcasting contents to C-profile linked contents, use absolute URI that start with "http: " or "https: ".

8.3.10.4 Operation of name spaces upon transition from C-profile link content to broadcasting content

When transitioning from C-profile linked contents to broadcasting BML document, always only use absolute URI that start with “arib-dc://-1.-1.-1”.
(ex: browser.launchDocument("arib-dc://-1.-1.-1/80/0000/startup.bml", "cut");)

8.3.11 Guidelines for operations of C-profile linked contents

8.3.11.1 Note on operations of C-profile linked contents

Service provider(including broadcaster) that supplies C-profile linked contents should consider that it is extremely difficult for them to present BML content to viewers in the same quality level as data broadcasting, due to the property of communication system used in C-profile linked contents distribution. Communication system properties are listed below.

- There is no guarantee that the transmission bandwidth is invariable.
- There are differences in useable transmission bandwidth depending on the receiver.
- Steady response to viewers cannot be guaranteed because the variable communication path can be selected.
- There is a possibility that data transmission is blocked due to proxy properties in the route.

Furthermore, there is a risk that response quality may greatly decrease due to heavy loads on servers. Note that due to these elements, presentation of C-profile linked contents distributed in communication systems is not invariable for all viewers and the receiver’s presentation may not perform correctly depending on the network situation.

Due to above reasons, it is recommended for BML contents fitting in items below to be distributed via data broadcasting rather than via communication.

- BML contents recognized execution timing such as video/audio sync as critical factor.
- BML contents recognized presenting all elements properly as critical factor. .

8.3.11.2 Guidelines for description of C-profile linked contents

Notes of description of C-profile linked contents are below.

- C-profile linked contents are described in profiles according to operations in 8.1.
- Note that there is difference between broadcasting contents and C-profile linked contents regarding the name space. Refer to 8.3.10 for details.
- It is recommended not to specify the port number for C-profile linked contents which are assumed to operate in link status.

- Always add “/” at the end when specifying C-profile linked contents by directory specifications. (ex: "http://localhost" ×: incorrect, "http://localhost/"O: correct)
- #fragment can be added to the URI that specifies C-profile linked contents. Refer to 8.2.3.4 regarding guidelines and behaviour of receiver for contents using #fragment. (ex: "http://localhost/test.bml#fragment")
- Following documents and files should be described in Shift-JIS, BML documents to compose C-profile linked contents, binary tables, ECMAScript files related in the “src” attribute of “script” elements and CSS files related in “link” elements.
- The BML Version described in C-profile linked contents is ”12.0”, as broadcasting contents. (ex: <?bml bml-version ="12.0" ?>)
- Extension of BML documents that compose C-profile linked contents should be bml. (ex: index.bml)
- By putting Mono-media in one page under the same directory, it can reduce receiver unit’s pass search process
- BML document should be specified for “href” attributes of the “a “element, arguments of launchDocument() and arguments of X_DPA_launchDocWithLink ()described in C-profile linked contents.
- Within C-profile linked contents, links to other BML documents or a procedure to quit it should always be described. This is because receiver units will have no procedure of transition to another status other than tuning, just in case receiver transit to a BML document with no link.
- Resources (mono-media such as JPEG, GIF, animation GIF and MPEG-2 AAC) in C-profile linked contents can be specified separately from the scope of the base URI directory.
- Because available functions are different in link status, it is different behaviour even in the same contents.
- Total amount of C-profile linked contents resources referred to in BML documents at the same time shall not exceed 256KBytes , considering that the area reserved for receiver’s module acquisition is 256KBytes.
- If the X_DPA_launchDocWithLink function is used, a host name or directory name described by the argument of its function is set as the URI directory, and goes in link status. Therefore, as specified in 7.10.5.1, available functions in link status can be used and can access to NVRAM, etc. Therefore, for contents description ,it should pay close attention to such issues.

8.3.11.3 Guidelines for receiver unit status transition operations

Receiver can be in the following 2 status as receiving BML contents.

(1) “Data broadcasting reception status” in which data broadcasting, video and audio are being received and played.

(2) “Link status” in which C-profile linked contents, video and audio are being received and played together.

Refer to 8.3.7 for link status. Receiver transitions betweenthese 2 statuses by the specification of contents or the operation of users. Figure 8-7 specified a situation of status transition.

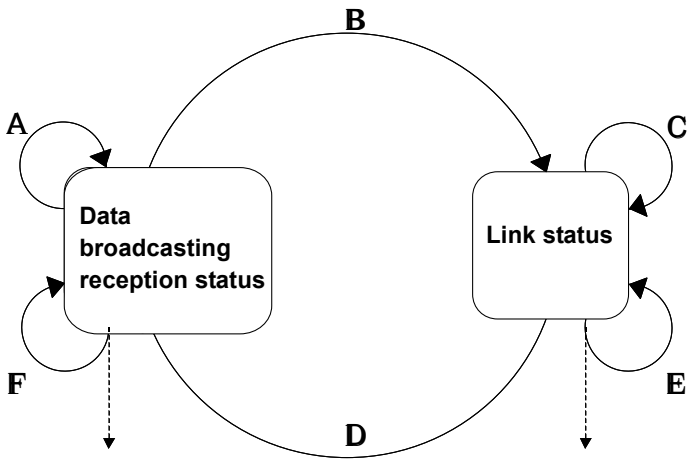


Figure 8-7 Status transition of receivers in C-profile linked contents reception

- Description of transition A (from data broadcasting reception status to data broadcasting reception status)

Transition A means a transition from data broadcasting reception status to data broadcasting reception status. Refer to chapter 4 for details of operations. In this transition, the URI scheme that can be specified for the “href” attribute of the “a” element and launchDocument() is "arib-dc: ".

- Description of transition B (from data broadcasting reception status to link status)

Transition from data broadcasting reception status to link status is possible by the specification of launchDocument( ), the “href” attribute of the “a” element or “action” attribute of the “form” element. There are 2 types of URI schemes to specify C-profile linked contents: "http: " and "https: ". Refer to 8.3.10 for details of name spaces in C-profile linked contents.

(ex: browser.launchDocument("http://localhost/hoge/index.bml", "cut");)

C-profile linked contents can specify the directory without specifying the resources. When the directory is specified, a server returns the C-profile linked contents that follow the setting. “index.bml”, for example. When specifying the directory, make sure to add “/” at the end to indicate that it is a directory.

(ex: browser.launchDocument("http://localhost/hoge/", "cut");)

BML browser fails and will present broadcasting contents following 8.3.11.4, if it is specified a document whose extension is “html” or “htm” by following method in data broadcasting status , the “href” attribute of the “a“ element and launchDocument() and the “action” attribute of the “form” element.

- Description of transition C (from link status to link status)

The scope of the document group in C-profile linked contents is the range specified under the first accessed URI’s host address and directory (base URI directory hereafter) or the range specified under the host name and directory name specified by X_DPA_launchDocWithLink() in link status. It does not depend on the scheme (protocol) specification. (ex: //localhost/hoge/)

When link status is kept by means of the base URI directory, the link status is kept even after transitioning from "http://localhost/hoge/index.bml" to “http://localhost/hoge/foo/test.bml” by using the “href” attribute of the “a” element or launchDocument(). Also, in case of a transition to “https://localhost/hoge/foo/test.bml”, the link status is kept in the same way.

However, if something other than the base URI directory was specified in the link status by the “href” attribute of the “a” element, the “action” attribute of the “form” element, and launchDocument(), BML browser will fail and will present broadcasting contents following 8.3.11.4. If a document whose extension is “html” or “htm” is specified by the “href” attribute of the “a” element, the “action” attribute of the “form” element, and launchDocument(), BML browser will fail and will present broadcasting contents following 8.3.11.4.

Note that when the link is kept by the base URI directory, the link status is still kept after transitioning from "http://localhost/hoge/index.bml" to “http://localhost/hoge/foo/test.bml” by utilizing X_DPA_launchDocWithLink(), but the base URI directory will be set as "//localhost/hoge/foo/".

It will be in link status when transitioned by X_DPA_launchDocWithLink(), even in cases where the host address is changed as in the transition from "http://localhost/hoge/index.bml" to “http://localhostdpa/dpa/foo/test.bml”, for instance. The base URI directory in that case is set as ”//localhostdpa/dpa/foo/.”

- Description of transition D (from link status to data broadcasting reception status)

Transition from link status to data broadcasting reception status can be performed by using the “href” attribute of the “a” element or launchDocument( ). Operation is an implementation dependent in cases where there is no BML document of transition destination or where the

BML document of transition destination is not acquirable due to reception status.

Always specify "arib-dc:" for the URI scheme to specify broadcasting contents. Note that only currently presented services such as "arib-dc://-1.-1.-1/80/0000/startup.bml" can be specified when transitioning using launchDocument().

(ex: browser.launchDocument("arib-dc://-1.-1.-1/80/0000/startup.bml","cut");)

Also, by using quitDocument(), the transition from link status to data broadcasting reception status is possible.

By tuning of the user's action, transition from link status to the data broadcasting reception status can also be executed. .

By using X_DPA_tuneWithRF(), transition from link status to the data broadcasting reception status is also executed.. However, an entry component is presented after transition in this case.

Also, if "arib-dc:" and "arib:" are specified by X_DPA_launchDocWithLink(), BML browser will fail and will present broadcasting contents following 8.3.11.4.

- Transition E (transition from link status to broadcasting reception status and other non-link status)

If X_DPA_startResidentApp() is executed in link status, the receiver's native application specified by argument will startup. When continuing presentation of BML browser such as simultaneously displaying with communication browser, the link status is kept and presentation of C-profile linked contents will continue. Refer to 8.1.15.8 for details of X_DPA_startResidentApp().

- Transition F (broadcasting reception status to broadcasting reception status and non-link status)

If X_DPA_startResidentApp() is executed in broadcasting reception status, the receiver's native application specified by argument will startup. When continuing presentation of BML browser, as simultaneously displaying with communication browser, the broadcasting reception status is kept and presentation of broadcasting content will continue. Refer to 8.1.15.8 for details of X_DPA_startResidentApp().

8.3.11.4 Guidelines for failure behaviour of receiver units

If the restricted function in status specified in 8.3.11.3 is used, the receiver will perform an operation failure.

Also, if a URI, which does not match the base URI directory, is specified by the “href” attribute of the “a” element and launchDocument() and the “action” attribute of the “form” element in link status, BML browser will fail and the receiver will display an error message.

In sections specified as “operation failure” and “failure behaviour” of BML browsers in 8.3.6 etc., receivers shall perform the following processes in order.

(1) BML browser ends once.

(2) An error message is displayed.

(3) Step 3 and later described in 4.1.5.1 “Receiver operation at the beginning of data broadcasting” are executed for the service which is tuned currently.

- Guidelines for receiver upon error response reception

The following guidelines are below when receiver requests C-profile contents and resources to consist of those from servers, and an error response[*1] is returned:

- Receiver behaviour upon reception of error responses in BML documents is an implementation dependent.
- Presentation of receiver upon reception of error response regarding resources that consist of BML documents is implementation dependent.

*1: If status code was in the 400’s or 500’s. Refer to RFC2616 for details.

8.3.12 Guidelines for connection and disconnection

The specification of this section do not apply to receivers which are always on Internet access. .

8.3.12.1 Guidelines for receiver upon automatic connection

If the receiver does not perform IP connection and the following events are occurred, the receiver performs an automatic connection to the communications system. Although in the case of mobile phone receivers should perform automatic connection to communications system when the following events are occurred, receivers may follow other specifications.

- If transition to a BML document on a server in link status is performed (transition by means of the “action” attribute of the “form” element, the “href” attribute of the “a” element, launchDocument(), and X_DPA_launchDocWithLink())
- If transmitTextDataOverIP() is used

8.3.12.2 Guidelines for receiver upon disconnection

Upon disconnection of communication by the quitDocument() function, the receiver shall completely disconnect the connection before the startup document is displayed. Within the

period between calling of quitDocument() function and presentation of the startup document, the timing of disconnection is an implementation dependent.

Upon disconnection of communication by tuning and use of the X_DPA_tuneWithRF() function, the receiver shall completely disconnect the connection before tuning is complete. Within the period between tuning or calling of X_DPA_tuneWithRF() function and completion of tuning, the timing of disconnection is implementation dependent.

8.3.13 Operation of certificates

In C-profile, the root certificate is not transmitted from the broadcasting station to the receiver. Instead, the root certificate already embedded and incorporated on the receiver shall be used. Contact the governing organization described in Appendix 13 regarding the root certificate to be operated by broadcasters.

- The hash function for signatures in the server certificate and intermediate certificate shall be limited to either MD5 or SHA-1.
- The receiver shall be implemented such that the user cannot voluntarily add a root certificate to the receiver. Alternatively, the receiver shall fail in TLS/SSL communications using a server certificate linked to a root certificate not operated by the broadcaster.
- Incorporation of update functions for the root certificate for when the certificate expires and for the root certificate information operated by the broadcaster to the receiver shall be a matter of product planning. However, it shall be desirable that the receiver incorporates both of these update functions.

8.3.14 Operation of User-Agent for identification of BML browsers and a browser for the C-profile contents on internet

As one of the method within servers to judge browsers, use of User-Agent of an HTTP Request header can be expected. Therefore, BML browsers and a browser for the C-profile contents on internet in receivers that have connectable features shall implement User-Agent

By using “User-Agent” as a procedure to judge receiver’s models within servers, broadcasters can get receiver’s various information and judge receiver’s models, and then can specify appropriate contents corresponding to the receiver. .

The following specifies information and the format setting for User-Agent. Also, in the case of mobile phones, if there are other specifications, they may follow those. .

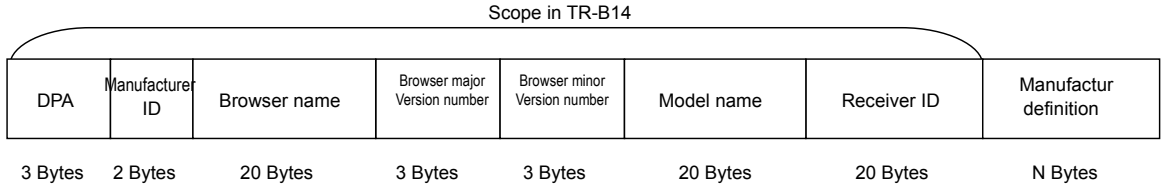


Figure 8-8 User-Agent format specified in TR-B14

- DPA: stands for request from browser specified in TR-B14. String that is a fixed length and “DPA”. When using User-Agent in BML browsers and browser for the C-profile contents on the internets, this shall always be added to the front.
- Manufacturer ID: stands for manufacturer’s ID managed by ARIB. Maximum size is 2 Bytes. This shall not be omitted in the specified format.
- Browser name: stands for browser’s name. Return string that is a maximum of 20 characters, using "0"-"9" and "A"-"Z" which are specified for each manufacturer. This shall not be omitted from the specified format.
- Browser major version number: stands for browser’s major version number. Return string that is described with a decimal number that is maximum of 3 digit. This shall not be omitted in the specified format
- Browser minor version number: stands for browser’s minor version number. Return string that is described with decimal number that is maximum of 3 digit. This shall not be omitted in the record.
- Model name: stands for receiver’s model name. Maximum size is 20 Bytes. This shall not be omitted from the specified format.
- Receiver unit ID: stands for receiver’s ID. Maximum size is 20 Bytes. This may be omitted from the specified format. However, it is recommended for the receiver to get a user permission before the fact in order to set the receiver’s ID.
- Manufacturer definition area: area which is defined and used by manufacturers and carriers.

➢ 1 For “manufacturer ID”, a hexadecimal string is returned, but it shall be 2 digit without adding the string implying it is a hexadecimal string, such as "0x" in front or "h" at the end. If it is less than 2 digit, “0” is added as necessary padding.

➢ 2 If “browser name” and ”model name” are less than the maximum Byte size, stuff with space (0x20).

- 3 Stuff with space (0x20) in case of omitting items that are allowed to be omitted.
- 4 For “browser major version number” and “browser minor version number”, if they are less than 3 digit, add “0” as padding.
- 5 Use “/” as a partition for each item.
- 6 “Manufacture ID”, “browser name”, “browser major version number” and “browser minor version number” shall be the same as the return value upon execution by getBrowserVersion() .
- 7 For “model name”, if it is mobile phones, it is recommended to set the model name that is set within User-Agent of carriers' proprietary browser. Set the model name that is managed by each manufacturer in the case of receivers other than mobile phones.

(Usage example)

User-Agent : DPA/1C/AAABBBCCC7 /023/007/DPA-H90001 /receiver unit ID/Manufacturer definition area…

8.3.15 Operation of X_DPA_getComBrowserUA()

By using information acquired by X_DPA_getComBrowserUA() for judgment of browser’s models, X_DPA_startResidentApp() can be used in changing the transition URI.

8.3.16 Content transition and browser startup

Figure 8-9 specified the contents transition and browser startup. Also, Figure 8-9 simply specified ”transition” of the contents and ”startup” of the browser. Refer to specifications in 8.1.4 - 8.1.7 for detailed information regarding startup and presentation of each browser and application.

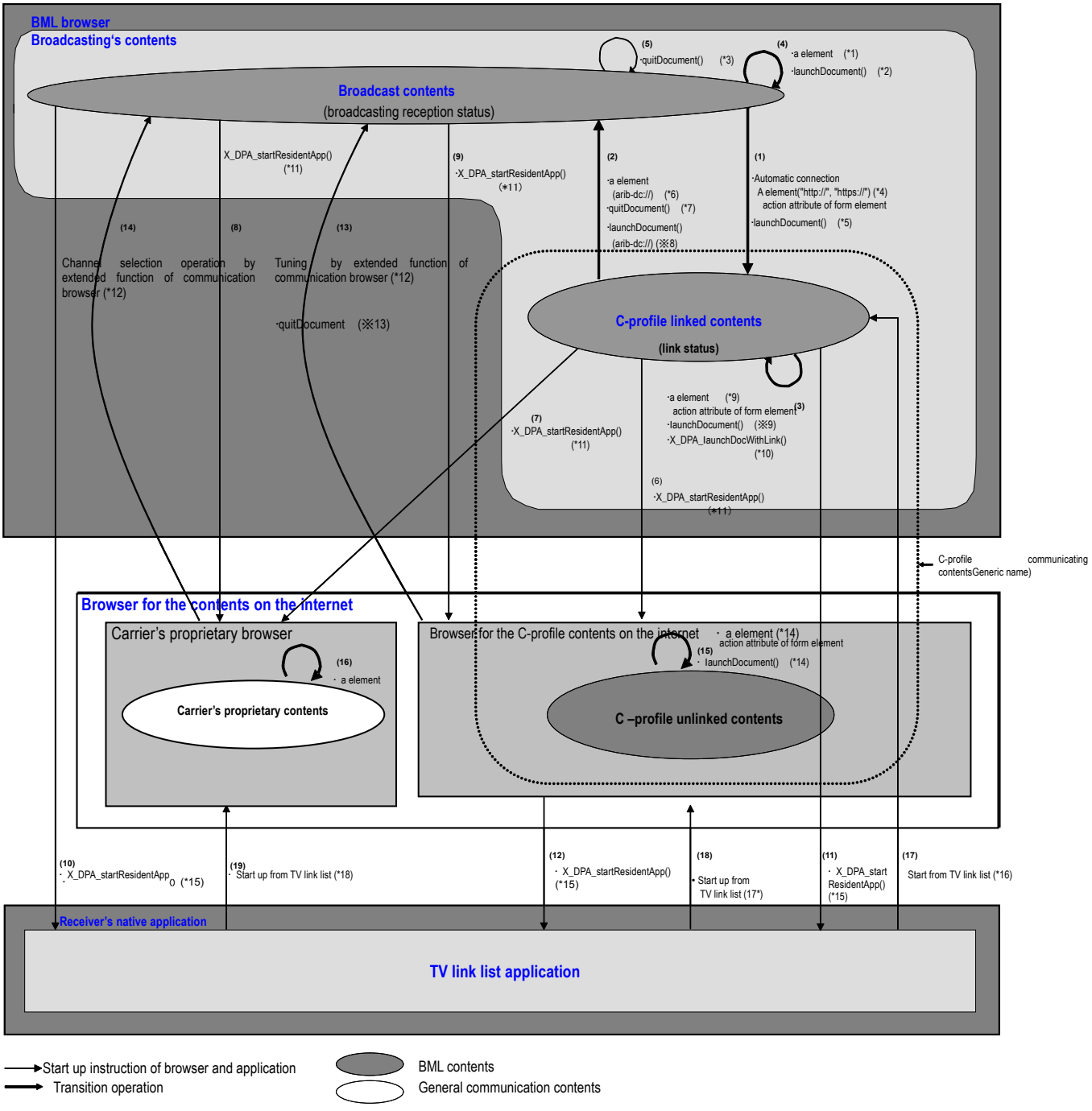


Figure 8-9 Transition and startup of each browser

*1: When broadcast contents are specified by the “href” attribute of the “a” element from broadcast contents. Use abbreviation (refer to ARIB STD-B24 Vol. 2: 9.2) for the name space description to specify broadcasting contents within BML documents.

*2: Transition from broadcasting contents to broadcasting contents by use of launchDocument(). Use abbreviation (refer to ARIB STD-B24 Vol. 2: 9.2) for the name space description to specify broadcasting contents within BML documents.

*3: Refer to 8.3.6.4 for quitDocument(). In this case, however, finish BML browser once and present entry component after restarting.

*4: Specify the URI of C-profile linked contents by the “href” attribute of the “a” element and the “action” attribute of the “form” element. Only "http: " and "https: " are specifiable for the URI scheme.

*5: Specify the URI of C-profile linked contents. It will be in link status if transitioned by this funcion. Only "http: " and "https: " are specifiable in the URI scheme.

*6: Specify only the URI of broadcasting contents by the “href” attribute of the “a” element. Only specify absolute URI that start with “arib-dc://-1.-1.-1" for the URI scheme.

*7: Refer to 8.3.6.4 for quitDocument(). In this case, however, finish data broadcasting browser once and present the entry component after restarting. However, upon this, the connection may be disconnected if already connected. Also, upon specification of quitDocument(), receivers may judge whether or not to disconnect the connection , depending on situations such as communicating with something other than BML browser, etc.,.

*8: Specify only the URI of broadcasting contents. Only specify absolute URI that start with “arib-dc://-1.-1.-1" for the URI scheme.

*9: Specify the URI of C-profile linked contents by the “href” attribute of the “a” element, launchDocument() and the “action” attribute of the “form” element. However, the URI in this case shall be within the base URI directory.

*10: Specify the URI of C-profile linked contents. If transitioned by X_DPA_launchDocWithLink(), the host name or directory name specified by X_DPA_launchDocWithLink() is set as the base URI directory and becomes a link status. Only "http: " and "https: " are specifiable for the URI scheme.

*11: X_DPA_startResidentApp() is executed in broadcasting reception status and link status, and after that a browser for the contents on the internet is started up. Perform presentation according to 8.1.4 to 8.1.7 upon this.

*12: Refers to behaviour when the tuning is specified by the extended function of a browser for the contents on the internet.

*13: If quitDocument() is used, finish the Browser for the C-profile contents on the internet once and present the entry component after restarting, not transitioning. After the browser is finished, the channel to tune shall be the last channel. However, connection may be disconnected if already connected upon this. Also, upon specification of quitDocument() , receivers may judge whether or not to disconnect the connection, depending on situations such as communicating with something other than BML browser, etc.,.

*14: Specify the URI of the C-profile unlinked contents by the “href” attribute of the “a” element, launchDocument() and the “action” attribute of the “form” element. Only "http: " and "https: " are specifiable for the URI scheme. URI of broadcasting contents cannot be specified.

*15: In broadcasting contents, C-profile linked contents and C-profile unlinked contents ,when X_DPA_startResidentApp() is executed by specifying "BookmarkList" for argument “appName”, TVlink list applications as one of the receiver’s native applications starts up.

*16: If TVlink set as CproBMtype=1 is selected by users in the TVlink list application as one of the receiver’s native applications, present C-profile linked contents according to 3.7.7.

*17: If TVlink set as CproBMtype=2 is selected by users in the TVlink list application as one of receiver’s native applications, present C-profile unlinked contents according to 3.7.7

*18: If TVlink set as CproBMtype=3 or CproBMtype=4 is selected by users in the TVlink list application as one of the receiver’s native applications, present the communication company specification contents according to 3.7.7

9 Functions with which C-profile basic receiver unit that only receive video and audio

This chapter specifies functions with which C-profile basic receiver unit that only receive and present video and audio.

9.1 Configuration of the receiver unit

Below are specifications of each processor from the viewpoint of hardware configuration necessary for reception of video and audio data.

9.1.1 Hardware configuration

Figure 9-1 specifies the hardware configuration of receiver units.

Digital broadcasting signals that enter into C-profile basic receiver units are transformed into a transport stream by a tuner and demodulator. The transport stream, by transport stream decoding process, is then demultiplexed into video and. Video stream is output to a video decoding process, and audio stream is output to an audio decoding process. With the above processes, C-profile basic receiver units perform playback of video and audio in a stream format.

From the viewpoint of such hardware processing operations, the following specifications are necessary.

(1) Receivable data by the transport decoder
(2) Playback audio data of the stream system
(3) Playback video data of the stream system
(4) Presentation of video

(1) for the TS decoding function, and (2), (3) and (4) for presentation functions, are especially specified.

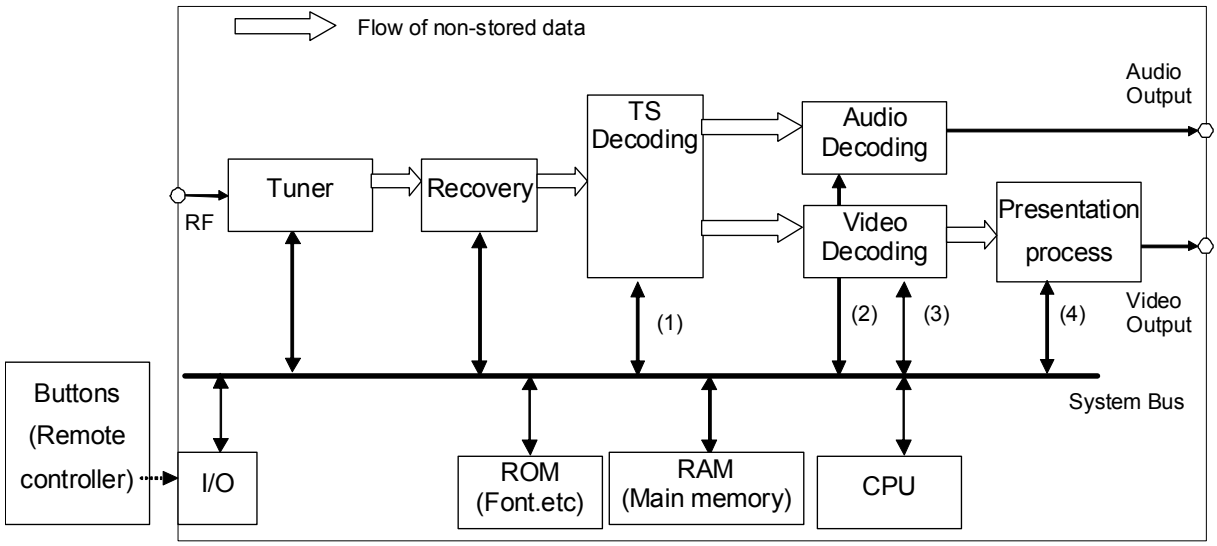


Figure 9-1 Hardware configuration of a C-profile basic receiver unit (video data/audio data code)

9.2 Presentation functions

9.2.1 Video presentation function

Table 9-1 indecates specifications for each coding formula of visual mono-media, whose presentation is desired in C-profile basic receiver unit that only receive video and audio.

Table 9-1 Visual mono-media whose presentation is desired

Coding formula		Contents of specification	
Video coding	H.264|MPEG-4 AVC	Transmission method	Video PES; stream format identifier = 0x1B
		Video size	QVGA(320x240 (4:3), 320x180 (16:9))
		Scaling	Implementation dependent

9.2.2 Audio playback function

Table 9-2 indecates specifications for coding formula of audio mono-media, whose presentation is desired in C-profile basic receiver units that only receive video and audio. Operation details for the coding method refer to Chapter 5.

Table 9-2 Audio mono-media whose presentation is desired

Coding method	Contents of specification	
MPEG-2 AAC	Transmission method	Audio PES; stream format identifier = 0x0F
	Sampling frequency	24kHz,48kHz

9.3 Guidelines for reception operation

1. If PMT 2nd loop includes components of component_tag=0x81, it shall be considered as a low-frame-rate and low-resolution picture (H.264|MPEG-4 AVC) stream and played back. Also, if it includes component_tag=0x83, it shall be considered as a MPEG-2 AAC (sampling frequency=24kHz) sound stream and played back. In the same manner, if it includes component_tag=0x85, it shall be considered as a MPEG-2 AAC (sampling frequency=48kHz) sound stream and played back.
2. It is not necessary to follow data (data carousel / event message) changes other than the video / audio stream.

9.4 Guidelines for presentation

In this section, guidelines for video and audio presentation in C-profile basic receiver units that only receive video and audio are specified.

9.4.1 Presentation of video / audio

9.4.1.1 Display position and display size of video

- Display position and display size of video placed within display devices of receiver units are implementation dependent.
- The receiver units do not necessarily need to display video transmitted by QVGA in the same pixel size. Specifically, they are allowed to display video with less pixel size and frame rate, if the receiver unit's performance or processing capability are insufficient. On the other hand, they are allowed to display enlarged video for receiver units whose pixel size is above QVGA.

9.4.1.2 Optimizing video display size

With receiver units whose display device possesses pixel size other than 320 dots in the horizontal direction, it is recommended to display video by scaling it according to display device's pixel size.

9.4.1.3 Display of QVGA 16:9 video

When displaying QVGA (16:9) video, areas within the display device that the video possess is 60 dots smaller in the vertical direction compared to QVGA (4:3) video. With receiver units with no multimedia presentation function, the display of contents, applications etc. supplied by broadcaster in this area should not be performed.

9.4.2 Presentation upon TV reception function startup and tuning

With C-profile basic receiver units that only receive video and audio, only video and audio originated in broadcasting should be presented and contents or applications other than broadcasting may not be displayed on display devices of receiver units upon TV reception function startup. The contents and applications that were displayed right before tuning should be invisible.

9.4.3 Restriction of mixed display of non-broadcasting screen and broadcasting screen and simultaneous display

9.4.3.1 Principle of mixed display prohibition

It is prohibited to possess a mixed display function in receiver units. This “mixed display” refers to such cases where a content supplier, to another content supplied by another supplier, purposely simultaneously displays their own content in relation to the other content, or such cases where different contents by multiple suppliers are displayed by influencing the display, etc. in order to mislead viewers into thinking that they are the same contents. The function which enables such displays is called mixed display function.

9.4.3.2 Prohibition of simultaneous display on receiver units that only receive video and audio

With receiver units that only receive video and audio, broadcasting contents and other contents supplied by other providers should not be simultaneously displayed. They should only display broadcasting contents. To display contents supplied by other providers, the broadcasting screen should be turned off and a full-screen switch should be performed.

However, the following screens may be simultaneously displayed as exceptions.

- The following screens for e-mail applications
  - e-mail composition screen
  - e-mail send screen
  - e-mail reception notification excluding subject line

- e-mail sender, subject line and content of e-mail from senders whose address entry is in the address book of receiver unit

- Screens which are displayed by receiver unit specific applications which were implemented in the receiver unit at the time of purchase and do not execute communication. (Including when receiver unit specific application was updated using communication in order to execute bug fix, etc.) For the messages, etc. refer to section 9.4.3.3.

9.4.3.3 Simultaneous display of messages displayed by receiver unit

The messages displayed by receiver unit as the following examples may be simultaneously displayed with broadcasting screens.

- System status display for receiver units
- Set-up screen of receiver units
- Operation support information for users
- Reception notification of sound communication and transmitter phone number notification
- Supplemental information for transmission, such as display of entered phone number upon user calling operations
- Display of approval request displayed when user permission is necessary upon connecting to communication and user interface for user to approve.
- Alert for operations that may generate disadvantages for the user
- Other messages displayed by receiver units not by communication with a third person

9.4.3.4 Operation of broadcasting contents upon simulaneous display

- It is recommended for video and audio of broadcasting to continue presenting, even while displaying contents whose simultaneous display is indicated in section 9.4.3.3.

Appendix 1 Regards for NVRAM access

Refer to Appendix-6 of [Section 2]

Appendix 2 Module compression format

Refer to Appendix-2 of [Section 2]

Appendix 3 DTD for operation scope checking for basic service

```
<!-- ======== Broadcast Markup Language (BML) for Mobile DTD [OPERATABLE]
======== -->
<!ENTITY %      ContentType      "CDATA" >
<!ENTITY %      Charset          "CDATA" >
<!ENTITY %      Character        "CDATA" >
<!ENTITY %      LinkTypes        "NMTOKEN" >
<!ENTITY %      Number           "CDATA" >
<!ENTITY %      URI              "CDATA" >
<!ENTITY %      Script           "CDATA" >
<!ENTITY %      StyleSheet       "CDATA" >
<!ENTITY %      Text             "CDATA" >

<!-- ================ Character mnemonic entities
========================== -->

<!ENTITY quot    "&#34;" ><!-- double quote -->

<!ENTITY % Events.attrib
"onclick        %Script;        #IMPLIED
onkeydown       %Script;        #IMPLIED
onkeyup         %Script;        #IMPLIED"
>
<!ATTLIST form
onsubmit        %Script;        #IMPLIED
>
<!ATTLIST input
onchange        %Script;        #IMPLIED
>
```

```
<!ATTLIST body
onload          %Script;            #IMPLIED
onunload        %Script;            #IMPLIED
>
<!ATTLIST object
bml:onfocus         %Script;        #IMPLIED
bml:onblur          %Script;        #IMPLIED
bml:accesskey       %Character;     #IMPLIED
>

<!ENTITY % Core.attrib
"id ID #IMPLIED
class CDATA #IMPLIED
style %StyleSheet;          #IMPLIED"
>
<!ENTITY % Common.attrib
"%Core.attrib;
%Events.attrib;"
>

<!ENTITY % Inlstruct.class "br | span" >
<!ENTITY % Inline.class
"%Inlstruct.class;
| a"
>
<!ENTITY % Blkstruct.class "p | div | pre" >
<!ENTITY % Block.class
"%Blkstruct.class;"
>
<!ENTITY % Boxed.mix
"%Block.class;
| img
| object
| form
| input
| textarea"
```

```
>

<!ENTITY % BlkNoForm.mix
"%Block.class;
| img
| object
| input
| textarea"
>

<!ENTITY % Br.content "EMPTY" >
<!ELEMENT br %Br.content; >
<!ATTLIST br
%Core.attrib;
>
<!ENTITY % Span.content "( #PCDATA | br )*" >
<!ELEMENT span %Span.content; >
<!ATTLIST span
%Core.attrib;
>

<!ENTITY % Div.content "( %Boxed.mix; )*" >
<!ELEMENT div %Div.content; >
<!ATTLIST div
%Core.attrib;
>
<!ENTITY % P.content "( #PCDATA | %Inline.class; )*" >
<!ELEMENT p %P.content; >
<!ATTLIST p
%Core.attrib;
>

<!ENTITY % Pre.content "( #PCDATA | %Inline.class; )*" >
<!ELEMENT pre %Pre.content; >
<!ATTLIST pre
%Core.attrib;
```

```
>

<!ENTITY % Script.content "( #PCDATA )" >
<!ELEMENT script %Script.content; >
<!ATTLIST script
src              %URI;                #IMPLIED
>

<!ENTITY % A.content "( #PCDATA    | br )*" >
<!ELEMENT a %A.content; >
<!ATTLIST a
%Common.attrib;
href             %URI;                #IMPLIED
accesskey        %Character;          #IMPLIED
>

<!ENTITY % Img.content "EMPTY" >
<!ELEMENT img %Img.content; >
<!ATTLIST img
%Core.attrib;
src          %URI;            #REQUIRED
alt          %Text;           #REQUIRED
>

<!ENTITY % Object.content "EMPTY" >
<!ELEMENT object %Object.content; >
<!ATTLIST object
%Common.attrib;
data           %URI;                              #IMPLIED
type           %ContentType;                      #REQUIRED
bml:streamstatus ( stop | play | pause )          #IMPLIED
>

<!ENTITY % Form.content "( %BlkNoForm.mix; )+" >
<!ELEMENT form %Form.content; >
<!ATTLIST form
```

```
%Core.attrib;
action          %URI;           #REQUIRED
method          (get|post)          'get'
>

<!ENTITY % InputType.class
"( text | password | submit )"
>
<!ENTITY % Input.content "EMPTY" >
<!ELEMENT input %Input.content; >
<!ATTLIST input
%Common.attrib;
type                %InputType.class;           'text'
name                CDATA                       #IMPLIED
readonly            (readonly)                  #IMPLIED
value               CDATA                       #IMPLIED
maxlength           %Number;                    '40'
accesskey           %Character;                 #IMPLIED
>

<!ENTITY % Textarea.content "( #PCDATA )">
<!ELEMENT textarea %Textarea.content; >
<!ATTLIST textarea
%Common.attrib;
accesskey           %Character;         #IMPLIED
name                CDATA               #IMPLIED
readonly            (readonly)          #IMPLIED
>

<!ENTITY % Title.content "( #PCDATA )" >
<!ELEMENT title %Title.content; >
<!ENTITY % Meta.content "EMPTY" >
<!ELEMENT meta %Meta.content; >
<!ATTLIST meta
name            NMTOKEN         #IMPLIED
content         CDATA           #REQUIRED
```

```
>

<!ENTITY % Link.content "EMPTY" >
<!ELEMENT link %Link.content; >
<!ATTLIST link
href        %URI;          #IMPLIED
>

<!ENTITY % Head.content "( title, meta?, link?,   script*, bml:bevent? )" >
<!ELEMENT head %Head.content; >

<!ENTITY % Body.content
"( div | form | p | pre )+"
>
<!ELEMENT body %Body.content; >
<!ATTLIST body
%Core.attrib;
>
<!ENTITY % Html.content "( head, body )" >
<!ELEMENT html %Html.content; >

<!ENTITY % Bevent.content "( bml:beitem )+" >
<!ELEMENT bml:bevent %Bevent.content; >
<!ATTLIST bml:bevent id ID #IMPLIED>
<!ENTITY % BMLEventType
"(EventMessageFired|ModuleUpdated|ModuleLocked|TimerFired
|DataEventChanged|MediaStopped|MainAudioStreamChanged)"
>
<!ENTITY % BMLTimeMode
"(absolute|origAbsolute)"
>
<!ENTITY % Beitem.content "EMPTY" >
<!ELEMENT bml:beitem %Beitem.content; >
<!ATTLIST bml:beitem
Id                  ID                  #REQUIRED
type                %BMLEventType;      #REQUIRED
```

```
onoccur              %Script;              #REQUIRED
es_ref               %URI;                 #IMPLIED
message_group_id     (0 | 1)               '0'
message_id           %Number;              #IMPLIED
message_version      %Number;              #IMPLIED
module_ref           %URI;                 #IMPLIED
time_mode            %BMLTimeMode;         #IMPLIED
time_value           CDATA                 #IMPLIED
object_id            CDATA                 #IMPLIED
subscribe            (subscribe)           #IMPLIED
>

<!-- End of BML DTD -->
```

Appendix 4 Default style sheet

```
@media tv {
/* margin */
div, p, pre, form, input, textarea, object, img { margin: 0 !important }
/* padding */
div, form, object, img { padding-top: 0 !important; padding-right: 0 !important; padding-bottom: 0 !important; padding-left: 0 !important }
/* border */
div, p, pre, form, input, textarea { border-width: 0; border-top-color: transparent; border-right-color: transparent; border-bottom-color: transparent; border-left-color: transparent; }
object, img { border-width: 0 !important; border-style: none !important }
/* display */
html, head, title, meta, script, link, bevent, beitem { display: none !important }
body, div, pre, form, input, textarea, object, img { display: block !important }
p { display: block }
br, span, a { display: inline !important }
/* position */
div, p, pre, form, input, textarea, object, img { position: absolute !important }
br, span, a { position: static !important }
/* top, left, width, height */
div, p, pre, form, input, textarea, object, img { top: 0; left: 0; width: 0; height: 0 }
/* z-index */
body, div, p, pre, br, span, a, form, input, textarea, object, img { z-index: auto !important }
/* line-height */
br, span, a { line-height: inherit !important }
/* visibility */
body { visibility: visible !important }
span, a { visibility: inherit !important }
/* overflow */
div, p, pre, form, input, textarea, object, img { overflow: hidden !important }
/* color */
p, pre, input, textarea { color: black }
span, a { color: inherit }
/* background-color */
object, img { background-color: transparent !important }
body { background-color: white }
/* background-repeat */
body { background-repeat: repeat !important }
/* font-family */
p, pre, span, a, input, textarea { font-family: "round gothic " !important }
/* text-align */
p, input, textarea { text-align: left }
/* white-space */
p, input { white-space: normal !important }
pre, textarea { white-space: pre !important }
/* pixel size */
body { pixel size: 240x480 !important }
/* marquee */
p { -wap-marquee-loop: 1; -wap-marquee-dir: rtl !important }
}
```

Appendix 5 Operation scope for browser pseudo-object in Browser for the C-profile contents on the internets

Operated as Table Appendix 5-1.

Descriptions in the “operation” column are as follows.

“O” Basic function in this specification.

“O(*1)” Optional function in this specification. Therefore, when utilizing these functions in contents, inspect the availability of the concerned function in the receiver by the getBrowserSupport() function, and call the concerned function only when it is available.

“O(*2)” Basic function in this specification in principle. However, with receivers which cannot utilize communication functions in BML browsers, implementation of these functions is not necessary. If the function is called, failure is returned as the return value.

“O(*3)” Necessary for receivers that is available the partial TS playback function. When utilizing these functions in contents, inspect availability of the concerned function in receivers by the getBrowserSupport() function, and call the concerned function only when it is available.

“X” Execution is prohibited in broadcasting status, link status and by Browser for the C-profile contents on the internets.
If the function is called, failure is returned as the return value. Refer to 8.3.11.4 for specification of failure behaviours.

“-” Neither a basic function nor an optional function in this specification. When the function is called, the receiver will occur an error.

Table Appendix 5-1 Operation scope of browser pseudo-objects

	Function	Operations in Browser for the C-profile contents on the internet	Notes			
			Others	Operations in broadcasting status	Operations in link status	(Note in original table)
Ureg function						
	Ureg[]	X		O	O	
Greg function						
	Greg[]	O		O	O	Greg lifetime shall be the time period while reception function is operating

	Function	Operations in Browser for the C-profile contents on the internet	Notes			
			Others	Operations in broadcasting status	Operations in link status	(Note in original table)
EPG functions						
	epgGetEventStartTime()	O		O	O	
	epgGetEventDuration()	O		O	O	
	epgTune()	-		-	-	
	epgTuneToComponent()	-		-	-	
	epgTuneToDocument()	-		-	-	
	epgIsReserved()	-		-	-	
	epgReserve()	O(*1)		O(*1)	O(*1)	
	epgCancelReservation()	O(*1)		O(*1)	O(*1)	
	epgRecIsReserved()	-		-	-	
	epgRecReserve()	-		-	-	
	epgRecCancelReservation()	-		-	-	
program group index functions						
	grpIsReserved()	-		-	-	
	grpReserve()	-		-	-	
	grpCancelReservation()	-		-	-	
	grpRecIsReserved()	-		-	-	
	grpRecReserve()	-		-	-	
	grpRecCancelReservation()	-		-	-	
	grpGetNodeEventList()	-		-	-	
	grpGetERTNodeName()	-		-	-	
	grpGetERTNodeDescription()	-		-	-	
	epgXTune()	-		-	-	
Series reservation functions						
	seriesIsReserved()	-		-	-	
	seriesReserve()	-		-	-	
	seriesCancelReservation()	-		-	-	
	seriesRecIsReserved()	-		-	-	
	seriesRecReserve()	-		-	-	
	seriesRecCancelReservation()	-		-	-	
Non-volatile memory functionss						
	readPersistentString()	-		-	-	
	readPersistentNumber()	-		-	-	
	readPersistentArray()	X		O	O	
	writePersistentString()	-		-	-	
	writePersistentNumber()	-		-	-	
	writePersistentArray()	X		O	O	
	copyPersistent()	-		-	-	
	getPersistentInfoList()	-		-	-	
	deletePersistent()	-		-	-	
	getFreeSpace()	-		-	-	
Functions for controlling access-controlled areas						
	isSupportedPersistentType()	-		-	-	
	setAccessInfoOfPersistentArray()	-		-	-	
	checkAccessInfoOfPersistentArray()	-		-	-	
	writePersistentArrayWithAccessCheck()	-		-	-	
	readPersistentArrayWithAccessCheck()	-		-	-	

	Function	Operations in Browser for the C-profile contents on the internet	Notes			
			Others	Operations in broadcasting status	Operations in link status	(Note in original table)
Interaction channel functions						
Interaction channel functions - Delayed call functions						
	registerTransmission()	-		-	-	
	registerTransmissionStatus()	-		-	-	
	getTransmissionStatus()	-		-	-	
	setDelayedTransmissionDataOverBasic()	-		-	-	
Interaction channel functions - Communication Functions assuming BASIC procedure						
	connect()	-		-	-	
	disconnect()	-		-	-	
	sendBinaryData()	-		-	-	
	receiveBinaryData()	-		-	-	
	sendTextData()	-		-	-	
	receiveTextData()	-		-	-	
Interaction channel functions - Communication functions assuming TCP/IP						
	setISPParams()	-		-	-	
	getISPParams()	-		-	-	
	connectPPP()	-		-	-	
	connectPPPWithISPParams()	-		-	-	
	disconnectPPP()	-		-	-	
	getConnectionType()	O(*2)		O(*2)	O(*2)	
	isIPConnected()	O(*2)		O(*2)	O(*2)	
	saveHttpServerFileAs()	-		-	-	
	saveHttpServerFile()	-		-	-	
	sendHttpServerFileAs()	-		-	-	
	saveFtpServerFileAs()	-		-	-	
	saveFtpServerFile()	-		-	-	
	sendFtpServerFileAs()	-		-	-	
	sendTextMail()	-		-	-	
	sendMIMEMail()	-		-	-	
	transmitTextDataOverIP()	O(*2)	User approval shall be acquired by the receiver prior to implementation. Transmission shall not be implemented without approval.	O(*2)	O(*2)	
	setDelayedTransmissionData()	-		-	-	
	getTransmissionStatus()	-		-	-	
	getTransmissionResult()	-		-	-	
	setCacheResourceoverIP()	-		-	-	

	Function	Operations in Browser for the C-profile contents on the internet	Notes			
			Others	Operations in broadcasting status	Operations in link status	(Note in original table)
Interaction channel functions - Deleyed call functions applicable to BASIC procedure and IP connections						
	getDelayedTransmissionStatus()	-		-	-	
	getDelayedTransmissionResult()	-		-	-	
Interaction channel functions - Function for obtaining line connection status						
	getPrefixNumber()	-		-	-	
Interaction channel functions - Communication functions using the mass call reception service						
	vote()	-		--	-	
Interaction channel functions - Functions for encrypted communication using CAS						
	startCASEncryption()	-		-	-	
	transmitWithCASEncryption()	-		-	-	
	endCASEncryption()	-		-	-	
Interaction channel functions - Functions for communication with public key encryption not using CAS						
	setEncryptionKey()	-		-	-	
	beginEncryption()	-		-	-	
	endEncryption()	-		-	-	
Operational control functions						
	reloadActiveDocument()	O		O	O	
	getNPT()	-		-	-	
	getProgramRelativeTime()	X		O	O	
	isBeingBroadcast()	-		-	-	
	lockExecution()	-		-	-	
	unlockExecution()	-		-	-	
	lockModuleOnMemory()	-		-	-	
	unlockModuleOnMemory()	-		-	-	
	setCachePriority()	-		-	-	
	getTuningLinkageSource()	-		-	-	
	getTuningLinkageType()	-		-	-	
	getLinkSourceServiceStr()	-		-	-	
	getLinkSourceEventStr()	-		-	-	
	getIRDID()	-		-	-	
	getBrowserVersion()	O		O	O	
	getProgramID()	X		O	O	
	getActiveDocument()	O		O	O	
	lockScreen()	O		O	O	
	unlockScreen()	O		O	O	
	getBrowserSupport()	O		O	O	

	Function	Operations in Browser for the C-profile contents on the internet	Notes			
			Others	Operations in broadcasting status	Operations in link status	(Note in original table)
	launchDocument()	O	Only those name spaces starting with http:// and https:// can be specified.	O	O	
	launchDocumentRestricted()	-		-	-	
	quitDocument()	O		O	O	
	launchExApp()	-		-	-	
	getFreeContentsMemory()	-		-	-	
	isSupportedMedia()	-		-	-	
	detectComponent()	X		O	O	
	lockModuleOnMemoryEx()	X		O	X	
	unlockModuleOnMemoryEx()	X		O	O	
	unlockAllModulesOnMemory()	X		O	O	
	getLockedModuleInfo()	X		O	O	
	getBrowserStatus()	O		O	O	
	getResidentAppVersion()	O		O	O	
	isRootCertificateExisting()	-		-	-	
	getRootCertificateInfo()	-		-	-	
	startResidentApp()	-		-	-	
	getDataDisplayAreaSize()	-		-	-	
	setFullDataDisplayArea()	X		O(*1)	O(*1)	
Receiver sound control						
	playRomSound()	O		O	O	
Timer functions						
	sleep()	-		-	-	
	setTimeout()	-		-	-	
	setInterval()	O		O	O	
	clearTimer()	O		O	O	
	pauseTimer()	-		-	-	
	resumeTimer()	-		-	-	
	setCurrentDateMode()	X		O(*3)	O(*3)	
External character functions						
	loadDRCS()	-		-	-	
	unloadDRCS()	-		-	-	
Functions for controlling external devices						
	enumPeripherals()	-		-	-	
	passXMLDocToPeripheral()	-		-	-	
Other functions						
	random()	O		O	O	
	subDate()	O		O	O	
	addDate()	O		O	O	
	formatNumber()	O		O	O	
Subtitle presentation control functions						
	setCCStreamReference()	-		-	-	
	getCCStreamReference()	-		-	-	
	setCCDisplayStatus()	-		-	-	
	getCCDisplayStatus()	-		-	-	

	Function	Operations in Browser for the C-profile contents on the internet	Notes			
			Others	Operations in broadcasting status	Operations in link status	(Note in original table)
	getCCLanguageStatus()	-		-	-	
Directory management functions						
	saveDir()	-		-	-	
	saveDirAs()	-		-	-	
	createDir()	-		-	-	
	getParentDirName()	-		-	-	
	getDirNames()	-		-	-	
	isDirExisting()	-		-	-	
File management functions						
	saveFile()	-		-	-	
	saveFileAs()	-		-	-	
	getFileNames()	-		-	-	
	isFileExisting()	-		-	-	
File input/output functions						
	writeArray()	-		-	-	
	readArray()	-		-	-	
Inquiry functions						
	getDirInfo()	-		-	-	
	getFileInfo()	-		-	-	
	getCarouselInfo()	-		-	-	
	getModuleInfo()	-		-	-	
	getContentSource()	-		-	-	
	getStorageInfo()	-		-	-	
Data carousel storage functions						
	saveCarouselAs()	-		-	-	
	saveCarousel()	-		-	-	
	saveModuleAs()	-		-	-	
	saveModule()	X		O(*1)	O(*1)	
	saveResourceAs()	-		-	-	
	saveResource()	X		O(*1)	O(*1)	
Functions for controlling bookmark area						
	writeBookmarkArray()	-		-	-	
	readBookmarkArray()	-		-	-	
	deleteBookmark()	-		-	-	
	lockBookmark()	-		-	-	
	unlockBookmark()	-		-	-	
	getBookmarkInfo()	-		-	-	
	getBookmarkInfo2()	-		-	-	
	startResidentBookmarkList()	-		-	-	
Printing functions API - printing basic functions						
	getPrinterStatus()	-		-	-	
	printFile()	-		-	-	
	printTemplate()	-		-	-	
	printUri()	-		-	-	
	printStaticScreen()	-		-	-	
Printing functions API - memory card functions						

	Function	Operations in Browser for the C-profile contents on the internet	Notes			
			Others	Operations in broadcasting status	Operations in link status	(Note in original table)
	saveImageToMemoryCard ()	X		O(*1)	O(*1)	
	saveHttpServerImageToMemoryCard()	X		O(*1)	O(*1)	
	saveStaticScreenToMemoryCard()	-		-	-	
Digital terrestrial broadcasting specific functions						
	X_DPA_mailTo()	X		O(*1)	O(*1)	
	X_DPA_startResidentApp( )	O		O	O	
	X_DPA_phoneTo()	X		O(*1)	O(*1)	
	X_DPA_getRcvCond ()	X		O(*1)	O(*1)	
	X_DPA_getCurPos ()	X		O(*1)	O(*1)	
	X_DPA_saveExAppFile()	X		O(*1)	O(*1)	
	X_DPA_startExAV()	X		O(*1)	O(*1)	
	X_DPA_stopExAV()	X		O(*1)	O(*1)	
	X_DPA_tuneWithRF()	O(*1)	Finish browser after implementation	O(*1)	O(*1)	
	X_DPA_writeSchInfo()	X		O(*1)	O(*1)	
	X_DPA_getComBrowserUA ()	O(*2)		O(*2)	O(*2)	
	X_DPA_writeAddressBookInfo()	X		O(*1)	X	
	X_DPA_launchDocWithLink()	X		X	O(*2)	
	X_DPA_chkAVtype()	X		O(*1)	O(*1)	
	X_DPA_getIRDID()	X		O	O	
	X_DPA_writeCproBM()	X		O(*2)	O(*2)	

Appendix 6 Operation of the getBrowserStatus() argument

The table below shows functions available for inspections in combination of “statusname” and “additionalinfo” when “sProvider” is "DPA".

Table Appendix 6-1 String specified as getBrowserStatus() arguments

statusname	additionalinfo	Operations of getBrowserStatus()
IRDState	One of the following "Broadcast" "Link" "UnLink"	If browser is in the status specified by “additionalinfo”, return 1. "Broadcast": data broadcasting reception status "Link": link status "UnLink": status other than data broadcasting status and link status

Appendix 7 Communication restrictions for carriers

With mobile phones, there are times when the carriers has communication restrictions that are not specified in the this specification, based on agreements with the broadcaster. Specifically, limited domains available for communication from broadcasting contents per broadcaster affiliation to one, and limited communication from contents presented in BML browsers to be URI's under this domain (including sub-domains). (However, in cases like presenting contents by starting up browsers for the contents on the internet by X_DPA_startResidentApp(), if another application, started up from a BML browser API, performs communication, this domain restriction does not apply.)

A communication restriction is implemented as receiver features processing, using tables (correspondence table of domain name and affiliation ID that specifies affiliation) pre-stored in mobile phones. Therefore, when mobile phones communicate from contents presented in BML browsers, communication with servers outside of the domain specified by the table will fail, even if the communication destination is under the base URI. Also, as a result of this communication restriction, NVRAM areas for the affiliation will no longer be accessed from contents other than contents under the specified domain.

Such communication restrictions are not necessary for receivers other than mobile phones. However, broadcasters cannot avoid arbitrary receivers performing similar communication restrictions. If the communication destination will be limited, a table showing the correspondence between affiliation ID for identifying the affiliation and domain names can be acquired from the governing organization shown in Appendix 13.

Also, only with operations of this restriction, the direct specification of IP addresses as URI host names will result in a communication failure as servers outside of the domain.

Appendix 8 Arrangement of elementary stream and empty carousel operations in the PMT

In C-profile, the visible area for data broadcasting is defined in an independent area. Therefore, unlike in profile A, allocation and release of the area is assumed in C-profile. That is to say, specifications specified in Table Appendix 8-1 can be assumed as operation example of receivers here. In this example, receivers control the allocation and release of the display area by description of default components (component_tag=0x80) in the PMT.

In consideration of this example, when data broadcasting contents are not transmitted for a certain period of time, such as programs, it is effective for the receiver to delete the default component in the PMT, if this is to avoid securing unnecessary visible area for data broadcastings for the receiver, and to avoid letting the receiver to continue to receive data.

However, if the description of the default component in PMT changes in a short period of time, it may inconvenience users due to the blink of the visible area for data broadcasting. Transmission of empty carousels can be assumed as a countermeasure for this blink in the visible area for data broadcasting, if the time period is too short to not transmit data broadcasting contents that should be presented. Here, we need to be careful since there is a possibility that it will cause receivers to secure unnecessary visible area for data broadcastings, if empty carousels are transmitted for a long period of time.

Considering the above points, whether there is description of the default component in the PMT or not, and the transmission of empty carousel recommend to be operated with great care.

Also, this appendix is described in order to specify general reference information upon designing and operation of transmission systems. This appendix do not specify specific restrictions for receiver specifications, transmission system specifications and transmission operations.

Table Appendix 8-1 Relationship examples for reception conditions and the visible area for data broadcasting

Item	Reception condition			Presentation status			
				Presentation of broadcasting contents		Presentation of link contents	
	description for default component in PMT	Carousel status	Other causes	Presentation area	Presentation contents	Presentation area	Presentation contents
Upon tuning							
	No	-	-	No	-	-	-
	Yes	Empty	-	Yes	Black screen or built-int wallpaper	-	
	Yes	Not empty	-	Yes	Presentation of startup document	-	
Data event update							
(data_event_id change)	Yes	Not empty -> not empty	-	Yes	Present startup document of data event after update	Yes	Continue presenting document in presentation
	Yes	Not empty -> empty	-	Yes	Black screen or built-in wallpaper	Yes	Continue presenting document in presentation
	Yes	Empty -> empty	-	Yes	Black screen or built-in wallpaper	Yes	Continue presenting document in presentation
	Yes	Empty -> not empty	-	Yes	Present data event startup document after update	Yes	Continue presenting document in presentation
Upon PMT update							
(Change in description for default component)	Yes -> No	-	-	No	-	Yes	Continue presenting document in presentation
	No -> Yes	-	-	Yes(*1)	Startup document	Yes	Continue presenting document in presentation
Change in broadcasting reception status							
(RF level, PMT reception, DII reception, etc.)	-	-	Good -> Ng	No change	No change	No change	Continue presenting document in presentation
	-	-	Bad -> Ng	No change	Present latest content (*2)	No change	Continue presenting document in presentation

- : Cause or symptom not assumed

(*1) Operations worth on 0x80 existence upon tuning

(*2) If a data event update and PMT update is detected after recovering the reception status, standardize on normal process.

Mandatory reloading or startup document transition is not assumed.

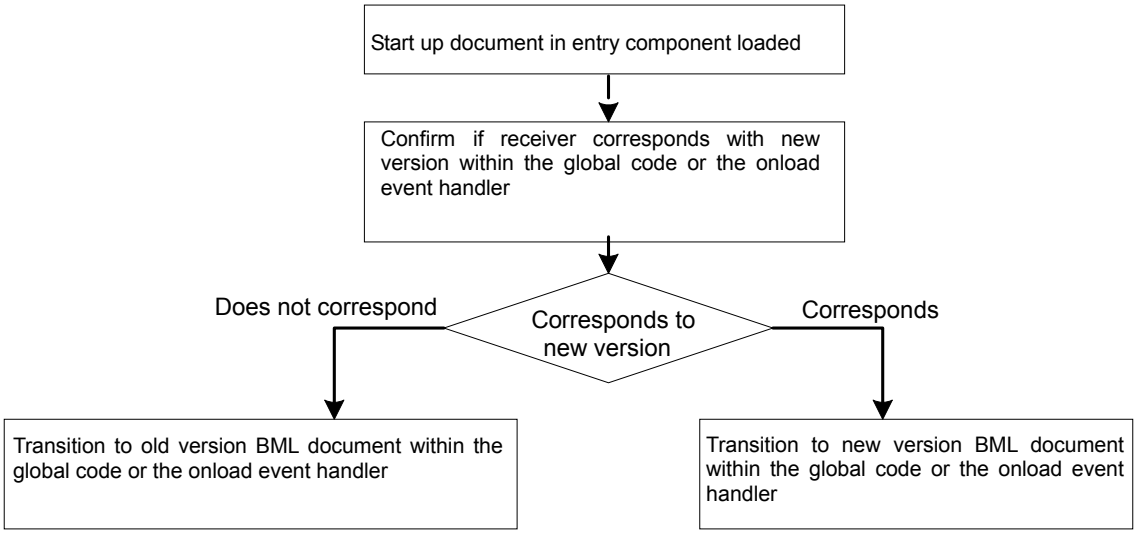
Appendix 9 Version update of the BML version

Because BML updates are assumed along the development of services, assumed examples for receiver behaviors and operations regarding BML version updates are defined below. Behavior necessary for current version receivers(major version=12, minor version=0) and operation of version numbers are specified based on the assumptions.

1. Assumed examples of contents operations when the different versions are transmitted

◆ If the “bml_major_version” of Data Component Descriptor in PMT is 12:

A. If the startup document is shared



B. If the startup document is not shared within the same component

- Standardize the new version startup document with a name that is different from "startup.bml”.
- Transmit multiple startup documents corresponding to each version to the same component

◆ If the “bml_major_version” of Data Component Descriptors in PMT are 12 and the other number:

C. When transmitting startup documents in another component

- Specify a new version entry component by component tag value other than 0x80.
- Transmit to the same TS with an arrangement of elementary stream that includes entry components corresponding to each version.

2. Behaviors necessary for current version receivers (major version = 12, minor version = 0)

The following specifies receiver behaviors in order to avoid occurring abnormal behaviors when assuming transmission of multiple version contents upon version updates specified in A-C above.

- An error shall not be occurred even if a BML document, whose version's operation does not correspond with the receiver, is included in components in presentation.
- An error shall not be occurred even if a resource, whose media type's operation does not correspond with the receiver, is included in components in presentation.
- An error shall not be occurred if a BML document, whose minor_version only is different, is presented. Operations of non-supported functions shall be implementation dependent.
- Behavior upon presentation of a BML document, whose major_version is different, is implementation dependent.
- Behavior upon presentation of resources of a media type which is not supported by receiver is implementation dependent.
- Startup "startup.bml" first and start contents presentation (in case of B).
- Start the startup document from components whose component tag value is 0x80 and start contents presentation (in case of C)
- Even if it is an ES with a undefined component tag value, it shall not be an error (in case of C).
- Behavior upon reception other than "bml_major_vesion" of Data Component Descriptor in PMT is implementation dependent (if major_vesion=12 is not included)

3. Operation of version numbers

- Minor version updates shall be limited to functions that can be implemented without failure behaviour even in older version receivers.
- Major version updates for functions that cannot be implemented without a failure behaviour even in older version receivers.
- The implementation shall be discussed with the governing organization described in Appendix 13.

Appendix 10 Guideline concerning the operation of personal information in NVRAM and communications in data broadcasting

1. Definition of personal information in data broadcasting

Under these guidelines, personal information shall mean "information about a living individual which can identify the specific individual by name, date of birth or other description contained in such information (including such information as will allow easy reference to other information and will thereby enable the identification of the specific individual)" as prescribed in Article 2 (1) of the "Act on the Protection of Personal Information".

2. Handling of information stored in NVRAM in data broadcasting

- Information writing/reading shall be performed exclusively by contents supplied by broadcasters.
- Personal information shall be handled upon user agreement.
- Stored information belongs to the user.
- Data such as “points” used for prizes/games/premium exchanges, etc. shall be handled carefully, from the viewpoint of user profit security.

3. Handling of personal information in TCP/IP communication

- TLS1.0 / SSL3.0 shall be used when handling personal information in Interactive Data Broadcasting Services and/or C-profile linked contents.

4. Guideline concerning the operation of personal information in data broadcasting content

- Utilization scope and objectives of personal information shall clearly be presented.
- Personal information shall be utilized under the permission of the viewer.

5. Guidelines for privacy policy (including “FAQ” and “help”, etc.) by data broadcasting contents

- “privacy policy” shall be specified per broadcaster, and shall be released by a standard procedure that can be acquired by viewers.
- It is recommended to describe the following items in “privacy policy”
  (1) Registered information is in receiver’s memory.
  (2) Personal information in receiver’s memory shall be managed in viewer’s responsibility, and needs to be erased by the viewers themselves upon transfer or disposal of the receiver.
  (3) Personal information shall be utilized only in the utilization scope and objectives specified by broadcater.
  (4) A disclaimer is necessary for data deletion.

Appendix 11 Guideline concerning the cache function and URI history

The following shows the guideline concerning the cache function of the receiver, storage of the URI history and transitions based on the URI history. In the case of mobile phone, separately defined prescriptions shall be followed if they exist.

1. Cache function in the receiver unit

If the receiver will implement a function for cacheing C-profile linked content via HTTP protocol, the receiver shall operate by referencing the following HTTP headers and fields.

- Operation of Cache-Control
  - In the event when the receiver receives a response message with "no-store" specified in the Cache-Control header from the interactive Web server, the receiver must not cache the file specified in the HTTP session.
  - In the event when the receiver receives a response message with "no-cache" specified in the Cache-Control header from the interactive Web server, the receiver shall operate in the same manner as when "no-store" has been specified.
  - The operation of the receiver when it receives values other than "no-cache" and "no-store" shall be implementation dependent.

- Operation of Pragma

  In the event when the receiver receives a response message with "no-cache" specified in the Pragma field from the interactive Web server, the receiver shall operate in the same manner as when the above "Cache-Control:no-cache" has been specified.

- Operation of other headers

  In addition to the cache function, the receiver shall also support "If modified since" of the request header as a means for reducing the communication transactions. The date/time format shall be the fixed length subset defined in RFC1123; and the receiver shall be able to interpret this format.

- Operation of cache-clear

  Cached content shall always be cleared if any of the following operations occurs:
  - When the channel has been switched
  - When the BML browser has terminated
  - In the case of a transition from C-profile linked content to broadcasting content

    * However, a receiver supporting "If modified since" shall not clear the cache if a transition is made from C-profile linked content to broadcasting content.

2. Storage of URI history by the receiver function and transitions based on the URI history

This document does not assume the receiver to have the function of recording the URI of displayed documents as URI history and allowing transitions to be made directly from URI history via user operation; and it has been created based on the premise that the broadcasted content does not have such a function either. For this reason, unexpected transitions may potentially occur on a receiver implementing the relevant function, causing the presentation of some content to fail. Therefore, it is not recommendable to implement such type of function. If such type of function will be implemented based on product planning, the following restrictions must at least be observed.

- Restrictions on transitions based on the URI history

  Transitions based on the URI history shall be limited to within C-profile linked content and must not be performed for data broadcasting content.

- Restriction regarding the storage of URI history

  URI history shall be cleared if any of the following operations occurs:

  - When the channel has been switched
  - When the BML browser is terminated
  - In the case of a transition from C-profile link content to broadcasting content

Appendix 12 Details of the communication function

The details of the communication function are as follows.

In the case of mobile phone, separately defined prescriptions shall be followed if they exist.

1. Operation of methods
   - If "http:" has been specified in the URI, the receiver and the interactive Web server shall perform communication based on HTTP/1.1 using the port specified by the URI.
   - If "https:" has been specified in the URI, the receiver and the interactive Web server shall perform encrypted communication based on HTTP/1.1 after having established a connection via TLS1.0 or SSL3.0 using the port specified by the URI.
   - If the port number was not specified in the URI, ports 80 and 443 shall be used for "http:" and "https:" respectively.

2. Operation of HTTP/1.1
   - The fixed length subset defined in RFC1123 shall be operated for the date/time formats.
     - It shall be desirable that the interactive Web server transmits dates to the receiver only in the date/time format of the fixed length subset defined in RFC1123.
     - The receiver shall be able to interpret the fixed length subset defined in RFC1123 as the date/time formats. In the event when a date in RFC1036 or ANSI C format is received, it shall be desirable that they be interpreted but they may also be ignored.

       Examples:

       Sun, 06 Nov 1994 08:49:37 GMT ; RFC1123
       Sunday, 06-Nov-94 08:49:37 GMT ; RFC1036
       Sun Nov 6 08:49:37 1994 ; ANSI C
   - Only the "Shift_JIS" character set shall be operated in request messages, response messages and entities. The operation of the receiver when it receives other character sets shall be implementation dependent.
   - "identity" shall be operated for content coding (content-coding). "deflate" shall be operated as an option. Operations of the receiver not supporting "deflate" when it receives "deflate" and when it receives any other values shall be implementation dependent.
   - "chunked" shall be operated if transfer coding (transfer-coding) will be specified. The operation of the receiver when it receives other values shall be implementation dependent.
   - If the browser name and version will be specified, the product token shall be consistent with the return value of getBrowserVersion()
   - Operation of quality values shall be implementation dependent.

- "ja" shall be operated for the language tag. The operation of the receiver when it receives other language tags shall be implementation dependent.
- The receiver shall implement "Keep-Alive".

3. Security

To protect personal information, the receiver shall implement the SSL3.0 and TLS1.0 protocols for encrypted communication.

(1) Cipher Suite

- The receiver shall incorporate the cipher suite described in Table Appendix 12-1. However, incorporation of cipher suites other than below shall be implementation dependent.

Table Appendix 12-1 List of cipher suites to be supported by the receiver

Cipher Suite	Key exchange	Data encryption	Hash
TLS_RSA_WITH_DES_CBC_SHA	RSA	DES	SHA
TLS_RSA_WITH_3DES_EDE_CBC_SHA	RSA	Triple-DES	SHA
SSL_RSA_WITH_DES_CBC_SHA	RSA	DES	SHA
SSL_RSA_WITH_3DES_EDE_CBC_SHA	RSA	Triple-DES	SHA

(2) Authentication function

- The receiver shall support server authentication in order to prevent "phishing" in various services.

(3) Certificate verification items

- The certificate of the interactive Web server shall have less than 5 hierarchical levels. The receiver shall be able to verify at least 5 certificates.
- The certificate shall comply with RFC3280, and it shall be verified according to the verification content shown in Table appendix 12-2. However, signature verification shall not be required for root certificates.

Table Appendix 12-2 Certificate verification content

Verification item	Root Certificate	Intermediate certificate	Server certificate
Basic Certificate Field			
Signature (signatureValue)	-	Verification shall be required.	Verification shall be required.
Version (version)	-	The version shall be 3.	The version shall be 3.
Issuer (issuer's DN)	-	It shall be a non-empty distinguished name (DN) matching the Subject of the certificate in the one higher level.	It shall be a non-empty distinguished name (DN) matching the Subject of the certificate in the one higher level.

Verification item	Root Certificate	Intermediate certificate	Server certificate
Expiration period (validity)	It shall contain the current time.	It shall contain the current time.	It shall contain the current time.
CN (CommonName) of Subject (Subject's DN)	-	-	It must match the FQDN of the access server.
Certificate Extensions -> They shall be confirmed when the following fields exist and are critical.			
Basic constraints (Basic Constraints)	—	【Conditions】 If "Basic Constraints" is critical: ・ "cA" must be true. ・ If "pathLenConstraint" is included, its value shall meet the depth restriction of the certificate chain.	—
Key application (KeyUsage)	—	—	【Conditions】 If "KeyUsage" is critical: Either "keyEncipherment" or "keyAgreement" shall be included.
Extension key application (ExtendedKeyUsage)	—	—	【Conditions】 If "ExtendedKeyUsage" is critical: "id-kp-serverAuth" shall be included.

* "-" indicates that verification is not required.

(4) In the event of TLS and/or SSL error

- The receiver shall display an alert message in the event when a TLS or SSL error occurs. The message display method shall be implementation dependent, but the message examples in Table Appendix 12-3 shall desirably be followed.
- The receiver shall not connect to the interactive Web server in the event when a TLS or SSL error occurs.

Table Appendix 12-3 Examples of alert messages to be displayed when a TLS or SSL error occurs

	Cause	Message example
1	If a root certificate is not set on the receiver side	"No root certificate is set on the receiver side. Connection cannot be established because the safety of the connection destination cannot be verified."
2	If a root certificate is set on the receiver side, but cannot be verified against the server certificate at the connection destination	"Connection cannot be established because the safety of the connection destination cannot be verified using the currently set root certificate."
3	If the root certificate is expired	"Connection cannot be established because the safety of the connection destination cannot be verified using the currently set root certificate."

	Cause	Message example
4	In the case of an authentication processing error due to network failure, etc. (i.e. a timeout process)	"Connection could not be established within the set time limit."
5	If the certificate at the connection destination is expired	"The certificate at the connection destination is expired. Connection cannot be established because the safety of the connection destination cannot be verified."
6	In the case of mismatch CommonName of the server certificate does not match	"The certificate at the connection destination does not contain the title of the page to be displayed. Connection cannot be established because the safety of the connection destination cannot be verified."
7	If the certificate is disabled on the receiver side	"The root certificate for authenticating the connection destination is currently disabled on the receiver side. Please enable the root certificate and try connecting again."
8	If the certificate is tampered with	"Connection will be aborted because falsified certificate has been detected at the connection destination."
9	In the case of an authentication error due to invalid chain, etc.	"Connection will be aborted because a problem has been detected in the certificate setting at the connection destination."

Appendix 13 Governing organization

The Association for Promotion of Digital Broadcasting
(http://www.dpa.or.jp/)

[Section 5] Operational provisions of common digital receiver units for terrestrial/BS/wide bandwidth CS

1 Introduction

[Section 5] provides functions required from digital terrestrial data broadcasting reception of common digital receiver units (common receiver unit hereafter) for terrestrial/BS/wide bandwidth CS.

2 Functions required in common receiver units

Common receiver units should at least be equipped with functions described in TR-B14 and TR-B15 Part 2 regarding the media they are capable of receiving (terrestrial/CS/BS).

2.1 RAM

Greg should be shared in all receivable media. Greg values should be kept even while presenting communication contents that do not support Greg. It is recommended to keep Greg values even while presenting other media that do not support Greg.

If Greg values cannot be kept for some reason during presentation of other media, etc., the receiver unit should initialize a Greg value upon first presenting media that support Greg afterwards.

2.2 NVRAM

Bookmark areas, registration calling areas and general-purpose root certificate areas should be shared with each media.

Access availability from other media in NVRAM areas for each media is as provided in Table 2-1.

Table 2-1 NVRAM's inter-media correlation in common receiver units

Media	Area name	Access from BS	Access from CS	Access from terrestrial
BS	Broadcaster common area	RW	×	R
	Broadcaster specific area	RW	×	×
	Broadcaster specific, broadcasting and communication common area (2KB per broadcaster)	RW (Only BML3.0)	×	×
CS	Broadcaster common area	×	RW	×
	Broadcaster specific area	×	RW	×

	Broadcaster specific, broadcasting and communication common area	×	RW (Only BML3.0)	×
Terrestrial	Broadcaster common area	R (Only BML3.0))	×	RW
	Broadcaster specific area	×	×	RW
	Affiliate-specific area	RW (Only BML3.0) (Only BS broadcasting companies defined as same affiliate by the terrestrial Extended Broadcaster Descriptor)	×	RW
	Broadcaster specific, broadcasting and communication common area	×	×	RW
Common	General-purpose root certificate area	RW	RW	RW
	Registration calling area	RW (Only level 3)	RW (Only level 3)	RW
	Bookmark area	RW (Only level 2 and 3)	RW	RW

RW : Reading and writing both available
R : Only reading available
× : Reading and writing both unavailable

3 Contents operation guidelines assuming common receiver units

This chapter provides guidelines for contents descriptions assuming common receiver units.

3.1 Judgment of common receiver units

Contents can confirm whether a receiver unit corresponds with media other than terrestrial digital media (i.e. is a common receiver unit) or not by utilizing isSupportedMedia(). To utilize functions unique to common receiver units such as NVRAM reference in other media, channel selection in other media, etc., it should always confirm that the receiver unit corresponds with the object media by utilizing isSupportedMedia() beforehand.

3.2 NVRAM access for other media

When receiver units correspond with media other than terrestrial digital media, that media can access the NVRAM area within the range described in 2.2. However, pre-confirmation by isSupportedMedia() is always necessary. Refer to TR-B15 Part 2 Vol. 3 “8.2 Operation of NVRAM for common use in MM service in BS” for operation of name spaces upon accessing the NVRAM area of BS digital.

3.3 Channel selection of other media services

If the receiver unit corresponds with media other than terrestrial digital media, channel selection can be performed for such media services by epgTune() and epgTuneToComponent(). Refer to ARIB STD-B24 Vol. 2 Appendix 1 “8.5.1 Operation of operational control functions” for details.

3.4 Viewing reservation and programmed recording of other media services

If the receiver unit corresponds with media other than terrestrial digital media, viewing reservation and programmed recording functions (epgIsReserved(), epgReserve(), epgCancelReservation(), epgRecIsReserved(), epgRecReserve(), epgRecCancelReservation()) for such media services can be implemented. However, there is no guarantee that the receiver unit will acquire the SI information of the object media, and reservation will fail without acquisition.

3.5 Information sharing between media

Since the Greg[] value is shared with all receivable media forms, it can be utilized for information delivery between contents broadcasted in different media forms. It can also be utilized for information delivery between communication contents and broadcasting contents.

<Intentionally blank>

OPERATIONAL GUIDELINES FOR
DIGITAL TERRESTRIAL TELEVISION BROADCASTING

ARIB TECHNICAL REPORT

ARIB TR-B14 VERSION 3.8-E1
(Fascicle 1)　(2/2)
(Dec 12, 2008)

This Document is based on the ARIB technical report of “Operational Guidelines for Digital Terrestrial Television Broadcasting” in Japanese edition and translated into English in June, 2011.

Published by

Association of Radio Industries and Businesses

Nittochi Bldg. 11F
1-4-1 Kasumigaseki, Chiyoda-ku, Tokyo 100-0013, Japan

TEL 81-3-5510-8590
FAX 81-3-3592-1103

Printed in Japan
All rights reserved