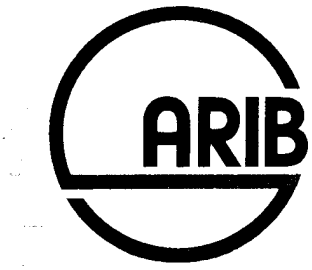


ENGLISH TRANSLATION

OPERATIONAL GUIDELINES FOR TERRESTRIAL MOBILE MULTIMEDIA BROADCASTING BY TRANSMISSION SYSTEM BASED ON CONNECTED SEGMENTS FOR VHF-HIGH BAND

ARIB TECHNICAL REPORT

ARIB TR-B33 Version 2.3

(Fascicle 1)

Established on March 28th, 2001
Revised on September 30th, 2015 Version 2.3

Association of Radio Industries and Businesses

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

1. Notes on Copyright

2. Notes on English Translation

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

- ARIB Standards and Technical Reports, in the original language, are made publicly available through web posting. 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.

Foreword

The Association of Radio Industries and Businesses (ARIB) investigates and summarizes the basic technical requirements for various radio systems in the form of “ARIB Standards”. These standards are developed with the participation of and through discussions amongst radio equipment manufacturers, telecommunication operators, broadcasting equipment manufacturers, broadcasters and users.

ARIB Technical Reports contain the concrete measurement methods, detailed explanation and remarks in respect to the operation and maintenance of the radio equipment and broadcasting equipment in order to ensure their compatibility and adequate quality, based on the ARIB Standards deriving from “governmental technical regulations” (mandatory standard) and “private technical standards” (voluntary standards).

This ARIB Technical Report is developed for the terrestrial television broadcasting and the terrestrial multimedia broadcasting by means of segment-connected transmission system using terrestrial basic stations through the use of radio waves with a frequency in the range of 207.5–222 MHz (VHF high band). In order to ensure fairness and transparency in the defining stage, the report was set by consensus at the ARIB Standard Assembly with the participation of both domestic and foreign interested parties from radio equipment manufacturers, telecommunication operators, broadcasting equipment manufacturers, broadcasters and users.

ARIB sincerely hopes that this ARIB Technical Report will be widely used by radio equipment manufacturers, telecommunication operators, broadcasting equipment manufacturers, broadcasters and users.

Integrated Contents

VOLUME 0

Basic Concept and Common Elements of Multimedia Broadcasting

Contents

Chapter 1 General Terms

1.1 Introduction

This volume describes the overview of the terrestrial television broadcasting and the terrestrial multimedia broadcasting by means of segment-connected transmission system using terrestrial basic stations through the use of radio waves with a frequency in the range of 207.5–222 MHz (VHF high band). (Hereinafter, when "multimedia broadcasting" is merely described without an explanatory note in this technical data, this shall mean "the terrestrial television broadcasting and the terrestrial multimedia broadcasting by means of segment-connected transmission system using terrestrial basic stations through the use of radio waves with a frequency in the range of 207.5–222 MHz (VHF high band)."

Note that a "program provider" and a "certified basic broadcaster" mean the same thing, as does a "consigned broadcaster" and a "basic broadcasting station provider."

<Blank Page>

Chapter 2 Service Overview

2.1 Multimedia Broadcasting

Multimedia broadcasting enables access to contents and services without caring about broadcast schedule or place of use, by expanding digital terrestrial television broadcasting through the combination with communications, based on the premise of the use of mobile receivers. There are two types of multimedia broadcasting: high-quality real-time broadcasting (conventional broadcasting using video, audio, data, or combination of these elements) and storage-based broadcasting which can store various types of contents in the receiver. Contents to be handled are video, audio, images, texts, data, or combination of these elements. However, broadcasters may be allowed to provide conventional video broadcasting services using only a part of these broadcasting types, depending on the classification of the broadcaster's license.

2.2 Broadcasting Types

2.2.1 Real-time broadcasting

Real-time broadcasting is a content transmission system which is primarily intended to enable all users to view broadcasted contents "concurrently" as with digital terrestrial television broadcasting.

Reference: Broadcasters with a license issued for terrestrial television broadcasting are allowed to provide real-time broadcasting only.

2.2.2 Storage-based broadcasting

Unlike digital terrestrial television broadcasting, storage-based broadcasting delivers contents to the receiver via broadcasting radio waves. Contents are stored in the receiver before viewed/utilized, hence they are sent and used at different times. This broadcasting system (multimedia broadcasting using the transmission system based on connected segments which is provided in a download format is called Storage-based Broadcasting in this technical report) enables users to view/utilize stored contents without caring about time or place of use. When stored contents are incomplete, transmission of contents can be complemented by compensating for missing data via communications (hereinafter referred to as Compensation of Stored Contents).

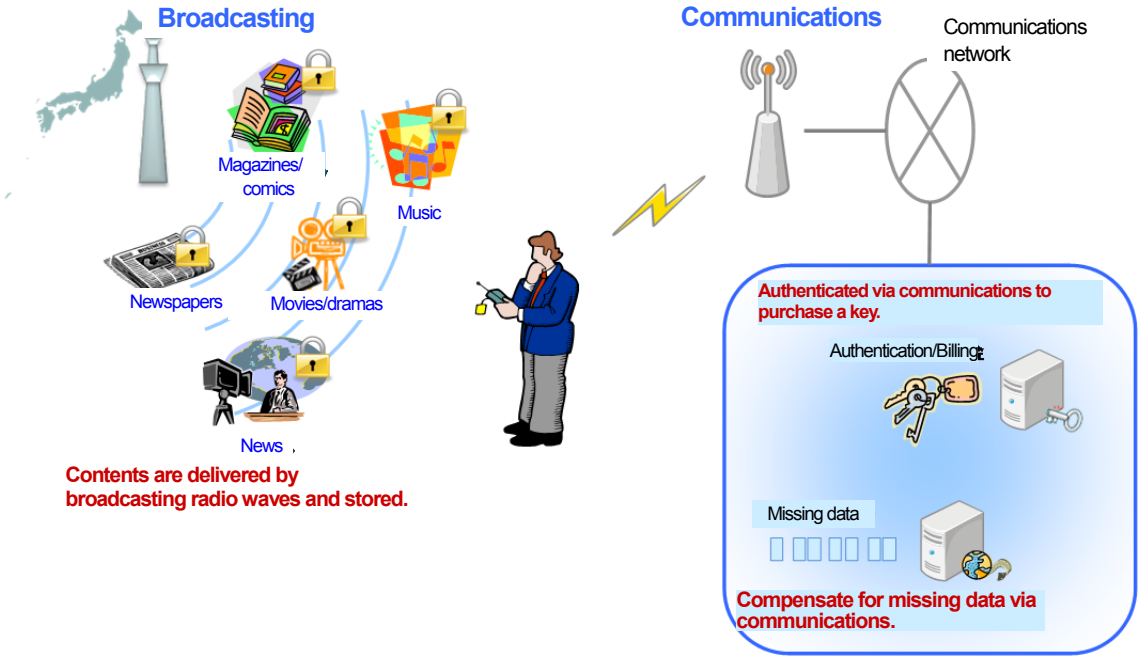


Fig. 2-1: Image of storage-based broadcasting linking broadcasting and communication

2.3 Available Bandwidth

The frequency band applied for multimedia broadcasting is the VHF band (207.5 MHz - 222 MHz). Specifically, there are 33 segments in total including two 13-segment bands and up to seven 1-segment bands.

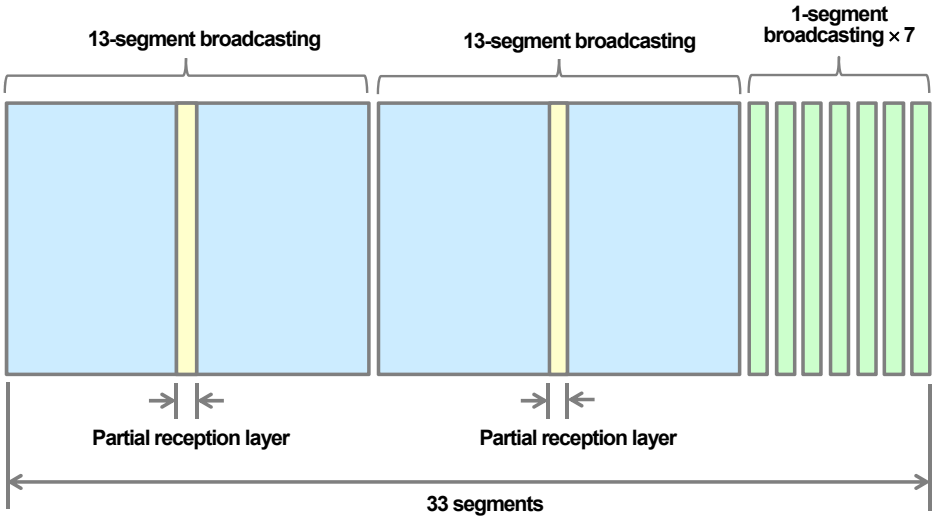


Fig. 2-2: Image of available bandwidth

2.4 Image of Frequency Band Utilization

To deliver more contents efficiently within the multimedia broadcasting band with a limited transmission rate, transmission capacities used for real-time broadcasting and storage-based broadcasting always need to be set at optimal rates. For this reason, fixed transmission rates are not set for each service, or specific segments are not allocated to individual services.

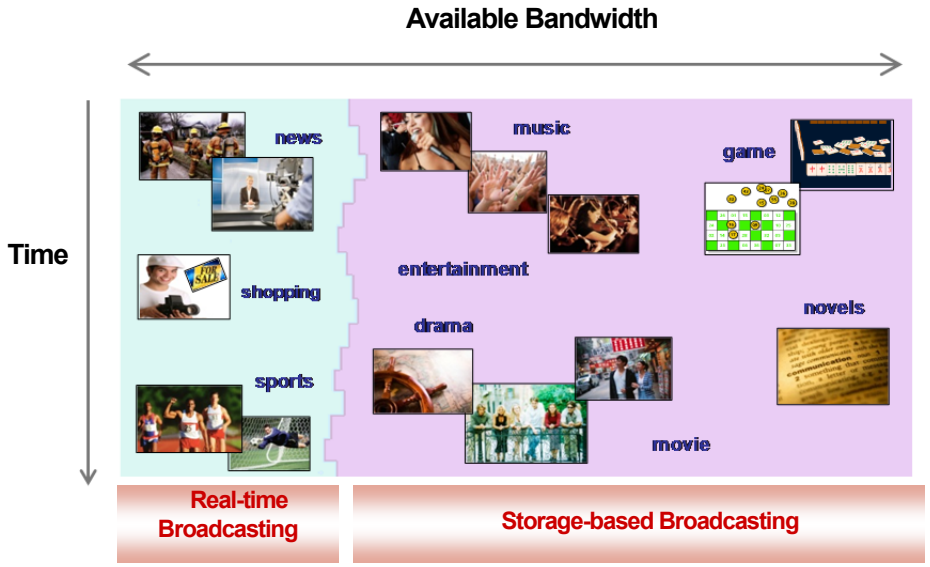


Fig. 2-3: Image of frequency band utilization (Utilization image in 13 segments)

2.5 System Image

A system image for multimedia broadcasting is shown below.

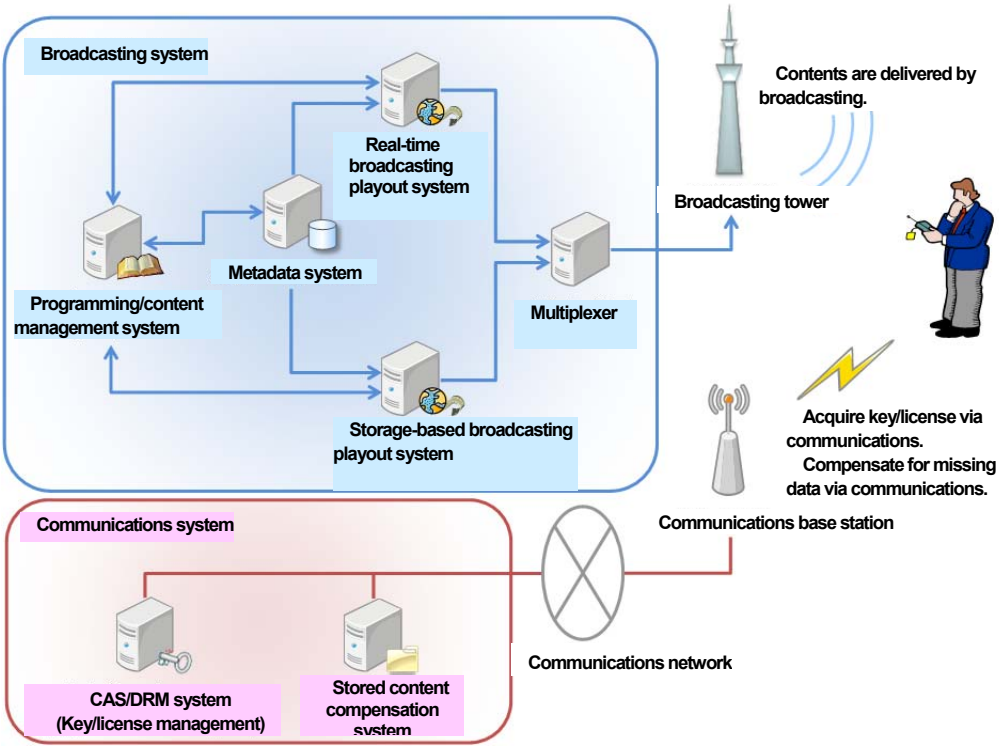


Fig. 2-4: System image for multimedia broadcasting

2.6 Content Types

2.6.1 Real-time broadcasting content types

Contents handled in real-time broadcasting are the same as those handled in digital terrestrial television broadcasting.

Table 2-1: Real-time broadcasting content types

Item No.	Type
1	Video (including audio and closed captions)
2	Audio
3	Data

2.6.2 Storage-based broadcasting content types

Storage-based broadcasting also handles still images and contents written in markup language in addition to video and audio. It should also be capable of handling complex contents in which one content includes several content elements. The manifest file defines scenario playback control for complex contents.

Table 2-2: Storage-based broadcasting content types

Item No.	Type
1	Video (including audio and closed captions)
2	Image
3	Markup languages
4	Other contents

<Blank Page>

Chapter 3 Technologies and Service Types

3.1 Technologies

Multimedia broadcasting adopts the ISDB-Tmm system, an extended standard of ISDB-T for mobiles employed in digital terrestrial television broadcasting.

<table>
<tr><td colspan="2">Real-time broadcasting contents</td><td>SI/PSI</td><td>Storage-based broadcasting contents</td><td>ECG/EPG metadata
Transmission control metadata</td></tr>
<tr><td rowspan="3">PES</td><td rowspan="3">Section</td><td colspan="3">FLUTE / AL-FEC</td></tr>
<tr><td colspan="3">UDP/IP / ROHC</td></tr>
<tr><td colspan="3">ULE</td></tr>
<tr><td colspan="5">MPEG-2 TS</td></tr>
<tr><td colspan="5">Physical layer (broadcasting)</td></tr>
</table>

Fig. 3-1: Protocol stack for multimedia broadcasting

3.1.1 Real-time broadcasting

Real-time broadcasting offers high-quality broadcasting at 720 × 480 pixel / 30 fps (frames per second). The video coding system is H.264 and the audio coding system is AAC. The assumed maximum number of audio channels is 5.1.

3.1.2 Storage-based broadcasting

Storage-based broadcasting uses the FLUTE protocol (RFC 3926) to multiplex IP packets using ULE (RFC 4326). After compiling the contents or metadata into a file, the system divides the file, generates FEC packets, adds the FLUTE header, and converts them into UDP/IP packets. Then, the UDP/IP packet header is compressed using the ROHC, and then encapsulated using the ULE in order to be transmitted in the MPEG-2 TS, and transmitted through the transmission channel for multimedia broadcasting. See Fig. 3-1 for the protocol stack for storage-based broadcasting in broadcast transmission channels.

3.2 Service Types

3.2.1 Real-time broadcasting

The following table shows service types in real-time broadcasting. See 14.3 Parameter used for the operation of hierarchical transmission for coding.

Table 3-1: Service types in real-time broadcasting

Item No.	Service type	Definition
1	Video real-time broadcasting service	Real-time broadcasting service mainly aimed for viewing video streams, which contains at least one video stream of stream_type = "0x1B."
2	Audio real-time broadcasting service	Real-time broadcasting service mainly aimed for viewing audio streams, which contains at least one audio stream of stream_type = "0x0F."
3	Independent data broadcasting service	Data broadcasting service mainly aimed for viewing data contents in real time, which contains at least one data carousel of stream_type = "0x0D." * When the partial reception layer contains a simplified moving image stream of stream_type = "0x1B," data carousel may not be contained.
4	Engineering service	Service for fixing receiver software, which performs bug fix; correction of failures caused by difference in interpretations on operations between the transmitter and the receiver; and improvement of display, response speed, and operability. It also updates program genre code lists, program characteristic code lists, and reserved words common to all receivers. See Volume 1 for more details.

3.2.2 Storage-based broadcasting

The following table shows service types in storage-based broadcasting.

Table 3-2: Service types in storage-based broadcasting

Item No.	Service type	Definition
1	Storage-based broadcasting service	Storage-based broadcasting service which contains at least one storage-based broadcasting data of stream_type = "0x91." Unlike general broadcasting services, this service delivers contents to the receivers via broadcasting radio waves, which are viewed/utilized after stored. The contents are sent and utilized at different times.
2	EPG/ECG metadata service	Service to broadcast EPG/ECG metadata which contains stream_type = "0x91."

Chapter 4　Access Control System

4.1 Basic Functions of Access Control System in Multimedia Broadcasting

The access control system in real-time broadcasting enables access control of contents which are broadcasted in a scrambled state in accordance with the access right information of each content based on the contract information of each subscriber stored in the receiver.

The access control system in storage-based broadcasting enables access control of contents when they are viewed/utilized by storing the contents which keep the encryption made at the time of broadcasting.

This function is achieved when the IC card or CAS Client, which is provided as a security module in the receiver, extracts the key and decrypts the encrypted contents with it according to the conditions issued by the license server.

<Blank Page>

Chapter 5 Utilization of Networks

5.1 Basic Rules for Network Utilization in Multimedia Broadcasting Receivers

In multimedia broadcasting, contents presented and utilized are mainly acquired by broadcasting, however, various types of other services utilizing characteristics of multimedia broadcasting receivers equipped with communications functions can also be provided by acquiring various resources (e.g. access right information of each content based on various keys and contract information, compensated data of storage-based broadcasting) via communications to compensate for the broadcasting.

5.1.1 Issuing licenses and keys

Contents broadcasted in real-time broadcasting and storage-based broadcasting are basically transmitted after encrypted. Responding to the request of the receiver, the license server which exists in the communications network issues licenses and keys based on the service conditions of contents. The receiver decrypts the contents using them.

5.1.2 Compensating for missing data in storage-based broadcasting

In storage-based broadcasting, contents are decrypted after stored to enable viewing/utilization. However, occurrence of missing reception data is also estimated depending on broadcasting radio wave reception environment. For this reason, missing data is compensated for using the communications function to store contents securely in order to complement transmission by broadcasting.

<Blank Page>

Chapter 6 Multimedia Broadcasting Receivers

6.1 Multimedia Broadcasting Receivers

Multimedia broadcasting receivers would include handy phone receivers mounted in cellular phones, mobile receivers mounted in car navigation systems, and reception terminals exclusive to multimedia broadcasting. However, other reception terminals including fixed TV sets are not excluded. These receivers must be capable of receiving 13-segment broadcasting (including reception at the partial reception layer) or 1-segment broadcasting, or both of them. To enjoy multimedia broadcasting services, having communications functions are recommended. Restrictions are imposed on the receivers without communications functions when they receive benefit from multimedia broadcasting services. However, the receivers with these restrictions imposed are not excluded.

<Blank Page>

Chapter 7 EPG and ECG in Multimedia Broadcasting

7.1 Concept of EPG and ECG

In multimedia broadcasting receivers, broadcasting services can be selected or program viewing/recoding can be scheduled using the EPG as with conventional digital broadcasting receivers. The ECG also enables scheduled storage and selection of contents available in storage-based broadcasting.

The EPG is a receiver application, which displays relevant information including broadcast schedule of contents broadcasted in real-time broadcasting.

The ECG is also a receiver application, which displays contents to be broadcasted in storage-based broadcasting and stored contents together with their relevant information.

Both the EPG and the ECG can also display a list of all contents or part of the contents using search, filtering, or sort function or according to user attributes etc. The EPG and the ECG can also be displayed comprehensively on the same screen.

Service conditions of each content are displayed according to the license reference information and the acquired license.

7.2 Role Sharing between SI and Metadata

In multimedia broadcasting, display of the EPG and the ECG, scheduled viewing/recording and channel selection of real-time broadcasting, and scheduled storage/playback and storage of storage-based broadcasting are performed in accordance with metadata.

On the other hand, to respond to potential changes in programming just before real-time broadcasting is delivered, program information for a few of the forthcoming programs is transmitted through SI.

<Blank Page>

Chapter 8 Content Protection

8.1 Basic Concept of Content Protection in Multimedia Broadcasting

To enable content protection, storage-based broadcasting contents are basically stored in the receiver in an encrypted state. Receivers to which a license (encryption key and service conditions – RMPI) was issued can decrypt the content before it is used in accordance with the RMPI and enable to play back (use) the content (conditional playback). Receivers which do not have playback conditions complying with the RMPI or a proper license cannot play back (use) the content, whereby a protection platform for multimedia broadcasting contents is realized.

8.2 Output Control Using RMPI

Protection in accordance with the output control conditions using the RMPI is also provided to external output using an interface which connects the receiver with a peripheral device, removable media, or a network to prevent unauthorized use such as duplication.

<Blank Page>

Chapter 9 Recommend

9.1 Recommend Function

The recommend function in multimedia broadcasting is enabled by communication between the receiver and the recommend server.

The recommend function would include automatic storage by the storage instruction from the recommend server and preferred presentation in the ECG or the EPG by the display instruction from the recommend server. Recommend would be classified into two methods: a method to recommend contents utilizing user attribute information, contract, access history, etc.; a method to recommend contents in accordance with the preset rules.

<Blank Page>

Chapter 10 Ads

10.1 Ads in Contents

Multimedia broadcasting also enables ads in contents as with conventional digital broadcasting.

10.2 Banner Ads

Banner ads (independent) displayed in the receiver user interface in multimedia broadcasting can be transmitted by storage-based broadcasting. Targeting ads are also available by linking banner ads (independent) with the recommend function.

<Blank Page>

Chapter 11 Operator and System Models

11.1 Operator Model

The operator model in multimedia broadcasting is shown below.

The following figure shows the relationship among platform providers who provide metadata aggregation functions and CAS/DRM key management functions for entire multimedia broadcasting; paid broadcasting administrators, main body of CAS/DRM operation, who issue and manage licenses; and program providers. It does not indicate that the operator shown with each frame provides the functions of the operators included in the frame; it represents that one platform provider corresponds to four paid broadcasting administrators, each of which corresponds to several program providers.

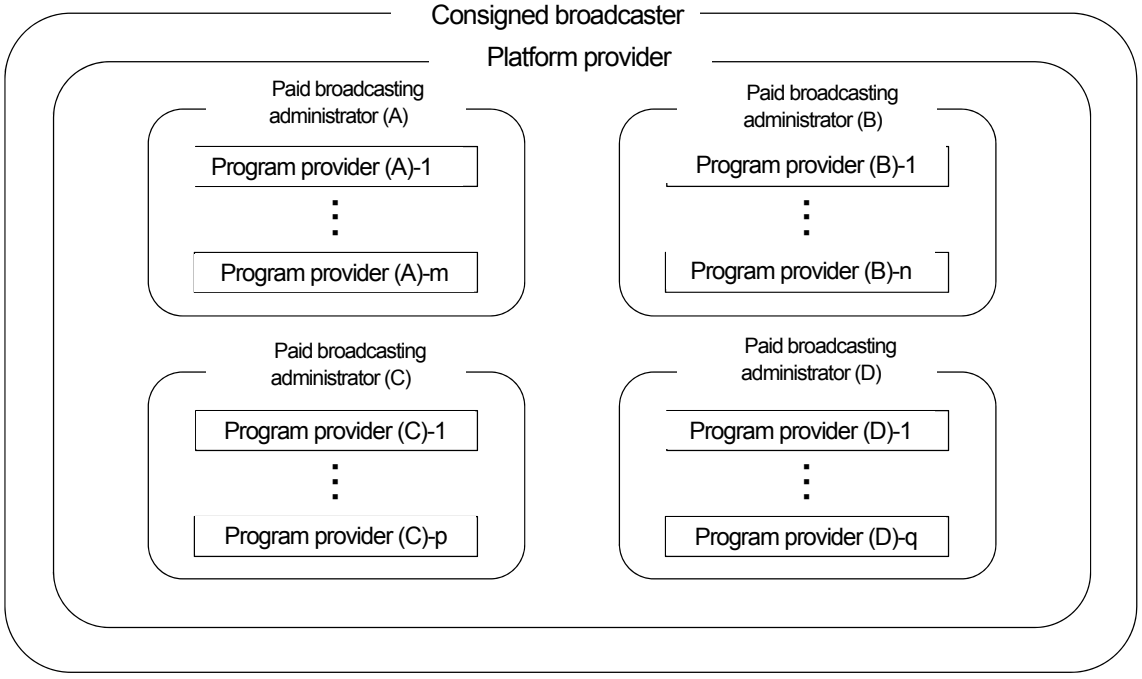


Fig. 11-1: Operator model in multimedia broadcasting

Platform provider: Unlike a conventional platform provider, it is supposed to supervise paid broadcasting administrators and program providers and provide metadata aggregation and CAS/DRM key management functions.

Paid broadcasting administrator: Main body to operate CAS/DRM, which is supposed to issue and manage licenses.

11.2 System Model

The system model in multimedia broadcasting is shown below.

The following figure shows the linkage between the receiver and the system which would be operated by a platform provider, the system which would be operated by a paid broadcasting administrator, and the system which would be operated by a program provider in multimedia broadcasting.

Arrows without any description show exchanges of information via communications.

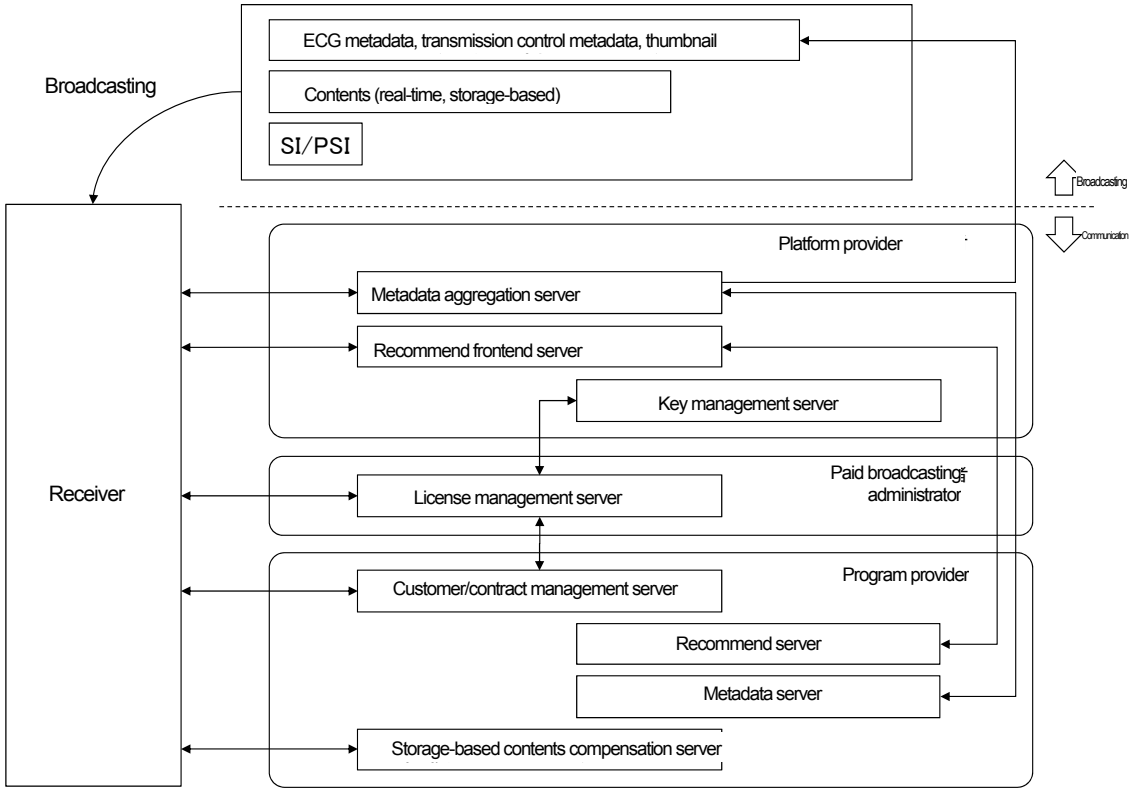


Fig. 11-2: System model in multimedia broadcasting

Metadata aggregation server: Aggregates and transmits ECG metadata, transmission control metadata, etc. from each program provider.

Recommend frontend server: Frontend server when the receiver receives recommend information.

Chapter 12 Receiver Processing Flow

12.1 Overview

A typical pattern to enjoy multimedia broadcasting services is shown below. A basic flow would involve the following four steps: (1) metadata acquisition; (2) storage scheduling; (3) receipt/storage of contents, and (4) viewing/utilization of contents. Contract to view/utilize contents (purchase of viewing/utilization right), acquisition of license, or compensation process when storage by broadcasting is incomplete would be required during these four steps depending on the content provision or reception conditions.

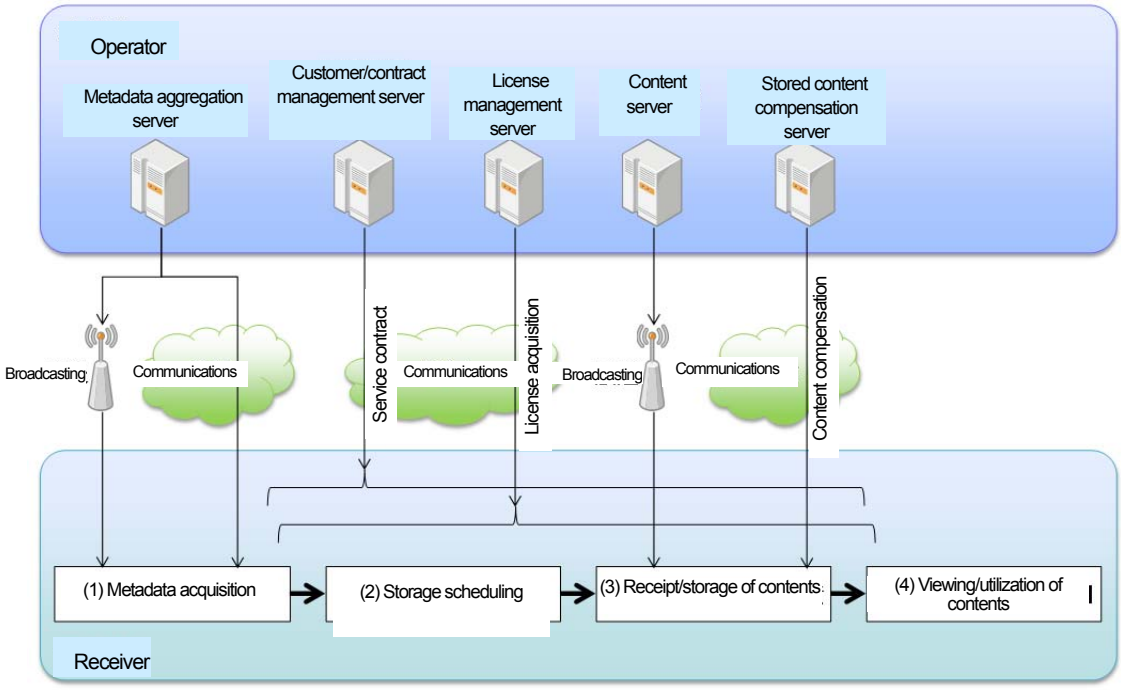


Fig. 12-1: Typical service flow

The receiver processing flow in multimedia broadcasting is described from ***12.2 Initial settings***. Multimedia broadcasting can be received by covering each process flow. When the processing flow is partly not executed, restrictions are imposed on the benefit received from the services.

12.2 Initial Settings

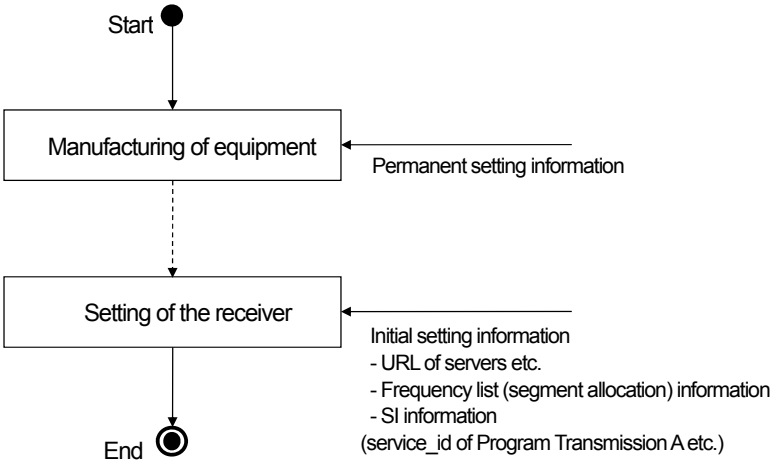


Fig. 12-2: Initial settings

12.3 Receiver Contract (Reception Setting)

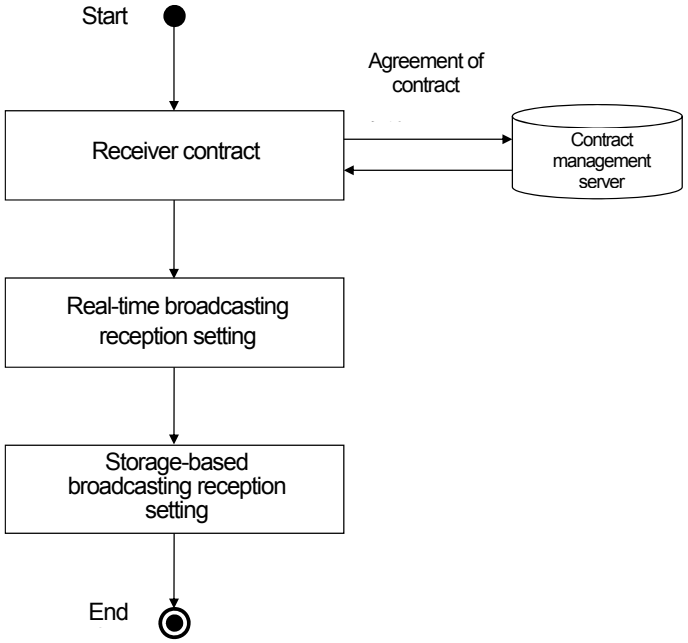


Fig. 12-3: Receiver contract (reception setting)

12.3.1 Real-time broadcasting reception setting

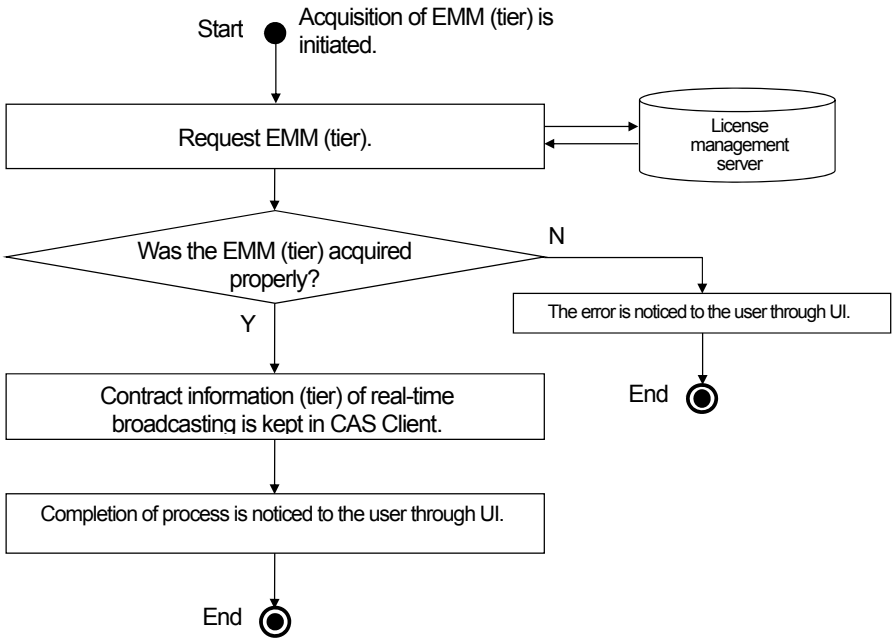


Fig. 12-4: Real-time broadcasting reception setting (acquisition of license information)

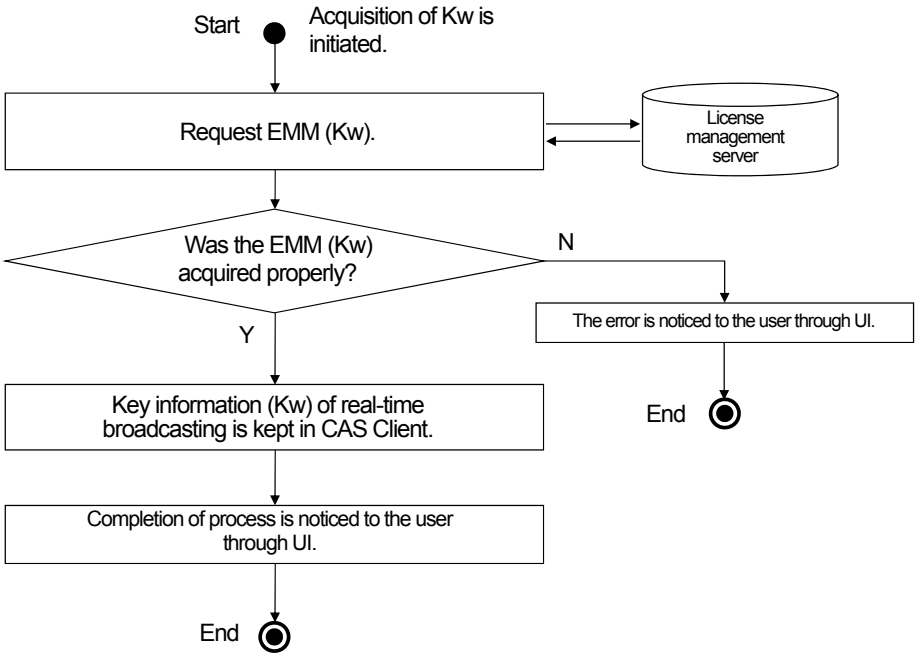


Fig. 12-5: Real-time broadcasting reception setting (acquisition of key information)

12.3.2 Storage-based broadcasting reception setting (CAS Client activation)

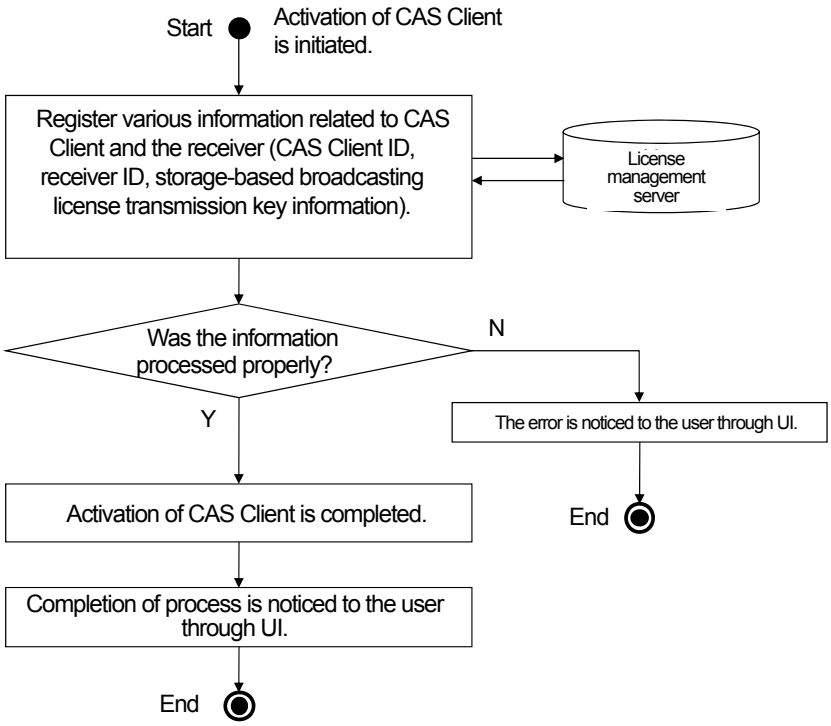


Fig. 12-6: Storage-based broadcasting reception setting (CAS Client activation)

12.4 Acquiring Frequency List

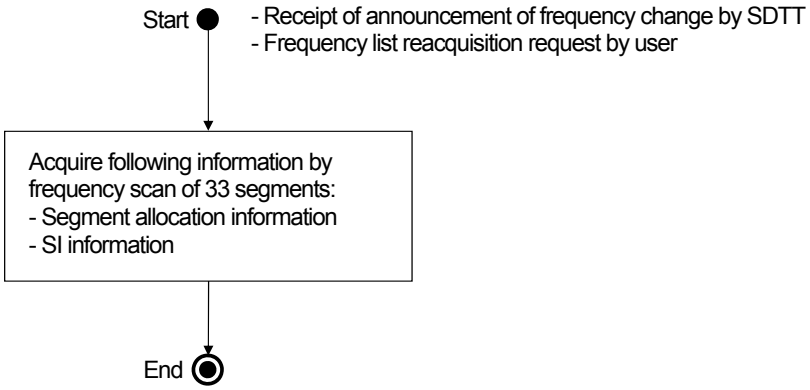


Fig. 12-7: Acquiring frequency list

12.5 Acquiring CS (Classification Scheme) Dictionary

CS dictionary file should be basically embedded when the receiver is manufactured as described in the technical report Volume 10, Annex 1 10 Guidelines for acquiring/managing

classification scheme. However, for receivers equipped with communications functions, the file may be embedded when the receiver is manufactured, or it may be acquirable by the following method: access the directory which manages the CS dictionary file in the metadata aggregation server using the server information that can be acquired at the initial settings, and acquire a set of CS dictionary in the directory using the HTTP GET method.

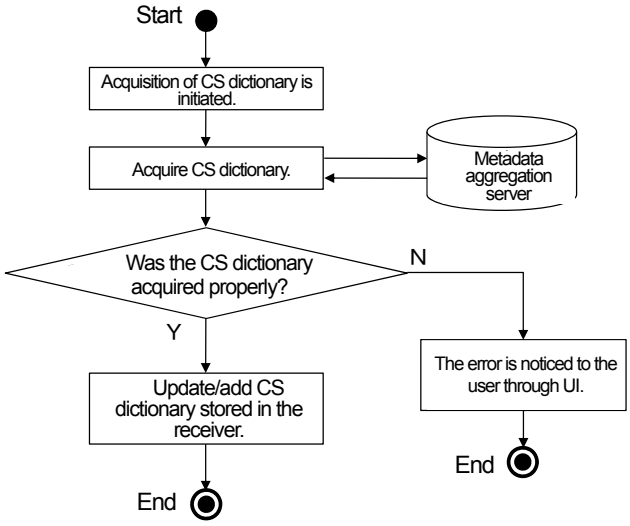


Fig. 12-8: Acquiring CS (Classification Scheme) dictionary

12.6 Receiving ECG Metadata (Program Transmission A)

Metadata (ECG, EPG, transmission control) is transmitted by "Program Transmission A" which uses Layer A and "Program Transmission B" which uses Layer B. Here, "Program Transmission A" transmits priority metadata (for example, metadata for two days from the present time).

At the frequency scan after the initial settings and receipt of real-time broadcasting during normal use, the metadata transmitted by "Program Transmission A" is acquired based on service_id included in "SI information" which was acquired at the aforementioned initial settings. For more details of "Program Transmission A" and its acquisition process, see the technical report Volume 11, 2.2.2 Program Transmission A.

Reception of "Program Transmission A" is basically performed once a day. When broadcasting radio waves cannot be received or no real-time broadcasting is received, necessary metadata should also be acquirable from the metadata aggregation server using communications. For more details of metadata acquisition process using communications, see the technical report Volume 10, Chapter 6 Operation of metadata delivery system via communications.

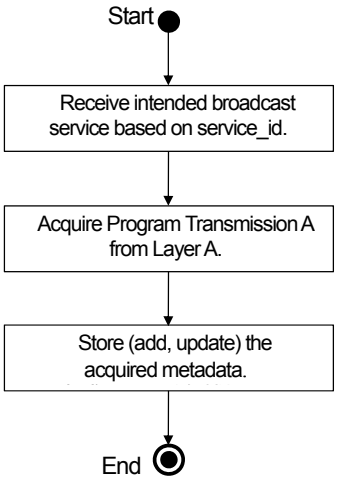


Fig. 12-9: Receiving ECG metadata (Program Transmission A)

12.7 Acquiring ECG Metadata (Program Transmission B)

12.10 Scheduled storage of storage-based broadcasting and 12.11 Receiving storage-based broadcasting/storing contents shall be followed.

12.8 Basic Contract

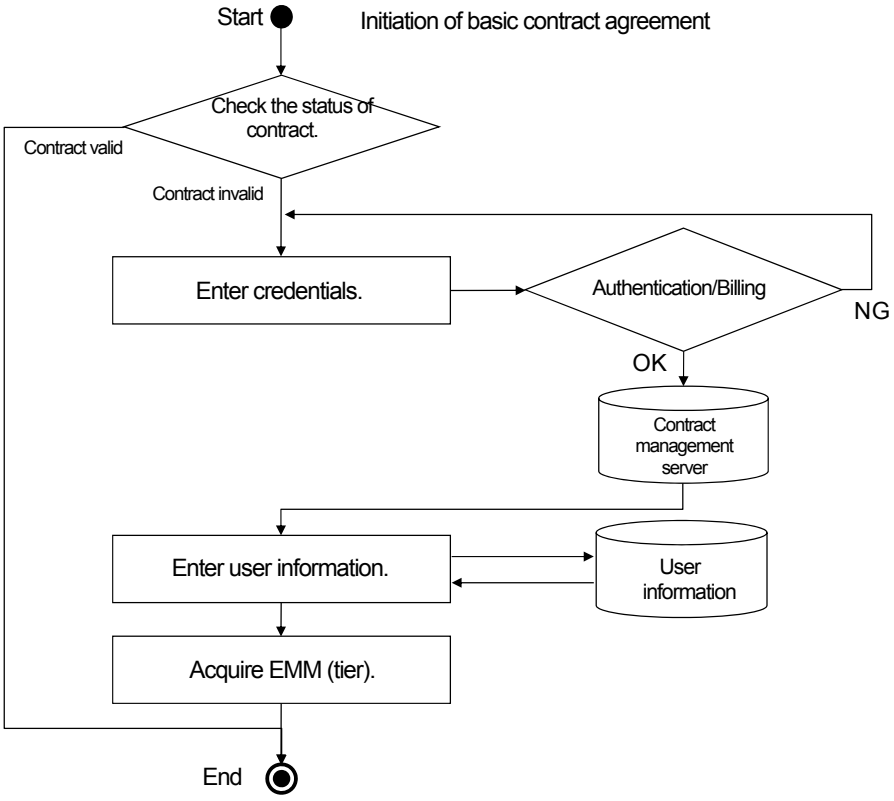


Fig. 12-10: Basic contract

12.9 Premium Contract

12.9.1 Real-time broadcasting

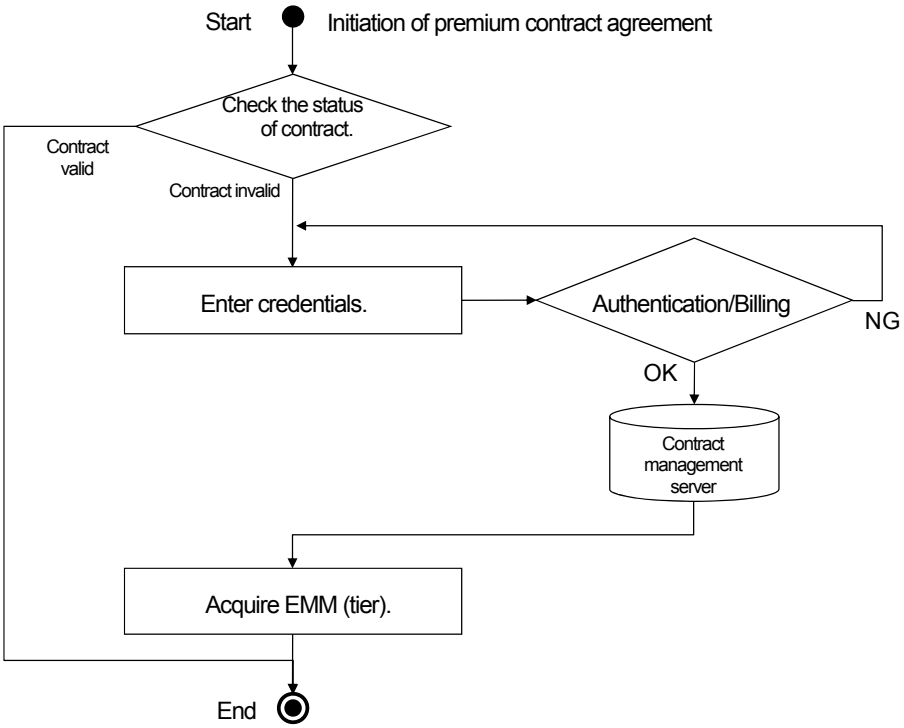


Fig. 12-10: Premium contract (real-time broadcasting)

12.9.2 Storage-based broadcasting

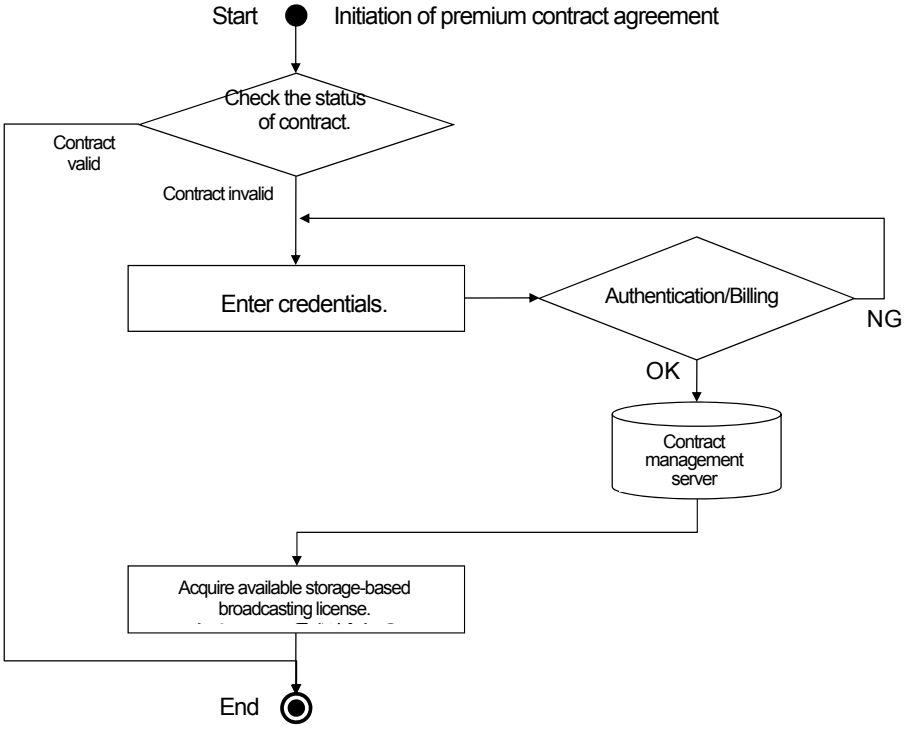


Fig. 12-11: Premium contract (storage-based broadcasting)

12.10 Storage Scheduling of Storage-based Broadcasting

This flow shall be followed also for Program Transmission B as for storage-based broadcasting content.

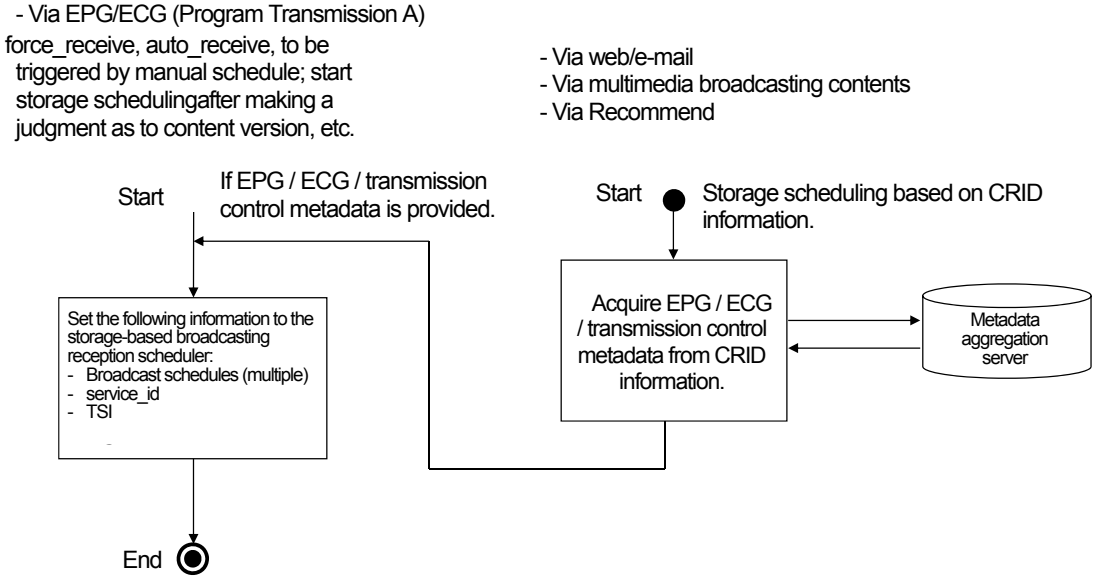


Fig. 12-12: Storage scheduling of storage-based broadcasting (service ID base)

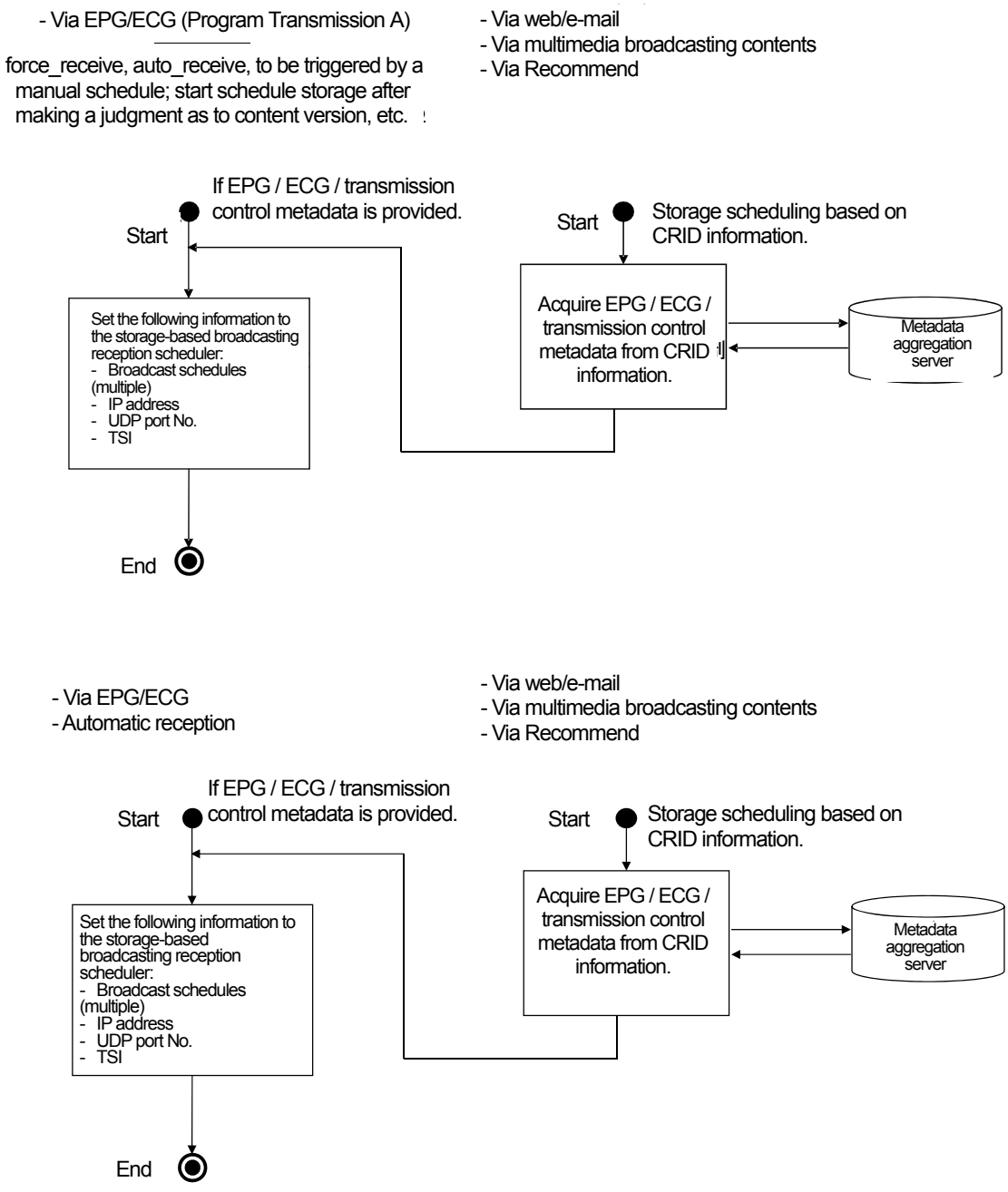


Fig. 12-13: Storage scheduling of storage-based broadcasting (IP base)

12.11 Receiving Storage-based Broadcasting/Storing Contents

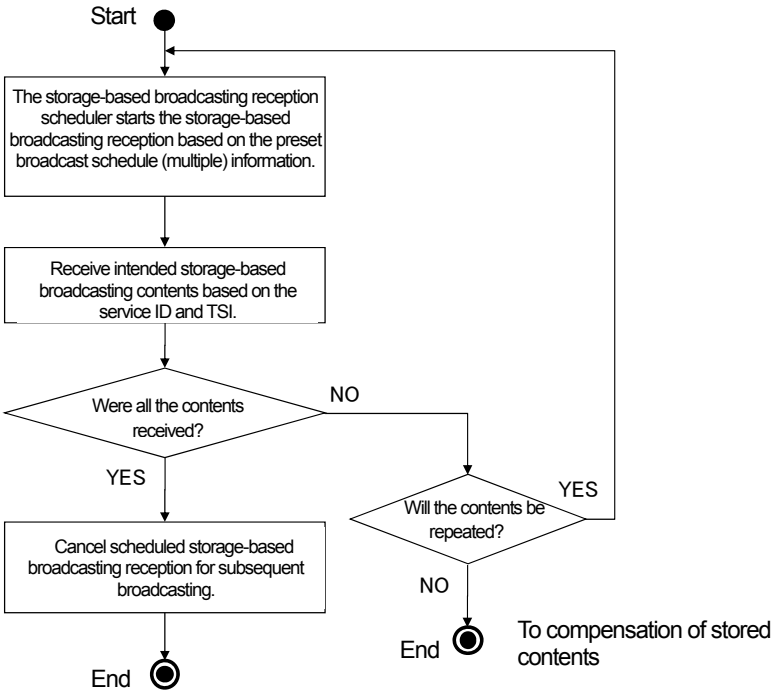


Fig. 12-14: Receiving storage-based broadcasting / Storing contents (service ID base)

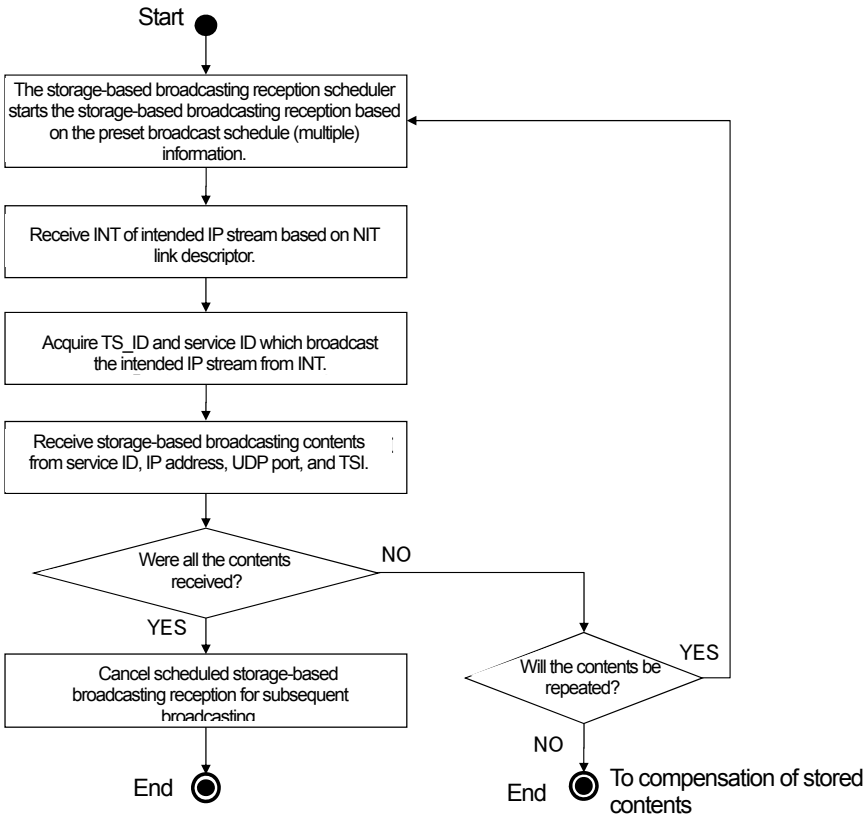


Fig. 12-15: Receiving storage-based broadcasting / Storing contents (IP base)

12.12 Compensating for Stored Contents

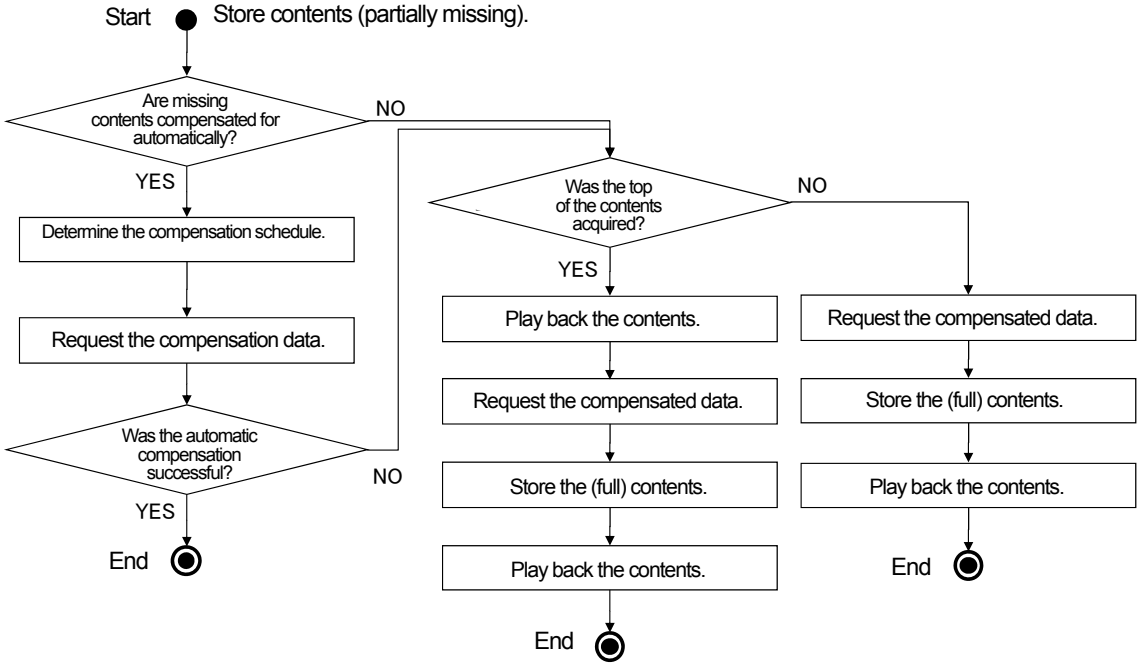


Fig. 12-16: Compensating for stored contents

12.13 Acquiring Storage-based Broadcasting License

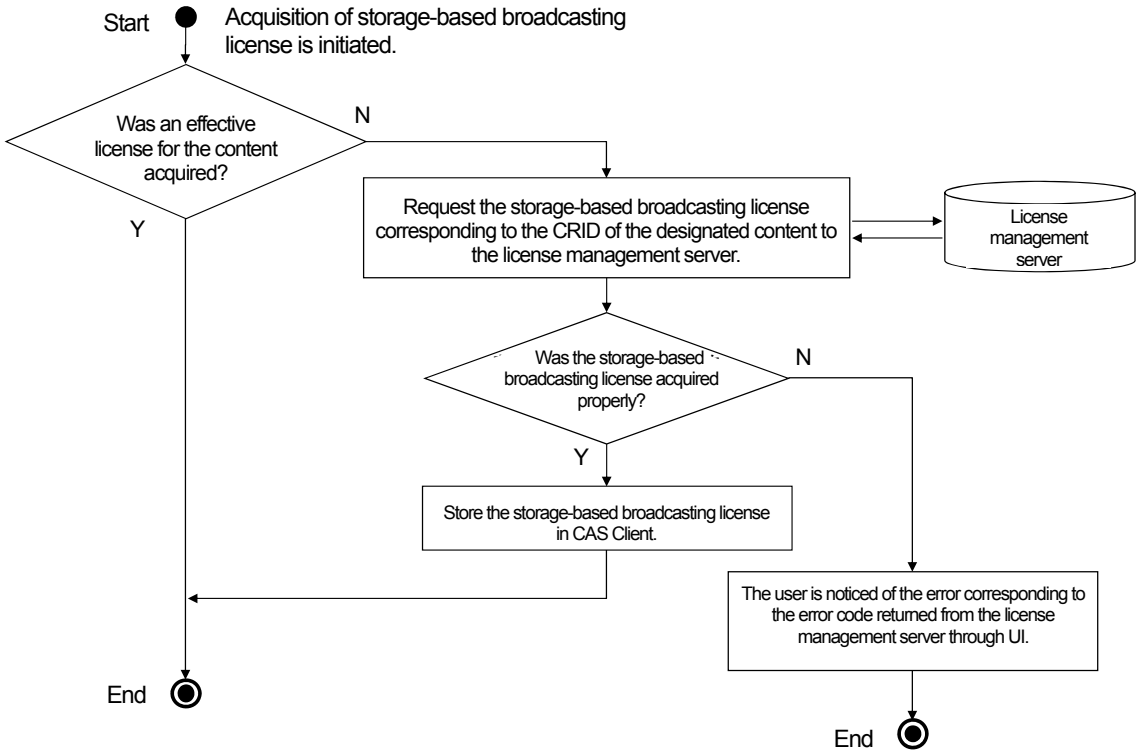
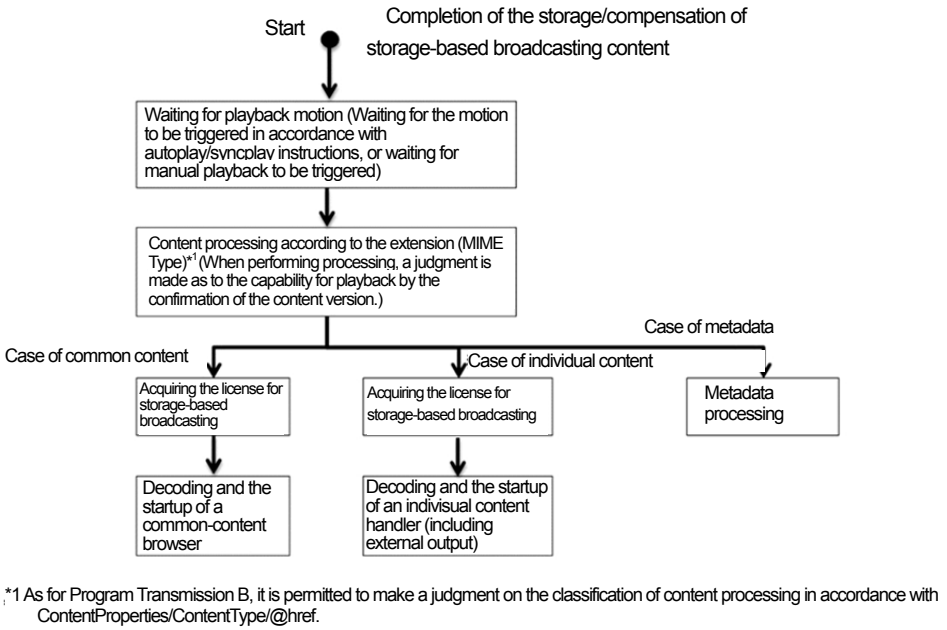


Fig. 12-17: Acquiring storage-based broadcasting license

12.14 Viewing/utilizing Storage-based Broadcasting Contents

This flow shall be followed also for Program Transmission B as storage-based broadcasting content.



*1 As for Program Transmission B, it is permitted to make a judgment on the classification of content processing in accordance with ContentProperties/ContentType/@href.

Fig. 12-19: Viewing/utilizing storage-based broadcasting contents

12.15 Scheduled Viewing/recording of Real-time Broadcasting

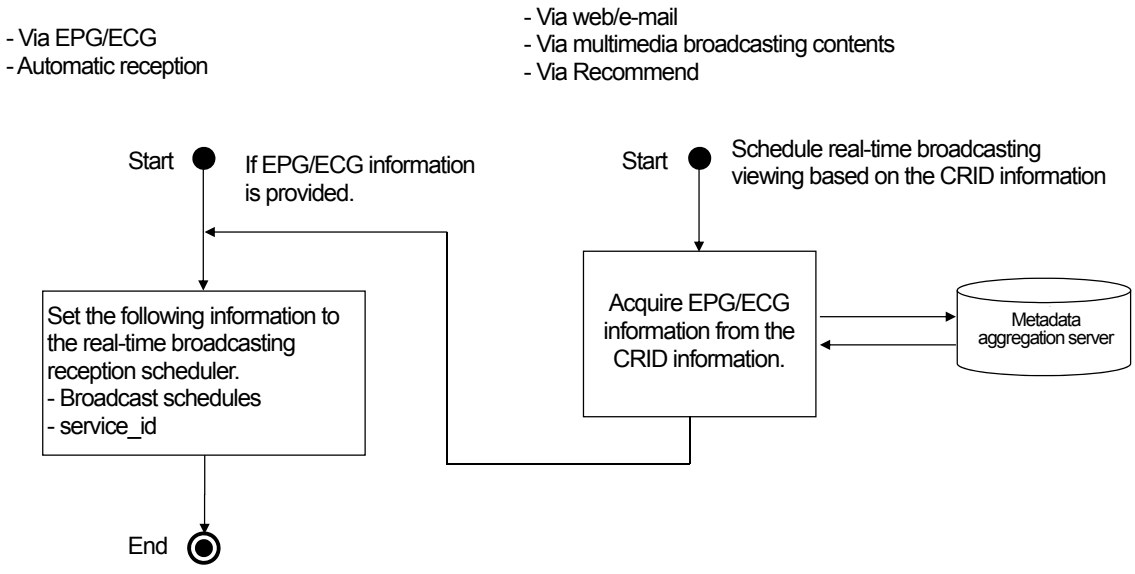


Fig. 12-20: Scheduled viewing/recording of real-time broadcasting

12.16 Receiving (viewing) Real-time Broadcasting

12.16.1 Viewing with EPG

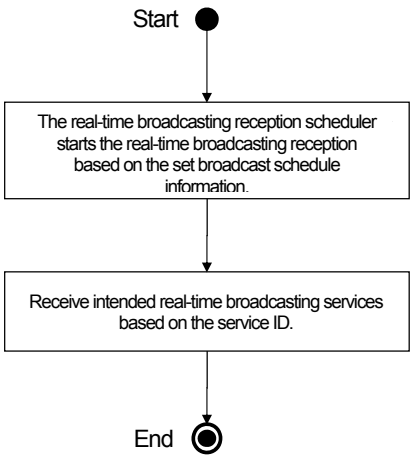


Fig. 12-21: Viewing with EPG

12.16.2 Zapping (viewing by channel selection)

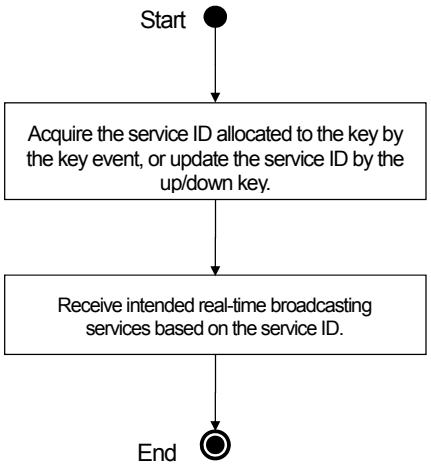


Fig. 12-22: Zapping (viewing by channel selection)

12.17 Updating Real-time Broadcasting Key/license Information

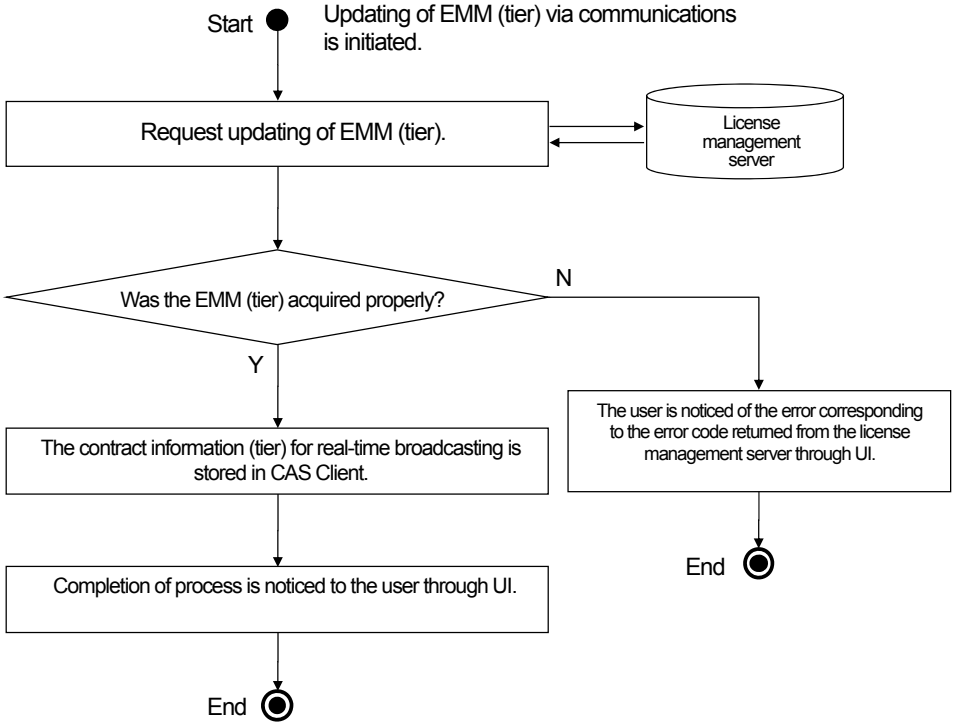


Fig. 12-23: Updating real-time broadcasting license information (communications)

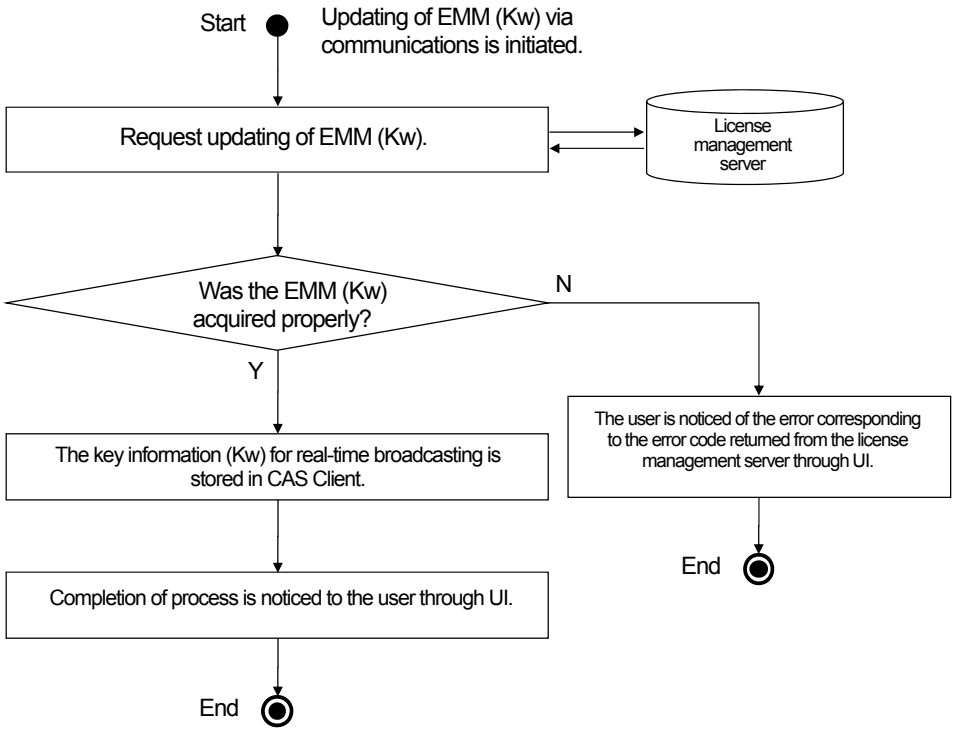


Fig. 12-24: Updating real-time broadcasting key information (communications)

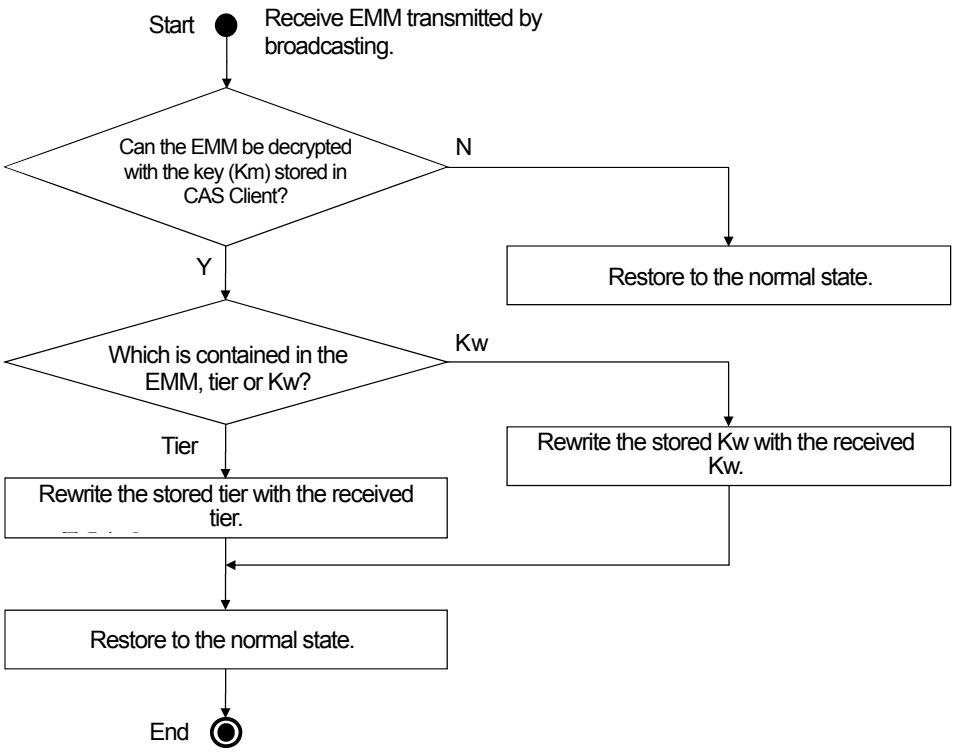


Fig. 12-25: Updating real-time broadcasting key/license information (broadcasting)

12.18 Reception Triggered by Programming Changes

12.18.1 Real-time broadcasting

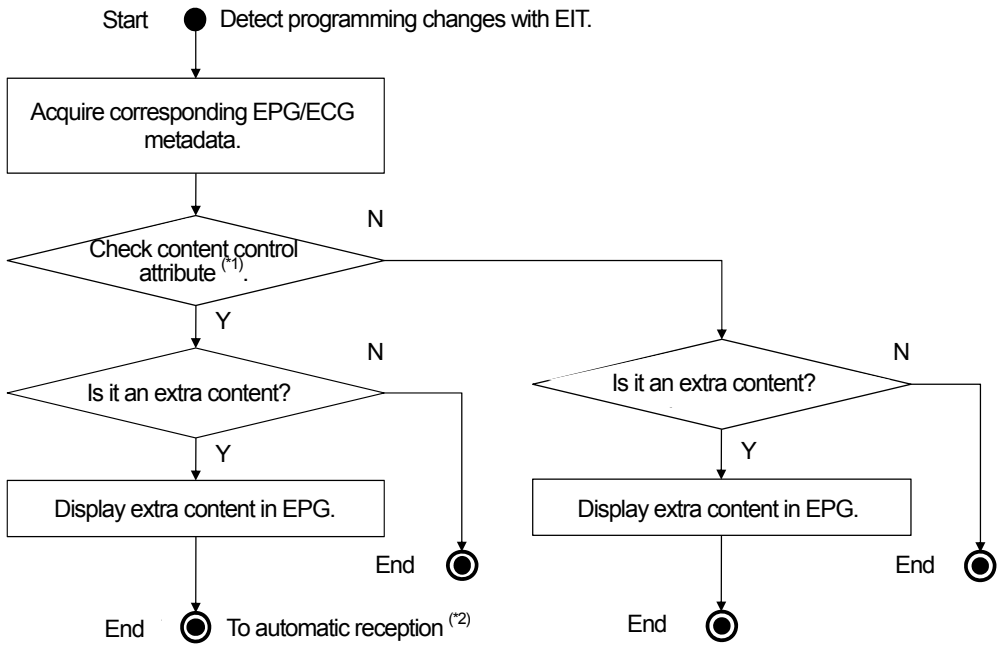


Fig. 12-26: Reception triggered by programming changes (real-time broadcasting)

*1 Check whether auto_receive:ON or force_receive:ON is set to the contents.

*2 See 12.15 Scheduled Viewing/recording of Real-time Broadcasting

12.18.2 Storage-based broadcasting

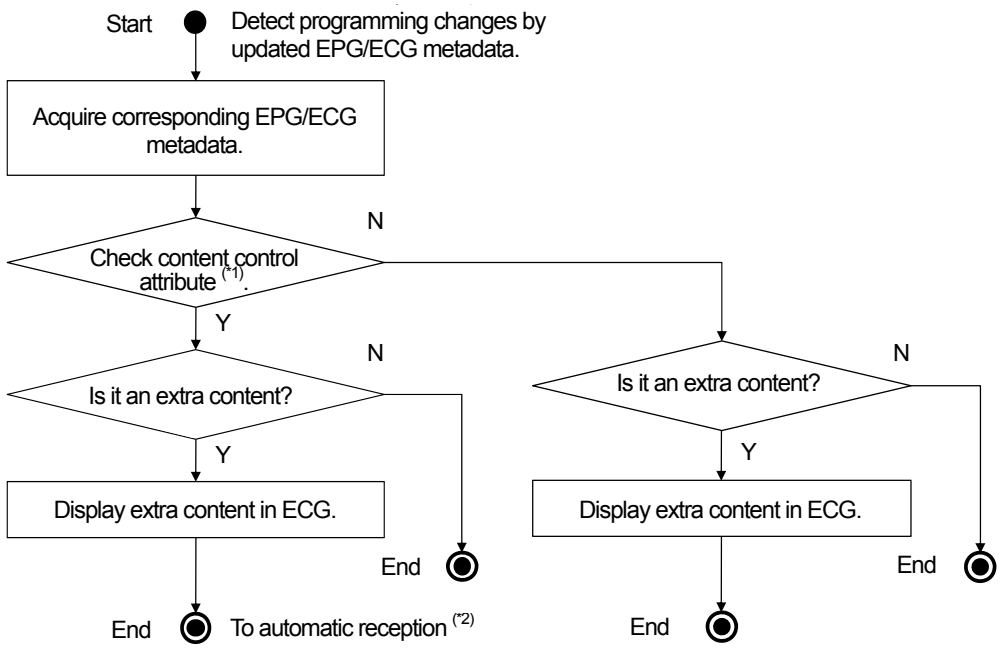


Fig. 12-27: Reception triggered by programming changes (storage-based broadcasting)

*1 Check whether auto_receive:ON or force_receive:ON is set to the contents.

*2 See 12.10 Storage Scheduling of Storage-based Broadcasting.

12.19 Cancellation

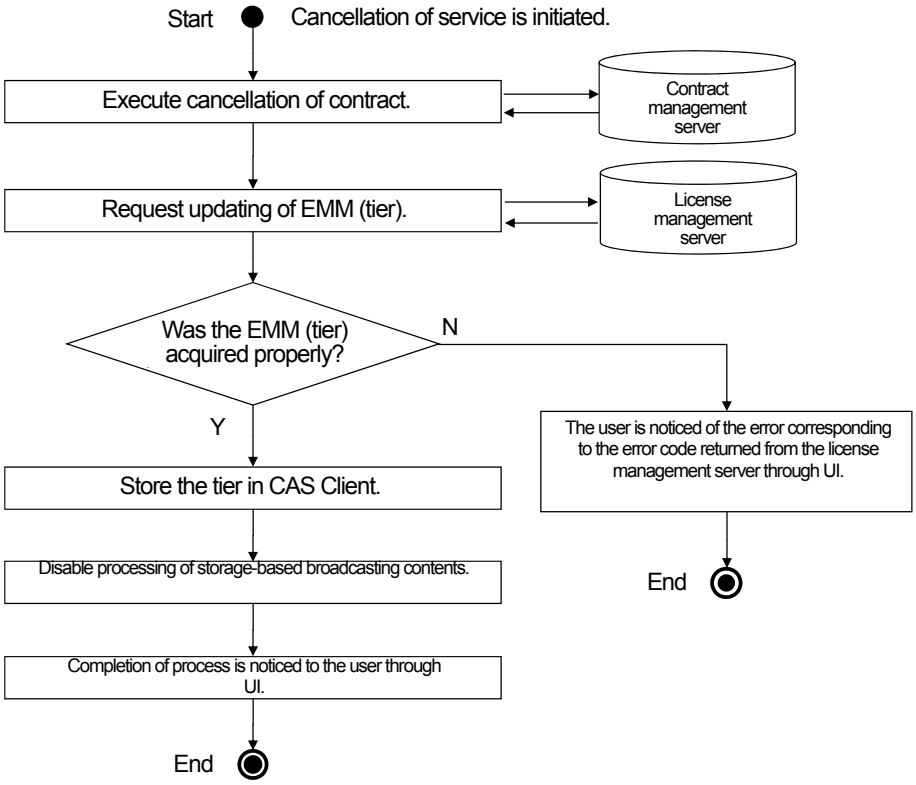


Fig. 12-18: Cancellation (communications)

12.20 Engineering Service

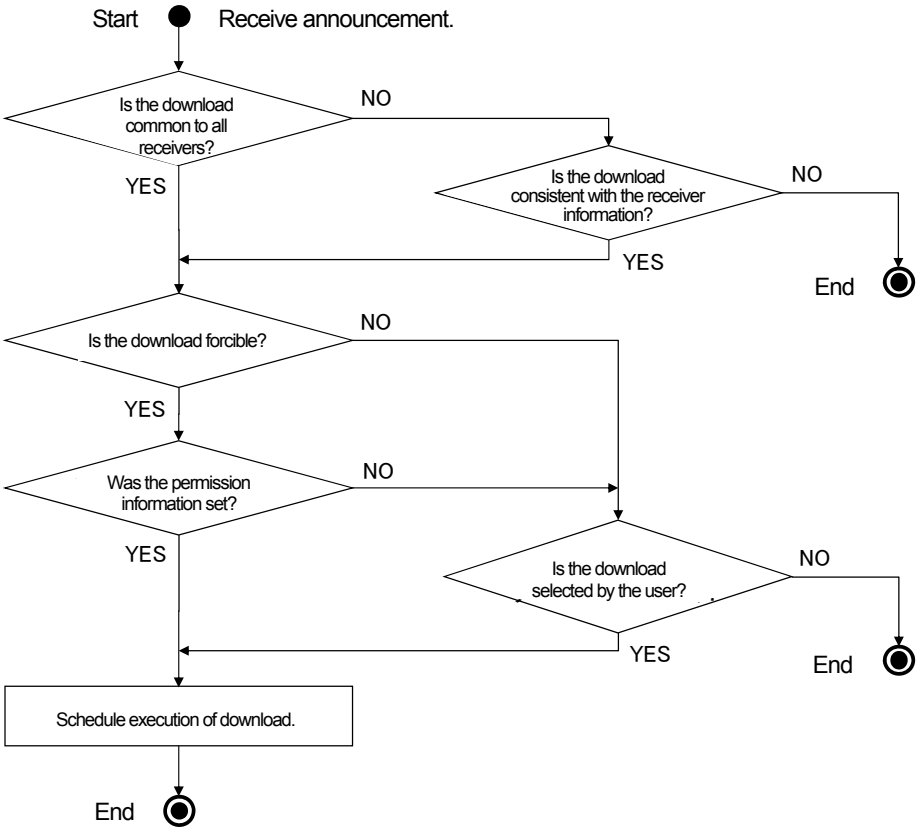


Fig. 12-29: Engineering service

Chapter 13 Description of the Technical Report

13.1 Organization of Technical Report

The technical report is composed of the following volume (Volume 0 to 12). Outline of each volume is described below.

(1) Volume 0: Basic concept and common elements of multimedia broadcasting

Prior to each subsequent volume below, this volume clarifies the basic concept and common elements of multimedia broadcasting.

(2) Volume 1: Operational guidelines for downloading

Updating services of information in the receiver are provided in accordance with ARIB STD-B21, ARIB STD-B24, and ARIB STD-B10. This volume sets forth operations for reception design in detail.

(3) Volume 2: Functional specifications for receivers

This volume summarizes receiver functions, and describes how to handle multimedia broadcasting contents received by broadcasting in the receiver and which user interface is required.

(4) Volume 3: Guidelines for multimedia coding

This volume mainly describes media coding used in multimedia broadcasting and sets forth the guidelines for signal transmission and receiver specifications for secure operations of multimedia broadcasting.

(5) Volume 4: Operational guidelines for PSI/SI

Service information used in multimedia broadcasting is operated in accordance with ARIB STD-B10. To ensure secure operations on the sending and receiving sides, this volume sets forth configurations of PSI and SI, signal types, basic data structure, operation and transmissions of identifiers.

(6) Volume 5: Operational guidelines for conditional access system (CAS) and receiver specifications

This volume describes technical requirements necessary for content protection and access control systems in multimedia broadcasting. All the requirements set forth in this volume should be fulfilled when multimedia broadcasting receivers are implemented. This volume, however, does not refer to specific implementation of receivers.

(7) Volume 6: Operational guidelines for bidirectional communication

(Not applicable in this report.)

(8) Volume 7: Operational guidelines for transmission

This volume mainly sets forth source coding, multiplex, transmission channel coding systems for operations and transmissions at broadcast stations. Broadcasters should be engaged in broadcasting in accordance with this guideline. Receivers need to be capable of using signals transmitted in accordance with this guideline in the assumed operational manner.

(9) Volume 8: Guidelines for content protection

Multimedia broadcasting contents are recorded in the recorder directly after received. To view or use the contents, the receiver reads these contents and decrypts them inside it. This volume describes the concept of decrypted content protection.

(10) Volume 9: Operational guidelines for sending

(Not applicable in this report.)

(11) Volume 10: Operational guidelines for metadata

This volume specifies metadata types, models, and utilization to be operated actually out of the metadata set forth in ARIB STD-B38.

(12) Volume 11: Operational guidelines for storage-based broadcasting

This volume sets forth the operation of storage-based broadcasting transmission systems in multimedia broadcasting.

(13) Volume 12: Compensation of stored contents

This volume sets forth the operation of compensation of stored contents utilized in storage-based broadcasting out of communication-related operations to deliver multimedia broadcasting services.

Compensation of stored contents is a function used when a file is not received fully by broadcasting radio waves in storage-based broadcasting by compensating for missing data from the server via communications to complete the file. Compensation of stored contents includes automatic compensation method and manual compensation method by users.

13.2 Reference Documents

This technical report specifies requirements for broadcasting for mobile devices based on the broadcasts described in the following documents. Relevant standard specifications are listed below.

(1) “Service Information for Digital Broadcasting System” ARIB Standard, ARIB STD-B10

(2) "Receiver for Digital Broadcasting" ARIB Standard, ARIB STD-B21

(3) "Data Coding and Transmission Specification for Digital Broadcasting" ARIB Standard, ARIB STD-B24

(4) "Conditional Access System Specifications for Digital Broadcasting" ARIB Standard, ARIB STD-B25

(5) "Transmission System For Digital Terrestrial Sound Broadcasting" ARIB Standard, ARIB STD-B29

(6) "Transmission System for Digital Terrestrial Television Broadcasting" ARIB Standard, ARIB STD-B31

(7) "Video Coding, Audio Coding and Multiplexing Specifications for Digital Broadcasting" ARIB Standard, ARIB STD-B32

(8) "Structure and Operation of Closed Caption Data Conveyed by Ancillary Data Packets" ARIB Standard, ARIB STD-B37

(9) "Coding, Transmission and Storage Specification for Broadcasting System Based on Home Servers" ARIB Standard, ARIB STD-B38

(10) "Content Download System for Digital Broadcasting" ARIB Standard, ARIB STD-B45

(11) "Transmission System for Terrestrial Mobile Multimedia Broadcasting based on Connected Segment Transmission" ARIB Standard, ARIB STD-B46

(12) "Receiver for Terrestrial Mobile Multimedia Broadcasting Based on Connected Segment Transmission (Desirable Specifications)" ARIB Standard, ARIB STD-B53

(13) "Operational Guidelines for Digital Terrestrial Television Broadcasting" Technical Report, ARIB TR-B14

(14) "Operational Guidelines for Digital Satellite Broadcasting" Technical Report, ARIB TR-B15

(15) "Digital Broadcasting System based on Home Server" Technical Report, ARIB TR-B27

(16) Report of the Telecommunication Technology Council, Deliberation No. 17

(17) Report of the Telecommunication Technology Council, Deliberation No. 74

(18) Ministerial Ordinance No. 87 of the Ministry of Internal Affairs and Communications, 2011

(19) Notification No. 298 of the Ministry of Internal Affairs and Communications, 2011

(20) Notification No. 299 of the Ministry of Internal Affairs and Communications, 2011

(21) Notification No. 302 of the Ministry of Internal Affairs and Communications, 2011

(22) ISO/IEC 13818-1 Information technology -- Generic coding of moving pictures and associated audio information: Systems (MPEG-2 Systems standard)

(23) "Domain names - concepts and facilities" IETF RFC-1034

(24) “Domain names - implementation and specification” IETF RFC-1035

(25) “Uniform Resource Locators (URL)” IETF RFC-1738

(26) “Uniform Resource Identifiers (URI): Generic Syntax” IETF RFC-2396

(27) “Hypertext Transfer Protocol -- HTTP/1.1” IETF RFC-2616

(28) “HTTP Over TLS” IETF RFC-2818

(29) “HTML 4.01 Specification” W3C-HTML 4.01

(30) “Phase-1 - Metadata Schema” ETSI TS 102 822-3-1

(31) “Phase-2 - Extended Metadata Schema” ETSI TS 102 822-3-3

(32) “The Content-MD5 Header Field” IETF RFC 1864

(33) “Hypertext Transfer Protocol - HTTP/1.1” IETF RFC 2616

(34) “FLUTE - File Delivery over Unidirectional Transport” IETF RFC 3926

Chapter 14 Parameters to Operate

14.1 Layer Structure to Operate and Parameters to Use

As with digital terrestrial television broadcasting, multimedia broadcasting enables hierarchical transmission which transmits signals having several transmission parameters concurrently.

Each layer consists of one or multiple OFDM segments (up to 13) having a single set of transmission parameters, enabling hierarchical transmission up to three layers.

14.1.1 Transmission parameters which can be designated for each layer

The following table shows transmission parameters which can be designated for each layer and for 13 segments entirely.

Table 14-1: Specified range of transmission parameters

Item No.	Item	Designated portion	
		Designated for 13 segments	Designated for each layer
1	Transmission mode	○	–
2	Guard interval ratio	○	–
3	Carrier modulation system	–	○
4	Convolutional encoding ratio	–	○
5	Time-interleaving length	–	○

14.2 Physical Layer Operational Parameters

Table 14-2: List of assumed operational parameters

Item No.	Item	Range in ARIB STD-B46	Assumed operational parameters
1	Transmission mode	1, 2, 3	3
2	Guard interval ratio	1/4, 1/8, 1/16, 1/32	1/4
3	Carrier modulation system	QPSK, 16QAM, 64QAM, DQPSK	QPSK, 16QAM
4	Convolutional encoding ratio	1/2, 2/3, 3/4, 5/6, 7/8	16QAM: 1/2 QPSK: 1/2, 2/3
5	Time-interleaving length	(Mode 1) I=0, 4, 8, 16 (Mode 2) I=0, 2, 4, 8 (Mode 3) I=0, 1, 2, 4	(Mode 3) I=4

14.3 Parameters Used for Operation of Hierarchical Transmission

Table 14-3: Parameters used for the operation of hierarchical transmission

<table>
<tr><th rowspan="2">Pattern</th><th rowspan="2">Layer used</th><th rowspan="2">Layer name</th><th rowspan="2">No. of segments</th><th rowspan="2">Transmission (See Table 14-4.)</th><th colspan="2">TV broadcasting</th><th rowspan="2">Download (See Table 14-7.)</th><th rowspan="2">SI/EPG (See Table 14-8.)</th><th rowspan="2">Data / bidirectional (See Table 14-9.)</th><th rowspan="2">CAS (See Table 14-10.)</th></tr>
<tr><th>Video (See Table 14-5.)</th><th>Audio (See Table 14-6.)</th></tr>
<tr><td>(1)</td><td>A</td><td>Weak layer</td><td>1</td><td rowspan="6">Parameters in Table 14-4 are available in all layers.</td><td rowspan="6">Parameters in Table 14-5 are available in all layers.</td><td rowspan="6">Parameters in Table 14-6 are available in all layers.</td><td>c</td><td>a</td><td rowspan="6">Parameters in Table 14-9 are available in all layers.</td><td>b</td></tr>
<tr><td rowspan="2">(2)</td><td>A</td><td>Partial reception layer</td><td>1 (Partial reception)</td><td>b</td><td>c</td><td>b</td></tr>
<tr><td>B</td><td>Weak layer</td><td>12</td><td>a</td><td>a</td><td>a</td></tr>
<tr><td rowspan="3">(3)</td><td>A</td><td>Partial reception layer</td><td>1 (Partial reception)</td><td>b</td><td>c</td><td>b</td></tr>
<tr><td>B</td><td>Strong layer</td><td>11 to 1</td><td>-</td><td>b</td><td>a</td></tr>
<tr><td>C</td><td>Weak layer</td><td>1 to 11</td><td>a</td><td>a</td><td>a</td></tr>
</table>

Layer used: A, B, and C indicate layers set forth in TMCC.

Layer name: Layer name used in this operational guideline for convenience

Table 14-4: Transmission

Mode/guard ratio (Note 1)	Time interleaving	Modulation / Error correction		
	(Mode 3)	16QAM	QPSK	
1/4	I=4	1/2	2/3	1/2
○	○	○	○	○

○: Available transmission parameters

Note 1: The mode and the guard ratio are designated in common for all layers, and cannot be designated for each layer.

Table 14-5: Video

Coding system	H.264			
Level	MP@L3	MP@L3	MP@L1.3	MP@L1.2
Size	720×480 p	352×480 p	320×180 p	320×180 p
Frame rate	30/1.001 Hz	30/1.001 Hz	30/1.001 Hz	15/1.001 Hz
Scanning mode	Sequential scanning	Sequential scanning	Sequential scanning	Sequential scanning
Transmission	○	○	○	○

○: Transmittable

Table 14-6: Audio

MPEG-2 AAC LC / MPEG-4 HE AAC							
48 kHz / 44.1 kHz / 32 kHz				Multiple ES (Note 1)	24 kHz / 22.05 kHz / 16 kHz		
Mono	Stereo	Multichannel	Dual Mono		Mono	Stereo	Dual Mono
○	○	○	○	○	○	○	○

○: Transmittable

Note 1: The number of sound ES referred to by each service sent in the layer.

Table 14-7: Download

Type	Download contents	SDTT for weak layer		SDTT for strong layer	
		For receiver software	For common data	For receiver software	For common data (Note 2)
a	△ (Note 1)	○	×	×	×
b	×	×	×	○	○
c	×	○	×	○	○

○: Transmittable
△: Not obligatory to all operators, but transmitted at least at 1 TS
×: Not transmitted

Note 1: Download contents are transmitted at least 1 TS, but not always transmitted by all operators.
Note 2: SDTT of frequency list and change information is assumed.

Table 14-8: SI/EPG

Type	Corresponding layer
	Basic
a	N-EIT
b	M-EIT
c	W-EIT

Table 14-9: Data/Bidirectional

Transmission of MM services		MM service transmission system	Transmitted monomedia					Bidirectional protocol	Closed caption / superimposed text
Data coding identification	BML version		MPEG-1 Video	MPEG-2 Video	H.264	MPEG-2 AAC	Others		
0x001B	14.0	- Data carousel - Event message	×	×	○	○	- MPEG-2 AAC (file) - JPEG - GIF/aGIF	- TCP/IP (HTTP)	○

○: Transmittable

×: Not transmitted

Table 14-10: CAS

Type	CAT	EMM	ECM
a	×	○	○
b	○	○	○

○: Transmittable

×: Not transmitted

14.4 Source Coding System in Real-time Broadcasting

14.4.1 Video coding system

The video coding system employs the system set forth in ITU-T Rec. H.264|ISO/IEC 14496-10. Corresponding video formats are shown below.

Table 14-11: Video formats operated in real-time broadcasting

Profile	Level	Format	Image size	Aspect ratio	Frame rate [/1.001Hz]	Scanning mode	Maximum bit rate
Main	1.2	QVGA	320 × 180	16:9	15	P	384 kbps
	1.3	QVGA	320 × 180	16:9	30	P	768 kbps
	3	525HHR	352 × 480	16:9	30	P	6 Mbps
	3	525SD	720 × 480	16:9	30	P	

14.4.2 Audio coding system

The audio coding system should be capable of using MPEG-2 AAC Audio (ISO/IEC 13818-7), MPEG-4 HE-AAC (ISO/IEC 14496-3:2001/Amd.1), and MPEG-4 HE-AAC v2 (ISO/IEC 14496-3:2005/Amd2:2006).

MPEG Surround (ISO/IEC23003-1) may be used. Though the receiver shall be capable of receiving audio data with surround, decoding may be limited to the ACC base.

Table 14-12: Key parameters for audio coding system in real-time broadcasting

Item	Value
Bitstream format	AAC Audio Data Transport Stream (ADTS)
Sampling rate	48 kHz, 44.1 kHz, 32 kHz, 24 kHz, 22.05 kHz, 16 kHz
Profile	Low Complexity (LC)
Maximum number of encoded channels	Up to 5.1 channels per ADTS
PES packets	Asynchronous with sound frame is permitted.
Mute flag	Not used. (Muted by input signal.)

14.5 Source Coding System in Storage-based Broadcasting

14.5.1 Video coding system

The video coding system should employ the system set forth in ITU-T Rec. H.264|ISO/IEC 14496-10. Corresponding video formats are shown below.

Table 14-13: Video formats operated in storage-based broadcasting

Profile	Level	Format	Image size	Aspect ratio	Frame rate [/1.001Hz]	Scanning mode	Maximum bit rate
High	3.1	QVGA	320 × 240	4:3	15	P	10 Mbps
					30	P	
			320 × 180	16:9	15	P	
					30	P	
		VGA	640 × 480	4:3	30	P	
		525SD	720 × 480	4:3	30	P	
					60	P	
				16:9	30	P	
					60	P	
		720HD	1280 × 720	16:9	30	P	
	4.0	720HD	1280 × 720	16:9	60	P	
		1080HD	1440 × 1080	16:9	30	I	
			1920 × 1080	16:9	30	I	
					30	P	
	4.2	1080HD	1920 × 1080	16:9	60	P	20 Mbps

14.5.2 Audio coding system

The audio coding system complying with MPEG-4 audio (ISO/IEC 14496-3) should be used. However, HE-AAC v1 (ISO/IEC 14496-3: 2003:Amd.1), HE-AAC v2 (ISO/IEC 14496-3: 2005:Amd.2), and MPEG Surround (ISO/IEC 23003-1) should be added.

Though the receiver must be capable of receiving audio data with surround, decoding may be limited to the ACC base.

The coding sampling frequencies should be 16 kHz, 22.05 kHz, and 24 kHz in addition to the input sampling frequencies (32 kHz, 44.1 kHz, and 48 kHz).

The reception terminal should be capable of playing back all the above audio coding formats. Playback of part of its functions is also allowed depending on the number of channels which can be played back by the terminal.

Annex 1　Service Image

Basic functions of multimedia broadcasting are viewing of real-time broadcasting and storage of storage-based broadcasting contents including metadata. By achieving scheduled storage of storage-based broadcasting contents using recommend information received via communications or viewing/utilization of contents by clicking links in e-mail or browser for terminal (media scheme), scheduled storage/viewing, and transfer to EPG/ECG, multimedia broadcasting aims for a smooth transition to contents. Additionally, real-time broadcasting contents, storage-based broadcasting contents, browsers for terminal, and EPG/ECG can also be linked mutually.

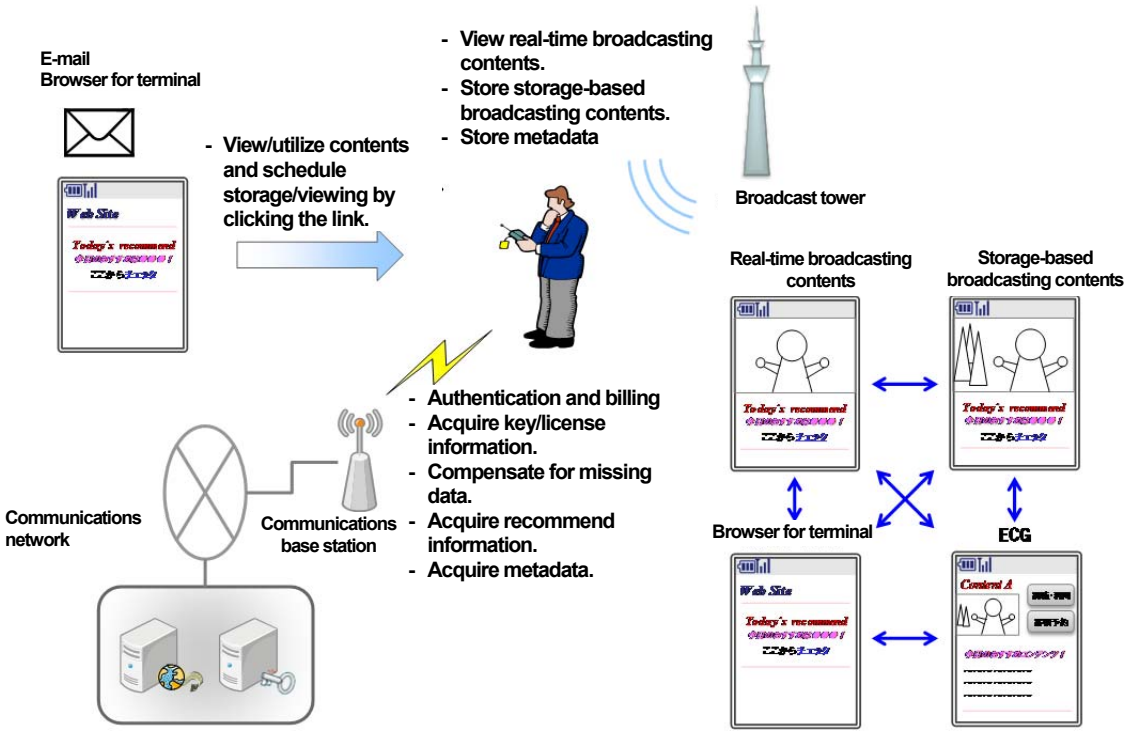


Fig. 14-1: Service image

ARIB TR-B33
Version 2.3-E1

VOLUME 1

Provisions for Download Operation for Multimedia Broadcasting

Contents

Chapter 1 General Terms

1.1 Introduction

The information updating service for receivers used in multimedia broadcasting is implemented according to the provisions of ministerial ordinance and notification, as well as according to the Association of Radio Industries and Businesses (hereinafter referred to as the "ARIB") standards, i.e., "Receivers for Terrestrial Multimedia Broadcasting based on Connected Segment Transmission" (ARIB STD-B53), "Data Coding and Transmission Specification for Digital Broadcasting" (ARIB STD-B24), and "Service Information for Digital Broadcasting Systems" (ARIB STD-B10). However, it is necessary to separately establish provisions for detailed operations in order to design receiver details, thus this "Provisions for Download Operation for Multimedia Broadcasting" was established. Further detail can be found in the service provisions of download service providers.

1.2 References

(1) "Service Information for Digital Broadcasting System" ARIB Standard, ARIB STD-B10

(2) "Receiver for Terrestrial Mobile Multimedia Broadcasting Based on Connected Segment Transmission (Desirable Specifications)", ARIB STD-B53

(3) "Receiver for Digital Broadcasting" ARIB Standard (Desirable Specifications), ARIB STD-B21

(4) "Data Coding and Transmission Specification for Digital Broadcasting" ARIB Standard, ARIB STD-B24

(5) "Conditional Access System Specifications for Digital Broadcasting" ARIB Standard, ARIB STD-B25

1.3 Terminology

Terms used in this Volume shall be defined as follows:

ARIB	Association of Radio Industries and Business: ARIB is an organization which standardizes technologies in relation to use of radio in Japan with participation by broadcasters, telecommunication operators, equipment manufacturers.
Engineering service	Service type 0xA4 information updating service in receivers notified by the SDTT procedure and transmitted by the data carousel.
Genre code	Code used in content_nibble_level_1 (4 bit) and content_nibble_level_2 field (4 bit) that specify genre. This is used in Content Descriptors, etc.
Schedule information	Download start time and download duration time stored in the SDTT schedule loop.
Updating service of the information in receivers	Service that distributes and stores receiver software, Common Data to All Receivers, logos, etc. in non-volatile memory, etc.
Module information	Information regarding downloaded modules that is set in module_info_byte in DII.
Approval information	Information that describes whether or not mandated downloading can be performed unconditionally depending on the update approval function of the receiver.
Partial reception/high/low protection layer	In hierarchical transmission, layers are referred to as high protection layers and low protection layers, respectively, in ascending order of the required CN ratio. The layer that carries out partial reception is referred to as a partial reception layer. The modulation system of the layer whose required CN ratio is equal to the lowest among the other layers or lower than those of other layers is applied to the partial reception layer.
SDTT for partial reception	SDTT transmitted via the partial reception layer or transmitted through single layer channel. The SDTT in the partial reception layer for reception software is not used for the transmission of download contents.
SDTT for the low protection layer	SDTT transmitted via the low protection layer or transmitted through single layer channel.
Mandated downloading	Downloading that requires mandated update
Notification Information	Information used for notifications such as downloading service_id, schedule information, receiver models for updating. Transmitted using SDTT.
Frequency list and modification information	Frequency list information and modification information that corresponds to frequency repacking, etc. (date of change, changed frequency, etc.).
Frequency repacking	For changing the current segment arrangement in order to structure an optimized network.
Receiver software	Conponents of the receiver software, such as, applications, librarys, OS, drivers, data, etc.

Product design	Receiver functions or actions which depend on the hardware design, the software design of the receiver planned by each manufacturer.
Selective (optional) downloading	Download that is implemented according to the viewer's selection after displaying downloads that can be implemented.
Common Data to All Receivers	Data stored in the receiver and commonly used in the receiver. Logo data, Genre Code Table, Program Characteristics Code Table, Reserved Word Table, Frequency List and Modification Information, etc.
Program characteristics code	Code used in user_nibble field (8 bit) that describes program characteristics. Used in Content Descriptors, etc.
Reserved word	Set terms such as "starring", "producer", "summary", etc. regarding program.

<Blank Page>

Chapter 2 Use and Premise of Downloading

- Software update for receivers

  This makes correction in software for receivers. It performs bug-fixing, correction of defects due to misunderstanding on operation between the transmitter and receiver, improvement of display, improvement in operation, etc.

- Updating Common Data to All Receivers

  (1) Updating Genre Code Table and Program Characteristics Code Table

  The Genre Code Table and Program Characteristics Code Table are not updated for multimedia broadcasting.

  (2) Updating the Reserved Word Table

  The Reserved Word Table is not updated for multimedia broadcasting.

  (3) Updating Frequency List and Modification Information

  Corresponding to frequency change is presumed to be possible causing as little trouble as possible for the viewers by notifying the changing frequency information to the receiver prior to the event, using an engineering service. Refer to each chapter in this volume for contents of SDTT, downloading contents and transmission procedure regarding the Frequency List and Modification Information for repacking correspondence. Also, refer to Vol. 2 for receiver operations regarding the Frequency List and Modification Information.

<Blank Page>

Chapter 3 Download Transmission Guidelines

3.1 Download Transmission Model for Multimedia Broadcasting

An overview of download transmission for multimedia broadcasting is described herein. Also, refer to the corresponding chapters for the format of each signal, transmission procedure, etc.

3.1.1 Download contents

There are two types of transmission procedures for downloaded content in multimedia broadcasting. They are both based on transmission in the low protection layer within layers used by each broadcaster (excluding simple logos). If single layer format is used, it shall be transmitted in that layer.

In the broadcasting target area, Common Data to All Receivers such as receiver software and genre, reserved words, frequency list are transmitted by at least 1 TS (specified by Service List Descriptor of the NIT).

The above is transmitted as an engineering service by using data carousels.

3.1.2 Notification information

The following two types of SDTTs (Software Download Trigger Tables) conduct downloading notification in the above and are transmitted on different PIDs. The transmission layer and transmission rate for each SDTT are according to Table 3-1.

Table 3-1: SDTT transmission layer and transmission rate

SDTT Type	Transmission layer	Model number	Standard description length (bytes)	Standard schedule number	Standard section size (bytes)	Cycle (seconds)	Maximum transmission rate (kbit/s)
SDTT for low protection layer	Low protection layer or transmission route whose layer number is 1	180	800	24	1287 *1 (Equivalent to 7TSPs)	180	10.5
SDTT for partial reception layer	Partial reception layer or transmission route whose layer number is 1	240	100	1	183 *1 (Equivalent to 1 TSP)	180	2.0

*1: Considering 1 byte for pointer field

Supplemental explanation:

- The SDTT for the low protection layer is transmitted on the low protection layer in its TS, and the SDTT for the partial reception layer is transmitted on the

partial reception layer (refer to Vol. 7, 5.3). If the SDTT is transmitted through a single layer channel, both the SDTT for the low protection layer and the SDTT for the partial reception layer are transmitted on the same layer.

- The SDTT in the partial reception layer for the receiver software shall only be used for receiver notifications, and the download contents shall not be accompanied. The SDTT for low protection layer, as the SDTT for BS/wideband CS digital broadcasting, have a function to notify receivers of the download content distribution date and to automatically conduct downloading. Refer to 3.3.1 “Transmission of notification iformation” for details.
- For Common Data to All Receivers, the Frequency List and Modification Information shall use the SDTT for the partial reception layer. Refer to 3.3.1 “Transmission of nofication iformation” for details.
- Model numbers here refer to model numbers when the SDTT section is transmitted in standard section size by all models at the maximum transmission rate. It is possible to transmit SDTT sections larger than the standard section size (4,096 bytes maximum), but the model number is reduced in that case. Refer to 3.3.1 “Transmission of Notification information” for details.

Table 3-2 describes the transmission pattern for signals regarding downloading. Also, Table 3-3 describes the transmission pattern for Common Data to All Receivers.

Table 3-2: Transmission pattern for signals regarding downloading

Pattern	Layer used	Segment No.	Download pattern *4	Download contents *1	SDTT			
					SDTT for the low protection layer		SDTT for the partial reception layer	
					Receiver software	Common data *2	Receiver software	Common data *3
(1)	A	1	c	×	○	×	○	○
(2)	A	1	b	×	×	×	○	○
	B	12	a	△	○	×	×	×
(3)	A	1	b	×	×	×	○	○
	B	11-1	—	×	×	×	×	×
	C	1-11	a	△	○	×	×	×

○: Transmittable

△: Not required for all broadcasters, but transmitted by at least 1 TS in the broadcasting target area

×: Not transmitted

*1: For download contents, at least 1 TS is transmitted in the broadcasting target area, but not necessarily transmitted by all broadcasters.

*2: SDTT for data used by all receivers such as the Genre Code Table, Program Characteristics Code Table, Reserved Word Table, etc.

*3: SDTT for the Frequency List and Modification Information for frequency repacking.

*4: For download patterns, refer to Table 14-3 and Table 14-7 in 14.3 of Volume 0.

Table 3-3: Transmission pattern for Common Data to All Receivers

Pattern	Layer used	Segment No.	Download pattern	Engineering service	SDTT for the partial reception layer
				Frequency List and Modification Information	Frequency List and Modification Information
(1)	A	1	c	×	○
(2)	A B	1 12	b a	× ○	○ ×
(3)	A B C	1 11-1 1-11	b - a	× × ○	○ × ×

3.1.3 Engineering services for multiple TS time shift transmission

When implementing the multimedia broadcasting engineering service on the specified TS as BS and wideband CS, the difficulty of engineering services where the TS cannot be received is expected at some point in some areas. It is ideal to implement the engineering service in multiple or all TS's in order to resolve this. When implementing the engineering service in multiple or all TS's, reception probablity will improve by each TS transmitted by differing the transmission time rather than transmitting the download contents at the same time due to the fact there are more opportunities for reception. Refer to corresponding chapters for operations of PSI/SI, SDTT, and download contents in case of engineering services for multiple TSs time shift transmission.

3.1.4 Downloading Frequency List and Modification Information

The Frequency List and Modification Information shall use both of the two following transmission procedures to distribute to receivers.

3.1.4.1 SDTT for the partial reception layer

Even receivers not capable of handling engineering services transmit the minimum necessary information that is possible to be handled (commencement date of change, work time period of change, etc.) and encourage receivers (viewers) to perform frequency scanning. At the same time, link information is transmitted to the following engineering services.

3.1.4.2 Engineering service

The further detailed information (frequency list before and after change, date of change, etc.) is transmitted to the receivers that can handle engineering services, thus making it possible to automatically follow the change of frequencies so that viewers can spend as less time and effort as possible.

The Frequency List and Modification Information can not only be used for frequency repacking but also in the cases as follows.

- Supporting at the receiver to detect new broadcastors join or start broadcasting using existing transmitters.
- Supporting to add stations to the reception table in the receiver whose broadcasting was on pause in the time of installation scanning.

For details, refer to the corresponding chapters of this volume and Vol. 2, 5.12 “Download”.

3.2 Transmission of PSI/SI Regarding Downloading

3.2.1 Operation of engineering service

The engineering service shall always perform the following operation in networks where it exists.

(1) “Engineering service type (0xA4)” shall be placed in the second loop of Service List Descriptor of the NIT.

(2) Data_component_id in Data Component Descriptor in PMT is 0x0009 (ARIB-data download procedure). Additional_data_component_info is not transmitted.

(3) The service_id of the engineering service shall be operated with the same ID value nationwide if the engineering services is transmitted in a single TS. However, the service_id is assigned for each schedule if a time shift transmission is scheduled on multiple TS’s. The network_id and transport_stream_id value shall be the value each transmitted network. Refer to 3.3.2 “Transmission of Download Content” for details on the service_id assignment.

(4) The original_network_id and transport_stream_id values, set in the NIT, PAT and BIT that describe the engineering service transmitted in the concerned network, shall be the values for each network transmitted. Therefore, in order to omit replacement operations by each broadcasting station of original_network_id and transport_stream_id values in the SDTT, the value of the original_network_id and transport_stream_id in the SDTT in case of transmission routes A and B provided in 3.3.1.2 “Operation of SDTT” are set to the value unique for maultimedia broadcasting. Refer to 3.3.1.2 “Operation of SDTT (Software Download Trigger Table)” for details.

(5) There may be cases where PCR is not transmitted in engineering service.

(6) A maximum of 1 second is permitted for the PMT repetition interval of engineering services.

3.2.2 Detection of download related table transmission

3.2.2.1 SDTT transmission detection

There are two types of SDTTs, one for the low protection layer (PID=0x0023) and one for the partial reception layer (PID=0x0028), but they are both transmitted in case of transmission, and the repetition rate is same as well.

In case of SDTT transmission in the concerned network, set SDTT (0xC3) as the table_id value of "SI Parameter Descriptor" placed in the second loop of the BIT. Therefore, receivers can detect the fact that the SDTT is transmitted in the TS in the concerned network by the fact that 0xC3 is set there.

3.3 Transmission of SDTT Procedure

3.3.1 Transmission of notification information

3.3.1.1 Transmission route

In principle, in all TSs, transmit the SDTT for the low protection layers in the low protection layer of its TS and the SDTT for partial reception layers in the partial reception layer (refer to Vol. 7, 5.3).

If transmitted in a single layer channel, transmit both the SDTT for low protection layers and the SDTT for partial reception layers in the layer. Also, it is possible that there may be some TSs whose notification information is not transmitted. The SDTT for low protection layers is transmitted in a PID of 0x0023, and the SDTT for partial reception layers is transmitted in a PID of 0x0028.

3.3.1.2 Operation of SDTT (software download trigger table)

(1) Operations of SDTTs for low protection layers and SDTTs for the partial reception layer

The operations of SDTTs for low protection layers and SDTTs for partial reception layers are shown in Table 3-4. Table 3-1 is shown again as Table 3-4 below.

Table 3-4: SDTT transmission layer and transmission rate

SDTT Type	Transmission layer	Number of models	Typical description length (bytes)	Typical schedule number	Typical section size (bytes)	Cycle (seconds)	Maximum transmission rate (kbit/s)
SDTT for low protection layer	Low protection layer or transmission route whose layer number is 1	180	800	24	1,287*1 (worth 7 TSPs)	180	10.5
SDTT for partial reception layer	Partial reception layer or transmission route whose layer number is 1	240	100	1	183*1 (Worth 1 TSP)	180	2.0

*1: Considering 1 byte for pointer field

Receiver software's SDTT for low protection layers shall be used for transmission of information necessary for the reception of download contents, and SDTTs for the partial reception layers shall be used to notify receiver software update and updating procedure for users of receivers that cannot receive low protection layers or receivers in the status that cannot receive low protection layers. Typical section size and schedule number for both SDTT are shown in Table 3-6. This "typical section size of SDTT" refers to the section length when divided evenly by number of models at the maximum transmission rate (refer to later estimate). However, it is possible to transmit sections whose size exceeds this (4,096 bytes maximum). Also, a typical schedule number is only a guide line; operation of schedule numbers over this is also possible under the restriction of the maximum transmission rate.

When the original_network_id of download contents transmission routes set in the SDTT is 0x7FFF, it shall correspond to content transmission A and B described in Table 3-5. In other cases, it implies that the download contents are being transmitted via a transmission route specified by the SDTT original_network_id, transport_stream_id. Also, when the original_network_id is 0x7FFF, the transport_stream_id shall indicate the following meanings.

transport_stream_id:

0xFFFF: partial reception layer SDTT, as well as content transmission route specification B

0xFFFE: low protection layer SDTT, as well as content transmission route specification B*2

0xFFFD: partial reception layer SDTT, as well as content transmission route

specification A

0xFFFC: low protection layer SDTT, as well as content transmission route specification A

*2: Not operated in multimedia broadcasting

Table 3-5: Content transmission route specifications

Transmission route specification	Type of content transmission specification	service_id set in SDTT	Meaning
0	Content transmission route specification A	0xFFF0 (Refer to 3.3.2.1 "Transmission Route" for details)	Download contents are transmitted using the service specified by the service_id of the TS that conducts multimedia engineering services common to the entire country.
1	Content transmission route specification B	0xFFFF	Download contents are not transmitted in broadcasting.

SDTT sub-table of Common Data to All Receivers shall be a maximum length of 1 section and 4,096 bytes maximum, and shall only be transmitted in the SDTT for low protection layers. Also, multiple sections are possible in each sub-table in case of receiver software updates, but the entire section number including receiver software update and Common Data to All Receivers shall be 180 maximum in the SDTT for low protection layers and 240 maximum in the SDTT for the partial reception layers.

(2) SDTT operation upon multiple TS time shift transmission

When conducting time shift transmission with multiple numbers of TS, the service_id set in the SDTT shall be operated in 0xFFF0 for the time being. For schedule information, set the schedule information of download contents whose service_id is 0xFFF0. Also, operate SDTT schedule time-shift information (refer to ARIB STD-B53).

Operation of schedule time-shift information shall be as follows.

4bit	Meaning
0	This shows that although engineering service is being implemented in multiple numbers of TS, download contents are being distributed at the same schedule for everything with no time shift.
1 -12	This shows that engineering service is implemented in multiple numbers of TS, and schedule time shift unit is 1 to 12 hours.
13 -14	Reserved
15	Engineering service is implemented in single TS.

(3) For receiver software update

Set one of “today“, “today and tomorrow” and “tomorrow” for the SDTT schedule information with download contents distribution. SDTT schedule information content without download contents distribution shall be for the private use of receiver makers that implement downloading. For example, use of a start date of information updating service in receivers through different communication lines and start dates of version upgrading service for stores can be expected. There are no provisions for the number of schedule repetetion, the number of Download Content Descriptors, the types of information, text length, module number and etc in the Download Content Descriptor. However, the number of schedule repetition is restricted by the broadcasting program. Also, the download_id shall all be the same regardless of the number of Download Content Descriptors. Also, it is possible to set the number of download content as other than 1 in order, for example, to display different messages for each group by using the group_id, but the schedule and download_id shall be the same.

Furthermore, assigned id’s as the maker_id (numbered by ARIB) shall be used. Also, operations of model_id, version_id, and group_id in case of downloads such as receiver software updates, etc. based on application by receiver makers shall be managed by each receiver maker and the content is not specified in this document.

Also, the use of multiple sections is possible in each sub-table in order to transmit download content that is different for each group by operating the group_id.

Table 3-6: Id’s for update target

	No. of bits	
maker_id	8	Value to discriminate makers
model_id	8	Value to discriminate receiver models
group_id	4	Value to group download
version_id	12	Value of system version

Refer to ARIB STD-B53 for the structure and meaning of SDTT.

Refer to ARIB STD-B21 for examples of notification information operations.

(4) For Common Data to All Receivers

Operation of the SDTT for Common Data to All Receivers is as follows. Furthermore, specific operations for the Genre Code Table, Program Characteristics Code Table and Reserved Word Table is described in (a), and specific operations for Frequency List and Modefication Information is described in (b).

SDTT sub-table of Common Data to All Receivers is constructed from 1 section. The loop

number of schedule information shall be “0”, and the SDTT shall only be transmitted while the carousel of the download contents are being transmitted. Operation of the version_id is described in 3.3.1.6(2). The group_id shall always be “0”, and the group_id is not operated for Common Data to All Receivers.

Also, the num_of_contents shall be “1”. The compatibility_flag and text_info_flag in the Download Content Descriptor shall basically be “0”, and this information can be ignored by receivers. The data in module_info_byte shall be the same as module_info_byte in DII. The private data length shall also basically be “0”, and this information can be ignored by receivers. Furthermore, add_on of the Download Content Descriptor shall also always be “0” and not be used.

(a) For the Genre Code Table, Program Characteristics Code Table and Reserved Word Table

The Genre Code table, Program Characteristics Code Table, and Reserved Word Table are not updated for multimedia broadcasting.

(b) For Frequency List and Modification Information

The SDTT for the partial reception layer shall be used with the maker_id and model_id fields set to (0xFFF8). Along with that, version_id of the Frequency List and Modification Information shall be managed separately from the above (a), and its operation shall be according to the description in 3.3.1.6 (2).

The module_id value in the Download Content Descriptor of the download contents shall be the fixed value of 64. Accordingly, receivers do not detect its service area is with the terrestrial distribution descriptor, service_id values and etc. in the received broadcasting waves.

The module_info_byte can be assumed to contain Control Descriptor only, and receivers can ignore the descriptors other than that. A Control Descriptor is capable of transmitting the information showing all segments in the event of a change made to the schedule for reasons such as new broadcasters or frequency repacking, or even in the event of no schedule.

If new station placement or a frequency change is scheduled in the area, set the same schedule information as the corresponding download contents in the module_info_byte. Also, set the same version number that is given to the corresponding download contents. The module_id shall always be set in ascending order and separated into several descriptors continuously if the download_content_descriptor length exceeds 254 bytes. Also, the Control Descriptor number per one area (module) shall be 1. The maximum value of the schedule loop described in the Control Descriptor shall be 8. Even if there is no modification in that region (module), the length of the Control Descriptor shall be 2; and download_data_type and module_data_version shall be sent out. The syntax of the Control Descriptor is described in

Table 3-7.

Table 3-7: Control Descriptor

Syntax	No. of bits	Mnemonic
Control_descriptor(){		
descriptor_tag	8	uimsbf
descriptor_length	8	uimsbf
download_data_type	8	uimsbf
module_data_version	8	uimsbf
for(i=0; i< N; i++){		
start_date	16	uimsbf
duration	8	uimsbf
}		
}		

download_data_type: Content is as follows:

download_data_type	Meaning
0x01	To use the Control Descriptor for the frequency list and modification information
Other than 0x01	Reserved

module_data_version: Version value of frequency information. This value is the same as the value given to download contents.

start_date: Specifies the start date of changing operations.

duration: Duration required for modification is displayed in a number of days (1-254) . 0x00 and 0xFF are reserved.

3.3.1.3 Transmission layer, interval and bandwidth

The transmission cicle of the SDTT shall be once per 3 minutes. The Bandwidth shall be 10 kbit/s or less for the low protection layer, and 2 kbit/s for the partial reception layer (refer to the Bandwidth estimate below)

However, no transmission is performed if there is no downloading schedule. If receivers cannot receive the SDTT for 3 minutes in the TS that transmits the SDTT indicated in the second loop of the BIT, the SDTT for the day can be considered non-existent. However, it is desirable to operate supposing that the SDTT may not be transmitted due to a facility failure, etc.

Bandwidth estimate

Low protection layer	Partial reception layer
○ Calculation premise	
- Typical length for each section of SDTT is 1,288 bytes	- Typical length for each section of SDTT is 184 bytes
- Maximum number of sections is 180	- Maximum number of sections is 240

- Multi section transmission in TS packet is not performed.

- Transmitted once every 3 minutes

○ TS packets necessary in each section is	
1,288/184 = 7 TS packets	184/184 = 1 TS packets
○ TS packets necessary for the entire table is	
7×180 = 1,260 TS packets	1×240 = 240 TS packets
○ Average TS rate to transmit this once every 3 minutes is	
(1,260×188×8)÷180=10.53kbit/s	(240×188×8)÷180=2.01kbit/s

Note: Typical section length here includes pointer field of 1 byte.

(1) For receiver software updates

Schedule information in the SDTT shall be set so that the maximum sub-table size, interval and Bandwidth are consistent.

(2) For Common Data to All Receivers

The number of schedule information loop shall be 0, and the SDTT shall be transmitted only while the carousel of the download content is being transmitted.

3.3.1.4 SDTT update

Fundamentally, the SDTT shall only be updated once a day at AM 0:00. However, whenever there is need for urgent update, it shall be updated as necessary.

3.3.1.5 Provisions for SDTT TS packeting and transmission

This shall be based on Vol. 4 of the document.

3.3.1.6 Download content version number

(1) For receiver software updates

Version numbering procedure, management and such are not provided.

(2) For Common Data to All Receivers

Regarding version_id operation, use the common version number of Common Data to All Receivers that is managed by broadcasters who transmit Common Data to All Receivers. For the Frequency List and Modification Information, all Frequency List and Modification Information nationwide are transmitted, and the version number is incremented by 1 in case of any change in transmission content. It's up to product design for how much information are stored in receivers at the time of shipment.

When the version_id exceeds 4,095, it shall be wrapped around to 0. In that case, divide the version_id into domains that are from 0 to 2,047 and 2,048 to 4,095 in order to manage the version value from 0 again. If the current version value is in the domain from 2,048 to 4,095 and the downloaded common data version value is between 0 and 2,047, the receivers shall consider that it is higher than the current version value and implement downloading.

3.3.2 Transmission of download content

Refer to ARIB STD-B53 for additional provisions to the DSM-CC data carousel.

Refer to ARIB STD-B53 for DII Module Info operation.

Also, multi-section transmission is possible for transmission of download content.

3.3.2.1 Transmission route

The layer that transmits download content shall be the low protection layer, and it shall be transmitted in different transmission routes other than the original_network_id and transport_stream_id set in the SDTT. Refer to 3.3.1.2 "Operation of SDTT" for details. The service_id assigned for the download content by this procedure shall be 0xFFF0-0xFFF5.

The service_id for single TS transmission is operated in 0xFFF0 nationwide. Receivers acquire download content based on the service_id set in the SDTT. Even when time shift transmission is performed with the multiple service_id by multiple TS's, the service_id set in the SDTT shall be operated in 0xFFF0 nationwide. The maximum number for the service_id assigned for time shift transmission shall be 6. Meaning, it can be operated from 0xFFF0 to 0xFFF5. Receivers extract the engineering service transmitted in the service_id +5 or less that includes the service_id set in the SDTT. With engineering services transmitted in service_id's that are different from the one set in the SDTT, operation of presumed time shift transmission for that difference is being conducted.

If there is an engineering service whose service_id is other than 0xFFF0-0xFFF5, receivers may ignore this. Refer to Figure 3.3 "Transmission model of multiple TS time shift transmission".

3.3.2.2 Bandwidth

The Bandwidth shall be 0.35207 Mbit/s (corresponds to 0.63 segment in a transmission model that is 16QAM, convolution 1/2, and guard ratio 1/4) at maximum and 0.1 Mbit/s at minimum for the TS rate. Broadcasters can select specific values in this range related to downloading. When transmitting engineering services with multiple TS's, bandwidth shall be the same for each.

[Bandwidth estimate]
Estimation formula for the number of transmittable models under the transmission model of 16QAM, convolution 1/2, guard ratio 1/4, and used segment 0.63 is shown below. Download content is assumed to be 10 MB.
Because the net bandwidth in 1 segment is 561.66 kbit/s,
Useable bandwidth = useable bandwidth in 1 segment × used segment
= 561.66×0.63 = 353.84 [kbit/s]
Time needed for one transmission = Download content/useable bandwidth
=10×1,024×1,024×8 / (353.84×1,000) = 237.07 [sec]

Transmittable model number (times)= 24 hr./time needed for one transmission/carousel times
=86,400/237.07/2 = 182.2 times
(In actual transmission in the TS, it will include a TS header section, adaptation field, etc., so the value will be smaller than the calculated value.)

3.3.2.3 Transmission time, period and interval

Downloaded content should be transmitted in succession during the airtime on a interval of 24 hours at maximum.

Single downloaded content should be transmitted using more than one unit and the duration of one unit is 10 seconds.

(1) For receiver software updated

An example of a downloaded content transmission schedule is shown in Figure 3-1. Receiver makers that would like to download content apply to a download organization for the transmission term, desired number of times of transmission, and duration (in units of 10 seconds). The transmission time changes depending on the number of models for which content is downloaded. The download organization will decide the schedule after adjustments are made between all receiver makers that will download the content. A carousel is transmitted for the number of maximum transmittable cycles (integer) within the duration (in units of 10 seconds) applied by a receiver maker. After the downloaded content is transmitted, dummies are inserted until the duration ends. Dummy is to be a null packet or an empty carousel packet, or the content of dummy is to be only an adaptation field.

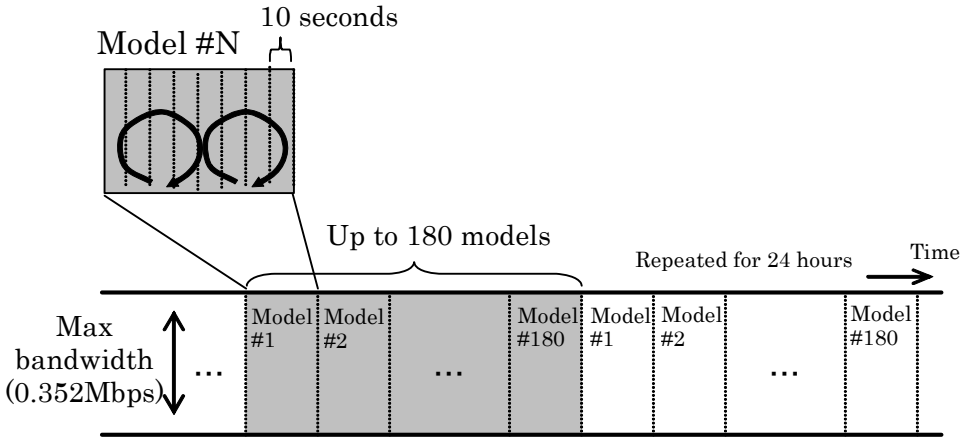


Figure 3-1: Reference model for downloaded content transmission schedule
(When there is no Common Data to All Receivers)

(2) For Common Data to All Receivers updated

An example of a downloaded content transmission schedule is shown in Figure 3-2. When Common Data to All Receivers needs to be updated, a broadcaster or an organization that would like to download content will apply to a download organization for the transmission period. The transmission time changes depending on the number of models for which content is downloaded. The download company will decide the schedule after adjustments are made between all receiver makers that will download the content.

A carousel is transmitted for more than two cycles and data for receiver software is transmitted at the same time. After the downloaded content is transmitted, dummies are inserted until the duration ends. Dummy is to be a null packet or an empty carousel packet, or the content of dummy is to be only an adaptation field.

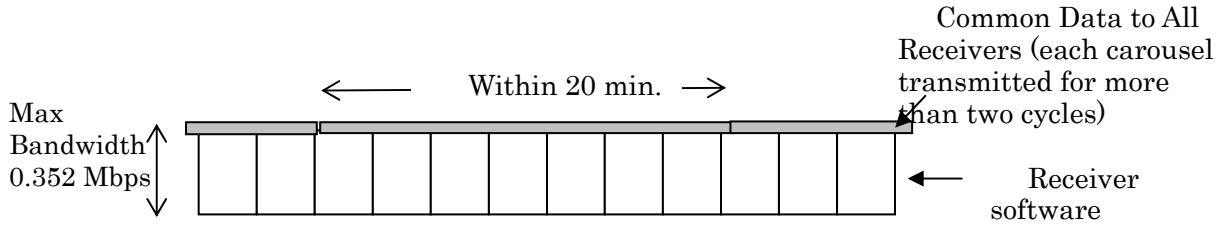
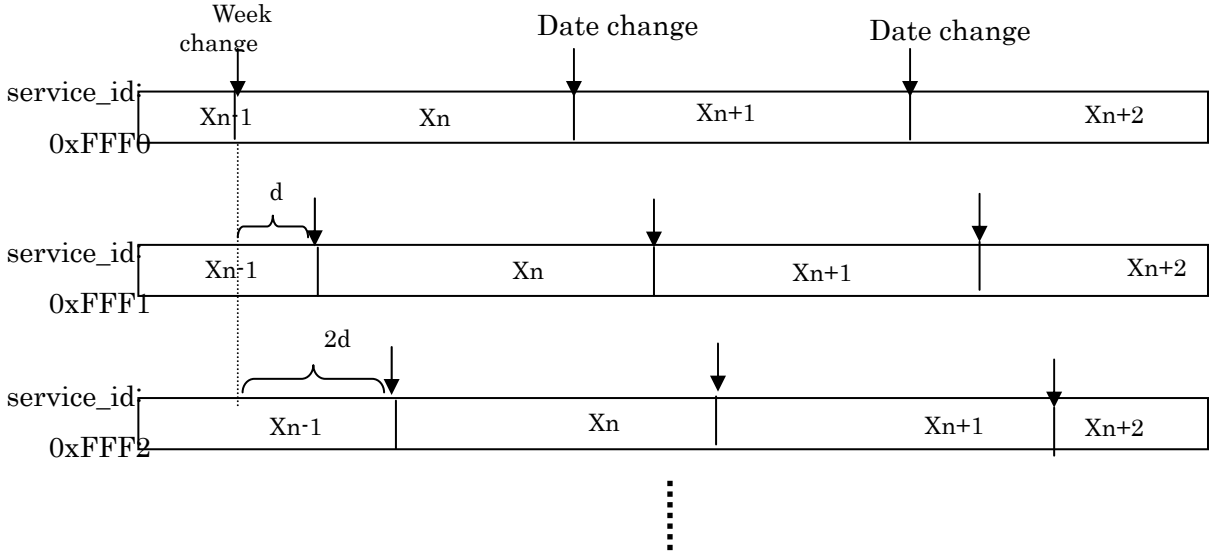


Figure 3-2: Reference model for downloaded content transmission schedule

(3) For more than 1 TS time shift transmission



Xn: Downloaded content transmitted on the n-th day.

d: Time shift unit (Specified with schedule_time_shift_information (Tentative name) in the SDTT)

Figure 3-3: Transmission model of multiple TS time shift transmission

Figure 3-3 shows a transmission model when downloded content is transmitted in more than 1 TS at different times. The schedule whose service_id is 0xFFF0 is the basic schedule and its schedule information is descirbed in the SDTT. When the time shift unit specified with schedule_time_shift_information in the SDTT is d hours, transmission of the TS whose service_id is 0xFFF1 starts d hours, 0xFFF2 is 2d hours and 0xFFF3 is 3d hours respectively, after transmision of the TS in the 0xFFF0 schedule has started.

- In the 0xFFF0 schedule information, the 0xFFF0 schedule for the day and the next day is described. The transmission start time of the 0xFFF1 schedule is worked out by adding d hours to the transmission start time of the 0xFFF0 schedule.
- The product of the number of service_id's and the amount of time shifted should be within 12 hours.

3.3.2.4 Assignment of PIDs and component_tag Values to downloaded content

In order for receivers to capture downloaded content reliably, a total of 12 different types of pairs (PID and component_tag value); i.e., two different types for Common Data to All Receivers ([Frequency List and Modification Information] for actual broadcasting and prior test broadcasting, respectively) and eight different types for receiver software are applied to the downloaded contents.

Eight different pairs of a PID and a component_tag value are allocated to receiver software in a circular order of transmission.

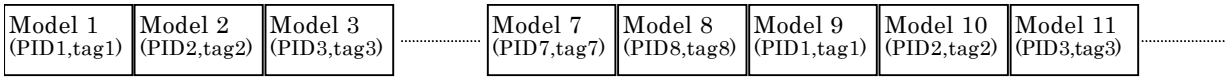
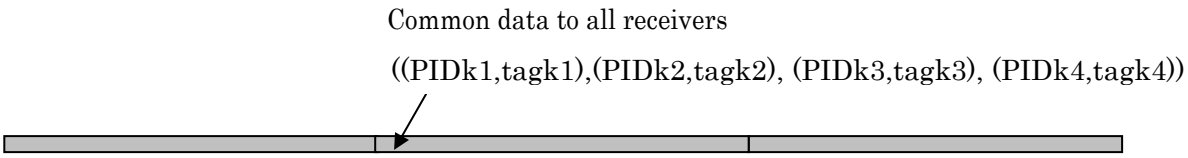


Figure 3-4: Assignment of PIDs and component_tag values
(without Common Data to All Receivers)

The following figure shows an example of when data for receiver software and Common Data to All Receivers is transmitted simultaneously.



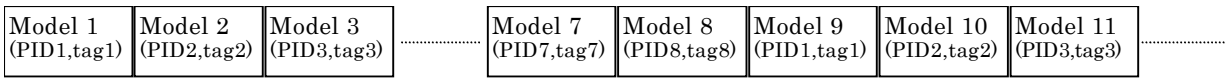


Figure 3-5: Assignment of PIDs and component_tag values
(when data for receiver software and Common Data to All Receivers is transmitted simultaneously)

When software for 8 models or less is updated, dummies are inserted as shown below. One dummy is transmitted for 10 seconds and a dummy is to be a null packet or an empty carousel packet, or the content of dummy is to be only an adaptation field.

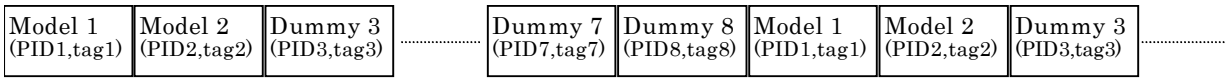


Figure 3-6: Assignment of PIDs and component_tag values when software of 8 models or less is updated

When the number of models for which data is transmitted is not in multiples of 8, dummies should be transmitted so that the number will become multiples of 8 and this should be repeated.

Ex: When software for 70 models is updated, two dummies should be transmitted (PID 7 and PID 8) and this cycle is repeated for the number of 72 models.

Additionally, when the day changes, adjustments should be made using dummies as shown above, so that PIDs and component_tag values are allocated in circular manner.

3.3.2.5 Module and carousel structures

(1) For receiver software updated

Modules that make up a carousel are not specified in this document.

(2) For Common Data to All Receivers updated (Genre Code Table, Program Characteristic Code Table and Reserved Word Table)

The Genre Code Table, Program Characteristics Code Table, and Reserved Word Table are not updated for multimedia broadcasting.

(3) For Common Data to All Receivers (Frequency List and Modification Information)

All of the data for the entire network of multimedia broadcasting is transmitted as download content.

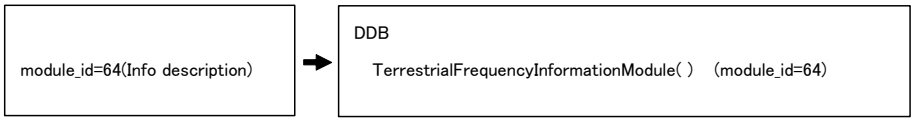


Figure 3-7: Example of a carousel structure of common data
(Frequency List and Modification Information)

3.3.2.6 Operation of DII (DownloadInfoIndication)

(1) For receiver software update

For the purpose of reliable downloading, transmission of the Compatibility Descriptor that specifies the model for which data is downloaded is mandatory. The values of maker_id, model_id, version_id, group_id and download_id should be the same as the ones of the id's in the SDTT. Other details are not specified.

(2) For Common Data to All Receivers update (Genre Code Table, Program Characteristic Code Table and Reserved Word Table)

The Genre Code Table, Program Characteristics Code Table, and Reserved Word Table are not updated for multimedia broadcasting.

(3) For Common Data to All Receivers update (Frequency List and Modification Information)

For the purpose of reliable downloading and transmission of module information, the transmission of the Compatibility Descriptor that specifies the model for which data is downloaded is mandatory. The values of maker_id, model_id, version_id, group_id and download_id should be the same as the ones of id's in the SDTT. The description in module_info_byte should be identical as the one in module_info_byte in the SDTT and only the description of the Control Descriptor is mandatory.

3.3.2.7 Operation of DDB (DownloadDataBlock)

(1) For receiver software is update

The details are not specified.

(2) For Common Data to All Receivers update

The data format of the Frequency List and Modification Information are shown in Table 3-8.

Table 3-8 Syntax of the requency List and Modification Information

Syntax	No. of bits	Mnemonic
TerrestrialFrequencyInformationModule(){		
area_code	8	bslbf
module_length	16	uimsbf
module_data_version	8	uimsbf
reserved	4	bslbf
num_of_network_loop	12	uimsbf
for(i=0; i< num_of_network_loop; i++){		
network_id	16	uimsbf
version_number	8	uimsbf
reserved_future_use	4	bslbf
number_of_affiliation_id	4	uimsbf
for(j=0; j< number_of_affiliation_id; j++) {		
affiliation_id	8	uimsbf
}		
num_of_schedule	8	uimsbf
for(k=0; k< num_of_schedule; k++) {		
start_date	16	uimsbf
duration	8	uimsbf
num_of_detail_information	16	uimsbf
for(l=0; l< num_of_detail_information; l++) {		
transport_stream_id	16	uimsbf
transmitter_id	24	uimsbf
zip_code	24	uimsbf
change_date	8	uimsbf
change_time	8	uimsbf
simul_duration	8	uimsbf
old_frequency	16	uimsbf
old_transmission_power	8	uimsbf
new_frequency	16	uimsbf
new_transmission_power	8	uimsbf
}		
}		
num_of_unchange_transmitter	16	uimsbf
for(m=0; m< num_of_unchange_transmitter; m++) {		
transmitter_id	24	uimsbf
zip_code	24	uimsbf
physical_ch	8	uimsbf
transmission_power	8	uimsbf
}		
text_info_length	8	uimsbf
for(n=0; n< text_info_length; n++){		
text_char	8	uimsbf
}		
}		
for(p=0; p< N; p++){	8	bslbf
reserved_future_use		
}		
}		

area_code: Area identification value that shows the service area. The area identification value is a fixed value of 64 (0x40) for multimedia broadcasting.

module_length: Data length after the module_data_version field (in bytes)

module_data_version: The version number of the module data. Incremented by one each time the content is changed. Same as the value described in module_info_byte in the SDTT.

num_of_network_loop: Number of broadcasters in the broadcasting area. Multimedia broadcasting shall be conducted as 0x001 fixed.

network_id: Network identification of the relevant multimedia broadcasting. Same as the value set in the NIT.

transport_stream_id: The transport stream identification of the relevant multimedia broadcasting. Same as the value set in the NIT.

version_number: Version number of the frequency information for the relevant network.

number_of_affiliation_id: Number of identifications of affiliations to which the relevant network belongs. "0" shall be described for multimedia broadcasting because of no identification for belonging affiliation.

affiliation_id: Same value as the affiliation identification value described in the BIT. Can be used to identifify such as affiliation station enabling successive reception of services provided by affiliation station while moving. No assignment shall be given for multimedia broadcasting because of no identification for belonging affiliation.

num_of_schedule: Describes the number of schedules such as frequency change and establishment of a new station. "0" is set when there is no change.

start_date: Modification start date (in MJD notation).

duration: The duration of time that is required to make a change is shown in the number of days (in Hex notation. 1 to 254 in decimal). 0x00 is set when the duration of time is unknown. 0xFF is set when the duration of time is 255 days or more.

num_of_detail_information: Number of transport streams to which changes are made regarding operation during the above duration of time in the relevant network.

transmitter_id: The identification of a transmitter to which changes are made. However, transmitters that re-transmit in low power in the same frequency due to the restrictions on the size of memory of receivers can be omitted. A unique value is set to the transmitter_id in nationwide. Multimedia broadcasting shall be conducted with "0x000001" fixed because SFN operation is conducted. The operation on the side of receivers shall be conducted with this value fixed.

zip_code: Describes the zip code value of a transmitter to which changes are made in 7 digits (in decimal notation) in binary form. Multimedia broadcasting shall be conducted with "0xFFFFFF" fixed because SFN operation is conducted. The operation on the side of receivers shall be conducted without using this value.

change_date: Describes a specific work date using a difference value (in Hex notation) from the start_date. However, receivers must take into consideration the actual work date is maybe a little different from the described work date. Additionally, 0xFF is set when unknown. The

change_date is not a day when test signal is transmitted and is a day when the construction is completed, and it should be transmitted continuously and stably after change_date as a rule.

change_time: Describes change time in units of hours (in Hex notation). However, receivers must take into consideration the actual change time may be a little different from the described change time. 0xFF is set when unknown.

simul_duration: In simultaneous operations, when data is transmitted in the old frequency for a certain period of time, this period of time is described by the number of days (in Hex notation. 0 to 254 in decimal). 0xFF is set when unknown. Please note that the simul_dulation indicates the simul operation duration after the change_date.

old_frequecy: The frequency before the change shall be described. The unit shall be 1/7 MHz.

old_transmission_power: Transmission power before change in dBm (in Hex notation). The value is rounded off. 0xFF is set for a newly established station and 0dBm does not mean output is nothing.

new_frequency: The frequency after the change shall be described. The unit shall be 1/7 MHz.

new_transmission_power: Transmission power after change in dBm (in Hex notation). The value is rounded off. 0xFF is set for a station that stopped transmission such as when broadcasting ends. Please note that normal broadcasting with reduced power (for example, when the transmission power is changed temporarily as the transmitting site in Tokyo Tower is changed to a backup one due to maintenance) is not regarded as a transmission power change.

num_of_unchange_transmitter: The number of transmitters to which no changes are made regarding operation during the above duration of time in the relevant network. Multimedia broadcasting shall be conducted with “0x0” fixed because SFN operation is conducted.

transmitter_id: Identification of a transmitter to which no change is made regarding operation during the above duration of time. However, transmitters that re-transmit in low power in the same frequency can be omitted due to the restrictions on the size of memory in receivers. Multimedia broadcasting shall be conducted with “0x000001” fixed because SFN operation is conducted. The operation on the side of receivers shall be conducted with this value fixed.

zip_code: Describes the zip code value of a transmitter to which no change is made in 7 digits (in decimal notation) in binary form. This shall be conducted with “0xFFFFFF” fixed because SFN operation is conducted for multimedia broadcasting. The operation on the side of receivers shall be conducted without using this value.

physical_ch: A physical channel number of a transmitter to which no change is made (in Hex notation).

transmission_power: Transmission power of a transmitter to which no change is made in dBm (in Hex notation). The value is rounded off. 0xFF is set when a station that stopped transmission needs to be described.

text_char: When establishment of a new station or frequency repacking, etc. is planned in the relevant network, character strings for notification to viewers can be described but this will not be used for the time being, and receivers must skip this even if the value of text_info_length is other than 0.

It is presumed that when receivers have the GUI to present these strings in the future, they will be able to present information such as that a new station will be established or frequency will be changed in the area. Therefore, receivers must safely ignore any strings in this field as of now.

3.3.3 Timing for transmitting notification information and download contents

3.3.3.1 For receiver software update

The transmission start time of download contents is specified with the start_time in the SDTT. So, it is desirable that control is performed so that schedule confirmation by re-acquiring the SDTT and receiver wakeup control to acquire download contents, etc. are completed before start_time and no data in the first time download contents is missing.

3.3.3.2 For common data to all receivers update

Whether download content is being transmitted can be judged by whether the SDTT is being transmitted. So, it is desirable that control is performed so that re-acquisition of the SDTT and receiver wakeup control to acquire download contents, etc. are completed before acquiring the DII and no data in the first time download content is missing.

3.3.4 Emergency suspension of the receiver built-in information update service (SDTT method)

Transmission of download contents may be stopped due to an emergency stop of downloading. There are cases where the SDTT is transmitted even under an emergency suspension, so it is desirable for receivers to perform a timeout process.

3.3.5 Suspension of receiver built-in information update service (SDTT method)

When there are no download contents at all, transmission of the SDTT and download contents other than the content transmission route specification B shown in Table 3-7 is suspended. However, even in such a case, transmission of the PMT for the engineering service will continue and the PMT_PID will remain in the PAT. Furthermore, in both content transmissions specification A and B, regardless of whether download contents exists or not, when SDTTs for both receiver software and Common Data to All Receivers are not transmitted, sections whose maker_id and model_id are 0xFF00 are transmitted, in order to make full-time

monitoring of the SDTT transmission function easier. Receivers may ignore this information.

3.3.6 Trial transmission of receiver built-in information update service (SDTT method)

For receiver software updates, targets that are not in the market can be specified using the model_id, version_id, etc. to be used for testing.

To test Common Data to All Receivers for the Frequency List and Modification Information, 0xFFF7 can be used. Common Data to All Receivers for actual broadcasting and for test broadcasting before actual broadcasting may be transmitted simultaneously.

3.3.7 Operation of day light savings (SDTT method)

The start_time in SDTT and the time management in related transmission systems shall be based on the "UTC (Coordinated Universal Time) + 9 hours" regardless of whether or not daylight savings time has been introduced.

3.3.8 SDTT method security

3.3.8.1 Receiver software updates

Encryption of downloaded content in relation to receiver software updates for confidentiality shall be up to product design.

For guidelines regarding security, refer to ARIB STD-B21.

3.3.8.2 Common Data to All Receivers updates

Security is not particularly considered.

Chapter 4 Receiver Guidelines for Receiving Download Content

4.1 Receiver Guidelines (SDTT Method)

4.1.1 Provisions for memory

(1) Memory buffers that are capable of keeping pace with the transmission speed should be ensured to receive notification information and download contents.

(2) The capacity of 250 bytes shall be secured as the memory domain for the Common Data to All Receivers and as the memory for the Frequency List and Modification Information (relevant description provided in Commentary 1).

(3) To deal with an accidental failure in the downloading process when receiver software is updated, a memory configuration of either the 2 bank method that enables returning to the status before downloading using two non-volatile memory banks, or the 1 bank + α method which has one memory bank to be updated when downloading and which always has a specific program on the memory should be used.

Please refer to ARIB STD-B53 Part1 12.3.2 for the size and performance of the hardware.

Please refer to ARIB STD-B21 for how to implement flash memory.

4.1.2 Provisions for operation

4.1.2.1 Reservation function

Function to receive Notification Information in accordance with the operation status and to reserve download software to be received.

(1) The content of received Notification Information is analysed, and whether Common Data to All Receivers will be downloaded or receiver software that matches with receiver information will be downloaded is decided and then, whether downloading can be reserved is decided.

(2) When the content of Notification Information is a “mandated downloading” and when the user’s permission has been set up, downloading is reserved while the user is not aware of that.

(3) When the content of Notification Information is an “optional downloading”, the details of downloading options are presented to the user with possible operating methods being provided, and based on the user’s selection, downloading is reserved. However, the influence on program viewing should be kept to a minimum, for example, the user is notified with a small mark on the screen when operation using a menu becomes available and when the viewer operates the menu after recognizing the mark, options

are displayed for the first time.

(4) When the download content transmission path described in the Notification Information is other than broadcasting (specifically, content transmission specification B in Table 3-5), information, such as that receiver built-in information update service is provided in another transmission path e.g. in telecommunication line, is presented. In such a case, the influence on program viewing should be kept to a minimum as in case (3).

(5) Function to make a list of service_id's of engineering services receivable in the NIT as well as to work out all the start times for downloading data that can be received, based on the schedule information and time shift units described in the SDTT and to reserve more than 1 TS, when more than 1 TS is transmitted at different times.

(6) Function to select a TS such as a TS with good reception C/N, and to acquire download content, when the engineering services with the same service_id can be received in more than 1 TS.

Please also refer to ARIB STD-B53 for functions required for receivers.

Please also refer to ARIB STD-B21 for examples of download availability and result judgments.

4.1.2.2 Reception function

Function to receive download content transmitted in a DSM-CC data carousel based on Notification Information and to store the downloaded content in non-volatile memory.

(1) The validity and adequateness of the content of the received downloaded content are evaluated, and the downloaded content is stored in non-volatile memory based on the Notification Information.

(2) Content is downloaded based on Notification Information. When there is schedule information[1], it is downloaded based on the schedule information during stand-by. When there is no schedule information[1], downloading is attempted when the power is turned off. However, reception of download content must not affect program viewing (including reserved viewing). Additionally, when the writing of data into non-volatile memory starts, operation of the receiver by a viewer may be rejected until the writing is completed.

(3) Function to receive download content starting from the TS that will be transmitted at the nearest time based on Notification Information and the current time, when more

[1] "When there is schedule information" is when the number of loops of schedule information in the SDTT is not 0, and "when there is no schedule information" is when the number is 0. However, in either case, the SDTT is of the content transmission path specification A, not for the content transmission path specification B.

than 1 TS is transmitted at different times.

Please also refer to ARIB STD-B53 for functions required for receivers.

Please refer to ARIB STD-B21 for operational scenarios.

4.1.2.3 Execution permission function

Function to provide operational methods to permit execution of downloading functions in the initial setting, and to keep permission information of viewers.

Please also refer to ARIB STD-B53 for functions required for receivers.

4.1.2.4 Status recovery function when abnormalities occur

Function to identify abnormal status such as turning on/off the power, interruption of the process, identification of data abnormalities while receiving download content, and to recover a receiver to normal status.

(1) When an abnormality is identified while receiving download content, the content of the downloaded content that has been stored in memory is made invalid and settings required to re-acquire the download content is ensured.

(2) When the two-bank method, in which two areas are kept for downloaded data when the receiver software is updated to secure safety, the receiver software before downloading should be executed when an abnormality occurs. When the 1 bank+α method that has a specific program area that is not updated and an area to be updated with downloaded data is applied, the specific program is executed and the minimum function should be ensured when an abnormality occurs.

(3) For the downloading process of Common Data to All Receivers, measures should be taken so that data before downloading will not be lost at least when an abnormality occurs.

Please also refer to ARIB STD-B53 for functions required for receivers.

Please refer to ARIB STD-B21 for operational scenarios.

4.1.2.5 Power control function

Power control function to turn on the power of the circuits needed to download using a timer based on Notification Information, download software and turn off the power when downloading is completed. Also, power control function to keep required circuits energized when the power is OFF, download software and turn off the power when downloading is completed. These functions should be operated in the same way as the power control function to acquire EMMs.

Please also refer to ARIB STD-B53 for functions required for receivers.

4.1.2.6 Version display function

Function to display valid versions for receiver makers and broadcasters to refer to when they support viewers over the phone when the downloading function did not work normally.

(1) It is desirable to provide protection such as that valid versions are not displayed through normal operation by the users and are displayed through multiple operations at the time of power on.

(2) It is desirable to display versions using numbers that are difficult for the users to interpret.

Commentary

Commentary 1 Memory Estimates for Frequency List and Modification Information

The memory requirement of receivers for the Frequency List and Modification Information is estimated based on the following conditions. Pay attention, as each of the numerical values shown in (1) conditions are needed for the estimation of the total amount of data per module of data carousel that receivers must secure; this is not the value that stipulates the upper limit of each parameter within the module.

(1) Conditions for estimating the memory amount

No. of region ID: 1

Number of networks: 1

Number of affiliations: 0

Number of changes in the schedule: 1

Number of detailed descriptions per one change in the schedule: 9 (number of segments arranged)

Number of transmitting stations without changes: 0

Number of bytes in the explanation characters: 200

(2) Estimates

The estimate made regarding the amount of data (memory amount) based on the above conditions comes to about 250 bytes.

VOLUME 2

Multimedia Broadcasting

Function Specification for the Receiver

Contents

Chapter 1 General Terms

1.1 Introduction

This volume describes the functional specifications for multimedia broadcasting receivers. Multimedia broadcasters' services assume that the receiver function specifications defined in this volume are the standard implementation. The priorities set by broadcasters regarding receiver functions are shown in the table below (mandatory: A, optional: B, not used: -- and list only: blank).

It depends on the product design of each manufacturer whether or not to install functions as defined, to provide equivalent functions using methods other than that specified, or to install functions beyond those specified in this specification. There are no restrictions in this regard. However, if a problem occurs as the result, broadcasters may not be able to provide solutions.

If the presentation of content is restricted because a manufacturer chooses not to implement a part of these functions as their own product design, applicable receivers must indicate that fact. In this case, the presentation method is not defined.

Broadcasting programs in Japan provided through various services involves various rights such as copyrights and neighboring rights. It is important to establish a comprehensive system to protect these rights for the purposes of providing high quality programs and promoting sound development and progress of broadcasting. From this perspective, it is desired that receivers ensure the uniqueness of broadcasting programs and that programs are presented as intended.

For more information about ensuring uniqueness, see "Description 2 Guarantee of Uniqueness of Broadcasting Programs and Contents" in this volume.

Table 1-1: Priority of receiver functions

Contents of function specification for the receiver	Priority	Comment
4 User interface requirements		
4.1 Time Management	A	
4.2 Initial Settings		
4.2.1 Initial scan	B	
4.2.2 Receiver functions		
4.2.3 User settings		
4.2.3.1 Antenna settings	B	
4.2.3.2 Aspect ratio	B	
4.2.3.3 Settings for whether or not to allow download	B	
4.2.3.4 Selection of whether or not to display superimpose	B	
4.2.3.5 Clear function of user defined information	A	

Contents of function specification for the receiver	Priority	Comment
4.3 Real-time Broadcasting Reception Function	A	
4.3.1 Conditional access service	See Vol. 5.	
4.3.2 ES switching		
4.3.2.1 Default ES	A	
4.3.2.2 Selection of audio elementary streams	A	
4.3.2.3 Selection of caption and superimpose display	A	
4.3.3 Reception of data broadcasting service		
4.3.3.1 Requirements for the receiver	A	
4.3.3.2 Startup and shutdown of data broadcasting service processing	A	
4.3.3.3 Reception of caption and superimpose	A	
4.3.4 Use of HTML	B	
4.4 Storage-based Broadcasting Reception Function	A	
4.4.1 Storage-based function	A	
4.4.1.1User interface for scheduled storage	A	
4.4.1.2 Storage of the linked-content	A	
4.4.1.3 User interface for the linked-content storage	A	
4.4.1.4 Execution of scheduling simultaneous storage	A	
4.4.2 Complementation of storage-based content	A	
4.4.3 Deleting content	A	
4.4.4 Storing metadata	A	
4.5 Content navigation		
4.5.1 EPG	B	
4.5.2 ECG	A	
4.6 Parental Control Function	A	
4.7 Extra Content Notification Function	B	
4.8 Program Channel Selection		
4.8.1 Selecting a channel	A	
4.9 Scheduling Functions		
4.9.1 Scheduling method	A	
4.9.1.1 Manual scheduling	A	
4.9.1.2 Auto scheduling	B	
4.10 Storage-based Broadcasting Content Playback Function	A	
4.11 Storage-based Broadcasting Content Usage and Operation Function	A	
4.12 Preview function	A	
4.13 Contract Information Management Function	A	
4.14 Settlement Function	A	
4.15 Change/Replacement of the Receiver	A	
4.16 Other Functions	B	

Contents of function specification for the receiver	Priority	Comment
4.16.1 Web browser linkage function	B	
4.16.2 Mail linkage function	B	
4.16.3 Recommend function	B	
4.16.4 One-segment linkage function	B	
4.16.5 Service using GPS	B	
4.17 Contents Protection Function	See Vol. 8.	
4.18 Adaptation to Various Television Broadcasting Modes		
4.18.1 Reception of emergency warning broadcasting (under the emergency warning system (EWS))	B	"A" if TMCC signals can be received when power is off (standby).
4.18.2 Reception and playback of 3D broadcasting contents	B	
4.19 Function Restriction in Countries Other than Japan	A	
4.20 Messages	A	
5 Requirements for Hardware and Software		
5.1 Front End	A	
5.2 TS Decoder	A	
5.3 Video Decoding Process and Output	A	
5.4 Audio Decoding Process and Output	A	
5.5 Memory		
5.5.1 RAM	A	
5.5.2 NVRAM		
5.5.2.1 For downloading	A	
5.5.2.2 For data broadcasting	A	
5.5.2.3 For saving storage-based broadcasting content	A	
5.6 Character Font		
5.6.1 Data broadcasting service	A	
5.6.2 EPG/ECG	A	
5.7 Built-in Sounds of the Receiver	A	
5.8 High-speed Digital Interface		
5.8.1 Restriction of partial TS output	(A) *1	"A" if installing a high-speed digital interface.
5.8.2 Specifications for the operation of the PSI/SI table for partial TS output	(A) *1	"A" if installing a high-speed digital interface.
5.8.3 IEEE1394 control command	(A) *1	"A" if a serial interface is installed as a high-speed digital interface.
5.8.4 IP interface specifications	(A) *1	"A" if an IP interface is installed as a high-speed digital interface.
5.9 Removable Media	B	

Contents of function specification for the receiver	Priority	Comment
5.10 Copy Control		
5.10.1 Analog video output	(A)	"A" if installing analog video output function
5.10.2 Digital audio output	(A)	"A" if installing digital audio output function
5.10.3 High-speed digital interface output	(A) *1	"A" if installing a high-speed digital interface.
5.10.4 Digital video output	(A)	"A" if installing digital video outout function
5.10.5 Digital video/audio output	(A)	"A" if installing digital video/audio outout function
5.11 Complementation of the Storage-based Content Function	B	
5.12 Download	B	
5.12.1 Frequency repacking		
5.12.1.1 Overview of downloading the frequency list and modification information		
5.12.1.2 Guidelines for receiving the frequency list and modification information	B	
5.13 Other		
5.13.1 Screen display priority	A	
5.13.2 Processing to take place in power stand-by mode	B	
5.13.3 Reset button	B	
5.13.4 RGB analog terminal	B	
5.13.5 Digital video terminal	B	
5.13.6 Digital video/audio output terminal	B	
ANNEX		
1 Specifications for the Operation of PSI/SI for Partial TS Output	(A) *1	"A" if installing a high-speed digital interface.
2 Specifications for the Operation of Tables	(A) *1	"A" if installing a high-speed digital interface.
3 Specifications for the Operation of the IP Interface	(A) *1	"A" if an IP interface is installed as a high-speed digital interface.

*1) When copy_control_type in the Digital Copy Control Descriptor is "10" in the portable receiver, it should be noted that MPEG_TS output from the high-speed digital interface is prohibited regarding contents other than "copy free" contents. See Vol. 8 of the document for details.

1.2 References

This volume defines the requirements for the user interface, hardware, and software based on ARIB STD-B53 "Receiver for Terrestrial Mobile Multimedia Broadcasting Based on Connected Segment Transmission."

Related standards are listed below.

(1) "Receiver for Terrestrial Mobile Multimedia Broadcasting Based on Connected Segment Transmission (Desirable Specifications)" ARIB STD-B53

(2) "Receiver for Digital Broadcasting (Desirable Specifications)" ARIB STD-B21

(3) "Transmission System for Digital Terrestrial Television Broadcasting" ARIB STD-B31

(4) "Video Coding, Audio Coding and Multiplexing Specifications for Digital Broadcasting" ARIB STD-B32

(5) "Service Information for Digital Broadcasting System" ARIB STD-B10

(6) "Data Coding and Transmission Specification for Digital Broadcasting" ARIB STD-B24

(7) "Conditional Access System Specifications for Digital Broadcasting" ARIB STD-B25

1.3 Terminology

The terms used in this specification is defined as shown below.

ARIB	Association of Radio Industries and Business: The ARIB is an organization that standardizes technologies in relation to the use of radio in Japan, with participation by broadcasters, telecommunications operators, and equipment manufacturers.
Product/merchandise design	Functions and operations that depend on the receiver or merchandise
Superimposition	Subtitling service that is not synchronized with the main video, audio, or data, e.g., news flashes, schedule remarks, time signals, earthquake early warnings, etc.
Caption	A service related to video content among the service that is provided by superimposing textual information over video content distributed in real-time broadcasting
ES	Elementary Stream; corresponds to encoded video, sound, and independent data in a PES packet; one ES is transmitted by a PES packet with an identical stream ID.
EPG	Electronic Program Guide; method used to enable the presentation of program information, such as the timetable, and the selection of programs for real-time broadcasting services
ECG	Electronic Content Guide; the method used to enable the presentation of the content information provided in multimedia broadcasting as well as the selection of content
Channel information	The information that is displayed after selecting a channel; the information indicates a broadcaster name or other data.
One-touch channel selection	A method to directly select a service by pressing a button, which is assigned to a broadcaster or service, on a receiver
Up/down channel selection	A method for switching a channel using the UP/DOWN buttons on the remote controller in the service_id order
GPS	Abbreviation for "Global Positioning System"; a satellite positioning system for searching the current location
CAS	Abbreviation for "Conditional Access System" (limited reception system); a system to restrict viewing services (programmed channels) or events (programs). This system is necessary for broadcasting chargeable programs and free programs with content protection.
Emergency warning system (EWS)	Emergency Warning System: Broadcasts disaster-related information; this system enables a receiver to automatically start the broadcast by sending an activation control signal, etc.
Bluetooth	Close-proximity wireless communication technology in such devices as mobile phones, etc., specified by Bluetooth SIG
Partial TS	The partial transport stream (Partial TS) is defined as a bit stream obtained by removing transport packets that do not relate to specially selected one or more programs from MPEG transport packets.
IEEE1394	IEEE Std 1394-1995 : a serial bus interface suitable for high-speed, real-time transfer, standardized by the IEEE Standard for a High Performance Serial Bus.
DLNA	Digital Living Network Alliance:a group that formulates and promotes guidelines for home network equipment implementation.
MPEG_PS	Program Stream defined by ISO/IEC 13818-1 MPEG-2 Systems

DTCP	Abbreviation for Digital Transmission Content Protection. DTCP is a standard for the content transmission and recording control system using a digital interface for authentication and encryption.
DVI	Digital Visual Interface: an interface standard specified by the DDWG.
HDCP	Abbreviation for the High-bandwidth Digital Content Protection System. The HDCP is a standard for the copyright protection system used when transmitting digital video signals and digital video/audio signals in DVI/HDMI.
HDMI	Abbreviation for the High-Definition Multimedia Interface: a digital interface standard specified by the HDMI founder, is under the control of HDMI Licensing and LLC (Limited Liability Company) for standard and licensing.
VESA	Video Electronics Standards Association: a group that formulates and promotes the standards for displays and display interfaces.
DDWG	Digital Display Working Group: an industry group that promotes the standardization of digital display interfaces.
DIT	Discontinuity Information Table: the table inserted at the transition point where the partial transport stream becomes discontinuous.
SIT	Selection Information Table: the table that lists partial TS stream information and the information regarding services and events transmitted in the stream.
Multimedia broadcasting tuner	A device that has a function to output a baseband signal by selecting/demodulating a received channel or selecting/decoding a desired program from the RF signal; this tuner is also referred to as an “STB” or “IRD.”
Storage-based broadcasting	One of the terrestrial multimedia broadcasting services based on connected segment transmission, which is provided as downloading
Entrusted broadcaster	Same as “certified infrastructural broadcaster"
Entrusting broadcasters	The same meaning as "basic broadcasting station provider".

<Blank Page>

Chapter 2 System Assumption

Regarding the standard and operation of broadcast signals received by the receiver described in this specification, a reference shall be made to "Transmission System for Digital Terrestrial Television Broadcasting" (ARIB STD-B31), "Video Coding, Audio Coding and Multiplexing Specifications for Digital Broadcasting" (ARIB STD-B32) and Vol. 7 of the document. In addition, the receiver shall have a function to identify broadcasting signal based on "Figure 14-1 Identification flow of broadcasting/non-broadcasting" in Chapter 14, Part 1 of ARIB STD-B53 using a system management descriptor that is sent as the PSI as defined in ARIB STD-B10 "Service Information for Digital Broadcasting System."

The receiver shall appropriately receive broadcast signals that should be received using a system management descriptor and properly process an unexpected signal received with a system management descriptor. Note that a different value from the one used in multimedia broadcasting is used in digital terrestrial television broadcasting. For details, see ARIB STD-B10.

<Blank Page>

Chapter 3 Receiver Profiles

This section provides the conceptual definition of each category of receivers.

This specification provides guidelines for product development. However, it depends on the product design of each manufacturer regarding whether or not they installed functions as defined, provided equivalent functions using methods other than that specified, or installed functions beyond those defined in this specification. There are no restrictions in this regard. Should a problem occur as the result, however, broadcasters may not be able to provide solutions.

3.1 Portable Receivers (such as the portable receiver unit)

A portable receiver that can receive data when moving around, such as when people are carrying it with them, etc. Therefore, it is designed to be small in size, light in weight, and high in energy savings. As high priority is given to these features, the supported functions may be limited.

<Blank Page>

Chapter 4 User Interface Requirements

4.1 Time Management

The Japan Standard Time (JST = Coordinated Universal Time (UTC) + 9) information transmitted in the TOT can be used to set current Japan standard time, with an error of ±500 ms (If a leap second adjustment is to be made, a discrepancy of up to about 1.5 sec from JST may occur during a few minutes before and after the leap second insertion.). If a method other than the TOT is available to set Japan standard time, there is no problem with the use of the method. For instance, the following methods that are different from the TOT allow for obtaining the Japanese standard time.

- Time information provided from a mobile phone network
- Time information included in the ECM if applicable; details on the ECM are provided in Section 4.11, Vol. 5 of this standard.

If thedaylight saving takes place, receiving Local Time Offset Descriptor in the TOT and adding an offset value is required to display offset time.

(The time information other than Local Time Offset Descriptor in the TOT shows Japan Standard Time.) If the time obtaining method other than TOT is used, appropriate handling of daylight saving shall also be used.

4.2 Initial Settings

Receiver shall have a screen to enable Initialization described below. All the set values shall be recorded in NVRAM.

4.2.1 Initial scan

It depends on the product design of each manufacturer regarding whether or not a "receivable frequency table" is used or if it is possible to perform the initial scan using a physical channel because it is expected that the reception status of a radio wave changes frequently when a portable or mobile receiver is used. Each manufacturer can also decide a data storage method based on its own product design.

4.2.2 Receiver functions

The re-scan is designed to switch the program service using a button on the receiver.

4.2.3 User settings

4.2.3.1 Antenna settings

It is preferable to perform "bar display," etc., according to the reception status of an antenna level value so that the antenna reception status can be known. Each manufacturer can decide on a display method based on its own product design. One possible example of a reference antenna value is the

value equivalent to twice the C/N value (dB, 5.6 MHz band) of the input carrier. However, the absolute values to be indicated will follow the product design.

4.2.3.2 Aspect ratio

Basically the ratio is set to 4:3 or 16:9 (wide). However, the setting depends on the product design of each manufacturer.

4.2.3.3 Settings for whether or not to allow download

The settings control whether the viewers allow the receiver to automatic update the downloaded frequency list and modification information. For more information, see Vol. 1 of the document. This depends on each manufacturer's product design whether it is set "Yes" or "No" as default for allowing update. Regardless of these settings, all receivers shall be capable of downloading common data including the frequency list and modification information, and reflecting common data downloaded in their settings other than frequency list and modification information.

4.2.3.4 Selection of whether or not to display superimpose

Superimpose display modes include "Selective Display upon Reception" and "Selective Display during Replay" (see "3.3. Operation of caption and superimpose encoding" of Vol. 3 of the document). Since superimpose transmission is not synchronous with video, audio and data (main information) transmission, the receiver shall have the function during initialization to allow the user to specify "Yes" or "No" for superimpose display. The default is "Yes".

4.2.3.5 Clear function of user specific information

The receiver shall have the initialization function to completely clear private information recorded in NVRAM during broadcasting such as those listed below.

- All areas for business operators/TV links that are used in data broadcasting.
  For details, see "3.4.2 Operation of NVRAM" in Vol. 3.

In addition to the clear function above, a function to delete the information in each company's own area and the personal information set by a viewer and to reset to a factory default setting shall be supported. It is preferable to set this function in a relatively deeper hierarchy in the operation menu to prevent the users from accidentally taking incorrect action.

4.3 Real-time Broadcasting Reception Function

Real-time broadcasting is classified into the following broadcasting content.

Table 4-1: Classification of real-time broadcasting

Broadcasting type	Video	Audio	Data broadcasting	Remarks
Audio broadcasting	×	○	*	e.g.: Radio
Video broadcasting	○	○	*	
Independent data broadcasting	×	×	○	e.g.: Weather, news, stock prices

* It is also expected that data broadcasting is applied.

4.3.1 Conditional access service

For details on conditional access service, see Vol. 5.

4.3.2 ES switching

Although the functions specified in this section are mandatory, it depends on each manufacturer's product design as to how to implement these functions.

4.3.2.1 Default ES

Regarding how to select the first ES (default ES) to be decoded from among multiple ES' when a receiver has selected a service unless otherwise specified, see "3.1.4 Default ES" in Vol. 7 of the document.

4.3.2.2 Selection of audio elementary streams

Since multiple audio elementary streams may be carried simultaneously, receivers shall come with a user interface that can extract a target audio stream. It is desired that when switching between channels, receivers can select a default ES and enables cyclic switching between mono/SAP and audio elementary streams using the audio button on the receiver controller. Audio information shall be displayed on a user interface, on which a viewer selects audio from the menu, by conforming to the ProgramInformationTable/ProgramInformation/AVAttributes/AudioAttributes/AudioLanguage information in the EPG metadata.

4.3.2.3 Selection of caption and superimpose display

A maximum of one caption elementary stream and one superimpose elementary stream each may be present. Regarding the caption elementary stream, receivers shall be able to enable the selection of one of two modes: no display of the caption or the display of one of the languages present in the caption elementary stream. The default shall be set to "no display of caption". It is preferable to enable cyclic switching using a button, etc., on a receiver. Language information shall be displayed on a user interface through which a viewer selects the caption ES from the menu based on information described in ProgramInformationTable/ProgramInformation/BasicDescription/CaptionLanguage in the EPG metadata. Whether or not to display superimpose will be selected from the initialization

menu. However, the superimpose is automatically displayed when specified so by the broadcaster. For details on how to use superimposition, see "Description 1" in Vol. 7. The caption and superimpose elementary streams shall be displayed at the same time.

4.3.3 Reception of data broadcasting service

Conformance to the following specifications is required in the reception of data broadcasting service.

4.3.3.1 Requirements for the receiver

(1) Portable receivers

The following functions and specifications described in "Chapter 2 Functions that basic receiver units should be equipped with" in Vol. 3 must be satisfied.

* Section 2.2: Display Function
* Section 2.3: Buttons
* Section 2.4: Memory that Must be Installed in the Receiver

4.3.3.2 Startup and shutdown of data broadcasting service processing

(1) Portable receivers: See "3.1.1.5 Performance of Related Receiver" in Vol. 3 of the document.

4.3.3.3 Reception of caption and superimpose

(1) Portable receivers

See "3.3 Operation of Captions and Superimposed Encoding" in Vol. 3.

4.3.4 Use of HTML

If HTML is used in real-time broadcasting, metadata is sent as described in "portrait_html:URI/OFF" and "landscape_html:URI/OFF" in "2.3 Description of various control information using Keywords [@type="other"]" in Annex 4, Vol. 10 “Metadata Operation Guidelines in Multimedia Broadcasting.” Portable receivers presentvideo, BML, and HTML based on the metadata.

4.4 Storage-based Broadcasting Reception Function

Storage-based broadcasting distributes moving image, music, and still pictures, etc., as content.

“Table 4-2: Storage-based broadcasting content types” provides the detailed types. For each media coding, see "5.2 Operation of Media Encoding" in Vol. 3. A receiver shall provide a playback function (player) that supports various types of content. Although, detailed specification is not defined in this document, it depends on the product design of each manufacturer regarding whether or not a function to play back content, such as ECG, and a function to create a link among various playback functions are supported.

Table 4-2: Storage-based broadcasting content types

Item no.	Content type	File format	Explanation/remarks
1	Video (with audio and captions)	MP4	It is also expected to support radio broadcast with audio content but without video.
2	Markup language	HTML, JavaScript	
3	Image	JPEG, GIF	
4	Others	Not defined	It is expected to support notification from a broadcaster.

4.4.1 Storage-based function

Receivers shall support a function to store the content distributed in a storage-based broadcasting service. For details on how to schedule a function to store the contents, see "4.9 Scheduling Functions" in this volume.

Whether or not content can be stored is determined based on the ECG metadata. For details, see 4.3 in Appendix 1. Contents are stored inside a receiver or in removable media. For the directory structure when storing, see 8.2.3.3 in Vol. 10.

It is expected to support the ECG and the BML of the real-time broadcasting reception function as a user interface for the storage function.

4.4.1.1 User Interface for scheduled storage

It is preferable to provide the following functions while executing the scheduled storage.

- Display the storage execution status.
- Display a message indicating that the storage may be interrupted if a user instructs channel selection.
- Provide a user interface to stop executing the storage.

4.4.1.2 Storage of linked-content

The linked-content is the storage-based broadcasting contents transmitted as an add-on service to the real-time broadcasting using the BML function. The calling from the BML is performed using the linked-content storage indicator function (X_TMM_ecgStoreStart ()).

(1) Before storing, whether or not content can be stored is checked according to the instruction written in Appendix 1, 4.3 "Determination of Whether or not Content Can be Stored." If content cannot be stored, then abort the process without storing content.

(2) If content can be stored, store the storage-based broadcasting content that is being transmitted in the specified ES.

In at least the following case, the function calling is aborted after returning an error return value, and then the receiver does not store content.

・The specified service has not been transmitted.

When calling the function, if there is no detectable error (including the errors mentioned above), the function calling is complete after it returns a successful return value, and then the receiver stores the content.

(3) If the following event occurred before completing the storage, the receiver finishes storing the content and creates a storage end event, showing the status that indicates that storage has failed. Note that the storage does not end if an event is updated or if transition to another BML occurs before the storage completes.

・Updating a data event in a specified ES

・Disappearance of a specified ES

・Calling the storage stop function (X_TMM_ecgStoreTerminate())

(4) If another channel is selected before completing the storage, the receiver suspends the storage.

(5) If the storage is complete, create a storage end event with the status indicating that the content has been stored successfully.

It depends on the product design of each receiver regarding whether a function is supported that simultaneously performs the linked-content storage for multiple storage-based broadcasting contents.

4.4.1.3 User interface for the linked-content storage

It is preferable to provide the following functions when executing the linked-content storage.

・Display of the storage execution status

・Display of the storage progress status

・Display of a message indicating that the storage may be interrupted, when a user instructs channel selection

・Provision of a user interface to stop executing the storage

4.4.1.4 Execution of scheduling simultaneous storage

If content specified to be associated with other content, then a receiver shall have a function that enables scheduling the storage of the specified content when registering the content that is originally scheduled to be stored. Content that is simultaneously stored is specified by specifying "ISDBTMMHowRelatedCS:28" in the HowRelated/@href of the ProgramInformation /BasicDescription/RelatedMaterial element in the ECG metadata and by describing the CRID of the content in MediaLocator/mpeg7:MediaUri.

4.4.2 Complementation of storage-based content

The content, which is stored in a receiver via broadcast waves, may not be stored successfully due to a cause such as the condition of the broadcast waves or the expiration of a distribution time. If this happens, a receiver shall notify users using an icon, etc., showing that the content is incomplete in the stored content list (it is expected that the ECG may also be used). Receivers shall also support a function to compensate for missing data in content via communication based on the user selection, etc. For details on the operation of the compensation of stored content, see Vol. 12 "Compensation of Stored Contents."

4.4.3 Deleting content

A receiver shall have a function that enables users to delete the content stored within it. For details see Section 4.5.2. An agreement shall be obtained from users regarding the deletion of licenses associated with applicable content when deleting.

4.4.4 Storing metadata

The following metadata shall be stored for EPG/ECG creation information and for controlling the transmission of storage-based broadcasting content.

Table 4-3: Metadata type

Metadata type	Purpose
EPG/ECG metadata (EPG)	This metadata is used as information to create the EPG.
EPG/ECG metadata (ECG)	This metadata is used as information to create the ECG.
Transmission control metadata	This metadata is used to store storage-based broadcasting content.

For details on metadata storage control, see "7 Metadata Storage Control" in Vol. 10. For metadata transmission, see "2.2 Transmission method of EPG/ECG metadata" in Vol. 11.

If received metadata is in the BiM format, then it needs to be stored in the NVRAM on a receiver after decompressing it in a receiver. For details on BiM, see "3.3 Metadata transmission coding

scheme" in ARIB STD-B38.

For the timing at which to delete metadata, see "7.3 Deleting metadata" in Vol. 10.

4.5 Content Navigation

4.5.1 EPG

The EPG is a navigation function for content distributed in real-time broadcasting. The EPG is presented by creating it in a receiver based on the EPG metadata. For details on EPG metadata distribution, see "Metadata Operational Standard" in Vol. 10.

In principle, EPG user interface is up to a manufacturer's product design, however, the following table provides the EPG function model list, taking into account viewer convenience and the guidelines for manufacturing EPG metadata. The function model list gives examples of function models and does not restrict the EPG function of a receiver. For EPG creation, see Appendix 1, Guidelines for Creating the EPG/ECG Using EPG/ECG Metadata, also.

Table 4-4: EPG function model list

EPG function	Description
Presentation of distribution program information and schedule	Presents a program name, distribution date and time, and other information For details on the display information, see "Table 4-5 Content information."
Program channel selection	Selects a currently distributed program for viewing
Viewing reservation	Manages the viewing reservation of a program to be distributed and the confirmation/deletion of viewing reservation information
Search function	Displays content by specifying it from the content-related information added to the content (title, actor, theme song, writer, producer, genre, rating, and other related information)
Series reservation function	Manages viewing the reservation/scheduling of the programs in a series and the confirmation/deletion of the reservation information

The following table provides correspondence between the information displayed in the EPG and the EPG metadata.

Table 4-5: Content information

Item no.	Information to be displayed	Information source
1	Program name	For details, see "1.1 EPG" in Annex 3 in Vol. 10 "Metadata Operational Standard."
2	Still picture thumbnail	
3	Broadcast date and time	
4	Program description	
5	Program genre	
6	Program language	
7	Program video, audio mode	
8	Playback time of a program	
9	Price (charged/free)	
10	Package that corresponds to a program	
11	Purchase status	See "5 License Related Information" in Appendix 1.
12	View limit License status (conditions of use)	
13	Viewing reservation status	Information created by a receiver
14	Scheduling status	
15	Recording status	

4.5.2 ECG

The ECG is a navigation function for content distributed in storage-based broadcasting. The ECG is created based on the ECG metadata in a receiver to be presented. For details on the distribution of ECG metadata, see Vol. 10 "Metadata Operational Standard."

In principle, ECG user interface is up to a manufacturer's product design, however, the following table provides the ECG function model list, taking into account viewer convenience and the guidelines for manufacturing ECG metadata. The function model list gives examples of function models and does not restrict the ECG function of a receiver. For ECG creation, see Appendix 1, Guidelines for Creating the EPG/ECG Using EPG/ECG Metadata, also.

Table 4-6: ECG function model list

ECG function	Description
Presentation of distribution content information and schedule	Presents a distribution content name, distribution date and time, and other information For details on the display information, see "Table 4-7 Content information" and "Table 4-8 Group information (series, package)"
Storing distribution content	Selects content that is currently being distributed and manages the confirmation/cancellation of storage and storage information
Storage schedule	Manages the storage schedule of content to be distributed and the confirmation/deletion of the scheduled storage information
Search function	Displays content by specifying it from the content-related information added to the content (title, actor, theme song, writer, producer, genre, rating, and other related information)
Stored content management function	Manages the confirmation/use (playback)/deletion, etc., of the content stored in a receiver
Group unit timer function	Manages the confirmation/deletion, etc., of the scheduled storage of the contents belonging to a group (series/pack) and the scheduled storage information

The following table provides the correspondence between the information displayed in the ECG and the ECG metadata. For details on each ECG metadata descriptor, see Vol. 10 "Metadata Operational Standard."

Table 4-7: Content information

Item no.	Information to be displayed	Information source
1	Content name	For details, see "1.2 ECG" in Annex 3 in Vol. 10 "Metadata Operational Standard." Note: As for the broadcasting period, metadata is used to obtain the entire broadcasting period information, and transmission control metadata is used to obtain the actual broadcasting period information. For details on transmission control metadata, see Section 2.1.5.4 in Vol. 11.
2	Still picture thumbnail	
3	Moving image preview	
4	Broadcasting period	
5	Content description	
6	Content genre	
7	Content language	
8	Content video, audio mode	
9	Content playback time	
10	Price (charged/free)	
11	Package that corresponds to content	
12	Purchase status	See "5 License Related Information" in Appendix 1.
13	Whether or not playback can be performed, view limit,	

	license status (conditions for use)	
14	Schedule status	Information created by a receiver
15	Storage status	
16	Storage capacity	See "4.4.5 Ensuring storage capacity when scheduled to record" in Appendix 1.

Table 4-8: Group information (series/package)

Item no.	Information to be displayed	Information source
1	Group name	For details, see "3.4.5 Use of purchase information element" in Volume 10 "Metadata Operational Standard."
2	Still picture thumbnail	
3	Group description	
4	Group genre	
5	Contents included in a group	
6	Package price (charged/free)	
7	Package purchase status	See "5 License Related Information" in Appendix 1.
8	Scheduling status	Information created by a receiver
9	Storage status	Information created by a receiver

4.6 Parental Control Function

Receivers are required to have a function to input parental control settings and a function to set a password lock to secure the settings. Receivers shall maintain an input parental level (minimum viewing age). A password shall be authenticated when changing a parental level (e.g. minimum viewing age) that was previously set. If the password does not match, the parental level cannot be changed.

The value set to a receiver shall be able to be specified in one-year units from four to 20 years.

For details on the functions restricted by the parental control, see Section 9 in Appendix 1. If parental control is applied when presenting the EPG/ECG, a parental rate set in the EPG/ECG metadata (ParentalGuidance/ParentalRating) is referenced. For details, see "9.2 Overview of Parental Control Function" in Appendix 1.

When receiving a real-time broadcasting service, as a channel selection method that does not use the EPG exists, a parental rate, which is described in the conditional access system descriptor of the PMT's first loop, shall be referenced.

In the factory default setting, both the password and parental level are not set to a receiver.

Unlike a function for inputting parental control settings in a receiver, it is optional to have a function that can control the parental control set by an entrusted broadcaster.

4.7 Extra Content Notification Function

In multimedia broadcasting, extra content (e.g., emergency news), which is not scheduled content, may be broadcasted. The extra content is distributed via real-time broadcasting and storage-based broadcasting. In real-time broadcasting, the extra content is broadcasted by replacing a scheduled content or interrupting the scheduled content. In storage-based broadcasting, the extra content is not only broadcasted by replacing or interrupting the scheduled content, but also broadcasted so as not to interrupt another broadcasting schedule.

It is preferable that receivers have a function to notify users that extra content is being broadcast or scheduled to be broadcast if it detects the extra content. A notification function and notification method vary depending on the product design of each manufacturer.

Extra content shall be identified by referencing the Genre attribute of EPG/ECG metadata (ProgramInformation/BasicDescription/Genre).

4.8 Program Channel Selection

This section describes the basic operation of program channel selection. However, channel selection is not limited to this method. Each manufacturer can decide on a selection method based on their own product design.

Basically, one-touch channel selection using the one-touch button is provided. However, taking into account viewer convenience, a better user interface shall be provided by supporting other functions, such as the up/down channel selection using the channel up/down buttons. The selected channel information (such as broadcaster name) shall be displayed as a banner after selecting a channel. It is preferable to present a program name or other information by obtaining the EIT before presenting audio and visual.

It is also preferable to shorten the time to provide audio and visual by memorizing the last PMT.

The channel selection operation is performed based on the PSI. If receivers can easily recognize the broadcasting status, such as suspension of broadcasting, they shall display a message indicating the status.

For the definition of suspension of broadcasting, see "6.3 How to Handle Broadcasting Suspensions" in Vol. 7.

4.8.1 Selecting a channel

(1) One-touch channel selection

- It is preferable that the service ID used by each broadcaster when the broadcast starts is set to the one-touch button setting and that it can be arbitrarily changed. When selecting a channel, a specified service_id is used.
- When startup, use the last mode for tuning to the lastly selected channel information..

(2) Up/down channel selection using the channel up/down buttons

- Select the adjacent service using the channel up/down buttons on a remote controller.

- Select the upper adjacent service if the up button is pressed. However, if the current value is the maximum value in the service list, then the service with the minimum value is selected.
- Select the lower adjacent service if the down button is pressed. However, if the current value is the minimum value in the service list, then the service with the maximum value is selected.
- When startup, use the last mode for tuning to the lastly selected channel information.

(3) Channel selection using the EPG

- Select a program channel from the EPG.

4.9 Scheduling Functions

Receivers shall have a function to set a reservation to store the content that will be broadcast in storage-based broadcasting and to use the content that will be broadcast in real-time broadcasting. The following table shows the timer types.

Table 4-9: Timer time

Storage-based broadcasting		Real-time broadcasting	
Scheduling storage		Viewing/using reservation	
Manual	Auto	Manual	Auto
○	○	○	○

Manual: Users manually register a scheduling.

Auto: Receivers automatically register a scheduling. However, the information that is used by the system for the registration is the information registered by users beforehand.

4.9.1 Scheduling method

4.9.1.1 Manual scheduling

Manual scheduling is a method that allows users to manually schedule for storing, viewing, or using content (both storage-based broadcasting and real-time broadcasting are supported).

Manual scheduling is basically designed to be operated by users using the ECG/EPG. However, it is also designed to enable scheduling using another function or application in a receiver. Each manufacturer can decide a scheduling method based on their product design.

4.9.1.2 Auto scheduling

Auto scheduling is a method in which a receiver automatically sets a schedule for storing, viewing, or using the content that is selected based on the information registered by a user in advance and the content selected based on recommendations (both storage-based broadcasting and real-time

broadcasting are supported). Each manufacturer can decide the information that is registered by the user beforehand based on the decided product design. However, it is expected that content-related information is used.

Receivers with an auto-scheduling function shall be designed to select the enabling/disabling of the function in the user's settings. It is preferable that the default setting is "enabled."

A storage area may be insufficient when a receiver stores the content using the auto-scheduling function. To deal with this problem, a function that automatically deletes stored content can be supported. If such a function is supported, stored content shall be carefully deleted so as not to delete the content for which a license was purchased by users or the content in use. It is recommended that the method for deleting stored content be asked to users in advance.

4.10 Storage-based Broadcasting Content Playback Function

This function enables displaying the storage-based broadcasting content list stored in a receiver. When displaying the list, the purchase status of a license (whether or not a license has been obtained) shall be indicated by an icon, etc.

This function shall have a license acquisition function (issuance function). For details on how to obtain a license, see Vol. 5 "Conditional Access System Operational Regulations and Receiver Specifications." The license acquisition function includes a function to gain unobtained licenses and a function to extend the license period of an obtained license. If users try to use the content without a license, receivers shall notify the users that the content does not have a license and prompt them to obtain a license.

When playing back the content with a license, they shall be played back based on the license information. For details on the conditional playback service, see Vol. 5.

It is preferable to have the following trick plays to play back video in storage-based broadcasting content.

- Fast forward/rewind (2, 4, 8, 16, 32, or 64 times)
- Double-speed playback with sound
- Slow playback/rewind (1/3 times)
- Frame forward/rewind (stop per frame)
- Chapter forward/rewind
- Specifying a part from which playback starts using the seek bar
- Displaying a bookmark on the seek bar, bookmark forward/rewind
  Enable setting a bookmark per content/scene.

It is preferable to enable the playback of storage-based broadcasting content by supporting the

media schemes. For details, see Annex 13 in Vol. 3. It is preferable that the player, which is used to playback content, has a function to create a media scheme. A created media scheme is used by attaching to a message of the e-mail function or by using another function of the receiver.

4.11 Storage-based Broadcasting Content Usage and Operation Function

A function to display and delete the storage-based broadcasting content information that was saved in a receiver shall be supported.

When deleting content, a notification has to be made to users indicating that the obtained licenses will be deleted while deleting the content.

Content shall be moved or copied based on the license of the applicable content.

4.12 Preview Function

The real-time broadcasting reception function shall support the preview function. During the preview, the status shall be indicated by an icon, etc.

Receivers shall have a function to purchase viewing rights for preview content during the preview, at the completion of the preview, and when viewing is disabled. For details, see Vol. 5.

4.13 Contract Information Management Function

The following contract types exist.

Table 4-10: Contract types

Contract type	Description	Billing type
PPV	Individual contract for real-time broadcasting content	Indibisual billing
PPC	Individual contract for storage-based broadcasting content	
PPP	Total contract supporting multiple real-time broadcasting content and multiple storage-based broadcasting content	Pack unit billing
PPM	Monthly contract supporting real-time broadcasting content and storage-based broadcasting content	Monthly billing

Receivers shall have a function to enable users to check the current contract information and a function to update the current contract. It is also preferable to display any expired contract. Each manufacturer can decide on the retention period of the expired contract information or the amount of retention data based on their own product design.

4.14 Settlement Function

A settlement method is required for the real-time broadcasting reception function and storage-based broadcasting reception function when viewing or using charged content. If a receiver supports charged

content, it shall have a settlement function specified by a broadcaster.

4.15 Change/Replacement of the Receiver

In the storage-based broadcasting reception function, a license file in which the usage condition of each content is described is stored in a receiver. Therefore, it is necessary to move the license file from an old receiver to a new receiver when changing a receiver or replacing it due to a failure. Such processing is performed as an initial setting, such as user registration, and reissuing the licenses, after changing a receiver.

4.16 Other Functions

This depends on the product design of each manufacturer regarding whether or not a linkage function is supported that cooperates with another function (such as a Web browser and e-mail) in a receiver as other functions.

Expected linkage functions are described in the subsequent sections.

4.16.1 Web browser linkage function

As a mobile terminal is targeted for use as a receiver, it is expected that the terminal supports a Web browser function. The detailed specification is not defined in this document. It depends on the product design of each manufacturer regarding whether or not a function is implemented to make a Web browser function that can support multimedia broadcasting.

As provided in the table below, it is expected to have functions that can be operated by clicking a specified link on a Web browser. For the method used to realize the linkage function, see Annex 13 in Vol. 3.

Table 4-11: Expected operation in the linkage function

Type	Expected operation	Remarks
Real-time broadcasting	Viewing a program that is on-air	
	Scheduling the viewing of a program that will be broadcast	
Storage-based broadcasting	Storing the content that are on-air	
	Scheduling the storage of content that will be broadcast	
	Playback of the stored content	*1, *2
	Scheduling the playback of the stored content	*3

*1 Includes playing back from an arbitrary part (time) in the content

*2 Includes playing back after storage if the content is not stored

*3 Includes scheduling the playback after storing or scheduling the storage if the content is not stored

4.16.2 Mail linkage function

As a mobile terminal is targeted for use as a receiver, it is expected that the terminal supports a mail function. The detailed specification is not defined in this document. It depends on the product design of each manufacturer regarding whether or not a function is supported that allows for a mail function to support multimedia broadcasting.

The operation performed from a received e-mail is the same operation as provided in "Table 4-11 Expected operation in the linkage function" in 4.16.1.

4.16.3 Recommend function

The recommend function is a function to recommend content that attracts user concern or interest by selecting from various content dealt in the multimedia broadcasting. The recommend function enables users to, not only easily select, view, or use content, but also notice the presence of new content.

It is designed to support the following two recommendation methods: a method to recommend content using the user attribute information and purchase/usage history and a method to recommend content based on previously set rules. The detailed specification is not defined in this document. It depends on the product design of each manufacturer regarding whether or not the recommend function is implemented.

4.16.3.1 Recommend interface

The recommend function is realized by the communication between a receiver and the recommend server. A recommend interface is used to distribute the information sent from the recommend server to the receiver. The information that will be returned as a response to a request from a receiver is a CRID and its additional information.

For the operation between a receiver and the recommend server, see "Figure 4-1 Sequence diagram of the recommend function."

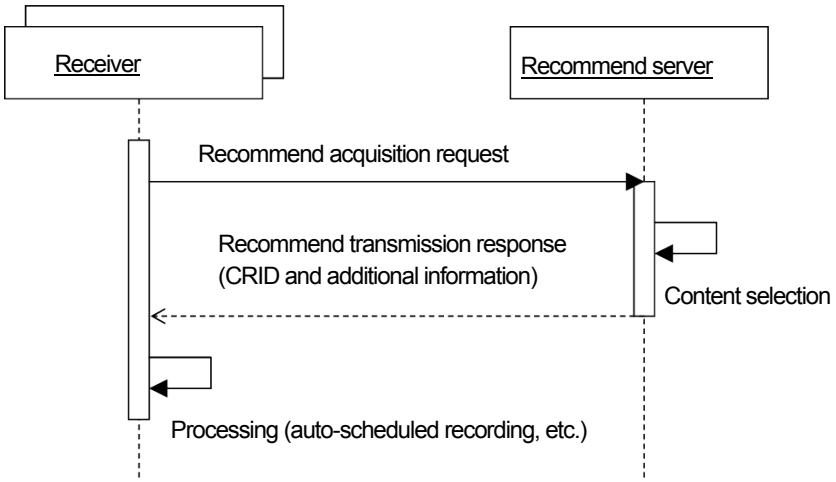


Figure 4-1: Sequence diagram of the recommend function

4.16.3.2 Other recommend interfaces

It is expected that a recommend interface that supports the Web browser linkage function described in 4.16.1 and the mail linkage function described in 4.16.2 will be applicable.

4.16.4 One-segment linkage function

As a mobile terminal is targeted for use as a receiver, it is expected that the terminal supports a one-segment function. As a linkage function between the one-segment function and the multimedia broadcast, it is expected that receivers support a function that performs the same operation as provided in "Table 4-11 Expected operation in the linkage function" in 4.16.1 by clicking a specified link in one-segment data broadcasting. However, the function is not used for the time being.

4.16.5 Service using GPS

(1) Service that stores the content distributed to limited areas, based on GPS information and the positioning information from a base station

(2) Service that distributes stored content particularly to the participants of an event, such as an area event, using the areas in which positioning information and ECG metadata are provided

(3) Activating an application that supports a GPS navigation or a map application by providing the location information included in content-related information or the location information obtained by GPS positioning

It is expected that the relationship between content and GPS information or area information is determined based on the Keyword attribute of the ECG metadata. For details, see 2.3 in Annex 4 in Vol. 10.

4.17 Content Protection Function

See Vol. 8 "Provisions for Contents Protection."

4.18 Adaptation to Various Television Broadcasting Modes

4.18.1 Reception of emergency warning broadcasting (under the emergency warning system (EWS))

See "5.10 Use of Emergency Warning Broadcasting (Emergency Warning System (EWS))" in Vol. 7 of the document.

The following steps (1) to (4) should be taken for receivers.

(1) After the start flag for emergency warning broadcasting of the TMCC signal changes from 0 to 1, receivers start monitoring the Emergency Information Descriptor in the descriptor area 1 in the PMT of the TS received.

(2) If the start_end_flag of the emergency information descriptor is set to 1 and if the area_code corresponds to an area code set in a receiver, receivers shall note that emergency warning broadcasting is on-air using a certain method.

(3) Receivers continuously monitor the PMT while the start flag for emergency warning broadcasting of the TMCC signal remains 1.

(4) When the start flag for emergency warning broadcasting of the TMCC signal has switched to 0 or when the Emergency Information Descriptor in the PMT has been deleted, emergency warning broadcasting comes to an end. However, it should be noted that as described in "5.10.4 Modification of the Information in the Emergency Information Descriptor" in Vol. 7 of the document, there is a possibility that emergency warning broadcasting may be resumed. Receivers must, therefore, continuously stay in the emergency warning broadcasting (EWS) reception mode for at least 90 seconds after the end of emergency warning broadcasting, and then, restore the pre-start state (note that the information regarding EWS reception service is not put into memory as the last service). If channel switching takes place during the reception of emergency warning broadcasting (EWS), the reception of emergency warning broadcasting (EWS) comes to an end. However, when the start flag for emergency warning broadcasting of the TMCC signal switches from 0 to 1, receivers start the reception of emergency warning broadcasting (EWS).

- Receivers shall not process the emergency warning broadcast when start_end_flag of the Emergency Information Descriptor is 0 since it is a test broadcast.
- If receivers are unable to receive TMCC signals when the power is off (stand-by mode), the receivers shall, after the power is switched on, monitor the Emergency Information Descriptor in the descriptor area 1 in the PMT of the TS received and start the reception of emergency warning broadcasting (EWS) when the start flag for emergency warning broadcasting of the TMCC signal is 1.
- If receivers are able to receive TMCC signals when the power is off (stand-by mode), the receivers shall receive abovementioned emergency warning broadcasting (EWS) when the power is off (stand-by mode).
- If the PMT in question becomes no longer present during the reception of emergency warning broadcasting, receivers can end the reception of emergency warning broadcasting.

However, it can be considered that an area code set in a receiver may be different from the actual location. Therefore, it is recommended to perform the activation operation regardless of the area_code checked in (2) above, except for the case where a reception area can be specified using another method. If a receiver has received broadcast waves but another application is displayed on the receiver's screen,

it is also effective to perform a warning operation, such as blinking a light on the receiver, to the viewers as an alternative method to the emergency warning broadcasting (EWS) reception process.

For details on the operation of the receiver and the modification of the emergency information descriptor, see "Figure 4-2 Modification of the emergency information descriptor" and the example of the operation of the receiver.

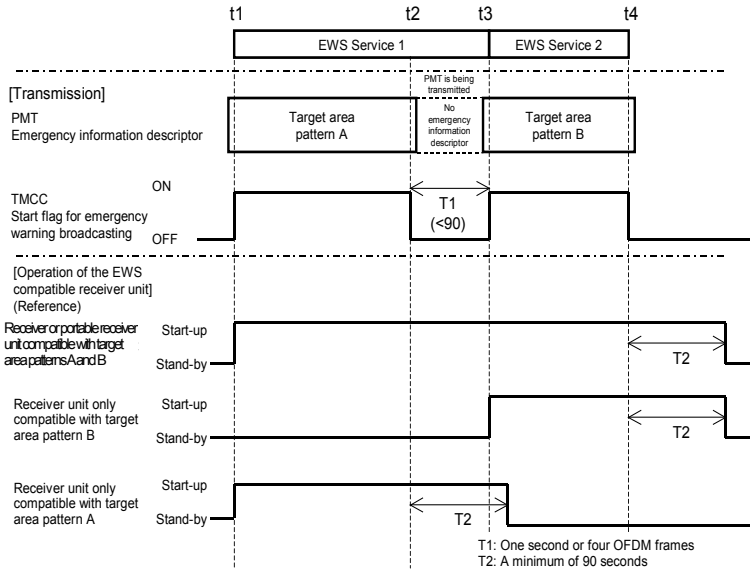


Fig. 4-2: Modification of the emergency information descriptor and an example of operation of the receiver

4.18.2 Reception and playback of broadcasting 3D content

For handling 3D video content, see "5.2.1.4 Handling 3D video" in Vol. 3 and "2.1.1.2.11 Handling 3D video" in Vol. 7. It is optional to support the 3D display of 3D video content in receivers. However, regardless of whether to support the 3D playback, receivers shall be able to receive/play back and display content.

If a receiver supports 3D display, it shall notify that 3D content is displayed. If a receiver does not support 3D display, then it shall notify that it does not support 3D display.

It depends on the product design of each manufacturer regarding whether or not captions and BML can be displayed while displaying 3D video content in real-time broadcasting using a receiver that supports 3D display. While displaying superimpositions, priority can be given to the superimpositions display, and the 3D content display can be stopped.

It also depends on the product design of each manufacturer regarding whether or not a 3D video of storage-based broadcasting content can be played back in the storage-based broadcasting browser in-line using a receiver that supports 3D display.

Each manufacturer can decide on the method to display 3D video content on a receiver, which does not support 3D display, according to their own product design.

4.19 Function restrictions in countries other than Japan

As multimedia broadcasting is a service provided in Japan, if a receiver is used in a country outside Japan, certain functions need to be restricted. Receivers are required to have a method to check that the multimedia broadcasting service is used in a country outside Japan. Hereinafter, when the multimedia broadcasting service is used in a country outside Japan, it is referred to as "overseas mode."

The following table lists whether each function can be used in overseas mode.

Table 4-12: Whether or not a function can be used in overseas mode

Function	Usage/Operation	Available/Unavailable	Remarks
Common	Conclusion of a contract	×	
	Cancellation of a contract	○	
Real-time broadcasting reception function	EPG usage	○	Including viewing reservation and search
	EPG acquisition by packet communication	×	
	Viewing	×	
Storage-based broadcasting reception function	ECG usage	○	Including storage scheduling and search
	ECG acquisition by packet communication	×	
	Content storage	×	
	Content playback	○	
	License acquisition	○	(Note 1)
	Compensation of stored content	×	

Note 1: This function can be used if each content allows using this function in overseas mode. For details, see "6.3.2 Acquisition of a storage-based broadcasting license" in Vol. 5.

If a user selects an item that cannot be used (×), a notification shall be sent to the user indicating that the item cannot be used in a country outside Japan. The same notification shall also be made to users so as not to use an item in which operation is not allowed in the EPG/ECG if such an operation is provided in the EPG/ECG and if the operation is requested in the EPG/ECG.

If packet communication is used to operate an available item above (○), the communication shall be established after informing the users that a communication fee applies and after obtaining their consent in advance.

4.20 Messages

If an error has occurred due to causes shown in the "Cause" column of Table 4-13, the screen shall display an error message. Since the error message displayed regarding the problem shown in Table

4-13 prompts the user to identify the cause as the first action, see each message example in the table (the use of the same messages as those shown in the table is not mandatory). It depends on each manufacturer's product design as to how the error message is displayed. For CAS-related messages, see Vol. 5 of the document.

Table 4-11: Error message list

Cause	An Example of An Error Message Shown on the Screen	Problem Solving Action
A program not broadcast has been selected.	Unable to view the program because the channel is not broadcast.	The channel (program) cannot be viewed.
There is a temporary down in the reception level. Antenna cable or connector connection is poor. The antenna direction is wrong.	The signal level is reduced.	Check the antenna line. Check the antenna direction.
There is a temporary interruption to reception. The antenna cable or connector is broken.	Unable to receive signals.	Check the antenna line. Check the receiver settings.
The broadcasting time has already ended.	Note 1	Check the broadcasting time, for example, using the program guide.
The selected channel cannot be found.	The channel does not exit.	Check the channel, for example, using the program guide.
The functions of the receiver are not compatible with the service on the channel.	The receiver is unable to receive the selected channel.	Select a channel different from the one selected.
(During the display of a data broadcast) The BML document cannot be obtained.	Unable to receive data.	First, select a channel different from the one currently viewed and then, select the data broadcasting channel on which an error has occurred again.
(During the display of a data broadcast) The BML engine of the receiver does not support the version of the BML document obtained.	The receiver is unable to display data.	Select another channel (give up viewing the selected data broadcasting channel.)
(During the display of a data broadcast) During the display of the contents, an execution error has occurred and external reference data cannot be obtained.	Data display has failed.	First, select a channel different from the one currently viewed and then, select the data broadcasting channel on which an error has occurred again.
(During the storage) Content cannot be stored because available capacity in the memory/removable media is lost while storing the content in it.	Storage failed due to the lack of a storage area.	Delete the stored content.

(During the storage) Content cannot be stored because an error, such as a write error, occurred in the removable media in which content is being stored.	Storage failed.	Reinsert the removable media or store the content using another removable media.
(During the storage) Content cannot be stored because an error, such as a write error, occurred in the removable media in which content is being stored.	Content cannot be stored because removable media has not been inserted.	Insert a removable media.
(During the storage) Storing content has failed because the receiver was out of communication range when starting content storage.	Storage failed because the reception has failed.	Check the antenna wires. Check the direction of the antenna.
(During the storage) Content that will be manually stored is not distributed.	Storage failed because the content is not distributed.	Confirm the distribution time in the ECG, etc.
(During the compensation of stored content) Compensation of stored content cannot be performed because the receiver was out of communication range.	Compensation of stored content cannot be performed because the receiver is out of the communication range.	Move to a communication range and perform the compensation of stored content again.
(Compensation of stored content) Compensation cannot be performed due to a communication error while performing the compensation of stored content.	Compensation of stored content cannot be performed due to communication error.	Retry the compensation of stored content.
(During the playback of storage-based broadcasting content) An execution error occurred while playing back the content.	Playback error occurred.	Play back the content again.

Note 1: While broadcasting is suspended, it is expected that a state occurred as described in "Table 4-14 Example of messages when broadcasting is suspended." To display an error message, it shall be displayed by referencing an example in the "message" field in "Table 4-14 Example of messages when broadcasting is suspended."

Table 4-14: Examples of messages when broadcasting is suspended

State	service_type	Message	Remarks
Although the channel list is in the NIT, no corresponding service is in the PAT (or a similar table)	Video service	Suspended	
	Data broadcasting service	Suspended	
Although the description of the corresponding service is in the PAT (or a similar table), the PMT does not exist (or the PMT cannot be obtained).	Digital TV service	Suspended	In the absence of an error*1 (Abnormal state)
	Special video service	Suspended	In the absence of an error*1 (State assumed possible in terms of provisions for operations)
Although physical carriers are received, only null packets are transmitted; PSI and other information do not exist.		Suspended	In the absence of an error*1
Suspension of waves (no physical waves are received)		Unable to receive waves	

*1: The absence of an error refers to when transport_error_indicator on the 9th bit (the 2nd byte MSB) from the header of the TS packet is set to 0 or refers to a similar state.
In the absence of an error, see Table 4-13 (E201, E202).

Chapter 5 Requirements for Hardware and Software

5.1 Front End

- The front end should have at least one RF input.
- Frequency range from 207.5 to 222 MHz bandwidth (V-HIGH) is supported.
- For the tuner specification, see Annex 4.1 to 4.5 of this volume.
- For the front-end signal processing, see Annex 4.6 of this volume.

5.2 TS Decoder

See "4.2.7 Transport Processing" in Chapter 5 of ARIB STD-B21.

5.3 Video Decoding Process and Output

For details on video decoding, see "2.1.1 Video" in Vol. 7 and "5.2.1 Video Encoding" in Vol. 3.

5.4 Audio Decoding Process and Output

In designing an AAC audio decoding circuit, precautions on the implementation of MPEG-2 AAC scheme in Description 4 of this volume shall be taken into account. As for the use of AAC coding tools and maximum instantaneous rate, sufficient discussion and consideration are desired regarding both the transmission operation of broadcasters and the decoding circuit design of manufacturers on the basis of a practical perspective on future progress in the coding technology.

When installing the digital audio output function, compliance to "AAC expansion" in IEC 60958 and IEC 61937 shall be observed. If digital audio will be outputted via the Bluetooth interface, the Bluetooth logo certification shall be acquired in order to guarantee compliance with the Bluetooth standard. For more information on audio output from the digital video/audio output terminal, see "5.13.6 Digital Video/Audio Output Terminal" in this volume.

For details on audio decoding, see "2.1.2 Audio" in Vol. 7 and "5.2.3 Audio Encoding" in Vol. 3.

Note: The formula for down-mixed audio signals was revised in ARIB STD-B53 1.2 Edition.

5.5 Memory

5.5.1 RAM

- If data broadcasting service is supported, see "2.4 Memory that should be installed in receiver units" in Vol. 3.
- To receive notification information and downloaded software, a memory buffer size enough to support the transmission speed should be allocated.
- It depends on each manufacturer's product design for determining the size of RAM to be allocated to resident programs in receivers, the most representative of which are SI/EPG.

- In order to receive storage-based broadcasting content, a memory buffer that supports the proper transmission speed shall be used.

5.5.2 NVRAM

5.5.2.1 For downloading

For details on software memory and memory for downloading data common to receivers, see "4.1.1 Provisions for memory" in Vol. 1.

5.5.2.2 For data broadcasting

(1) Portable receivers

See "2.4.2 NVRAM" in Vol. 3 of the document.

5.5.2.3 For saving storage-based broadcasting content

Each manufacturer can decide the memory capacity for saving storage-based broadcasting content as their own product design. This depends on the product design of a manufacturer regarding whether or not a NVRAM other than the NVRAMs described above is implemented.

5.6 Character Font

The following character fonts shall be available on receivers for data broadcasting service and SI/EPG display.

5.6.1 Data broadcasting service

(1) Portable receiver

See "2.2.5 Fonts" in Vol. 3 of the document.

5.6.2 EPG/ECG

It depends on each manufacturer's product design as to which font type and size to use for EPG display. For character sets, see "2.1 Character Sets" in Vol. 4 and "3.1 Use of Metadata Character Coding" in Vol. 10.

5.7 Built-in Sounds of the Receiver

For details, see "3.2.3.2 Built-in sound" and "3.3.8.3 Operation of built-in sound" in Vol. 3.

5.8 High-speed Digital Interface

To install a high-speed digital interface, see "Chapter 9 Specifications for High-speed Digital Interfaces" and "Appendix 2 High-speed Digital Interface" of ARIB STD-B21.

5.8.1 Restriction of partial TS output

When outputting a multimedia broadcasting program, a component that cannot be descrambled should not be output.

5.8.2 Specifications for the operation of the PSI/SI table for partial TS output

Assuming a connection with equipment that record MPEG streams, there are stipulated specifications for the partial TS to be received by receivers through a high-speed digital interface. For details, see "Specification for the Operation of PSI/SI for Partial TS Output" in Annex 1 of this volume.

5.8.3 IEEE1394 control command

See ARIB STD-B21.

5.8.4 IP interface specifications

When installing an IP interface as a high-speed digital interface, refer to "9.2 IP interface specifications" of ARIB STD-B21 and "High-speed digital interface" in Appendix-2 of ARIB STD-B21.

See "Appendix-3 Specifications for the operation of the IP interface" in this volume for operational details.

With regard to the operation when outputting in MPEG_PS, refer to "Appendix-3 Specifications for the Operation of the IP interface" and the provisions regarding MPEG_PS transmission described in the DLNA Guidelines.

When using wireless LAN, trouble may be caused by the connection of equipment not recognized by the users. When installing an IP interface, therefore, care should be taken to avoid disturbance to the users.

5.9 Removable Media

It is optional whether or not to support removable media. If supported, content shall be recorded based on the content protection method described in Vol. 8.

5.10 Copy Control

5.10.1 Analog video output

- Copy control systems for each analog video output format shall conform to the rules shown in Table 5-1.
- The following control information should be used: service_type (service type identifier information) in the Service List Descriptor, copy_control_type (copy control type information), digital_recoding_control_data (digital copy control information) and APS_control_data (analog output copy control information) in the Digital Copy Control Descriptor. (See "30.3.2.2 Digital Copy Control Descriptor" in Vol. 4 of the document.)

More specifically, for pseudo-sync pulses Macrovision and color stripes, APS_control_data (analog output copy control information) should be used, while for CGMS-A as the video ID signal, digital_recoding_control_data (digital copy control information) shall be used and for APS as the video ID signal, APS_control_data (analog output copy control information) should be used.

Table 5-1: Copy control systems for each analog video output format

Analog video output **1		Macrovision **2	Video ID signal **3
	480i composite	Seudo-sync pulses and color stripes	CGMS-A APS
	480i component	Seudo-sync pulses	CGMS-A APS
	480p component	--	CGMS-A APS
	720p component	--	CGMS-A APS
	1080i component	--	CGMS-A APS
RGB analog output **4		--	--

**1) Including cases where received video signals are format-converted by the receiver and output as analog video signals of various formats.

**2) A contract between broadcasters and Macrovision is required to use Macrovision's copy protection technology. No parameters are transmitted in this case.

**3) The video ID signal refers to signals transmitted in convoluted identification signal waveform carried by VBI, which include information including CGMS-A information and APS information.

**4) For more information on RGB analog output, see 5.13.4 in this volume. For more information on the operation of RGB analog output, see Vol. 8 of the document.

Output control of storage-based broadcasting content is based on the RMPI control information. For details on the RMPI, see "2.4.3 Use of RMPI" in Vol. 8.

5.10.2 Digital audio output

Whether to install the digital audio output function or not is optional. If installing the function, however, the copy control is performed using: service_type (service type identifier) in the Service List Descriptor, copy_control_type (copy control type information) and digital_recoding_control_data (digital copy control information) in the Digital Copy Control Descriptor.

- The output of linear pulse code modulated (PCM) audio shall conform to IEC 60958 and the output of AAC streams shall conform to AAC audio format specified in IEC 61937.

Output control of storage-based broadcasting content is based on the RMPI control information. For details on the RMPI, see "2.4.3 RMPI use" in vol. 8.

5.10.3 High-speed digital interface output

Whether to install a high-speed digital interface or not is optional. If installing the function, however, the following requirements shall be conformed.

- Control shall be performed based on: service type, copy_control_type, digital_recording_control_data and encryption_mode
- There will be cases where the MPEG_TS output is restricted during data service and special data service. In the case of IP interface, the output may be restricted even with MPEG_PS.
- See "30.3.2.2 Digital Copy Control Descriptor" and "30.3.2.4 Content Availability Descriptor" in Vol. 4 of the document.
- Broadcasters apply the DTCP technology to protect copyrights. If DTCP is used, DTCP_descriptor shall be inserted when outputting as MPEG_TS. PCP-UR shall be inserted with UR Mode = 10 and Content Type = 00 when outputting to the IP interface as MPEG_PS. See the DTCP Specifications for details.

To separately output standalone audio streams to a high-speed digital serial interface, the following requirements shall be conformed.

- Compliance to the IEC60958 conformant format (including the IEC61937 conformant format) of IEC61883-6 shall be ensured.
- Channel_status inserted in the IEC60958 conformant format of IEC61883-6 shall be set in accordance with digital_recording_control_data of the Digital Copy Control Descriptor.
- Control shall be performed based on: copy_control_type and digital_recording_control_data.
- See "30.3.2.2 Digital Copy Control Descriptor" in Vol. 4 of the document.
- Broadcasters should use DTCP as the copyright protection scheme.
- There will be cases where the MPEG_TS output is restricted during data service and special data service.
- When applying the DTCP technology, DTCP_descriptor should be inserted. For more information, see DTCP specifications.
- When copy_control_type is set to '11' during data service, the MPEG_TS output in compliance with IEC61883-4 is also possible.

5.10.4 Digital video output

- Protection technology shall be properly applied based on the HDCP specification when supporting the DVI and when outputting: the content restricted to be copied in the digital copy control descriptor, the content specified to be protected in the content usage descriptor, and the storage-based broadcasting content that is restricted to be copied and specified to be protected in the RMPI. For more information, see Vol. 8 of the document.

5.10.5 Digital video/audio output

- Protection technology shall be properly applied based on the HDCP specification when supporting the HDMI and when outputting: the content restricted to be copied in the digital copy control descriptor, the content specified to be protected in the content usage descriptor, and the storage-based broadcasting content that is restricted to be copied and specified to be protected in the RMPI. For more information, see Vol. 8 of the document.
- Protection technology shall be properly applied based on the HDCP specification when supporting wireless digital video audio output and when outputting: the content restricted to be copied in the digital copy control descriptor, the content specified to be protected in the content usage descriptor, and the storage-based broadcasting content. For more information, see Vol. 8 of the document.

5.11 Complementation of the Storage-based Content Function

The storage-based broadcasting reception function shall support the compensation of stored content function that enables the acquisition and compensation of the data through communications that could not be received via broadcast waves. For the usage, see Vol. 12 "Compensation of Stored Content."

As a compensation method, a method to automatically compensate and a method to manually compensate shall be supported.

For auto compensation and manual compensation, see "2.1 Compensation of stored content type" in Vol. 12.

The user operation standard regarding the compensation of stored content is as follows. The standard is applied for manual compensation. For executing auto compensation, only (1) and (4) are applied.

(1) Before compensation

- The display shall provide information showing that missing data exists in the ECG.

(2) When starting compensation

- It is preferable to inform users that packet communication will be performed.

(3) During compensation

- The display shall provide information showing that compensation is being performed.
- A notification shall be made to users when compensation is interrupted or failed.

(4) After compensation is complete

- The display shall provide information showing that missing data was complemented and is usable in the ECG.
- Content can be played back if applicable (i.e., if it has a license and can be played back).

5.12 Download

For more information, see "Chapter 4 Receiver guidelines for Receiving Download Content" in Vol. 1 of the document. Software update using the SDTT method is not mandatory if receivers support a communication function and a software update function via communication.

5.12.1 Frequency repacking

5.12.1.1 Overview of downloading the frequency list and modification information

Information concerning frequency repacking is transmitted to receivers using the combination of the two methods shown below.

(1) SDTT for partial reception hierarchy

Minimum required information (including the target area, change start date and change operation duration) is transmitted to all receivers. The link information to the following engineering service, which can be used with the minimum required information is also transmitted.

(2) Engineering service

More detailed information (including the pre and post modification frequency lists and modification date) is transmitted to perform automatic tuning of frequencies, saving viewer's operations.

The frequency list and modification information can also be used for the following purposes in addition to frequency repacking.

- Tuning of receivers when a new broadcaster has started broadcasting from an existing transmitting station.
- Relief of a station that has suspended broadcasting while scanning.

For more information on the SDTT for the frequency list and modification information, contents data format and transmission, see Vol. 1 of the document.

5.12.1.2 Guidelines for receiving the frequency list and modification information

Shown below are examples of operations required to receive the frequency list and modification information.

(a) Frequency modification has taken place as a result of frequency repacking.

- Receivers periodically monitor the SDTT and check whether module_data_version_code of module_id, which includes the information on the target reception area, has been modified.
- If above information has been modified, receivers obtain relevant frequency information contents.
- Receivers extract obtained contents and check information concerning the channel currently received.

- Receivers, by comparing the values for start_date, the duration and the current date, can determine whether frequency repacking will take place in the future, is currently underway, or has already taken place.
- If frequency repacking will take place in the future, receivers store relevant information in NVRAM, and if necessary, give a notification to users.
- It is considered highly likely that frequency repacking has already taken place when tuning to a desired channel on the conventional frequency is not possible during the repacking period. Further evidences of completed frequency repacking include: that there are descriptions about the transmitting station from which broadcasts are received (estimated from transmitter_id) and that tuning became impossible after change_date.
- If simul_duration includes the effective duration information, tuning to a channel is still possible on the old frequency during the period. However, to be more careful, receivers also try to tune to the same channel on the new frequency. If they can successfully receive the same channel, they perform the same processing as mentioned above on the assumption that frequency repacking has already taken place.

(b) A broadcaster is added to an existing transmitting station, a broadcaster starts broadcasting in an existing transmitting station, and a station that has suspended broadcasting while scanning is relieved.

- Obtain frequency information content. If a network that has not been received is described in the content, all the physical channels described in the content are scanned regularly and checked so that they cannot be received. It is preferable to perform a scan when obtaining an EIT, etc., and not to increase scan frequency more than necessary.
- The success of receiving a channel, which has not been previously received, means a detection of a new broadcasting channel. If automatic tuning is set to "yes" by users, receivers update the reception settings.

Modification information regarding transmission output power (transmission_power) is also transmitted as part of the engineering service. When transmission output power has been modified, there is a possibility that channels which have not been received may become receivable for some viewers. It is, therefore, preferable that receivers prompt viewers to perform a re-scan.

It depends on the product design of a manufacturer regarding whether or not frequency repacking is installed. If installed, the function will only need to have the ability to perform the above mentioned processing (a) and (b). Even if the auto setting is not supported, it is preferable to display the information (such as target area, year/month/date when a change is applied, a period of the operation

for the change) sent using the SDTT for partial reception hierarchy and to present a message that prompts the user to scan again. In either case, it depends on each manufacturer's product design as to what kind of information is included in the message to be displayed and as to how to install the re-scan function and automatic tuning function in response to frequency repacking.

It is expected that a receiver may not be able to download content. If this happens, the frequency repacking described in (a) and (b) may not be processed. It is preferable to display the information sent using the SDTT for partial reception hierarchy (such as target area, year/month/date when a change is applied, a period of the operation for the change) and to display a message prompting the user to rescan because the SDTT for partial reception hierarchy may be able to receive. However, it depends on each manufacturer's product design as to what kind of information is included in the message to be displayed and as to how to install the re-scan function.

5.13 Other

5.13.1 Screen display priority

In case various displays appear in the same area on the screen, the priority is as follows.

(1) Error message

(2) Superimpose

(3) Caption

5.13.2 Processing to take place in power stand-by mode

The following processing takes place in power stand-by mode.

(1) Various scheduling actions by viewers (Including program recording scheduling etc)

(2) Download

(3) Acquisition of the EIT information

5.13.3 Reset button

In anticipation of a situation where the receivers have been rendered inoperable, it is desirable to install a button (e.g. reset button) on the units to reset them.

5.13.4 RGB analog terminal

- Whether to install a VGA terminal or not is optional. If installing the terminal, however, a connector in compliance with the specifications in "4. Physical Connections" in the Enhanced Display Data Channel Standard (Version 1) issued by VESA should be installed and signals should be output in the format specified in "2. VESA Video Signal Definition" in the Video Signal Standard (Version 1, Rev. 1) also issued by VESA.

- Whether to install a DVI terminal with analog output or not is optional. If installing the terminal, however, it is recommended that a connector in compliance with "5. Physical Interconnect Specification" in the Digital Visual Interface DVI (Revision 1.0) issued by DDWG should be installed. On the other hand, signals shall be output in the format specified in "2.5 Analog" in "2. Architectural Requirements" in the Digital Visual Interface DVI (Revision 1.0).
- For more information on the operation of RGB analog output, see Vol. 8 of the document.

5.13.5 Digital video terminal

- Whether to install a DVI terminal or not is optional. If installing the terminal, however, it is recommended that a connector in compliance with "5. Physical Interconnect Specification" in the Digital Visual Interface DVI (Revision 1.0) issued by DDWG should be installed. On the other hand, signals shall be output in the format specified in "2. Architectural Requirements" in the Digital Visual Interface DVI (Revision 1.0).
- For more information on copyright protection technology, see "5.10.4 Digital Video Output" in this volume as well as Vol. 8 of the document.

5.13.6 Digital video/audio output terminal

- Whether to install an HDMI terminal or not is optional.
  If installing the terminal, however, the compliance of the terminal to the High-Definition Multimedia Interface Specification issued by HDMI Licensing, LLC. shall be ensured.
- It is optional to install the wireless digital video/audio output function.
- For more information on copyright protection technology, see "5.10.5 Digital Video/Audio Output" in this volume as well as Vol. 8 of the document.

Annex 1 Specifications for the Operation of PSI/SI for Partial TS Output

1 Type and Identification of Tables

1.1 Definitions of Tables and Descriptors

To partially output the TS for multimedia broadcasting, the tables as shown in Table S1-1 are inserted for use. For more information on each table, see ARIB STD-B10 and STD-B21.

Table S1-1: PSI/SI tables used for multimedia broadcasting

Table Name	General Information on the Function
PAT (Program Association Table)	Specifies the packet identifier of the TS packet that carries the PMT related to the partial TS.
PMT (Program Map Table)	Specifies the packet identifier of the TS packet that carries coding signals for each broadcasting program.
DIT (Discontinuity Information Table)	Indicates the transition point where the service information for the program carried by the partial TS may be discontinuous.
SIT (Selection Information Table)	Indicates the information concerning the program carried by the partial TS.

The PID values for transport stream packets that carry partial TS sections are shown in Table S1-2.

Table S1-2: Assignment of PID to PSI/SI

PID	Table
0x0000	PAT
Specified indirectly by the PAT	PMT
0x001E	DIT
0x001F	SIT

Regarding the values (table_id) assigned for the identification of tables of partial TS sections output by a multimedia broadcasting receiver, the following values are used from among those that are specified in Section 4.1 in Part 1 of ARIB STD-B10. These values are shown in Table S1-3.

Table S1-3: Assignment of table_id

table_id	Table
0x00	PAT
0x02	PMT
0x7E	DIT
0x7F	SIT

1.2 Type and Identification of Descriptors

Regarding descriptors used in the partial TS, the descriptors shown in Table S1-4 are used from among those that are specified in "3.2 Type and Identification of Descriptors" in Part 1 of Vol. 4 of the document. Descriptors not listed in Table S1-4 are not used in the partial TS.

Table S1-4: Descriptors used in the partial TSs

Descriptor Name	General Information on the Function
Stuffing Descriptor	See Vol. 4.
Service Descriptor	Same as above
Short Event Descriptor	Same as above
Stream Identifier Descriptor	Same as above
Digital Copy Control Descriptor	Same as above
Emergency Information Descriptor	Same as above
Data Component Descriptor	Same as above
Video Decode Control Descriptor	Same as above
TS Information Descriptor	Same as above
Content Availability Descriptor	Same as above
Partial Transport Stream Descriptor	Description regarding the partial TS
Network Identification Descriptor	Description regarding network identifiers
Partial TS Time Descriptor	Description regarding partial TS time
Broadcast ID Descriptor	Description regarding various broadcast IDs necessary for the replay of data contents

2 Operation of Common Items to All Tables

2.1 Operation of version_number

(1) Addition of version_number

version_number is independently added to each table.

Regarding the PMT carried by the partial TS, only when the PMT used for a broadcast can be inserted as it is as the PMT for the partial TS, the same version_number as that for the broadcast can be output. On the other hand, regarding version_number of the table when receivers reconfigure the PAT, DIT, SIT and PMT, any value specified by each receiver can be set as the initial value.

(2) Modification timing

When outputting the PMT for a broadcast as it is in the partial TS, the same modification timing as that used for the broadcast can be used. However, if using a table reconfigured by a receiver, modification should take place only when the necessity to modify the table information occurs. For more information on cases where the modification of a stream should be accompanied by DIT insertion, see DIT related information. If the modification of a stream is not accompanied by DIT insertion, it should be ensured that a delay in the modification of a stream should be kept to a minimum.

(3) Modification of the version number

Receivers should basically perform the management of version_number and can output any value as an initial value. The updating and modification of version_number usually involves adding 1. Although the PMT for broadcasting may not be updated correctly, it is still recommended that in such a case, correct version_number is used for output.

Adding 1 is also recommended when making modifications which are accompanied by DIT insertion.

2.2 Operation of current_next_indicator

In all tables, current_next_indicator shall be set to '1' for output. Tables with the current_next_indicator value set to '0' shall not be output. It shall be ensured that when receivers configure tables, the current_next_indicator value shall be always set to '1'.

2.3 Operation of running_status

The SIT shall be so configured that the running_status value is set to 0x0 (meaning "undefined") in all cases for output.

2.4 Operation of Reserved, ISO_reserved and reserved_future_use Items

All bits shall be set to '1' for output.

3 Repetition Rate of Each Table (Cycle of Re-transmission)

The maximum cycle of transmitting each table in the partial TS is shown in Table S1-5.

Table S1-5: Maximum cycle of outputting each table (cycle of repetition)

Table_id	Table	Value of the Recommended Maximum Cycle of Insertion
0x00	PAT	120 ms
0x02	PMT	120 ms
0x7E	DIT	Inserted when it will be necessary
0x7F	SIT	3.6 s

With regard to the interval for inserting each table, it is recommended that each table be inserted in a way to replace the table previously inserted in the broadcast stream. This means, more specifically, that the recommendation is that the PAT replaces the PAT previously transmitted in the broadcast stream in the same location and that the PMT replaces the PMT previously transmitted in the broadcast stream in the same location. The PMT needs to be created in a receiver accordingly because the transmission cycle for the PMT is long in the partial reception hierarchy. It is preferable to insert the SIT in the same cycle as the EIT [p/f] that was sent in broadcasting. The values of maximum cycles indicated in the table include 10% margin over maximum cycles stipulated for broadcasting. However, the exception is the SIT; the maximum cycle of transmitting the SIT specified for BS digital broadcasting receivers was applied instead. For more information on the output of each table in the partial TS, see "9.1. Detailed Rules for Placement of Sections in TS Packets" and "9.2. Details of TS Packet Transmission" in Vol. 4 of the document in order to ensure compliance.

<Blank Page>

Annex 2 Specifications for the Operation of Tables

1 PAT

1.1 Structure and Operation of the PAT

[Application]

The PAT specifies the identifier of the TS packet that carries the PMT and SIT related to the contents inserted in the partial TS.

[Structure]

The structure of the PAT is shown in Table S2-1.

Table S2-1: Structure of the PAT (program association table)

Data Structure	Number of Bits	How the Bit Stream is Expressed
program_association_section () {		
table_id	8	uimsbf
section_syntax_indicator	1	bslbf
'0'	1	bslbf
reserved	2	bslbf
section_length	12	uimsbf
transport_stream_id	16	uimsbf
reserved	2	bslbf
version_number	5	uimsbf
current_next_indicator	1	bslbf
section_number	8	uimsbf
last_section_number	8	uimsbf
for (i = 0;i< N;i++) {		
program_number	16	uimsbf
reserved	3	bslbf
if(program_number == "0x0000"){		
network_PID	13	uimsbf
}		
else{		
program_map_PID	13	uimsbf
}		
}		
CRC_32	32	rpchof
}		

[Semantics of each field]

The semantics of each field shall conform to the provisions in Section 5.2.1 in Part 2 of ARIB STD-B10 and the definitions in Section 2.4.4 of ISO/IEC13818-1.

[Output operation rules]

- Regarding the output repetition rate, provisions in Section 3 in Annex 1 in this volume shall be followed.

- The partial TS shall output one PAT.
- It is desirable, for example, that the PAT is output in a way to replace the PAT previously inserted in the broadcast stream.

Output operation rules of each field are shown in Table S2-2.

Table S2-2: Output operation rules of the PAT

Output operation rules of Each Field	
table_id	Set to "0x00".
section_syntax_indicator	Set to "1".
section_length	This field describes the section length of the PAT. Since the maximum length of the entire section is 1024 bytes, the maximum value for this field is 1021.
transport_stream_id	This field describes transport_stream_id of the original transport stream that contained the PAT in question. The value used by the original broadcast stream that included the partial TS is inserted for use as it is. However, since the PAT is not transmitted to the partial reception layer, the information obtained from the NIT is inserted.
version_number	This field is set to a value, which is incremented by one for every update. This field can be set to any value when the output of the partial TS starts.
current_next_indicator	Set to "1".
section_number	Set to "0x00".
last_section_number	Set to "0x00".
[program_loop]	This field describes the service in the target transport stream but does not specify the maximum number of loops.
program_number	This field describes service_id of the target service, and at the same time without fail, describes only one program_loop with program_number set to "0x0000" (the PID field that follows subsequently describes the PID ["0x001F"] of the SIT) in the PAT. However, since the PAT is not transmitted to the partial reception layer, the information obtained from the PMT is inserted.
network_PID	This field describes the PID ("0x001F") of the SIT.
program_map_PID	This field describes the PID of the PMT of the target service.

2 PMT

2.1 Structure and Operation of the PMT

[Application]

The PMT specifies the PID of the TS packet that transmits every program coding signal output in the partial TS.

[Structure]

The structure of the PMT is shown in Table S2-3.

Table S2-3: Structure of the PMT (program map table)

Data Structure	Number of Bits	How the Bit Stream is Expressed
program_map_section () {		
table_id	8	uimsbf
section_syntax_indicator	1	bslbf
'0'	1	bslbf
reserved	2	bslbf
section_length	12	uimsbf
program_number	16	uimsbf
reserved	2	bslbf
version_number	5	uimsbf
current_next_indicator	1	bslbf
section_number	8	uimsbf
last_section_number	8	uimsbf
reserved	3	bslbf
PCR_PID	13	uimsbf
reserved	4	bslbf
program_info_length	12	uimsbf
for (i = 0;i< N;i++) {		
descriptor()		
}		
for (i = 0;i< N;i++) {		
stream_type	8	uimsbf
reserved	3	bslbf
elementary_PID	13	uimsbf
reserved	4	bslbf
ES_info_length	12	uimsbf
for (j = 0;j< M;j++) {		
descriptor()		
}		
}		
CRC_32	32	rpchof
}		

[Semantics of each field]

The semantics of each field shall conform to the provisions in Section 5.2.3 in Part 2 of ARIB STD-B10 and the definitions in Section 2.4.4 of ISO/IEC13818-1.

[Output operation rules]

- When a stream is contained in the partial TS, the PMT shall be output without fail regarding the service described in the PAT in accordance with the repetition rate specified in Section 3 in Annex 1 of this volume.
- If the service output in the partial TS is configured in the same way as the original broadcast and meets the conditions set out by this specification, the PMT output for the broadcast can be transmitted as it is. When outputting an overwritten PMT, however, it is desirable, for example, that the PMT is output in a way to replace the PMT previously inserted in the broadcast stream. However,

if a multi-section PMT, which is differently from the PMT output in the partial TS, is transmitted in the broadcast stream, the multi-section PMT shall not be output without being modified. A PMT suited for output in the partial TS shall be structured.

- If a broadcasting service is suspended, the PMT may not be output. In this case, the partial TS that only contains the service is not output, and like the broadcasting service, no PAT is output.

Output operation rules of each field are shown in Table S2-4.

Table S2-4: Output operation rules of the PMT

Output operation rules of Each Field	
table_id	Set to "0x02".
section_syntax_indicator	Set to "1".
section_length	This field describes the section length of the PMT. Since the maximum length of the entire section is 1024 bytes, the maximum value for this field is 1021.
program_number	This field describes service_id of the target service. The value output in the broadcast stream is used as it is.
version_number	In normal times, this field is set to a value, which is incremented by one for every version update. However, if an abnormal system condition occurs while the PMT used for the broadcast is being inserted as it is, the value may be incremented by 1 or more. Even so, the value can be output as it is.
current_next_indicator	Set to "1".
section_number	Set to "0x00".
last_section_number	Set to "0x00".
PCR_PID	This field describes the PID of the TS packet that transmits the target PCR packet.
program_info_length	This field describes the loop length of 1st_loop. The maximum loop length is limited by section_length.
[1st(program) loop]	
[2nd(ES)_loop]	The maximum number of loops is 16.
stream_type	This field describes the stream type identifier of the target ES (specified in Vol. 4).
elementary_PID	This field describes the PID of the TS packet that transmits the related ES or payload.
ES_info_length	This field describes the length of the ES Descriptor that follows subsequently.

2.2 Descriptors Inserted in the PMT

The descriptors inserted in the PMT are shown in Table S2-5.

Table S2-5: Descriptors inserted in the PMT

Tag	Descriptor	Propriety of Insertion	loop
0x09	Conditional Access Descriptor	X	--
0xDE	Content Availability Descriptor	#	1

0x42	Stuffing Descriptor	O	D
0x52	Stream Identifier Descriptor	*	2
0xC1	Digital Copy Control Descriptor	#	D
0xC8	Video Decode Control Descriptor	#	2
0xFC	Emergency Information Descriptor	O	1
0xFD	Data Component Descriptor	#	2

: Means that the descriptor shall be inserted in the table without fail when it is inserted in the broadcast stream and relevant components are output.

O : Means that the descriptor should be inserted in the table if doing so is considered necessary.

X : Means that the descriptor shall not be inserted in the table.

* : Means that inserting the descriptor in the table is recommended.

1 : Means that the descriptor is inserted in the 1st loop.

2 : Means that the descriptor is inserted in the 2nd loop.

D : Means that the descriptor can be inserted either in the 1st or 2nd loop.

2.3 Descriptors Inserted in the 1st Loop of the PMT (Program Loop)

The following section provides explanations of descriptors. For more information of the structure of each descriptor as well as the semantics, and basic output operation rules of each field, see Vol. 4 of the document.

(1) Digital Copy Control Descriptor

See "30.3.2.2 Digital Copy Control Descriptor" in Vol. 4 of the document.

[Application]

This descriptor is used to provide control information regarding the digital copy or to describe the maximum transmission rate, or when both apply, for the entire service concerned. The descriptor in the broadcast stream is inserted as it is.

[Output operation rules]

This descriptor shall be inserted exactly as it was described in the original broadcast stream if the ES concerned is the digital copy control target and if the descriptor is described in the broadcast stream.

(2) Emergency Information Descriptor

[Application]

This descriptor means that the service is an emergency warning broadcasting or that an emergency warning signal test is underway.

[Output operation rules]

Although the output of the descriptor is optional, it is desirable to delete this descriptor.

(3) Content Availability Descriptor

[Application]

See "30.3.2.4 Content Availability Descriptor" in Vol. 4 of the document.

[Output operation rules]

This descriptor shall be used to provide control information regarding bound recording and output in combination with the Digital Copy Control Descriptor for the entire service concerned.

2.4 Descriptors Inserted in the 2nd Loop of the PMT (ES Loop)

The following section provides explanations of descriptors. For more information of the structure of each descriptor as well as the semantics, and basic output operation rules of each field, see "30.3.3 Descriptor Inserted in the Second Loop of PMT (ES Loop)" in [Section 3] of Vol. 4 of the document.

(1) Stream Identifier Descriptor

[Application]

This descriptor is used to attach a label to the ES concerned and use the label to reference what is described in the Component Descriptor in the SIT.

[Output operation rules]

If the Component Descriptor and Audio Component Descriptor are found in the service loop of the SIT, links should be created with component_tag to establish a relationship with the ES concerned.

Output operation rules of the Stream Identifier Descriptor are shown in Table S2-6.

Table S2-6: Output operation rules of the stream identifier descriptor

Output operation rules of Each Field	
descriptor_tag	Set to "0x52".
descriptor_length	This field describes the length of the Stream Identifier Descriptor.
component_tag	This field describes the tag value of the component, which is unique in the program concerned and which matches the tag value of the component of the Component Descriptor in the SIT.

(2) Data Component Descriptor

[Application]

This descriptor is used to indicate the data coding scheme for the ES concerned.

[Output operation rules]

This descriptor is always transmitted in the broadcast stream during data broadcasting (including caption and up-to-the-minute superimpose news). When data contents are inserted

in the partial TS for output, it is mandatory that this descriptor be inserted in the SIT.

(3) Digital Copy Control Descriptor

See "30.3.3.3 Digital Copy Control Descriptor" in Vol. 4 of the document.

[Application]

This descriptor is used to provide control information regarding the digital copy or to describe the maximum transmission rate, or when both apply, for the ES concerned. The descriptor in the broadcast stream is inserted as it is.

[Output operation rules]

This descriptor shall be inserted exactly as it was described in the original broadcast stream if the ES concerned is the digital copy control target and if the descriptor is described in the broadcast stream.

(4) Video Decode Control Descriptor

[Application]

This descriptor is used to control the video decode when the video coding scheme changes in same service_id as well as to indicate whether the ES concerned is comprised of MPEG-I framed still pictures.

[Output operation rules]

This descriptor should be inserted when it is described in the original broadcast stream and when the operation specified by this descriptor is requested in the video that forms the partial TS.

[Other special notes]

The Video Decode Control Descriptor contains information for the purpose of performing the following reception control.

1. Seamless video switching during video format change
2. Identification of MPEG-I framed still picture broadcasting

To perform the same control during a replay as that during the reception of a broadcast, it is recommended that information for the purpose of performing the two items above be output. Certain types of receivers are considered to use information particularly regarding 2. above to identify still picture broadcasting. It is therefore mandatory to insert this descriptor when the components of a still picture broadcast are output in the partial TS.

3 DIT (Discontinuity Information Table)

3.1 Structure and Operation of the DIT

[Application]

This DIT describes the points at which the partial TS becomes discontinuous. The DIT is inserted by receivers when a problem encountered by the receivers that output the partial TS causes a discontinuity in the partial TS.

[Structure]

The DIT contains a single section that uses the data structure shown in Table S2-7. This DIT section is transmitted in the transport stream packet with its PID value set to 0x001E. The table identification number is set to 0x7E.

Table S2-7: Structure of the DIT (discontinuity information table)

Data Structure	Number of Bits	How the Bit Stream is Expressed
discontinuity_information_section () {		
table_id	8	uimsbf
section_syntax_indicator	1	bslbf
reserved_future_use	1	bslbf
Reserved	2	bslbf
section_length	12	uimsbf
transition_flag	1	uimsbf
reserved_future_use	7	bslbf
}		

[Semantics of each field]

The semantics of each field shall conform to the provisions in 9.1.8.2(2) 1) in Chapter 9 of ARIB STD-B21 and the definitions in 7.1.1 of ETS 300 468.

[Output operation rules]

- The DIT shall be inserted at the following points: the point at which a discontinuity in the system time base of the stream occurs during the output of the partial TS (more specifically, a discontinuity of the PCR) and the point at which a discontinuity of continuity_counter in the transport packet header occurs in one of the packets that form the partial TS. The discontinuity of continuity_counter occurs when an ES is added or deleted. This kind of discontinuities, which usually occur in broadcast streams, are caused by broadcasters and the insertion of a DIT is not required in these cases.

  The DIT is inserted by receivers when a problem encountered by the receivers that output the partial TS causes a discontinuity in the partial TS.
- Discontinuities that cause the insertion of a DIT in the partial TS take place in the following cases.
  (1) When the output of the partial TS is started or stopped.
  (2) When channel switching, for example, during the output of the partial TS, changes the service in the stream, and this may be accompanied by a discontinuity of the ES/PCR.
  (3) When certain factors cause a change of the ES (such as the increase and decrease in the number of ES), though with no change to the output service, during the output of the partial

TS.

- When switching between streams, a DIT shall be inserted only once between the old and new streams. It is desirable that the DIT be inserted when the output is stopped and that the DIT be not inserted again when the output is started.
- During the output of the partial TS, a DIT shall not be inserted at a transition point (for example, point at which only the SI information changes) in the stream if the conditions above do not apply.
- When inserting a DIT, it shall be ensured that two transport packets in specified format be inserted consecutively. No other transport packet shall be allowed to intervene between these two packets. If a change of the stream is accompanied by the insertion of a DIT, the pre-DIT-insertion stream and table information shall not be present in the post-DIT-insertion stream. It shall also be made sure, whenever possible, that contradictions are not allowed to occur; for example, the post-DIT-insertion stream and table information should not be present in the pre-DIT-insertion stream.

<The first packet>

The adaptation field shall be inserted into the transport packet with its PID set to 0x001E. The value of payload_unit_start_indicator in the transport packet header shall be set to "0". The value of adaptation_field_control shall be set to "10" (adaptation_field only, no payload).

Only discontinuity_indicator in the adaptation field shall be set to "1" and all other flags shall be set to "0". The remaining field of the transport packet shall be designated as stuffing_byte.

<The second packet>

The DIT table defined by this specification shall be inserted. The PID of the transport packet header shall be set to 0x001E and the value of payload_unit_start_indicator shall be set to "1".

Output operation rules of each field are shown in Table S2-8.

Table S2-8: Output operation rules of the DIT

Output operation rules of Each Field	
table_id	Set to "0x7E".
section_syntax_indicator	Set to "0".
section_length	This field describes the section length of the DIT and is set to a fixed value of 0x001.
transition_flag	For more information on the operation of bits, see 9.1.8.2.(2) 1) in ARIB STD-B21.

4 SIT (Selection Information Table)

4.1 Structure and Operation of the SIT

[Application]

The SIT provides the partial TS information as well as a summary of SI information necessary to provide information on the service available in the stream.

[Structure]

The structure of the SIT is shown in Table S2-9.

Table S2-9: Structure of the SIT (selection information table)

Data Structure	Number of Bits	How the Bit Stream is Expressed
selection_information_section () {		
table_id	8	uimsbf
section_syntax_indicator	1	bslbf
reserved_future_use	1	bslbf
ISO_reserved	2	bslbf
section_length	12	uimsbf
reserved_future_use	16	bslbf
ISO_reserved	2	bslbf
version_number	5	uimsbf
current_next_indicator	1	bslbf
section_number	8	uimsbf
last_section_number	8	uimsbf
reserved_future_use	4	bslbf
transmission_info_loop_length	12	uimsbf
for (i = 0;i< N;i++) {		
descriptor()		
}		
for (i = 0;i< N;i++) {		
service_id	16	uimsbf
reserved_future_use	1	bslbf
running_status	3	bslbf
service_loop_length	12	uimsbf
for (j = 0;j< M;j++) {		
descriptor()		
}		
}		
CRC_32	32	rpchof
}		

[Semantics of each field]

The semantics of each field shall conform to the provisions in 9.1.8.2(2) 2) in Chapter 9 of ARIB STD-B21 and the definitions in 7.1.2 of ETS 300 468.

[Output operation rules]

During the output of the partial TS, the SIT should insert, when necessary, the stream information as well as the SI information for the service offered, and shall be output in the repetition rate specified in Section 3 in Annex 2 of the document.

Output operation rules of each field are shown in Table S2-10.

Table S2-10: Output operation rules of the SIT

Output operation rules of Each Field	
table_id	Set to "0x7F".
section_syntax_indicator	Set to "1".
section_length	This field describes the section length of the SIT. Since the maximum length of the entire section is 4096 bytes, the maximum value for this field is 4093.
version_number	In normal times, this field is set to a value, which is incremented by one for every version update.
current_next_indicator	Set to "1".
section_number	Set to 0x00.
last_section_number	Set to 0x00.
transmission_info_loop_length	This field describes the loop length of 1st_loop. The maximum loop length is limited by section_length.
service_id	This field describes service_id of the target program. The value transmitted in the broadcast stream is used as it is.
running_status	This field must be set to 0x0.
service_loop_length	This field describes the loop length of 2nd_loop. The maximum loop length is limited by section_length.

(1) Operation of running status

In the SIT, running status shall be set to 0x0 (meaning "undefined") in all cases.

(2) Updating of the table

The table information includes the description of only the event currently output and no description of other events (service) in the partial TS. When the event changes, the table should be updated and modified if necessary, and when any change to the SIT information takes place, version_number should be incremented.

The updating or modification of the table information is synchronous with the increment of version_number by 1. The value that comes after 0x1F should be 0x00. Any number can be assigned to version_number for transmission when the output of the partial TS is started.

It is also desirable that when a modification is accompanied by the insertion of a DIT, version_number is incremented by 1. However, the continuity of values is not always mandatory.

When inserting the Partial TS Time Descriptor in the 1st loop of the SIT, JST_time can be used to indicate the present time of transmitting the stream. In this case, however, every time a table to update JST_time is inserted, version_number of the SIT should be incremented. However, if the information of other descriptors in the SIT remains unchanged, the other_descriptor_status bit of the Partial TS Time Descriptor should be set to "0" in order to indicate that the information of other descriptors has not changed.

4.2 Descriptors Inserted in the SIT

The descriptors inserted in the SIT are shown in Table S2-11.

Table S2-11: Descriptors inserted in the SIT

Tag	Descriptor	Propriety of Insertion	Loop
0x42	Stuffing Descriptor	O	D
0x48	Service Descriptor	O	2
0x4D	Short Event Descriptor	*	2
0x63	Partial Transport Stream Descriptor	@	1
0x85	Broadcast ID Descriptor	#	2
0xCD	TS Information Descriptor	*	1
0xC2	Network Identification Descriptor	@	1
0xC3	Partial TS Time Descriptor	* *1	D

@: Means that the descriptor shall be inserted in the table without fail.

O: Means that the descriptor should be inserted in the table if doing so is considered necessary.

*: Means that the descriptor should be inserted in the table in principle.

#: Means that the descriptor shall be inserted in the table without fail when it is inserted in the broadcast stream and data are transmittted.

1: Means that the descriptor is inserted in the 1st loop.

2: Means that the descriptor is inserted in the 2nd loop.

D: Means that the descriptor can be inserted either in the 1st or 2nd loop.

*1: Means that the insertion of the Partial TS Time Descriptor in the 1st loop (transmission_info_loop) is optional. In principle, the program starting time and the program duration time of the EIT should be inserted in the 2nd loop (service_loop) of the SIT. If inserting data contents to the partial TS, JST time should be used.

4.3 Descriptors Inserted in the 1st Loop of the SIT (transmission_info Loop)

(1) Partial Transport Stream Descriptor

[Application]

This descriptor describes the partial TS information.

[Structure]

The structure of the Partial Transport Stream Descriptor is shown in Table S2-12.

Table S2-12: Structure of the partial transport stream descriptor

Data Structure	Number of Bits	How the Bit Stream is Expressed
partial_transport_stream_descriptor () {		
descriptor_tag	8	uimsbf
descriptor_length	8	uimsbf
reserved_future_use	2	bslbf
peak_rate	22	uimsbf
reserved_future_use	2	bslbf
minimum_overall_smoothing_rate	22	uimsbf
reserved_future_use	2	bslbf
maximum_overall_smoothing_buffer	14	uimsbf
}		

[Semantics of each field]

The semantics of each field shall conform to the provisions in 9.1.8.3(1) in ARIB STD-B21.

[Output operation rules]

This descriptor shall be inserted in the 1st loop of the SIT without fail.

Output operation rules of each field are shown in Table S2-13.

Table S2-13: Output operation rules of the partial transport stream descriptor

Output operation rules of Each Field	
descriptor_tag	Set to "0x63".
descriptor_length	This field describes the length of the descriptor.
peak_rate	This field describes the maximum momentary partial TS packet rate. The description of the upper limit of the peak rate is required as a minimum. This 22-bit field is coded by the positive integer in 400 bits/second units.
minimum_overall_smoothing_rate	This field describes the minimum overall smoothing buffer leak rate of the entire partial transport packets. This 22-bit field is coded by the positive integer in 400 bits/second units. The field is set to 0x3FFFFF; 0x3FFFFF represents an undefined state.
maximum_overall_smoothing_buffer	This field describes the maximum smoothing buffer size of the entire partial transport packets. This 14-bit field is coded by the positive integer in 1-bit units. The field is set to 0x3FFF; 0x3FFF represents an undefined state.

(2) Network Identification Descriptor

[Application]

This descriptor specifies the source network from which the partial TS was created.

[Structure]

The structure of the Network Identification Descriptor is shown in Table S2-14.

Table S2-14 Structure of the network identification descriptor

Data Structure	Number of Bits	How the Bit Stream is Expressed
network_identification_descriptor () {		
descriptor_tag	8	uimsbf
descriptor_length	8	uimsbf
country_code	24	bslbf
media_type	16	bslbf
network_id	16	uimsbf
for (i = 0;i< N;i++) {		
private_data	8	bslbf
}		
}		

[Semantics of each field]

The semantics of each field shall conform to the provisions in 9.1.8.3(2) in ARIB STD-B21.

[Output operation rules]

The insertion of this descriptor in the SIT is mandatory.

Output operation rules of each field are shown in Table S2-15.

Table S2-15 Output operation rules of the network identification descriptor

	Output operation rules of Each Field
descriptor_tag	Set to "0xC2".
descriptor_length	This field describes the length of the Network Identification Descriptor.
country_code	This field describes the country code of the country where the distribution system which created the partial TS is located. This field is set to 0x4A504E (the code for Japan).
media_type	This field describes the media type of the distribution system which created the partial TS. This field is set to 0x4D4D (the code for multimedia broadcasting).
network_id	This field describes the network identification value of the distribution system which created the partial TS. This field uses the value specified in the NIT.
private_data	This field is left blank.

(3) Partial TS Time Descriptor

[Application]

This descriptor, when inserted in the first loop of the SIT, describes time information regarding the creation of the partial TS.

[Structure]

The structure of the Partial TS Time Descriptor is shown in Table S2-16.

Table S2-16: Structure of the partial TS time descriptor

Data Structure	Number of Bits	How the Bit Stream is Expressed
partialTS_time_descriptor () {		
descriptor_tag	8	uimsbf
descriptor_length	8	uimsbf
event_version_number	8	uimsbf
event_start_time	40	bslbf
duration	24	uimsbf
offset	24	bslbf
reserved	5	bslbf
offset_flag	1	bslbf
other_descriptor_status	1	bslbf
JST_time_flag	1	bslbf
if (JST_time_flag ==1) {		
JST_time	40	bslbf
}		
}		

[Semantics of each field]

The semantics of each field shall conform to the provisions in 9.1.8.3(3) in ARIB STD-B21.

[Output operation rules]

- It is desirable that this descriptor is inserted.
- It is desirable that the JST_time insertion interval does not exceed 10 seconds.
- When inserting this descriptor in the SIT of the partial TS that carries one service, it is desirable to insert this descriptor in only one of the two loops. However, if inserting this descriptor a multiple number of times, the descriptor in the 1st loop shall describe JST_time and the descriptor in the 2nd loop shall describe time information regarding the service, instead of describing JST_time.
- If modifying only the Partial TS Time Descriptor and not modifying other descriptors in the SIT, other_descriptor_status shall be set to 0.
- If this descriptor is inserted in the 1st loop of the SIT, event_version_number, event_start_time and duration in this descriptor are invalid.
- The insertion time error of JST_time shall be within two seconds.
- Output operation rules of each field are shown in Table S2-17.

Table S2-17: Output operation rules of the partial TS time descriptor

Output operation rules of Each Field	
descriptor_tag	Set to "0xC3".
descriptor_length	This field describes the length of the Partial TS Time Descriptor.
event_version_number	If the Partial TS Time Descriptor is inserted in the 1st loop of the SIT, this field is invalid.
event_start_time	If the Partial TS Time Descriptor is inserted in the 1st loop of the SIT, this field is invalid. This field is set to 0xFFFFFFFFFF.
duration	If the Partial TS Time Descriptor is inserted in the 1st loop of the SIT, this field is invalid. This field is set to 0xFFFFFF.
offset	When summer time is applied to JST_time, the offset time must be inserted in this field. The value of local_time_offset in the Local Time Offset Descriptor of the TOT is expressed in 24 bits. More specifically, one byte (0x00) must be added, following 16-bit local_time_offset. If no offset time is applied, this field is set to 0x000000. For more information on the operation of summer time, see Vol. 4 of the document.
offset_flag	This field indicates whether to add or subtract the offset time to or from JST_time. "0": The value of the offset time is added to JST_time. "1": The value of the offset time is subtracted from JST_time.
other_descriptor_status	This field describes the state of descriptors other than the Partial TS Time Descriptor inserted in the SIT. "0": No changes have occurred in other descriptors. "1": Some changes have occurred in other descriptors.
JST_time_flag	This field describes whether JST_time will appear in the field that follows. If this field is set to "1", it means that the JST_time field will appear.
JST_time	This descriptor describes the time of the output of the partial TS. JST_time described in the TOT is used as it is. The time error when updating JST_time must be within two seconds.

(4) TS Information Descriptor

[Application]

This descriptor describes information associated with the source TS from which the partial TS was created.

[Output operation rules]

- It is desirable that this descriptor is inserted.
- If inserting this descriptor, the descriptor transmitted in the broadcast stream shall be used as it is.

4.4 Descriptors Inserted in the 2nd Loop of the SIT (service Loop)

The following section provides explanations of descriptors. For more information of the structure of each descriptor as well as the semantics, and basic output operation rules of each field, see Vol. 4.

(1) Service Descriptor

[Application]

This descriptor describes basic information regarding the service inserted in the partial TS such as

the service name and broadcaster name.

[Output operation rules]

- This descriptor shall be inserted only once for the target service.
- The broadcaster name and service name obtained from the broadcast stream shall be inserted as they are.

(2) Short Event Descriptor

[Application]

This descriptor describes the event name and short character-type information regarding the event.

[Output operation rules]

- In principle, this descriptor should be inserted.
- This descriptor shall be output only once for each event.
- The descriptor obtained from the broadcast stream shall be used as it is.

(3) Partial TS Time Descriptor

[Application]

This descriptor, when inserted in the second loop of the SIT, describes time information regarding the event inserted in the SIT.

[Structure]

For more information on the structure of the Partial TS Time Descriptor, see Table S2-16.

[Semantics of each field]

The semantics of each field should conform to the provisions in 9.1.8.3(3) in ARIB STD-B21.

[Output operation rules]

- In principle, this descriptor should be inserted.
- If inserting JST_time, it is desirable that the JST_time insertion interval does not exceed 10 seconds.
- When inserting this descriptor in the SIT of the partial TS that carries one service, it is desirable to use one descriptor. However, if inserting this descriptor a multiple number of times in the partial TS, the descriptor in the 1st loop shall describe JST_time and the descriptor in the 2nd loop shall describe time information regarding the service, instead of describing JST_time.
- If modifying only the Partial TS Time Descriptor and not modifying other descriptors in the SIT, other_descriptor_status shall be set to 0.
- The insertion time error of JST_time shall be within two seconds.

Output operation rules of each field are shown in Table S2-18.

Table S2-18: Output operation rules of the partial TS time descriptor

Output operation rules of Each Field	
descriptor_tag	Set to "0xC3".
descriptor_length	This field describes the length of the Partial TS Time Descriptor.
event_version_number	If the Partial TS Time Descriptor is inserted in the 2nd loop of the SIT, this field is valid. A change in the event information included in the specified service is accompanied by count increase by 1. Any value can be set as the initial value. It is desirable that the continuity of the version number is guaranteed when a change in the stream with a DIT inserted occurs and when the continuity of the service before and after the inserted DIT is maintained.
event_start_time	If the Partial TS Time Descriptor is inserted in the 2nd loop of the SIT, this field indicates the starting time of the broadcast of the specified event. start_time described in the EIT should be used. To set an invalid value, this field is set to 0xFFFFFFFFFF.
duration	If the Partial TS Time Descriptor is inserted in the 2nd loop of the SIT, this field indicates the duration time of the specified event. duration described in the EIT should be used. To set an invalid value, this field is set to 0xFFFFFF.
offset	When summer time is applied to event_start_time, the offset time is inserted in this field. The value of local_time_offset in the Local Time Offset Descriptor of the TOT is expressed in 24 bits. More specifically, one byte (0x00) is added, following the end of 16-bit local_time_offset. If no offset time is applied, this field is set to 0x000000. For more information on the operation of summer time, see Vol. 4 of the document.
offset_flag	This field indicates whether to add or subtract the offset time to or from event_start_time and JST_time. "0": The value of the offset time is added to event_start_time and JST_time. "1": The value of the offset time is subtracted from event_start_time and JST_time.
other_descriptor_status	This field describes the state of descriptors other than the Partial TS Time Descriptor inserted in the SIT. "0": No changes have occurred in other descriptors. "1": Some changes have occurred in other descriptors.
JST_time_flag	This field describes whether JST_time will appear in the field that follows. If this field is set to "1", it means that the JST_time field will appear.
JST_time	This descriptor describes the time of the output of the partial TS. JST_time described in the TOT is used as it is. The time error when updating JST_time is within two seconds. It is desirable that JST_time is inserted in the 1st loop.

[Other special notes]

When inserting multiple services into the partial TS, same descriptors may be inserted in the 2nd loop multiple numbers of times. If, in this case, JST_time is inserted in some descriptors but not inserted in other descriptors, processing may become complex. It is, therefore, desirable that JST_time is inserted in the 1st loop and only the time information regarding the service, instead of JST_time, is inserted in the 2nd loop. It is also possible that JST_time is inserted in all descriptors. However, in this

case, the same value should be inserted in all these descriptors.

(4) Broadcast ID Descriptor

[Application]

This descriptor describes various broadcast IDs necessary for the replay of data broadcasts.

[Structure]

The structure of the Broadcast ID Descriptor is shown in Table S2-19.

Table S2-19: Structure of the broadcast ID descriptor

Data Structure	bit	Identifier
broadcast_id_descriptor () {		
descriptor_tag	8	uimsbf
descriptor_length	8	uimsbf
original_network_id	16	uimsbf
transport_stream_id	16	uimsbf
event_id	16	uimsbf
broadcaster_id	8	uimsbf
}		

[Semantics of each field]

The semantics of each field should conform to Vol. 4 of the document.

[Output operation rules]

- If output data components (component tag values, 0x40 to 0xDF) via partial TS, this descriptor shall be inserted in the 2nd loop of the SIT.

Output operation rules of each field are shown in Table S2-20.

Table S2-20: Output operation rules of the broadcast ID descriptor

Output operation rules of Each Field	
descriptor_tag	This field must be set to "0x85".
descriptor_length	This field describes the length of the descriptor.
original_network_id	This field describes original_network_id to which the service output in the partial transport stream belongs.
transport_stream_id	This field describes transport_stream_id to which the service output in the partial transport stream belongs.
event_id	This field describes event_id of the event output in the partial transport stream.
broadcaster_id	This field describes the ID of the broadcaster to which the service output in the partial transport stream belongs.

[Supplementary explanation]

- Regarding the multimedia data service, the identification of data transmission modules and video and audio components may use the following fields: original_network_id, transport_stream_id, service_id and event_id. An example is shown below.

arib://<original_network_id>.<transport_stream_id>.<service_id>
[;<event_id>]/component_tag

- Also, regarding the multimedia data service, broadcaster-specific areas are assigned in NVRAM in order to enable the storage and retrieval of information on each broadcaster. Multimedia broadcasting uses broadcaster_id for the identification of each area in NVRAM.
- As in the case of receiving a broadcasting service, in order to normally operate the multimedia data service function described above when playing back content from a partial transport stream, this descriptor needs to be described in the SIT for the partial transport stream and to obtain a desired ID. As a service_id is described in the SIT, each ID excluding the sevice_id is described using a broadcasting ID descriptor.

Annex 3 Specifications for the Operation of the IP interface

1 Packet Format

For partial TS, multimedia broadcasting receivers shall use PCP (Protected Content Packet) described in DTCP Volume 1 Supplement E (DTCP V1SE) instead of the packet format described in "9.2.2.2 Packet format" in Chapter 9 of ARIB STD-B21.

The contents contained in one PCP shall be adjusted so that the time of content reception by the receiver would be in the range between 0.3 and 2 sec. However, the PCP immediately before Nc renewal and the last PCP of HTTP response are exceptions. When the entire content is shorter than 0.3 sec or when the content within the HTTP response is shorter than 0.3 sec for access involving content range specification, the entire content within the HTTP response is dealt as one PCP.

2 Operation of Partial TS Output

Operation of partial TS output from the IP interface shall comply with "Annex 1 Specifications for the operation of PSI/SI for partial TS output" and "Annex 2 Specifications for the operation of tables" in this volume. In the case of content transmission by HTTP, however, DITs shall not be inserted for the start and end of each HTTP response. For instance, no new DIT is inserted when discontinuity with respect to the preceding response is generated by access involving content range specification.

3 Operational Rules on Tuner Description

A tuner container and channel items shall be implemented in a multimedia broadcasting receiver based on the standard defined in the DLNA Networked Device Interoperability Guidelines (DLNA guideline). Described below are the property operational rules on tuner containers and channel items in a multimedia broadcasting receiver.

3.1 Operational Rules on Tuner Description in General

The property shall conform to the standard defined in the DLNA guidelines.

3.2 Operational Rules on Tuner Containers and Channel Items

The operational rules on the properties of tuner containers are listed in Table S3-1. Multimedia broadcasting receivers must incorporate properties of "mandatory" implementation level (denoted by @), in addition to the requirements of the DLNA Guidelines. The implementation of the properties of "optional" implementation level (denoted by O) shall also comply with this table.

Table S3-1: Operational rules on the properties of tuner containers

Property name	Implementation level	Property type	Property description
dc:title	@	String	Indicates the name of each broadcasting system. For multimedia broadcasting receivers, a notation of "multimedia broadcasting (マルチメディア放送)" shall be given in double-width characters.

@: mandatory

O: optional

The property of "dc:" is defined by the Dublin Core Metadata Initiative.

The operational rules on the properties of channel items are listed in Table S3-2. Multimedia broadcasting receivers must incorporate properties of "mandatory" implementation level (denoted by @), in addition to the requirements of the DLNA Guidelines. In such case, the operation shall be performed according to Table S3-2. The implementation of the properties of "optional" implementation level (denoted by O) shall also comply with this table.

Table S3-2: Operational rules on the properties of channel items

Property name	Implementation level	Property type	Property description
dc:title	@	String	Indicates the event name. In principle, the event_name_char information contained in the first loop of the Short Event Descriptor within the EIT event loop is inserted. It is desirable that the inserted information be updated simultaneously with the update of EIT information. When event_name_char is unknown, the same character string of upnp:channelName shall be inserted according to the rule of the DLNA Guidelines.
upnp:genre	@	String	Indicates the genre to which the event belongs. As a rule, the content of a Definition element specified in the GenreCS:[TermID] of EPG/ECG metadata is inserted. Multiple upnp:genres can be applied. If the Genre is not known, an "unknown (不明)" character string shall be inserted.
upnp:channelName	@	String	Indicates the name of service (service channel). In principle, the "char" information contained in the second loop of the Service Descriptor in the SDT service group is inserted. When "char" is unknown, it is desirable to insert the name of the previously set service channel.

upnp:channelNr	@	Integer	Indicates the channel number given by the following equation: upnp:channelNr = (one-touch channel selection number) x 10000 + (three-digit number) x 10 + (branch identifier) [One-touch channel selection number] The values set for the services (service channels) in the receiver are used. The value of 0 shall be used when no one-touch channel selection number is assigned to the service. [Three-digit number] As a rule, the service_id of a service list descriptor included in the NIT second loop (TS loop) is used. For instance, if a service_id is "101" (0x0065), the 3-digit number is "101." If the NIT cannot be acquired, it is desirable to insert the previously set three-digit number.
upnp:scheduledStartTime	O	String	Indicates the start time of the relevant event. In principle, the start time (start_time) of the relevant event in the EIT event loop is inserted after converting it from the MJD+BCD format into the following format, which is the same as the format for dc:date defined in the DLNA Guidelines. When TOT is not present, it is described as follows. CCYY-MM-DDTHH:MM:SS When the time offset value for implementing summer time is set in the TOT, TimeOffset (+/-HH:MM) is added as follows: CCYY-MM-DDTHH:MM:SS+09:00 if summer time is not implemented; CCYY-MM-DDTHH:MM:SS+10:00 if summer time (1 hour) is implemented. This property is not prepared if start_time is unknown value.
upnp:scheduledEndTime	O	String	Indicates the end time of the relevant event. In principle, the end time to be inserted is generated from start_time in the EIT event loop and the length (duration) of the relevant event, and then converted from the BCD format into the following format, which is the same as the format for dc:date defined in the DLNA Guidelines. When TOT is not present, it is described as follows. CCYY-MM-DDTHH:MM:SS When the time offset value for implementing summer time is set in the TOT, TimeOffset (+/-HH:MM) is added as follows: CCYY-MM-DDTHH:MM:SS+09:00 if summer time is not implemented; CCYY-MM-DDTHH:MM:SS+10:00 if summer time (1 hour) is implemented.

			This property is not prepared when upnp:scheduledEndTime cannot be generated because of the variability of start_time or duration.
dc:description	O	String	Indicates program description. In principle, text_char contained in the second loop of the Short Event Descriptor in the EIT event loop is inserted.
res@resolution	O	Pattern string	Indicates the resolution of the content to be output. The resolution is expressed as follows by using the numbers of horizontal and vertical pixels in single-width characters. (Number of horizontal pixels) x (Number of vertical pixels) Example: 1920 x 1080
upnp:rating	O	String	Indicates the age restriction of viewers. In principle, the rating value of the Parental Rating Descriptor is inserted after converting it to a hexadecimal number in the form of 0xXX. For example, "0x10" is inserted when the rating value is 10. This property is used for display only and the use of the property for other purposes is not guaranteed.
upnp:icon	O	URI	This property is not used.
upnp:icon@arib:resolution	O	Pattern string	This property is not used.

@: mandatory

O: optional

The properties of "upnp:" are defined by the UPnP Forum.

3.3 Properties that Should be Included in the Recorded Content

A receiver with a recording function shall have the properties provided in Table S3-3 when outputting recorded content based on "Specifications for the Operation of the IP interface" in Annex 3. Multimedia broadcasting receivers shall satisfy the requirements defined in the DLNA guidelines and implement a property for which the implementation level is "@" (mandatory). A mandatory property shall be implemented based on the description in Table S3-3. A property for which the implementation level is "O" (optional) is also implemented based on the description in Table S3-3.

Table S3-3: Properties of bound recorded contents

Property name	Implementation level	Property type	Property description
dc:title	@	String	This property indicates an event name. As a rule, the PI/BD/Title (@type="main") content of the EPG/ECG metadata is inserted. The event_name_char information, which is included in the first loop of the short format event descriptor in the EIT event loop, can be inserted. If a Title or event_name_char is unknown, the same character string as the upnp:channelName is inserted based on the standard defined in the DLNA guidelines.
upnp:genre	@	String	This property indicates a genre to which an event, which is expressed as dc:title, belongs. As a rule, the content of a Definition element specified in the GenreCS:[TermID] of the PI/BD/Genre/@href in EPG/ECG metadata is inserted. Multiple upnp:genres can be applied. If the content_nibble_level_1 of the Genre is not known, a character string of "unknown" shall be inserted.
upnp:channelName	@	String	This property indicates a service (programmed channel) name. As a rule, the char information included in the second loop of the service descriptor included in the SDT service loop is inserted. If the char is unknown, it is preferable to insert a programmed channel name that was previously set.
upnp:channelNr	@	Integer	This property indicates a channel number expressed as the following formula. A channel number is expressed in the following format: upnp:channelNr = one-touch channel number x 10000 + 3-digin number x 10 [one-touch channel number]. A value set for a service (programmed channel) in a receiver is used. "0" is used if a one-touch channel number is not assigned to the service. [3-digit number] As a rule, the service_id of a service list descriptor included in the NIT second loop (TS loop) is used. For instance, when the service_id is "101" (0x0065), the 3-digit number is "101." If the NIT cannot be obtained, it is preferable to insert a 3-digit number that was previously set.
dc:date	@	String	This property indicates the time at which to start recording the applicable content. The

			same formula as the upnp:scheduledStartTime of a channel item is used. For instance, if recording starts at 7:00 a.m. on June 17, 2005, the time information is described as 2005-06-17T07:00:00.
res@duration	@*1	String	Indicates the length of time spent for recording applicable content
upnp:scheduledStartTime	O	String	This property indicates the start time of a target event. As a rule, this property is inserted by converting the start_time, which represents the start time of a target event in an event loop in the EIT, from the MJD+BCD format, to the following format, which is the same as the dc:date defined in the DLNA guidelines. If the TOT does not exist, then describe as follows. CCYY-MM-DDTHH:MM:SS This property is described by adding TimeOffset (±HH:MM) as described below if a time off set value is set to the TOT when the summer time period begins because summer time is applied. CCYY-MM-DDTHH:MM:SS+09:00 → Without summer time CCYY-MM-DDTHH:MM:SS+10:00 → With summer time (1 hour) The content of PLT/BE/PublishedStartTime in ECG/EPG metadata can be described. This property is not used for storage-based broadcasting content or if an unknown value is set to the start_time.
upnp:scheduledEndTime	O	String	This property indicates the end time of a target event. As a rule, an end time is created based on the start_time in the event_loop in the EIT and the duration that represents the length of time required for a target event. Then, the end time is inserted after converting it from the BCD format to the same format as the dc:date defined in the DLNA guidelines. If the TOT does not exist, then describe as follows. CCYY-MM-DDTHH:MM:SS This property is described by adding TimeOffset (±HH:MM) as described below if a time off set value is set to the TOT when the summer time period begins because summer time is applied. CCYY-MM-DDTHH:MM:SS+09:00 →Without summer time CCYY-MM-DDTHH:MM:SS+10:00

			→With summer time (1 hour) The content of the PLT/BE/PublishedEndTime in ECG/EPG metadata can be described. This property is not used for storage-based broadcasting content or if the upnp:scheduledEndTime cannot be created because a start_time or duration is not set.
dc:description	O	String	This property indicates a program description. As a rule, the content of the PI/BD/Synopsis in ECG/EPG metadata is inserted. A text_char, which is included in the second loop of a short format event descriptor in the EIT event loop, can be inserted.
res@resolution	O	Pattern string	This property indicates the resolution of the content to be output. A resolution is expressed in a one-byte character string as follows using horizontal pixels and vertical pixels. (horizontal pixels) x (vertical pixels) e.g.: 1920 x 1080
upnp:rating	O	String	This property indicates viewing age restriction. As a rule, a recommended minimum viewing age, which is specified by the ARIBParentalRatingeCS:[TermID] of the PIT/PI/BD/parentalGuidance/mpeg7:ParentalRating/@href in EPG/ECG metadata, is inserted after conversion to a hexadecimal string. The rating value of a parental rate descriptor can be inserted after conversion to a hexadecimal string, 0xXX. For instance, if a rating value is "10," "0x10" is inserted. This property is used for display. Operation is not guaranteed if the property is used for other purposes.
upnp:icon	O	URI	This property is not used.
upnp:icon@arib:resolution	O	Pattern string	This property is not used.

@: mandatory

O: optional

*1: This attribute is not necessary if the length of time is unknown because content are being recorded or if an item does not have res.

4 Control of Content Selection

4.1 protocolInfo and MIME-Type

In stream output by HTTP from the digital media server (DMS) of a multiple broadcasting receiver, the control of content selection is expressed as follows by using protocolInfo defined in the Media

Management of the DLNA Guidelines and the recommended MIME-Type described in the DTCP V1SE.

protocolInfo consists of four fields

<protocol>":"<network>":"<contentFormat>":"<additionalInfo>

First field <protocol>: protocol used for the output of content.

Second field <network>: the definition of this field depends on the output protocol. Asterisk "*" is described in the case of HTTP.

Third field <contentFormat>: the definition of this field depends on the output protocol. The format of the content itself is indicated in the case of HTTP.

Fourth field <additionalInfo>: additional information is given.

For example, when the stream output protocol is HTTP and MPEG content in TS format with timestamp complying with "Annex 3 1 Packet Format" is transmitted upon protecting it using DTCP-IP, it shall be expressed in protocolInfo as shown below:

http-get:*:application/x-dtcp1;DTCP1HOST=(host);DTCP1PORT=(port);CONTENTFORMAT="video/vnd.dlna.mpeg-tts":DLNA.ORG_PN=DTCP_AVC_TS_MP_SD_HEAAC_L2_T;DLNA.ORG_FLAGS=01110000000000000000000000000000

Each field shall be inserted as follows according to the protocolInfo specifications given in the Media Management of the DLNA Guidelines.

First field: "http-get," which indicates that the output protocol is HTTP, is inserted.

Second field: asterisk "*" is inserted.

Third field: MIME-Type, explained below, is inserted.

Fourth field: If a stream format protected using DTCP-IP will be described in DLNA.ORG_PN, it shall be described in the format defined in "Link Protection Guideline" in Volume 3 of the DLNA Guidelines. It shall also be possible to insert other parameters defined in the DLNA Guidelines and/or manufacturer-specific parameters in the fourth field. See the DLNA Guidelines for rules on inserting multiple parameters.

In the third field, the recommended MIME-Type described in the DTCP V1SE shall be used as MIME-Type of the content to be transmitted with DTCP-IP protection; and the MIME-Type specified for the TS format with timestamp defined in "MPEG-2 MIME-Type Definition" in the DLNA

Guidelines shall be used as CONTENTFORMAT as expressed below:

application/x-dtcp1;DTCP1HOST=(host);DTCP1PORT=(port);CONTENTFORMAT="video/vnd.dlna.mpeg-tts"

Here, "(host)" represents the address of the host that performs AKE, and "(port)" represents the port of the host that performs AKE.

Although "DTCP1HOST=(host)" and "DTCP1PORT=(port)" are optional, they shall always be inserted if the content needs to be protected by DTCP-IP from the beginning or midstream of the content.

The MIME-Type of the content transmitted as the payload of PCP is specified for CONTENTFORMAT. When transmitting the TS format with time stamp described in "8.1.4 Transmission of MPEG video/audio in a time-stamped TS format" in Vol. 2 of ARIB STD-B24, "video/vnd.dlna-mpeg-tts" shall be specified. In the third field of protocolInfo of "Out" in CMS:GetprotocolInfo(), "DTCP1HOST=(host)" and "DTCP1PORT=(port)" are optional because the address and port of the host that performs AKE may vary with contents.

When describing the above-defined protocolInfo as an attribute of "res" (res@protocolInfo), the problem of double quoting arises. To evade this problem, either the outside quotation marks are expressed as single quotes or the inside quotation marks are escaped by using """ according to the XML syntax. The description of res@protocolInfo then becomes as follows.

protocolInfo='http-get:*:application/x-dtcp1;DTCP1HOST=(host);DTCP1PORT=(port);CONTENTFORMAT="video/vnd.dlna.mpeg-tts": DLNA.ORG_PN=DTCP_AVC_TS_MP_SD_HEAAC_L2_T '

or

protocolInfo="http-get:*:application/x-dtcp1;DTCP1HOST=(host);DTCP1PORT=(port);CONTENTFORMAT="Video/vnd.dlna.mpeg-tts":
DLNA.ORG_PN=DTCP_AVC_TS_MP_SD_HEAAC_L2_T "

The following shows an example of a CDS (ContentDirectory service) item for when transmitting partial TS:

```
<item id=”ID1” restricted=”1” parented=”0”>
    <dc:title>Title</dc:title>
    <upnp:class>object.item.videoItem</upnp:class>
    <upnp:genre>Genre</upnp:genre>
    <upnp:channelName>Channnel Name</upnp:channelName>
```

```
<upnp:channelNr>10000</upnp:channelNr>
<dc:date>2008-09-18T18:00:00</dc:date>
<res protocolInfo="http-get:*:application/x-dtcp1;DTCP1HOST=192.168.0.1;
DTCP1PORT=30000;CONTENTFORMAT=&quot;video/vnd.dlna.mpeg-tts&quot;:
DLNA.ORG_PN=DTCP_AVC_TS_MP_SD_HEAAC_L2_T;DLNA.ORG_FLAGS=011100000000
00000000000000000000">URL1</res>
</item>
```

The following shows another example for the case in which partial TS can be transmitted without protecting it using DTCP-IP (i.e. no encryption):

```
<item id="ID1" restricted="1" parented="0">
<dc:title>Title</dc:title>
<upnp:class>object.item.videoItem</upnp:class>
<upnp:genre>Genre</upnp:genre>
<upnp:channelName>Channnel Namer</upnp:channelName>
<upnp:channelNr>10000</upnp:channelNr>
<dc:date>2008-09-18T18:00:00</dc:date>
<res protocolInfo="http-get:*:video/vnd.dlna.mpeg-tts:DLNA.ORG_PN=
AVC_TS_MP_SD_HEAAC_L2_T;DLNA.ORG_FLAGS=01100000000000000000000000000000">
URL1</res>
</item>
```

4.2 URL of Content Protected by DTCP-IP

For the URL of content protected by DTCP-IP and described as a "res" value in the tuner description of a multiple broadcasting receiver, use of the recommended URI described in the DTCP V1SE is not mandatory.

4.3 Content-Type Header Field of HTTP Header

In an HTTP request or HTTP response related to content protected by DTCP-IP, the address and port of the host that performs AKE shall be included in the Content-Type header field inserted in the HTTP header, as specified in the DTCP V1SE. This MIME-Type is the same as the one given in 4.1 in this volume.

Content-Type header field shall be as follows for partial TS output:

Content-Type:application/x-dtcp1;DTCP1HOST=(host);DTCP1PORT=(port);CONTENTFORMAT=

"video/vnd.dlna.mpeg-tts"

Content-Type header field shall be as follows for MPEG_PS output:

Content-Type:application/x-dtcp1;DTCP1HOST=(host);DTCP1PORT=(port);CONTENTFORMAT="video/mpeg"

4.4 Access Involving Content Range Specification

When a multimedia broadcasting receiver supports range-specified access by TimeSeekRange.dlna.org described in the DLNA Guidelines, the relevant provisions in the Guidelines shall be complied with.

When supporting range-specified access by Range.dtcp.com described in the DTCP V1SE, the following requirements shall be satisfied.

If a multimedia broadcasting receiver supports range-specified access by Range.dtcp.com and if DLNA.ORG_PN is described in the fourth field of res@protocolInfo of the corresponding content, either cleartextbyteseek-full or lop-cleartextbytes flag of DLNA.ORG_FLAGS shall be set according to the provisions of the DLNA Guidelines.

The format of Range.dtcp.com is as follows:

- Range.dtcp.com = "Range.dtcp.com" ":" range-specifier
- range-specifier = byte-range-specifier
- byte-range-specifier = bytes-unit "=" byte-range-set
- bytes-unit = "bytes"
- byte-range-set = byte-range-spec
- byte-range-spec = first-byte-pos "-" [last-byte-pos]
- first-byte-pos = 1*DIGIT
- last-byte-pos = 1*DIGIT

The "first-byte-pos" item indicates the position of the first byte in the unencrypted content, and "last-byte-pos" indicates the position of the last byte in the unencrypted content.

Examples of Range.dtcp.com description are given below:

- Range.dtcp.com: bytes=1539686400-
- Range.dtcp.com: bytes=1539686400-1541710655

When the range requested by Range.dtcp.com for content of the TS format with time stamp in a multimedia broadcasting receiver does not match with the 192-byte-unit packet boundary of the TS format with time stamp, the response range is extended as follows to make it match with the

192-byte-unit packet boundary.

- When the start position of the requested range does not match with the start of the packet of the TS format with time stamp, the start position of the response range shall be changed to the start position of the packet of the TS format with time stamp so that the changed range includes the requested start position.
- When the end position of the requested range does not match with the end of the packet of the TS format with time stamp, the end position of the response range shall be changed to the end position of the packet of the TS format with time stamp so that the changed range includes the requested end position.

The response range to the range request using Range.dtcp.com for content other than TS format with time stamp shall follow Volume 3 of the DLNA Guidelines.

With regard to responding to the Range.dtcp.com request, refer to Table 8-29 for HTTP response codes in different situations described in Table S3-4.

Table S3-4: HTTP response codes for Range.dtcp.com request

Situation	Response code
When responding normally to the Range.dtcp.com request	200 (OK); 206 (Partial Content) shall not be used
When the range requested by Range.dtcp.com is invalid (e.g., the start position of the requested range exceeds the end of the content.)	416 (Requested Range Not Satisfiable)
When the Range.dtcp.com request is grammatically invalid	400 (Bad Request)
When the Range.dtcp.com request is not supported for the relevant content.	406 (Not Acceptable)

When responding with the response code of "200 (OK)," insert the Content-Range.dtcp.com header field described in the DTCP V1SE in the HTTP response header.

The format of Content-Range.dtcp.com is as follows:

- Content-Range.dtcp.com = "Content-Range.dtcp.com" ":" content-range-spec
- content-range-spec = byte-content-range-spec
- byte-content-range-spec = bytes-unit SP byte-range-resp-spec "/" (instance-length | "*")
- bytes-unit = "bytes"
- byte-range-resp-spec = first-byte-pos "-" last-byte-pos
- first-byte-pos = 1*DIGIT
- last-byte-pos = 1*DIGIT
- instance-length = 1*DIGIT

The "first-byte-pos" item indicates the position of the first byte in the unencrypted content, and "last-byte-pos" indicates the position of the last byte in the unencrypted content.

The "instance-length" item indicates the entire size of the unencrypted content. When the entire size is difficult to calculate, "*" may be used as allowed in Content-Range.

An example of Content-Range.dtcp.com description is given below:

- Content-Range.dtcp.com: bytes 1539686400-1541710655/9238118400

When accessing the content in the TS format with a timestamp in a multimedia broadcasting receiver using the Range.dtcp.com request, it is preferable to specify a range that matches the 192-byte-unit packet boundary in the TS format with a time stamp.

<Blank Page>

Annex 4 Specifications of Receiver

1 Input

- Input impedence: 75Ω
- Frequency range: VHF-HIGH 207.5 MHz to 222 MHz
- Center frequency

  13-segment type: 210+3/7 MHz, 216 MHz

  One-segment type: 219 MHz, 219+3/7 MHz, 219+6/7 MHz,
  220+2/7 MHz, 220+5/7 MHz, 221+1/7 MHz, 221+4/7 MHz

2 Center Frequency

Not stipulated

3 Synchronization Frequency Range of the Received Signal

- Synchronization frequency range of the received signal: ±30 kHz or wider

4 Synchronization Clock Range of the Received Signal

- Synchronization clock range of the received signal: ±20 ppm or wider

5 Characteristics of the Tuner

The tuner that receives multimedia broadcasting shall satisfy the following functions.

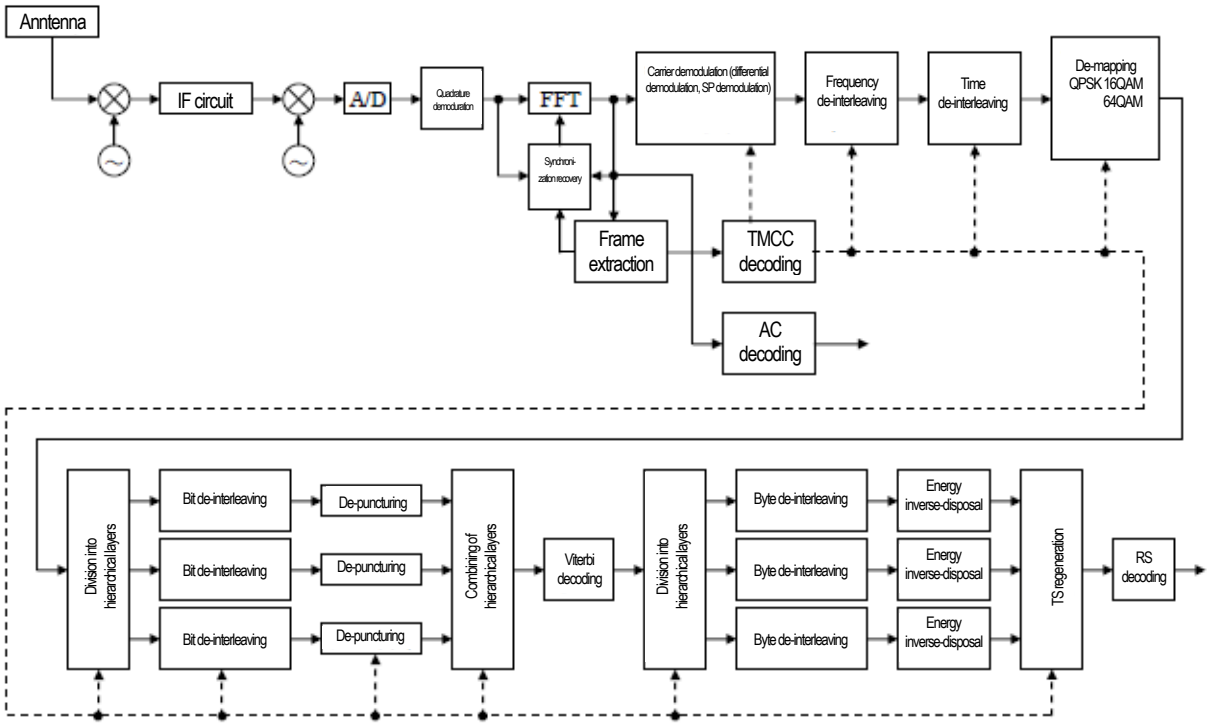
- Minimum input level: -85.5 dBm or lower (targeted value)
- Maximum input level: -20 dBm or higher

However, when the tuning unit is in the one-segment reception mode or partially receiving one segment located in the central part of the 13 segments, the level should be reduced by a factor equivalent to the bandwidth (i.e., $1/13^{th}$ or -11 dB).

Note: The parameters used for the measurement of the input level should be as follows: Mode 3, guard interval ratio of 1/4, time interleaving of 4, modulation of 16 QAM, and an inner-code coding rate of 1/2.

6 Front-end Signal Processing

- Overview of reception block diagram



- Channel selection: Super segment 1

  Super segment 2

  Super segment 3 (sub-channel 1, 4, 7, 10, 13, 16, 19)

For details on each block, see Chapter 4 "4.2.6 Front-end signal processing" in Part 1 in ARIB STD-B53.

Description

Description 1 Copy Generation Control for Analog Video Output (Copy Generation Management System – Analog (CGMS-A))

To output analog video signals from a multimedia broadcasting tuner, CGMS-A based copy control shall be used.

Regarding 480i (525i) streams and related copyright information, conformance to CPR-1204 of JEITA (EIAJ) and IEC 61880, respectively, shall be ensured. Regarding 480p (525p) streams and related copyright information, conformance to CPR-1204-1 of JEITA (EIAJ) and IEC 61880, respectively, shall be ensured. Regarding 720p (750p) and 1080i (1125i) streams and related copyright information, conformance to CPR-1204-2 of JEITA (EIAJ) and IEC 61880, respectively, shall be ensured.

1 Definition of CGMS-A

The definition of CGMS-A and recording control applicable to recording media is shown in Table C1-1.

Table C1-1: Definition of CGMS-A and recording control applicable to recording media

CGMS-A	Definition	Recording Method
0 , 0	Copy freely	Recording is done with the CGMS mode set to 0 , 0.
0 , 1	Not defined	
1 , 0	Copy One generation	Recording is done with the CGMS mode set to 1 , 1.
1 , 1	Copy Never	No recording can be done.

2 Transmission Method When Using CGMS-A

Copy generation control information shall be transmitted using the 1H of the vertical blanking interval of luminance signals. Reference signals, reduced to 70% of the white peak level, and 20-bit digital signals with an amplitude of 70% or 0%, shall be assigned to the 1H of the active video area, and using the 20 bits, the copy generation control information and video-related information are coded for transmission.

2.1 Analog Output using the Composite System (480i)

Analog transmission using the composite system (480i) shall conform to the specifications for the identification signal waveform in the following standard.

- EIAJ CPR-1204 "Video ID Signal Transmission Method Using a VBI (525 Line System)"

2.2 Analog Output using the Component System (480i)

Multiplex line	The 20H and 283H vertical blanking interval of luminance signals
Multiplex level	Logic 1: 70% ± 10% of the white peak level
	Logic 0: +10% and -5% of the black level
Clock frequency	fsc/8 = (455/16) fH = 447 kHz

It should be noted that fH represents the horizontal scanning frequency.

The transmission signal waveform is shown in Fig. C1-1. The cumulative time error from the rise of Ref bit to each bit shall be below 0.44μs.

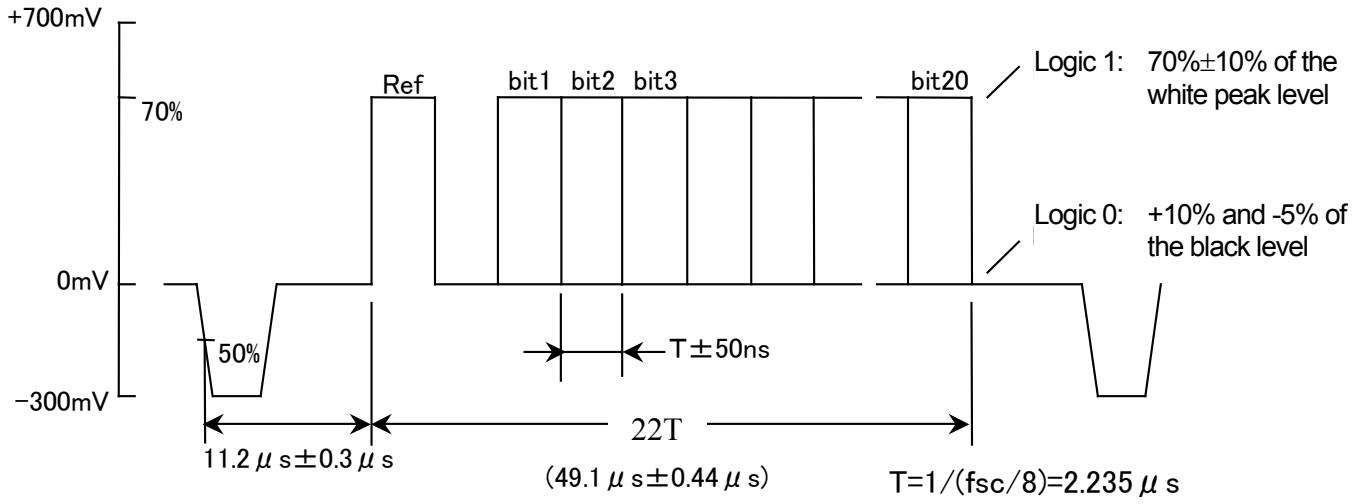


Fig. C1-1: Identification signal waveform when using the component system (480i)

2.3 Analog Output using the Component System (480p)

Analog transmission using the component system (480p) shall conform to the specifications for the identification signal waveform in the following standard.

- EIAJ CPR-1204-1 "Video ID Signal Transmission Method Using a VBI (525p System)"

2.4 Analog Output using the Component System (720p)

Analog transmission using the component system (720p) shall conform to the specifications for the identification signal waveform in the following standard.

- EIAJ CPR-1204-2 "Video ID Signal Transmission Method Using a VBI (750p and 1125i System)"

2.5 Analog Output using the Component System (1080i)

Analog transmission using the component system (1080i) shall conform to the specifications for the identification signal waveform in the following standard.

- EIAJ CPR-1204-2 "Video ID Signal Transmission Method Using a VBI (750p and 1125i System)"

3 Assignment of Identification Signals

The identification signal is comprised of 20-bit information, and the 20-bit data is comprised of WORD0=2-bit, WORD1=4-bit, WORD2=8-bit and CRCC=6-bit.

The detailed structure is shown below. The unspecified bits are considered not in use ("0").

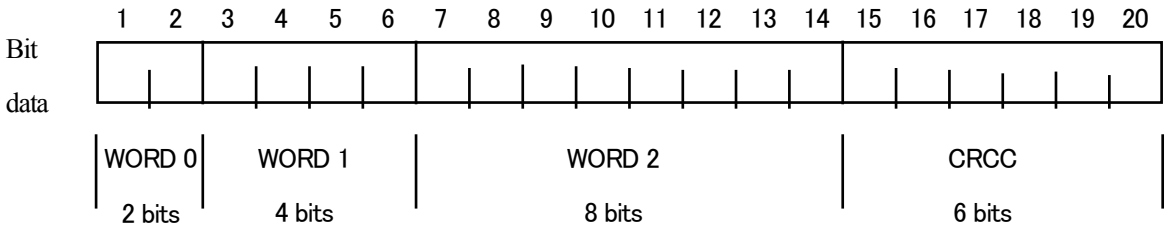


Fig.C1-2: Assignment of identification signals

WORD0 -- Information concerning the aspect ratio

Table C1-2: WORD0 -- Information concerning the aspect ratio

WORD 0		Detail
Bit 1	Bit 2	
0	0	Signal for a picture with an aspect ratio of 4:3
0	1	Signal for a letter box with an aspect ratio of 4:3
1	0	Signal for a picture with an aspect ratio of 16:9
1	1	Reserved

WORD 1 – Header that indicates the information to be transmitted in WORD2

Table C1-3: WORD 1 – Header that indicates the information to be transmitted in WORD2

WORD 1				Information Transmitted in WORD2
Bit 3	Bit 4	Bit 5	Bit 6	
0	0	0	0	CGMS-A information
1	1	1	1	No information
Other than above				Not defined

WORD2 – Information in bits 7, 8, 9 and 10

When bits 3 to 6 in WORD1 are set to 0000, CGMS-A information and analog output copy control information shall be respectively assigned to bits 7 and 8 and bits 9 and 10 in WORD2.

Table C1-4: WORD2 – Information in Bits 7 and 8

b7 b8	CGMS-A
0 0	0, 0
0 1	0, 1
1 0	1, 0
1 1	1, 1

Table C1-5: WORD2 – Information in Bits 9 and 10

b9	b10	Analog Output Copy Control Information
0	0	Copy Freely
0	1	A seudo-sync pulse is present
1	0	Seudo-sync pulse + 2 line inverted and split burst insertion
1	1	Seudo-sync pulse + 4 line inverted and split burst insertion

WORD2 – Information in bits 11 to 14

These bits shall be left undefined (Logic 0).

CRCC – Information in bits15 to 20

The CRC code is the error check code.

Generator polynomial G(x) is expressed as follows: G(x) = X6+X+1.

In Fig. C1-3, all preset values are "1".

SW1 is closed and SW2 is placed to the "a" position. Then, the first 14-bit data are entered.

From the 15th bit, SW1 is open and SW2 is placed to the "b" position to output CRCC.

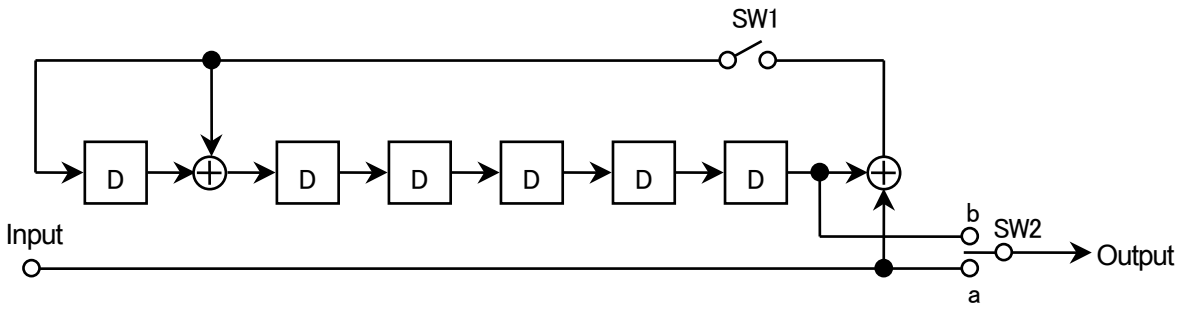


Fig. C1-3: CRCC

Description 2 Guarantee of Uniqueness of Broadcasting Programs and Contents

It is desirable to observe the following matters for receivers.

- Content shall not be presented by forcibly mixing the content that is not related to the service intended by a broadcasting station, except for a case where such an operation is performed by a user.
- It is prohibited to support a function to cut or skip a specific part of a broadcasting program or content independently from the user's operation using a broadcast signal or descriptor or data included in a broadcast signal, etc.

<Blank Page>

Description 3 Restrictions regarding the Functions for Using SI Information on Receivers

Regarding the use of SI information on receivers, the following restrictions should not be violated without the permission of broadcasters.

- Receivers should not come with the function to output only the SI information extracted from multiplexed broadcast signals. However, this does not apply to the output of the SI information related to the contents intended for private copy.
- Receivers should not come with the function to simultaneously display the SI information and the similar information from the other networks not specified in this document as if the information are identical. However, this does not apply to the cases where the SI information is present for the BS/wideband CS on common receivers.
- Receivers should not come with the function to display the SI information together with the contents other than those related to the SI information. One example of this is the display of advertisements issued via the Internet not related to the programs on the program table.

It is desirable that when displaying a broadcast program that includes recorded contents using the search function, information regarding the program broadcasting source such as channel information is displayed in a way that is readily understood by the viewers.

<Blank Page>

Description 4 Precautions for Implementation of MPEG-2 AAC

The following should be taken into consideration regarding implementation of the MPEG-2 AAC format:

- When judging the audio encoding mode, it should be kept in mind that PCE may not necessarily be being sent out at all times as prescribed in ARIB STD-B32, Part 2. Hence, the receiver should ideally be designed to judge the audio encoding mode based on "5.2.3 (2) Detailed provisions regarding coding mode in a single ADTS and ADTS configuration (order of transmission)" of the above standard, in the conditions other than the mandatory transmission conditions for PCE prescribed in "5.2.3 (3) Detailed provisions regarding transmission of PCE (Program Configuration Element)" of the above standard.
- Refer to "Annex 4: Precautions associated with implementation of MPEG-2 AAC standard" in ARIB STD-B32, Part 2. The appendix is quoted here for the convenience of designers who may use this volume as a guide.

(Start of quotation from Annex 4 of ARIB STD-B32, Part 2)

This reference explains the characteristics of coding tools used in the MPEG-2 AAC standard and presents precautions associated with the implementation of the standard.

(1) Treatment of AAC coding tools

The MPEG-2 AAC standard specifies three coding tools that can be used in the AAC LC profile: M/S Stereo, Intensity stereo, and TNS. Note that "Prediction" and "Gain Control," which are beyond the usable range in the LC profile, cannot be used and that use of "Coupling Channel" is prohibited in Section 5.2.2 of Part 2 of the present standard.

The AAC standard stipulates that these three coding tools shall be treated in the decoder according to bit streams. Namely, decoding shall be carried out according to the bit stream no matter which coding tool is being used. This specification should be taken into account when designing and implementing decoders.

(2) TNS

When designing and implementing decoders, care should be taken on TNS, which may require a large number of steps for decoding. Although the highest TNS filter order in long window mode is restricted to 12 in the LC profile, compared with 20 in the Main profile, it is still possible that a large number of processing steps will be required. The factors that directly affect the number of steps for TNS decoding include the number of filters, filter order, filter length, and the number of channels.

In the case of a decoder DSP that performs fixed-point calculation, the required precision of

operation may not be ensured when a high scaling level is adopted to avoid overflow or underflow caused by filtering (i.e., TNS may not be effective in improving sound quality). When designing and implementing decoder systems, therefore, care should be taken on overflow and underflow caused by filtering.

(3) Huffman decoding

The processing load of Huffman decoding tends to increase with bit rate. When designing and implementing decoders, theoretical maximum instantaneous rate should be taken into account in reference to the buffer model of the AAC standard.

In the case of encoders, care should be taken to avoid excessively high maximum instantaneous rate when bit rate is relatively high.

(End of quotation from ARIB STD-B32, Part 2)

Description 5 Notes When Using Receivers without Some Functions

Note the following points if a receiver is implemented that does not support certain functions, due to the product design of each manufacturer.

- Receivers without a communication function
  - As a license for storage-based broadcasting is provided only via communication by a broadcaster, the content cannot be played back if limited playback is applied to the content to limit viewing.
  - The compensation of stored content is provided only via communication by a broadcaster. Therefore, the compensation of stored content cannot be performed.
  - Among the licenses for real-time broadcasting (EMM), the licenses provided by a broadcaster via communication cannot be used. Therefore, viewer convenience shall be ensured by separately applying a method that is set by a broadcaster.

- Receivers without the storage-based broadcasting reception function
  - This standard defines that the necessary information to create the EPG/ECG is provided mainly as metadata in storage-based broadcasting by a broadcaster. The information transmitted using the EIT is limited to the information that needs to be transmitted immediately, etc. Therefore, attention should be paid so that the content of the information is limited when creating the EPG using the EIT. The necessary information to create the ECG is not sent using the EIT.

- Receivers without a touch panel
  - The BML browser defined in Vol. 3 of this standard is designed to mainly use a touch panel as a user interface. Therefore, if a receiver that does not have a touch panel is used, it is necessary to individually support a pointing device or other function. Each manufacturer can decide whether or not to implement a pointing device and the type of the pointing device based on their own product design.

- Function to automatically switch between the longitudinal-type screen and lateral-type screen
  - As the BML browser defined in Vol. 3 of this standard, the following two standards exist: a standard for the longitudinal-type screen and a standard for the lateral-type screen. Broadcasters can apply the BML for each standard. Note that two different types of BMLs may be applied if a receiver is designed

to output only to the lateral-type screen or the longitudinal-type screen.

- Caption/superimposition display function

➢ Note that receivers without the caption/superimposition display function may not be able to receive important information, such as telops.

Specifically, the receivers without some functions are the following types of the receivers.

✓ Simple real-time broadcasting receivers

This type of receiver outputs video/audio to a presentation device using a presentation device, HDMI cables, IP, etc. Receivers do not always support either/both communication function or/and storage-based broadcasting reception function. A touch panel and a function to automatically switch between the longitudinal-type screen and the lateral-type screen cannot be expected to be supported in this type of receiver.

✓ Separate-type receivers

This type of receiver is connected using a presentation device and a local access bus. Such receivers may not support one or many of the following functions: communication function, storage-based broadcasting reception function, touch panel, and function to automatically switch between the longitudinal-type screen and the lateral-type screen.

✓ Vehicle-mounted receiver

This type of receiver uses Mobacas as a method to update content for vehicle-mounted receivers and signage. The vehicle-mounted receivers may not have one or many of the following functions: communication function, storage-based broadcasting reception function, touch panel, and real-time broadcasting reception function. A function to automatically switch between the longitudinal-type screen and the lateral-type screen cannot be expected to be supported in this type of receiver.

Appendix

Appendix 1 Guidelines for Creating the EPG/ECG Using EPG/ECG Metadata

1 Program Channel Selection

1.1 Selecting a Real-time Broadcasting Service Channel

A URI (URI including <original_network_id>, <transport_stream_id>, <service_id>, and <event_id>), which is described in ProgramLocationTable/BroadcastEvent/ProgramURL, is used to select a channel from the metadata for real-time broadcasting content. It is recommended to use the CRID of the metadata in a real-time broadcasting content in crid://<authority>/<data>/<service_id>/<event_id> format.

1.1.1 Data broadcasting

In metadata for real-time broadcasting content, correspondence between a program and data broadcasting can be expressed by specifying ISDBTMMHowRelatedCS:24.X (X is selected from the values defined in 3 in Annex 1 of Vol. 10 "Metadata Operational Standard") in ProgramInformation/BasicDescription/RelatedMaterial/HowRelated/@href. A data broadcasting URI or a program URI is described in the RelatedMaterial/MediaLocator/mpeg7:MediaUri above.

A real-time broadcasting content type is indicated by specifying the ISDBTMMContentTypeCS:4.Y (Y is an optional number, which is selected from the values defined in 5 in Annex 1 in Vol. 10 "Metadata Operational Standard") in ProgramInformation/BasicDescription/ContentProperties/@href. If content only for data broadcasting is handled as real-time broadcasting content, an applicable content type is specified using this attribute (ISDBTMMContentTypeCS:4.3 indicates "program-linked data broadcasting" and 4.4 indicates "independent data broadcasting").

1.2 Selecting a Storage-based Broadcasting Service Channel

There is no ES that should be presented during recording in a storage-based broadcasting service. Therefore, it is preferable that receivers notify users that a storage-based broadcasting service is being received. However, if a storage-based broadcasting service is received during the real-time broadcasting service, notification shall be made using an icon, etc., and influence on the presentation of real-time broadcasting content shall not occur.

2 EPG and Video Recording in Multimedia Broadcasting

Basically, each manufacturer can decide on the EPG function (such as program guide display, program search, and programmed recording) using metadata or the EPG user interface based on their own product design. However, taking into account user convenience and the guidelines for creating metadata, some EPG display examples are provided below.

2.1 Basic EPG Function

(1) Video recording needs to be scheduled by event unit from the EPG. For details, see "2.3.2 Scheduled recording by event unit" in Appendix 1.

(2) Video recording needs to be scheduled by a series unit from the EPG. For details, see "2.3.3 Scheduled recording by series unit" in Appendix 1.

(3) The EPG functions, such as program search and detailed information display, need to be available regardless of the program characteristic information.

(4) It is required to be able to display program characteristic information by event unit, service pattern, and the attributes, such as recording capacity.

- The program characteristic information and service pattern are determined based on the content described in ProgramInformation/BasicDescription/Keyword[@type="other"] (for details, see 2.3 in Annex 4 of Vol. 10 "Metadata Operational Standard").
- As the recording capacity of real-time broadcasting content is estimated based on a bit rate obtained from the ProgramInformation/AVAttributes/BitRate, the capacity is only a rough estimate. Therefore, it is optional to display the recording capacity. For details, see Vol. 10 "Metadata Operational Standard."

(5) It is preferable that the series attributes of a series belonging to a selected program can be displayed from a program table. The series' attributes are the series name, the expected date when the series ends, the number of episodes in the series, the last episode number, and program resending information. For details, see Vol. 10 "Metadata Operational Standard."

- The CRID of series metadata can be obtained from the MediaLocator/mpeg7:MediaUri of ProgramInformation/BasicDescription/RelatedMaterial/HowRelated/[@href="ISDBTMMHowRelatedCS:23"] in the program metadata.
- By referencing the series metadata above, the series name can be obtained from GroupInformation/BasicDescription/Title[@type="main"], the expected date when the series ends can be obtained from GroupInformation/Period[@type="display"]/End, and the last episode number can be obtained from GroupInformation/@numOfItem.
- By referencing program metadata, the number of series can be obtained from ProgramInformation/MemberOf/@index, for which ProgramInformation/MemberOf/@crid has the CRID of the series metadata, and the program resending information can be obtained from ProgramLocationTable/BroadcastEvent/RepeatLabel.

For any items other than the items described above, see "ARIB TR-B14" and "ARIB TR-B15."

2.2 Activating a Broadcaster's Portal from the EPG

It is desirable to support a function that activates a braodcaster's portal without changing the status of the selected broadcasters, services, and programs presented in a program table.

- The method for providing the URI of a broadcaster's portal provided by each broadcaster depends on the service. For instance, the URI is stored in a terminal in advance as the initial setting.

- The purchase of an applicable service or program can be recommended to users by confirming whether or not recording can be performed and by activating a broadcaster's portal if recording cannot be performed.
- A site ID, which is used when purchasing a service or program, is obtained from the PricingServerURL of PurchaseInformation/@purchaseIdRef, which has the purchaseId described in the //PurchaseList/PurchaseIdRef for service metadata or program metadata.

2.3 Scheduled Recording of Real-time Broadcasting Content

This section describes the overview of the scheduled recording of real-time broadcasting content using metadata.

2.3.1 Registering scheduled recording

(1) A recording can be scheduled from the EPG and BML content. The recorded content is managed equally in a receiver regardless of whether a scheduling method is used.
(2) Scheduling can be registered by event unit or series unit.
(3) Note that an event_id or series_id should be a reusable id.

2.3.2 Scheduled recording by event unit

It is expected to manage CRID, service ID, broadcasting time, duration, URI for viewing as scheduled recording information. A CRID is obtained from ProgramInformation/@programId (a CRID is described in crid://<authority>/<data>/<service_id>/<event_id> format), the service ID is obtained from BroadCastEvent/@serviceIdRef, the broadcasting time is obtained from BroadCastEvent/PublishedStartTime and BroadCastEvent/PublishedEndTime, the duration is obtained from BroadCastEvent/PublishedDuration (or ProgramInformation/BasicDescription/Duration), and the URI for viewing is obtained from BroadCastEvent/ProgramURL (the namespace of an event is specified: the arib://<original_network_id>.<transport_stream_id>.<service_id>.<event_id> format).

2.3.3 Scheduled recording by series unit

Scheduled recording by series unit means to schedule the recording based on the program metadata and series metadata for real-time broadcasting service by scheduling the recording of a series program.

(1) Specify an event that corresponds to a specified series based on the series metadata information and implement a function to schedule the recording of the event. Specifically, the following functions are necessary. For details, see "3.9.2 Correspondence between EIT and metadata" in Vol. 10. To obtain series metadata from program metadata, obtain the CRID of the series GI (the CRID is described in crid://<authority>/<data>/<series_id> format) from the MediaLocator/mpeg7:MediaUri of the program metadata 's ProgramInformation/BasicDescription/RelatedMaterial/HowRelated[@href="ISDBTMMHowRelatedCS:23"]. As for series metadata, a "series_id" is described in the MediaLocator/mpeg7:MediaUri of the GroupInformation/BasicDescription/RelatedMaterial/HowRelated[@href="ISDBTMMHowRelatedCS:2

2"].

(2) Obtain the schedule pattern of an applicable series from the information as described in GroupInformation/BasicDescription/Keyword[@type="other"] of the series metadata (for details, see 2.3 in Annex 4 of Vol. 10 "Metadata Operational Standard"). Set a schedule to record an event, which is not yet recorded and which will be broadcast on the nearest date and time, among the events belonging to the selected series. Specifically, the program metadata, for which the CRID of the applicable series metadata is described in MemberOf/@crid (GroupInformation/@groupId), is searched (MemberOf search; for details, see "Section 6.3 Search request" in Vol. 10). An event that will be broadcasted on the nearest date and time is obtained by referencing the BroadCastEvent/PublishedStartTime and BroadCastEvent/PublishedEndTime of the obtained program metadata. The event that should be recorded next is programmed by checking whether or not the event that belongs to a series can be recorded. The total number of the programs in an applicable series is obtained from the GroupInformation/@numOfItems of series metadata. Scheduling is set to record the programs (excluding resent programs) based on the obtained value. For details on the series event usage, see "Chapter 17 Operation of Series" in Vol. 4. It is not guaranteed that the program metadata corresponding to all events related to the series will exist when a user program has been set to record the series. Therefore, it is desirable that the receiver checks whether there is an event related to the series by updating metadata once a day until the preset time to start recording the series.

(3) It is preferable to support a function that enables users to temporarily suspend or cancel a schedule to record a series.

- Temporary suspension of schedule to record a series means to temporarily suspend a schedule to automatically record an event that belongs to the series. As a premise, the schedule shall be able to be restarted via the user's direction.
- Cancellation of schedule to record a series means to cancel all the schedules to automatically record the series and discard the registered details. In this case, it should be noted that the application status of the service does not change.

2.4 Auto-scheduled Recording

As part of the product design of each manufacturer, a receiver can have an auto-scheduling function based on any information other than a series.

For instance, it is possible to implement a function to automatically record an available unit of content using a keyword or genre described in metadata (such as ProgramInformation and GroupInformation).

2.5 Ensuring Recording Capacity When Scheduling

(1) Receivers ensure necessary capacity when setting a schedule for recording and can provide the information to the users.

- Note that the capacity to record real-time broadcasting content is estimated based on the bit rate value obtained from ProgramInformation/AVAttributes/BitRate and the duration obtained from BroadCastEvent/PublishedDuration (or ProgramInformation/BasicDescription/Duration).

(2) It is desirable that a recording can be scheduled even if enough capacity cannot be ensured.

2.6 Confirmation of a Scheduled Recording and Resetting the Schedule

(1) The receiver has a function that enables users to confirm a scheduled recording status. Each manufacturer can decide on the details on how to confirm a scheduled recording status based on their own product design.

- Receivers shall have a method to check any overlap with a scheduled time for starting the recording of a program/content when newly registering the recording of a program/content.
- It is preferable that users can check the overlap of a scheduled recording at any time. It is also desirable to provide a method to eliminate overlapping. Note that a scheduled recording may be overlapped when recording starts, even if the overlap of the scheduled recordings is not detected when the recording is scheduled.
- As a method to eliminate any overlap, programs can be resent. It is desirable that the receivers detect resent program schedules and notify the users.
- In real-time broadcasting content, whether or not playback can be performed can be checked using the BroadCastEvent/RepeatLabel. The metadata of a program, which is resent using the series_id and episode_number described in ProgramInformation/BasicDescription/Keyword[@type="other"], is obtained (for details, see Section 2.3 in Annex 4, Vol. 10 of this technical document). Note that the CRID of the program that is resent is different.

(2) Receivers have a function to cancel a scheduled recording. This function shall support both cancellations by event unit and by series unit.

3 Recording Function

3.1 Managing Real-time Broadcasting Content in a Receiver

(1) Recorded real-time broadcasting content is managed by associating with a content reference identifier (CRID). Specifically, when an applicable CRID is specified, corresponding real-time broadcasting content is managed by retaining identification.

(2) Each resource, which configures real-time broadcasting content, is managed by being associated with a URI that is used for referencing the resource. The recording method for an actual recording device and the management method vary depending on the receiver. However, as a possible method, recorded real-time broadcasting content can be stored/managed by module unit.

3.2 Executing a Scheduled Recording

Scheduled recording is a recording method performed by a receiver that automatically records based on the scheduling registered by a user beforehand. A scheduled recording is registered with the event unit.

The recording time is controlled based on the metadata or SI information.

The method to decide priority order when multiple scheduled recording times overlap or when a scheduled recording overlaps with another scheduling varies depending on the receiver.

3.2.1 Recording events

(1) All available units of content, which are transmitted in an event to be recorded, are recorded. Specifically, a recording starts/ends based on the start/end time of an event indicated in BroadCastEvent/PublishedStartTime and BroadCastEvent/PublishedEndTime.

(2) The start/end of an event to be scheduled for recording is determined based on the EIT [p/f] information and performed according to the description in "18.3 Basic Rules regarding Event Progress" in Vol. 4 of this technical document.

(3) Determining whether or not a recording can be performed per event is based on the following requirement.

- Only one real-time broadcasting content exists in an event.

(4) Determining whether or not a recording can be performed per available unit of content is based on the following requirements.

- A recording operation continues from the start time of an event to the end time. If the receiver does not detect an error during the recording, it is determined that the recording was successfully performed. Note that it is acceptable if a recording could not be made five seconds after the event starts and 20 seconds before the event ends.
- Receivers do not have a method to completely check whether available units of content were recorded without error. Therefore, receivers can determine that an event was recorded successfully if no error was found within the receiver detection range. Each manufacturer can decide the detection level based on their own product design.

(5) Even if the recording of an event has failed, the partially recorded real-time broadcasting content can be used.

3.2.2 Supporting event programming modification

The following action is taken if event programming related to real-time broadcasting content is changed. For how to send the SI when an event programming is modified, see "18.5 Guidelines for the Transmission of a Change of Programming" in Vol. 4 of this technical document.

For the meanings of the terms "current event" and "next event," used below, see the description in "ARIB STD-B10." For the meaning of the terms "event suspension" and "event interruption," see the description in "18.5 Guidelines for the Transmission of a Change of Programming" in Vol. 4 in this technical document.

(1) When an event is extended or ends early

- If the duration value of the present program information in the EIT [p/f] for the event being recorded is changed, the receiver recognizes that an event is extended or ends early and records for a period of time that corresponds to the start_time and duration set in the EIT [p/f].

(2) When the start time of an event is delayed

- If the start_time of the following program information set in the EIT [p/f]t is changed when the next event is scheduled to be recorded, the receiver recognizes that the start time of the event is delayed and records for a period of time that corresponds to the start_time and duration of the following program information set in the EIT [p/f].
- If the start time of an event is changed to a time later than when the next event starts, metadata is obtained again and the scheduling is reset by confirming the changed schedule.

(3) Support for event suspension/interruption

- If the current event information in the following program information in the EIT [p/f] (the information described previously in the present program information in the EIT [p/f]) appears while the receiver is recording the current event, then the receiver recognizes that an event was suspended.
- While suspending an event, it is preferable that the receiver continues to record based on the following information of the EIT [p/f] (the event information that has been originally recorded).
- If the recording is stopped while suspending an event, the recording operation shall be able to be restarted when the suspended event is replayed. When the recorded content is played back, the content recorded before/after the event was suspended shall be available.
- If the EIT [p/f] is updated while suspending an event, and if the suspended event information is described in the present program information in the EIT [p/f], the receiver recognizes that the suspension of the event is over and that the suspended event is restarted, and then continues to record based on the present program information (suspended event) in the EIT [p/f]. After that, the receiver performs the same recording processing as normal events.
- If the EIT [p/f] is updated and the suspended event information disappears from the EIT [p/f] during the suspension of the event, the receiver terminates the recording processing of the event.

(4) Support for undefined events

- If both start_time and the duration of the following program information in the EIT [p/f] are changed to "all 1" when a scheduled recording is programmed, the receiver recognizes that the event status is changed to undetermined and cancels the scheduled recording.

(5) Support when scheduled recording overlaps

- If start_time of the following program information in the EIT [p/f] is changed when a program for which the recording time is close to that of the other program that is currently scheduled to be recorded, then it is expected that the recording time overlaps that of the other program. Each manufacturer can decide the priority order of the programs to be recorded based on their own product design.

3.3 Storing Metadata

For details on metadata acquisition and storage, see "Chapter 7 Metadata Storage Control" in Vol. 10 of this technical document.

4 ECG

This section explains the ECG/navigation function for storage-based broadcasting content.

4.1 Management of Various Information that Configures the ECG

4.1.1 Managing metadata in receivers

The receivers that support the ECG shall be designed to create and manage a metadata database based on the metadata. The method of managing metadata in a receiver depends on the implemented receiver.

(1) Receivers receive/store metadata sent via broadcasting or communication and manage such so that it can be displayed as a list, etc.

(2) An association between a resource, which configures an event or available unit of content, and metadata and an association between the metadata for each content, which configures a content group, such as package and series, are properly created.

(3) Receivers need to update the latest status in the metadata database in the receiver based on the latest obtained metadata. Note that the applicable metadata may be referenced from the ECG during the update (for details, see "7.2 Adding/updating metadata" in Vol. 10).

(4) If a presentation period is specified, the metadata cannot be used after the specified period. However, this cannot be applied when corresponding content is stored. Metadata for which the presentation period has passed should be deleted immediately (for details see "Section 7.3 Deleting metadata" in Vol. 10).

4.1.2 Information created/managed in a receiver

It is preferable that receivers can manage the following information in order to present such to the ECG.

(1) Storage status of each content (not stored, storing, suspending, stored, etc.)

(2) Viewing status of each content (viewing, etc.)

(3) Distribution route for stored content

(4) Size of the stored content in a storage device

(5) License ID list of purchased content that is notified by the X_TMM_getPurchaseInfo () calling from the BML

4.2 Displaying the ECG

(1) Receivers shall have a list display function to play back/use the content stored in them.

(2) The following two types of list displays are supported: Display by available unit of content unit and

display by group (series/package) unit.

(3) Whether to display (1.1 to 1.4) or not (1.5) in a list is controlled based on the //ProgramInformation/Genre/@href of metadata or the value of the ISDBTMMDeliveryStatusCS that is specified in the //GroupInformation/Genre/@href (for details, see 6 in Annex 1 of Vol. 10 "Metadata Operational Standard").

(4) Each manufacturer can decide on the display order of the listed content based on their own product design of a receiver. It is preferable that such is ordered so that a significant difference does not exist between the content lists of each broadcaster. For the sort specification when searching a metadata server, see "6.3.2.1 Common format" in Vol. 10.

(5) Each manufacturer can decide on a method for displaying other lists and a sorting method based on their own product design. However, they should be creative in developing the method, such as in using hierarchical expression, as the number of content may be large.

(6) The location of each available unit of content (the content is stored or can be obtained via broadcasting in the future) is presented to users.

- As for stored content, the relationship between the CRID and the actual location where the content is stored in a receiver is managed in the receiver.
- As for storage-based broadcasting content, the storage-based broadcasting schedule of applicable contents can be obtained from the Session Description (for details, see Chapter 2 in Vol. 11) of the transmission control metadata that is described in OnDemandService/OnDemandProgram/ProgramURL.

4.2.1 Presenting a still picture thumbnail

(1) A still picture thumbnail that corresponds to an available unit of content can be displayed using ProgramInformation/BasicDescription/MediaTitle/TitleImage information. A still picture thumbnail that corresponds to a group (series/pack) can also be displayed using GroupInformation/BasicDescription/MediaTitle/TitleImage information. In addition, a still picture thumbnail that corresponds to a coupon can be displayed using CouponDescription/CouponImage and CouponDescription/CouponThumbnail information.

(2) In the metadata described above, a URI that represents a still picture resource starting with "crid:" or "http:" is described. If a URL starting with "http:" is specified, receivers can display a still picture thumbnail by obtaining the resource above using HTTP.

(3) If a still picture thumbnail is displayed, it can be displayed by scaling within the extent that the impression of the original picture is not largely altered. The aspect ratio shall be maintained, and the picuture shall not be trimmed.

(4) It depends on the product design of each manufacturer whether to display a default image or not if a specified still picture resource does not exist in a receiver or if a specified still picture resource could not be obtained during the communication.

4.2.2 Presenting a preview moving image

(1) A preview moving image that corresponds to available unit of content can be displayed using ProgramInformation/BasicDescription/MediaTitle/TitleVideo information. A preview moving image that corresponds to a group (series/pack) can also be displayed using GroupInformation/BasicDescription/MediaTitle/TitleVideo information.

(2) In the metadata above, a URI (rtsp://xxx, http://xxx) that indicates an AV resource on a streaming server or the CRID (crid://xxx) of other available unit of content (including an AV resource) is described.

(3) When preview moving images are displayed, they can be displayed by scaling. However, they shall be displayed retaining the aspect ratio of the video parts. For instance, if the aspect ratio of a display area and the aspect ratio of a video signal are different, images shall be displayed retaining the aspect ratio of the video by adding a side panel, etc.

(4) The preview playback of real-time broadcasting content is controlled based on the preview control descriptor of the ECM. The information described in LicenseInformation/RMPIDescription/PreviewDescription is used for display, etc., in the ECG.

4.2.3 Content search function

Receivers can search content by title or keyword using obtained metadata.

4.3 Determination of Whether or not Content Can be Stored

It is preferable to make a decision on whether content can be stored after confirming the type of applicable content involved (for details, see "2.2 Description of Various Control Information Using Genre[@type="other"]" in Annex 4 in Vol. 10) and referencing the version information of the contents (for details, see 2.3 Description of Various Control Information Using Keyword[@type="other"]" in Annex 4 in Vol. 10).

4.4 Scheduled Storage of Storage-based Broadcasting Content Using the ECG

This section describes an overview of the scheduled storage of storage-based content using metadata.

4.4.1 Registering scheduled storage

(1) Storage can be scheduled from the ECG or the BML content. The stored content is managed equally in a receiver regardless of the method used.

(2) A scheduled storage can be registered by content unit and group (series/pack) unit.

(3) It is preferable to prohibit the scheduled storage after checking whether or not that content can be stored and if the result shows that the scheduled storage is not admitted at the time of scheduling the storage. Even if the storage of contents is not admitted, storage can be scheduled by displaying a message indicating that storage may fail. Whether or not contents can be stored can be determined by checking that a specific item is stored or purchased (see "5 License Related Information" in Appendix 1).

4.4.2 Scheduled storage by content unit

It is expected that CRID, service ID, session information, and schedule information are managed as storage reservation information. A CRID is obtained from the ProgramInformation/@programId, service IDs are obtained from the OndemandService/@serviceIdRef, and transmission control metadata is obtained from the OnDemandService/OnDemandProgram/ProgramURL. Session information and schedule information are obtained from the transmission control metadata (for details, see Chapter 2 in Vol. 11). It is desirable that any content can be scheduled to be registered regardless of whether a service is applied.

4.4.3 Scheduled storage by group unit (series/pack)

Scheduled storage by group unit means to schedule and register the storage by group unit based on the content metadata and group metadata information.

(1) The following functions are necessary to specify the content, which belong to a specified group (series/pack) based on the group metadata information, and implement a function to schedule the storing of the content.

- Among the content belonging to a selected group, the content that is not stored is scheduled to be recorded. Specifically, the program metadata for which ProgramInformation/BasicDescription/MemberOf/@crid has the CRID of the applicable group metadata (//GroupInformation/@groupId) is searched (MemberOf search; for details, see "6.3 Search Request" in Vol. 10).
- The transmission control metadata, which is indicated by the OndemandService/OndemandProgram//ProgramURL, of the acquired content metadata is obtained. Then, session information and schedule information from the transmission control metadata are obtained (see "4.4.2 Storage reservation by content" in Appendix 1).
- The storage scheduling algorithm varies depending on the receiver.

(2) The metadata for all content that belongs to a group may not always exist when a user schedules the storing of the content by group unit. Therefore, a receiver updates the metadata once a day until the scheduled date, in order to check whether there is any content belonging to the group.

(3) It is preferable to support a function that enables users to temporarily suspend or cancel the scheduled storing of content by group unit.

- Temporary suspension of the scheduled storing of content by group unit means to temporarily suspend the schedule to automatically store the content belonging to a group. As a premise, the suspended schedule shall be able to be restarted upon the user's direction.
- Cancellation of the scheduled storage of content by group unit means to cancel all the scheduled storage by group unit and discard the registered details. In this case, it should be noted that the application status of the service does not change.

4.4.4 Auto-scheduled storage

As the product design of each manufacturer, a receiver can have an auto-scheduled storage function based on any information other than series.

For instance, a receiver can have a function to automatically store the available unit of content using a keyword or genre described in metadata (such as ProgramInformation and GroupInformation).

4.4.5 Ensuring storage capacity when scheduling storage

(1) Receivers can ensure the necessary capacity when scheduling storage and can provide the information to the users.

- The storage capacity of a storage-based broadcasting content can be obtained from ProgramInformation/BasicDescription/ContentProperties/FileProperties/FileSize. Note that the total capacity described in the link above is not always necessary depending on the storage status of a receiver (for instance, a case where thumbnails are stored together or an update is performed by overwriting stored content).

(2) It is desirable that storage can be scheduled even if enough capacity cannot be ensured.

4.4.6 Confirmation of storage reservation and resetting the reservation

(1) Receivers support a function to confirm the reservation status that is operated by a user.

The manufacturer of the receiver can decide the details on how to confirm the reservation status as their own product design. As for storage-based broadcasting content, multiple schedule information may be described in the transmission control metadata that corresponds to applicable content. In this case, the schedule is modified by referencing the schedule information.

(2) The receiver has a function to cancel storage reservation. Both cancellation by content unit and group unit shall be supported.

4.4.7 Failure of scheduled storage

Scheduled storage may fail in the following situation.

・ Lack of available capacity in a memory/removable media in which content is stored

If a scheduled storage has failed because the receiver was out of the communication range, each manufacturer can decide on how to notify users based on their own product design.

4.5 Application/Cancellation of Service

(1) Receivers have a function that enables the user to select services by displaying package or content (PPC/PPV) services that can be applied on a screen.
- For the information that should be displayed on each package or content (PPC/PPV), see "4.5 Content Navigation."
- For how to check whether or not service has been applied for a package or content (PPC/PPV), see "5 License Related Information" in Appendix 1.

(2) Receivers display a screen to apply the package or content (PPC/PPV) services selected in the ECG.
- A site ID, which is described in GroupInformation/BasicDescription/PurchaseList/PurchaseIdRef of a package or PurchaseInformation/Purchase/PricingServerURL of PurchaseInformation that corresponds to ProgramInformation/BasicDescription/PurchaseList/PurchaseIdRef of a certain content, is used.

  * The management method for purchase site URLs in a terminal varies depending on the service provider.

4.6 Content Activation from the ECG

An available unit of content is played back from the ECG.

5 License Related Information

This section describes a method to identify the license related information of storage-based broadcasting content and real-time broadcasting content in the EPG/ECG.

5.1 License Information of Real-time Broadcasting Content

(1) License usage status

If a corresponding ECM exists, the status is checked by reading the usage status field from the ECM.

(2) Correspondence between a content and item

The correspondence between a content and item is expressed using the set of the content ID (LicenseInformation/ProgramCRID/@crid) and item ID (LicenseInformation/PurchaseIdRef) described in the license reference information.

(3) Note

Note that if the date of a real-time broadcasting license expires, the license cannot be used even if the ECM is effective.

5.2 License Information of File Content

(1) License usage status

The status is checked by reading the "usage status" from the license information that is read using the "license information acquisition message" from the storage-based broadcasting license.

(2) Correspondence between content and items

The correspondence between content and items is expressed using the set of content ID (LicenseInformation/ProgramCRID/@crid) and item ID (LicenseInformation/PurchaseIdRef) described in the license reference information.

6 Playback of Content

6.1 Storage-based Content Activation Processing

The resource file of applicable content is obtained using a specified content reference identifier (CRID). Each manufacturer can decide on how to store storage-based broadcasting content in a receiver based on their own product design. However, the management shall be done using a method to uniquely obtain available units of content corresponding to the metadata. For details, see "8.2.3.3 Location resolution in a receiver" in Vol. 10.

6.2 Playback Restriction Control When Performing Trick Play

Whether or not trick play can be performed is based on the conditions of the use of each content (RMPI). When a trick play, such as fast forwarded/rewound, is performed, the number of viewings (playbacks) is not counted. For details, see Vol. 5.

7 Operating Stored Content

7.1 Deleting Stored Content

7.1.1 Deleting stored content by user operation

(1) A function to delete stored content by available unit of content unit is supported. It depends on the product design of each manufacturer regarding whether or not they support the function of deleting stored content by package unit.

(2) Receivers shall not support a function that enables users to individually delete a resource that configures an available unit of content.

(3) When deleting an available unit of content, all the corresponding licenses stored in a receiver are deleted.

(4) When deleting an available unit of content, the corresponding still picture thumbnails, preview moving images, and transmission control metadata are deleted if they are stored.
(5) To delete an available unit of content that has a valid license when the deletion is ordered, the content is deleted after obtaining the user's consent without exception.
(6) If a card storage area or receiver storage area is full when obtaining a new license or at another timing, a notification should be made to the users and a user interface that enables deleting content should be presented.
(7) It needs to be noted that deleted content may be stored again via broadcasting (regardless of user intention).

7.1.2 Auto delete processing of stored content

(1) Receivers can implement a function to automatically delete any stored content that satisfies conditions (2) to (4) below.
(2) The available units of content that have a current and valid license shall not be automatically deleted. A license may be valid even if the expiration date, at which such content can be purchased, has expired.
(3) Content shall not be automatically deleted while a user is viewing the stored content or when content is locked.
(4) The operation that deletes content shall not affect other operations, such as viewing by user and storage.

7.1.3 Deleting content in a reference relationship

It is desirable to display a warning on the screen if any content is referenced by ProgramInformation/BasicDescription/RelatedMaterial while deleting content. Receivers can support a function to automatically delete related content. Sufficient attention shall be paid to support the function because complicated structures can be formed; for instance, a content with RelatedMaterial is described in RelatedMaterial.

7.1.4 Deleting metadata

It is preferable that corresponding metadata is deleted when deleting content. It is desirable to support a function that automatically deletes metadata in which the expiration date (FragmentExpirationDate) has expired. However, content is not deleted when the content that corresponds to applicable metadata is stored although the expiration date has expired. Metadata cannot be deleted individually. The entire available unit of content is deleted. For details on the requirements for deleting metadata, see "Section 7.3 Deleting metadata" in Vol. 10.

7.1.5 Protection from the operation of deleting content

This depends on the product design of each manufacturer regarding whether or not they support a function to protect other data when deleting applicable content. However, receivers shall support a lock function for available units of content from the BML content.

8 Functional Requirements for Metadata

In addition to the metadata-related functions described in the sections other than this section in this volume, receivers shall have a function to process metadata in the receivers, a function related to location resolution, and a function related to metadata reception. For details, see Vol. 10 "Metadata Operational Standard."

9 Parental Control Guidelines

The parental control function is a function that can restrict the display or purchase of applicable content based on user intentions if content with a viewing age limit (adult content and R-rated content) can be included in the broadcasting content provided by a broadcaster. This function is provided based on user intentions and on the mutual recognition between users and broadcasters. This function is also necessary to provide content and services that ensure safety for users. This section provides the necessary guidelines for realizing the parental control function.

9.1 Concept of Parental Control in Multimedia Broadcasting

This function can control content for which users cannot view, use, or purchase based on the registered basic information.

9.2 Overview of the Parental Control Function

To realize the parental control function, the following two items shall be set in a receiver as the parental control information.

(1) Parental level (minimum viewing age)
(2) Password for changing the parental level

(When content with a viewing age limit is displayed, the status is expressed as "with a viewing age limit." When content without a viewing age limit is displayed, the status is expressed as "without a viewing age limit.") In addition, it is necessary that a parental rate (ParentalGuidance/ParentalRating) is described in the metadata.

The parental control function is used to set a viewing limit on content with a viewing age limit based on the rules that are separately defined for part of the displayed content, such as the purchase button, preview button, and content title, if the result obtained by comparing the parental level set in the receiver and the parental rate is a parental rate value that is greater than the parental level.

(1) Hide
(2) Rewriting content
(3) Disabling button selection
(4) Masking

9.3 Parental Control Restriction Status and Operations Allowed for Receivers

"Table A1-1 Parental control restriction status and operations allowed for receivers" shows an overview of the relationship between the parental control restriction status and the expected operations allowed for receivers. The legends used in this table are as follows:

○: Available, X: Unavailable

Table A1-1: Parental control restriction status and operations allowed for receivers

Parental rate (R)	Receiver setting*1 (age)		EPG/ECG display, List display	Purchase	License acquisition	Storage (scheduled storage)	Viewing/usage, preview (recording)
G	–		○	○	○	○	○
20*2	No limit (including undefined)		○	○	○	○	○
	With limit	R = age (20)	○	○	○	○	○
		Age < R (4 to 19)	X	X	X	X	X
12 to 19	No limit (including undefined)		○	○	○	○	○
	With limit	R ≤ age	○	○	○	○	○
		Age < R	○	X	X	X	X
4 to 11	–		○	○	○	○	○

*1: The table above describes the parental control rule. The GUI for realizing the parental control function set in a receiver varies depending on the receiver.

*2: In multimedia broadcasting, adult content is not provided. However, adult content is mentioned in the explanation of this standard.

VOLUME 3

Digital Terrestrial Television Broadcasting
Multimedia Coding Regulation

Contents

Chapter 1 General Terms

1.1 Introduction

In multimedia broadcasting, the following data is transmitted:

- Image data and audio data: Simplified video (H.264|MPEG-4 AVC), MPEG-2 AAC
- Multimedia data: Data coding compliant with ARIB STD-B24 "Data coding and transmission specification for digital broadcasting", Vol. 2, Attachment 4
- Closed captions and superimpose
- Storage-type data: See Chapter 7 in this volume

The receiver function can be roughly divided as follows, depending on the type of broadcast:

・ Real-time-type broadcast reception function

This function receives real-time-type broadcasts. The image, sound and data should be in one piece as a broadcast content, and in principle, the receiver needs to present the whole of them.

・ Storage-type broadcast reception function

This function receives a storage-type broadcast, stores it, displays the stored contents and plays them back.

This volume regards the receiver equipped with both real-time-type and storage-type reception functions as a basic receiver. The functions to be provided for a real-time-type broadcast receiver are specified in Chapters 2-4. The functions to be provided for a storage-type broadcast receiver are specified in Chapters 2, 5 and 6.

1.2 Operation guidelines specified by this volume

1.2.1 Profile

Multimedia broadcast specifies the X profile in the real-time-type broadcast reception function, as a basic operation profile of data broadcasting services, targeting mobile phone receivers.

1.2.2 Basic functions and optional functions

- The basic functions refer to the functions that should be provided in the receivers that are equipped with the real-time and storage-type broadcast reception functions.
- Functions other than the basic functions are optional. This volume specifies the operation guidelines that should be at least observed when a receiver realizes an optional function.

When it is described as “preferable” in this volume, it refers to implementations (specifications) that are not required but are preferable from the broadcaster point of view.

1.3 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 (preferred specifications)” 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

1.4 Terminology

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.
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-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). 13-bit stream identification information of the TS packet header block. It shows the attributes of the individual stream of this relevant packet.
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). In multimedia broadcasting, 1TS is allocated to each of 13-segment broadcast and 1-segment broadcast.
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
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”.
BML-engine	Receiver software which receives and interprets multimedia 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 writing service	Service to broadcast "bookmark write contents" with the following features. A bookmark icon is presented, according to the timing designated in the contents in advance, to match 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 X-profile, refer to the "TVlink".
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.
Link state	Status where receivers 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.
VIDEO PES	Data component used to transmit encoded video as standardized in ISO/IEC 13818-1.
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.
Common fixed color	Color specified as common on the receiver color palette for things such as logo display.
Viewing reservation	Used to make reservations to view programs by event, based on SI information.
Closed caption	Service related to visual contents that display captions on top of the television broadcasting image.
Optional feature	When featured functions and performance depend on the receiver or product.
Server	Web server capable of handling HTTP/1.1 on TCP/IP networks.
Communication contents	BML contents located on servers
Independent data program	Data program principally involving multimedia data. In some cases, the video/audio components of TV programs are shared.
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 message	Unit of request for HTTP/1.1 servers.
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 receivers. (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.
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.
BML browser	The BML browser that operates according to the guideline specified in these operation guidelines. This operates in the data broadcast reception, linking status, and presents the contents received via broadcasting signals or communication lines.
Communication browser	General term to refer to the browser for the X-profile contents on the internet and the carrier’s proprietary browsers. Mixed display with broadcast contents and execution of some scripts will be restricted.
X-profile communication browser	The communication browser having the specifications that include the X-profile specified in these operation guidelines. However, execution of part of the scripts is 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 X-profile	The character encoding method for closed caption used in X-profile, this is subset by restricting the use of control codes based on the “8-bit character codes” used in Profile A.
X-profile communication contents	General term for BML contents located on servers specified in X-profile. Classified as X-profile linked contents and X-profile unlinked contents.
X-profile linked contents	Communication contents written under the assumption that a receiver in accordance with X-profile and with a linked status will present them.
X-profile unlinked contents	Communication contents written under the assumption that a receiver in accordance with X-profile and with an unlinked status will present them from a Browser for the X-profile contents on the internet.
Broadcaster contents	General term for broadcast contents and X-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 receivers.
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.
Multimedia	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 multimedia. 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 receivers. 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.
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.
Supplementing storage contents	If the contents received from the broadcasting waves are incomplete in storage-type broadcasting, this function supplements the incomplete part
MIME	Multipurpose Internet Mail Extensions: Protocol of application layer. Contents architecture that makes it possible to handle the multimedia data of text, audio and image in formats other than the US-ASCII format on Internet mail.
FLUTE	File Delivery over Unidirectional Transport: Protocol of distributing files for one-direction communications (RFC3926)
UDP	User Datagram Protocol: Format used for data transmission on the Internet. After data is transmitted, this communication protocol does not confirm whether the data has been transmitted to the destination.
TSI	Transport Session Identifier: Address port number of IP packet that stores the transmission file attribute information packet.
TOI	Transport Object Identifier: Object identification information. When an FDT instance is transmitted, the identification information value is set to 0.
INT	IP/MAC Notification Table: In the stream that configures the storage-type services, a receiver IP address of the transport stream ID/service ID/component tag/object with respect to the platform ID is designated.
AL-FEC	Application Layer Forward Error Correction: FEC format of the application layer.
ROHC	RObust Header Compression: Format to compress the headers in the IP layer or higher, defined in RFC3095.
SBN	Source Block Number: Numbers of source blocks obtained by dividing an object to be transmitted, using the information of transmission length, source block length and encode symbol length.

FDT instance	File Delivery Table instance: Object to transmit the File Delivery Table (FDT) which describes various attributes on the file distributed using FLUTE defined by IETF RFC 3926.
Transmission control metadata	Data in XML document format that describes the information necessary to receive and download storage-type broadcasting contents.
User Service Description	One of three description types specified by the transmission control metadata, which includes Session Description and Associated Delivery Procedure Description.
Session Description	One of three description types specified by the transmission control metadata
Associated Delivery Procedure Description	One of three description types specified by the transmission control metadata
SDP	Session Description Protocol: SDP format data that holds information that is unique to the download transmission method used for user services
ECG	Electronic Contents Guide: Description language-type metadata defined by ARIB STD-B38. Used to perform contents guide and contents navigation.
URI	Uniform Resource Identifier: Describing method that shows the location of the resource. Concept that includes URL
CRID	Content Reference Identifier. This identifier is not dependent on location
IANA	Internet Assigned Numbers Authority: One of the organizations making up IETF which standardizes the Internet
LDPC	Low Density Parity Check Code: Error-correcting codes that makes the part that could not be received recoverable using the data for correction.
UEP	Unequal Error Protection: Efficient error correction using unequal error protection
ULE	Unidirectional Lightweight Encapsulation: IP packet capsulation function to achieve IP transmission on the MPEG-2 Systems
CID	Context Identifier: ID to identify the flow of the IP packet whose header was compressed (aggregation of IP packets whose values in the five areas - IPv4 packet protocol or IPv6 packet next header and source address, destination address, source port and destination port – have the same combination).
HTML	HyperText Markup Language: Markup language for describing web pages.
XHTML	eXtensible HTML: HTML standard that makes HTML4.0 conform to the grammar of XML
XML	Extensible Markup Language: One of the description languages established by W3C

Storage-type broadcasting	Of the terrestrial multimedia broadcasts based on connected segment transmission, these broadcasts are offered by the downloading method
Manifest file	One of files that make up the storage-type broadcasting contents: XML document-type data in which the information on configuration file control and scenario control is described.
Consigning broadcaster	Same as “accredited basic station broadcaster”.
Consigned broadcaster	Same as “basic station broadcast provider”.

Chapter 2 Functions that Basic Receivers Should Have

2.1 Receiver configuration

The basic receivers that receive real-time-type broadcasts are compliant with ARIB STD-B24 “Data coding and transmission specification for digital broadcasting”, Vol. 2, Attachment 4. The following will specify each processing block when the basic receiver is viewed from the hardware configuration and the resources inside the receiver using a reference decoder.

The product planning of receiver manufacturers is not limited, so that the receivers are not equipped with the specified functions, the equivalent functions are achieved using another measure, and the functions exceeding the specifications are provided. However, if any failure occurs due to such planning, it may be assumed that some broadcasters cannot deal with it. When some function is not provided due to specific product planning and it is clear that some failure will occur, it is preferable to make it well known to viewers as specified in this technical data, Vol. 2.

2.1.1 Hardware components

Basic receiver hardware components are shown in Figure 2-2-1.

The digital broadcasting signal input into a basic receiver 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 receivers. For a data stream, the data in the data carousel needs once to be transferred to the main memory or nonvolatile memory, etc. and is then processed by the CPU. In addition to usual video and audio playback processes, it may be possible that characters and graphics are processed, and at the same time, the data in the main memory is transferred to the audio decoder, where audio is subject to the playback process.

For storage of a storage-type broadcast reception function, the storage-type data stream is derived after the transport stream decoding process, once transferred to the main memory or nonvolatile memory, etc. to be processed by the CPU, and is then stored in the removable medium of flash memory.

It is also forecast that more interactive operation than usual television watching occurs by using bidirectional communication lines. From the viewpoint of the above hardware processing operations, it is necessary to specify the following functions:

(1) Data receivable by transport decoders

(2) Playback of the stream system and accumulated audio data

(3) Playback of stream system 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 receivers such as fonts.

(8) Memory size for obtaining and decoding data.

(9) Guidelines for operations by buttons or remote controller

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.

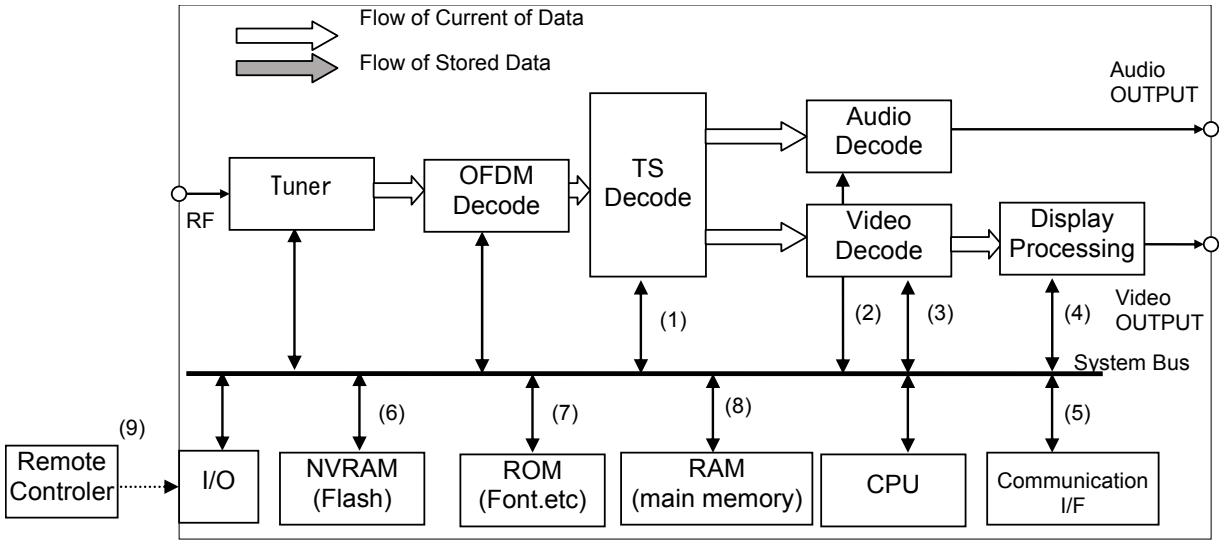


Figure 2-2-1: Hardware components of basic receivers

2.1.2 Receiver reference model

2.1.2.1 Real-time-type broadcast reception function

A reference model of the receiver in real-time-type broadcasting is used to clarify the resources inside the receiver and specify the receivable limits of data broadcasting in basic receivers. The receiver reference model is shown in Figure 2-2-2.

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 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. As for the mono-media content for video systems, on the other hand, the videos, stills, and text graphics are respectively decoded, then allocated in the display areas, and presented via the presentation devices. The presentation of closed captions and superimpose specified in Chapter 3 is a matter of product planning of receivers.

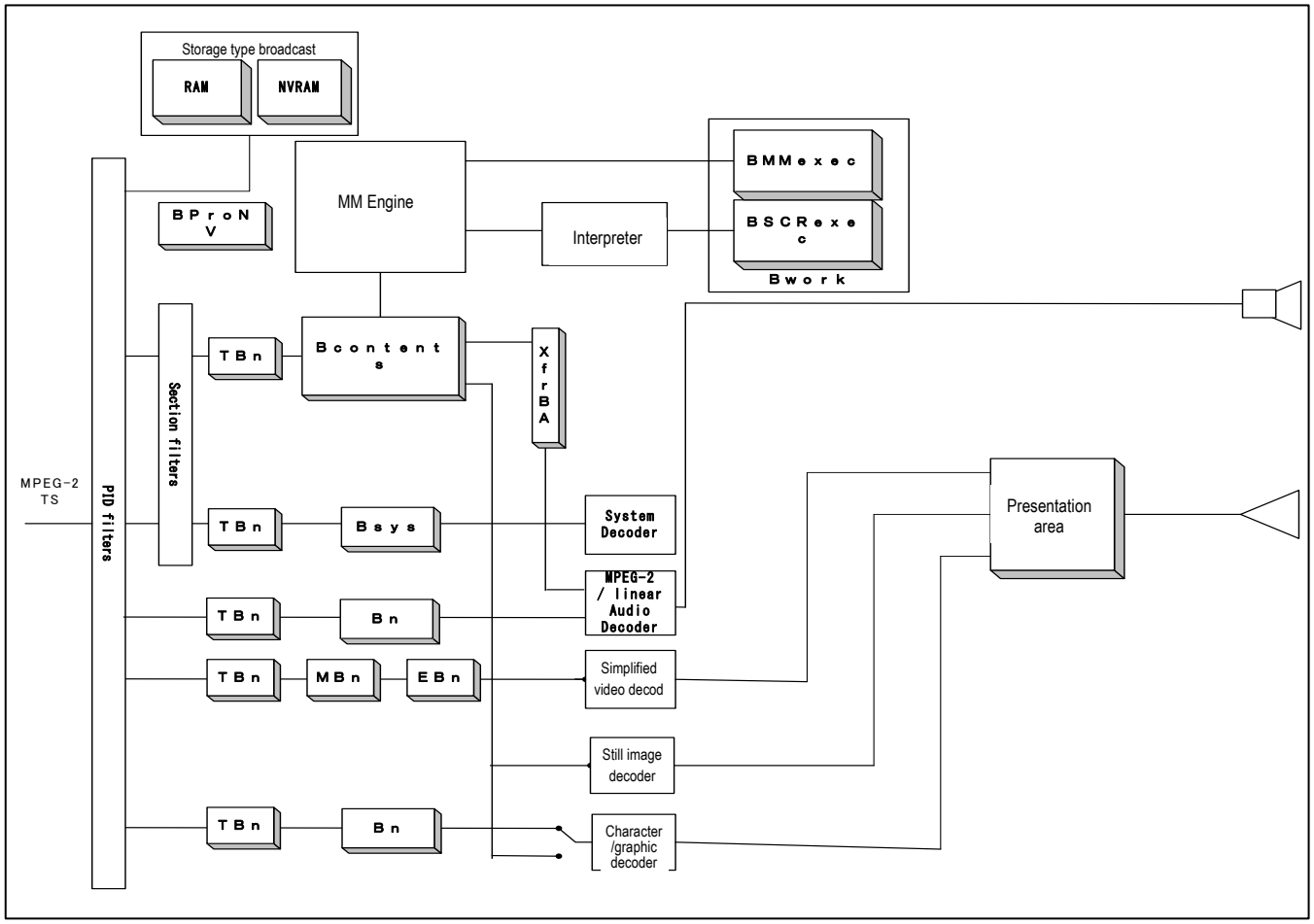


Figure 2-2-2: Reference model of basic receivers

TBn	Transport buffer of the elementary stream *n*
Bn	Primary buffer of the elementary stream inside the decoder.
Bcontents	Buffer of multimedia 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 multimedia code
BSCRexec	Execution memory of script.
Bwork	Multimedia 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.

2.1.2.2 Storage-type broadcast reception function

In the reference model of a receiver in storage-type broadcasting, the received MPEG-2 TS is supplied to the PID filter, where it is received as a ULE data stream, and stored inside the receiver. Refer to Vol. 11 for details.

2.2 Presenting function

The presenting the function of basic receiver's real-time-type broadcast reception functions is modeled by the virtual plane that outputs the decoder results of each display system mono-medium and the display buffers that are dependent on the receiver's display devices. The configuration of display devices is a matter of product planning. For how to present image and data in the display area, follow the guidelines of the presentation shown in ARIB STD-B24 Vol. 2, Attachment 4, section 5.1, and Chapter 4 of these operational specifications.

As for the presenting function of the basic receiver's storage-type broadcast reception functions, basically, no presentation is made during storage, and therefore, it is not defined in this volume. However, it is also assumed that storage of storage-type broadcasting is made during reception of a real-time-type broadcast; in this a case, storage should not affect the real-time-type broadcast reception function. To present the stored contents, see Vol. 2 because ECG is assumed to do so. Playback of stored contents is not defined in this volume, because it is covered by the specifications of the player mounted in the receiver.

2.2.1 Receiver's presenting method

As the method of presenting the real-time-type broadcast reception function, the receiver should present images, closed captions, superimpose and BML contents in accordance with the following procedure.

1. Decode the video stream with a video decoder and output it to the virtual video plane (Y,Cb,Cr 4:2:0 format). Assume that this output result is (A).
2. Decode closed captions and superimpose data with a closed caption decoder and a superimpose decoder respectively, then output them to the corresponding virtual closed caption planes. Assume that this output result is (B). The method of presenting closed caption data and superimpose data at the same time is a matter of product planning. When either one is prioritized for presentation, superimpose should have a priority.
3. Following the direction of presentation (vertical, horizontal), the receiver decodes

the BML contents that fit the virtual plane for the browser for data broadcasting with a data broadcasting browser, and outputs it to the virtual plane for the data broadcasting browser. Assume that this output result is (C).

4. Implement scaling and color conversion for each of outputs (A), (B), and (C) to suit the receiver's own display formats and output them to the receiver's display buffers. It is not essential for the virtual plane for data broadcasting browser and the virtual plane for closed captions to be displayed at the same time. This means that it is not mandatory to operate the data broadcasting browser and closed caption decoder in parallel. For BML contents, displaying part of them is also allowed. At that time, the function to view the voluntary part on the virtual plane should be provided using the receiver's unique scroll function. This is shown in Figure 2-2-4 and Figure 2-2-5.

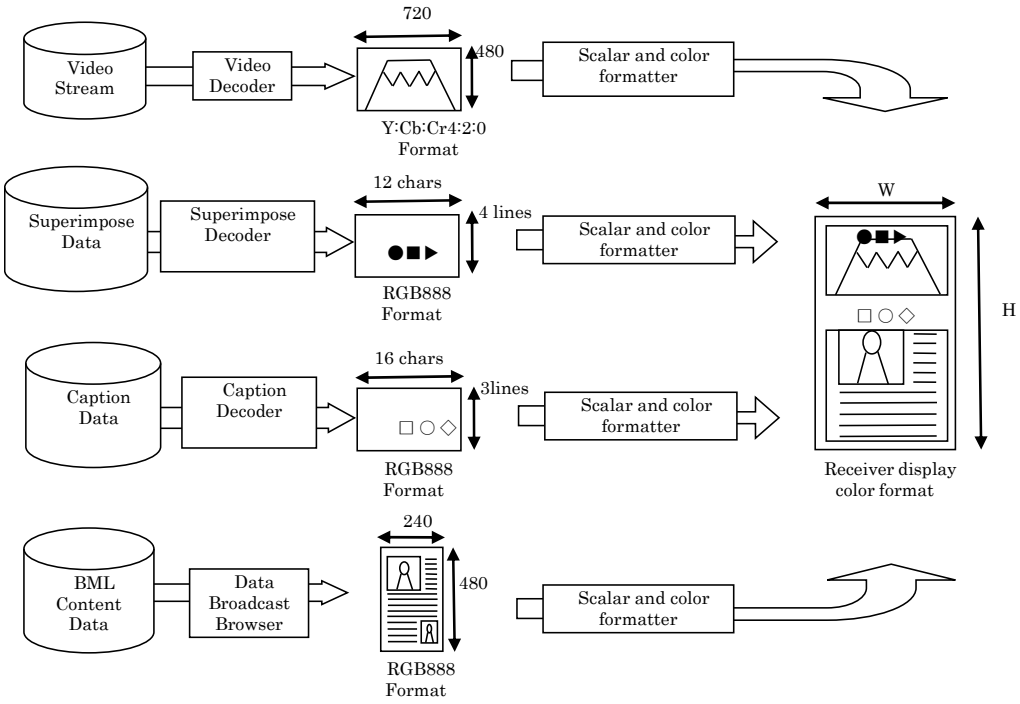


Figure 2-2-3: Receiver's presenting method (Vertical display)

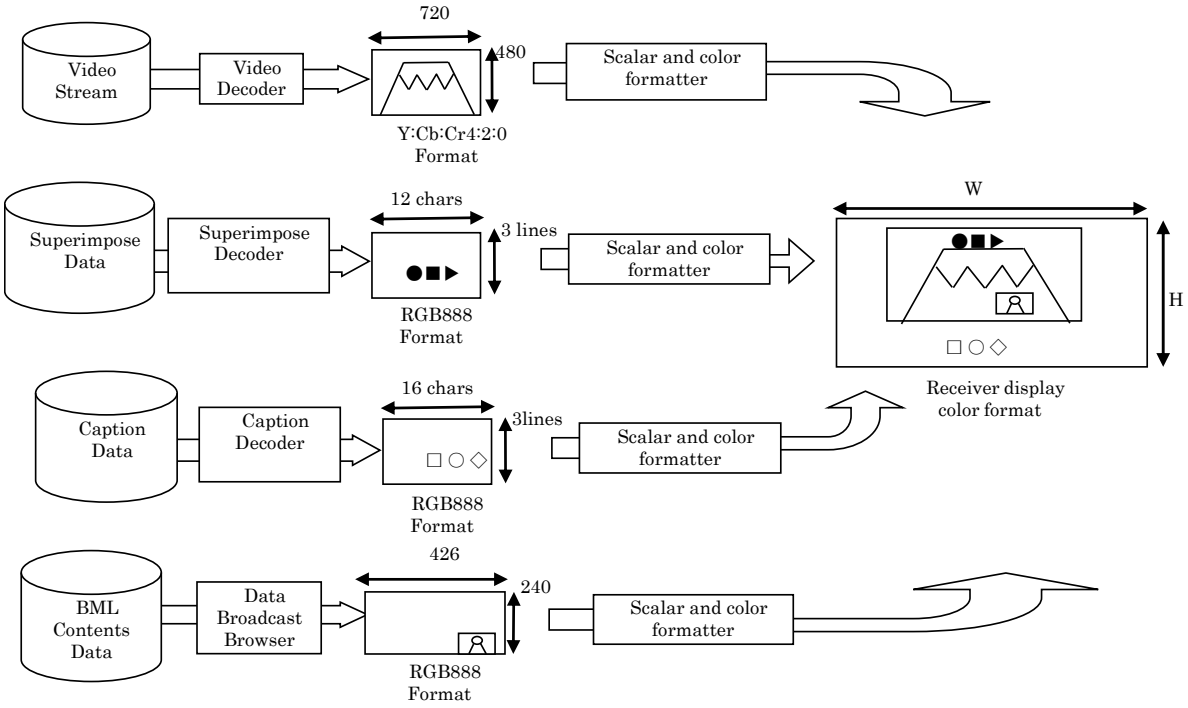


Figure 2-2-4: Receiver's presenting method (Horizontal display)

A specific method for presenting the storage-type broadcast reception function is not described. However, it is also assumed that multiple storage-type contents are handled as one content, and at this time, the function to play back each content from the contents configured by the markup language (HTML, etc.) becomes mandatory.

2.2.2 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 2-1.

Table 2-1: Pixel size of screen planes

Item	Description of provision	
Video virtual plane	pixel size	720 x 480 x 16, YCbCr(4:2:0), 16:9
Virtual plane for the data broadcasting browser	pixel size	240 x 480 x 24, 8 bits for each RGB (vertical display) (*1) 426 x 240 x 24, 8 bits for each RGB (vertical display) (*1) (*3)
Closed caption virtual plane	pixel size	Character direction standard size 12 characters or more x 4 lines, or character direction standard size 16 characters or more x 3 lines, 8 bits for each RGB (*2)(*4)

(*1) For color resolution of pixels, assume that there are some receivers that are expressed with a resolution of 8 bits for each RGB, and when color differences are explicitly expressed in producing contents, it is better to focus on the use of distinctly different colors.

(*2) See section 3.3.3 for details.

(*3) For horizontal display, perform scaling on the virtual plane for data broadcasting and allocate it on the video virtual plane. However, make consideration so that the resolution of the video virtual plane is not degraded during synthesis when the video virtual plane is synthesized.

(*4) As broadcasters generate closed caption data assuming that the virtual closed cation plane has the size of 16 characters by 3 lines, so the recievers, which has the different display sizes, will not be able to exactly reproduce the presentation that the broadcasters intened.

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

Table 2-2: Items to be restricted in presentation on the screen plane

Item	Description of provision	
Video virtual plane	Presentable mono-media code	H.264|MPEG-4 AVC
	Presentation position	Matter of product planning of the receiver for even pixels to odd pixels (*1) on the plane for both the x and y coordinates
	Size	Number of even pixels for both the x and y coordinates
Virtual plane for data broadcasting browser	Presentable mono-media code	JPEG, GIF, aGIF
	Presentation position	From arbitrary pixels to artery pixels for both the x and y coordinates
	Size	Arbitrary number of pixels for both the x and y coordinates
	Overlapping	There is restriction on aGIF and marquee (See Section 3.4.11.)
Closed caption virtual plane	Presentable mono-media code	8-bit code for the X-profile
	Presentation position	Control characters used for closed caption and superimpose are restricted as defined in 3.3.5 of this document. Presentation position indication (SDF, SDP) by content data is not operated, as the behavier of the recievers, which receives these control character, is not guaranteed.
	Size	Medium-size font (but this is a matter of product planning of the receiver)
Presentation switching effect	Process of presentation switching effect is implementation dependent.	

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

2.2.3 Presentable mono-media coding

The following shows the provisions on each coding method for presentable mono-media that

are to be presented by a basic receiver. Details of the operation of each coding method are specified in Chapter 3.1.

Table 2-3 shows the provisions on each coding method for video mono-media that are to be presented by a basic receiver.

Table 2-3: Presentable mono-media to be presented

Coding system		Provisions	
Video coding	H.264|MPEG-4 AVC	Transmission method	Video PES: Stream type identification = 0x1B
		Image size	720x480 (16:9)) (*1)
		Scaling	PRODUCT PLANNING

(*1) Values are in the order of horizontal x vertical.

Table 2-4 shows the provisions on each coding method for video mono-media that are to be presented by the data broadcasting browser on a basic receiver. Basically, it is compliant with the X-profile.

Table 2-4: Presentable mono-media to be presented by the Data Broadcasting Browser

Coding system		Provisions	
Still image coding	JPEG	Transmission method	Data carousel: Stream format identification = 0x0D
		Image size	Max 240x320 (vertical display) (*1) Max 426×240 (horizontal display) (*1)
		Scaling	Product planning. Not specified from multimedia.
Graphic coding	GIF	Transmission method	Data carousel: Stream format identification = 0x0D
		Image size	Max 240x320 (vertical display) (*1) Max 426×240 (horizontal display) (*1)
		Scaling	Product planning. Not specified from multimedia.
	aGIF	Transmission method	Data carousel: Stream format identification = 0x0D
		Image size	Max 240x240 (*1) (*2)
		Scaling	Product planning. Not specified from multimedia.
Text coding	Shift JIS	Transmission method	Data carousel: Stream format identification = 0x0D
		Font	Best if selectable from three types: small, medium and large.
		Miscellaneous	For details of operation, see ARIB STD-B24 Vol. 2, Annex 4.

(*1) Values are in the order of horizontal x vertical.

(*2) Max 240 for both the x and y coordinates. See section 3.2.2.2.11 for details.

Table 2-5 shows the provisions on each coding method for presentable mono-media that are to be presented in closed captions and and superimpose by a basic receiver. The detailed operation of each coding method are specified in section 3.3.

Table 2-5: Presentable mono-media to be presented by closed captions and superimpose

Coding system		Provisions	
Text coding	8-bit code for the X-profile	Transmission method	Independent PES: Stream format identification = 0x06
		Font	Medium (But this is a matter of product planning of the receiver.)

2.2.4 Audio playing function

Table 2-6 shows the provisions on each coding method for audio mono-media that are to be presented by a basic receiver. The details of operation of each coding method are specified in section 3.2.3.

Table 2-6: Audio mono-Media to be presented

Coding system	Provisions	
MPEG-2 AAC	Transmission method	Audio PES: Stream format identification = 0x0F Data carousel: Stream format identification = 0x0D
	Sampling frequencies	24kHz, 48kHz
	Maximum file size that can be played back continually	128KB
Built-in audio	The coding system is a matter of product planning of the receiver.	

2.2.5 Fonts

Table 2-7 shows the provisions on the fonts that are to be mounted in a receiver.

Table 2-7: Fonts

Item	Description of provision
Number of font styles	Matter of product planning of the receiver
Character type	Kanji (1st, 2nd level), Hiragana, Katakana, alphanumeric, symbols, etc. (*1)
Character size	Best if selectable from three types:: small, medium, and large. The number of dots is not specified.
Grayscale font	Matter of product planning of the receiver (not specified from contents)

(*1) For details of the character types, see ARIB STD-B24 Vol. 2, Attachment 4.

2.3 Buttons

2.3.1 Keys to be used in X-profile reception

Keys expected to be used in reception are classified into the following three groups.

- Group 1: Key events handling by BML contents
- Group 2: Key events handling by BML browsers
  (Not handled by BML contents)
- Group 3: To be used for receiver features.

Table 2-8 shows the type of keys to configure each group, contents production, and the guidelines at the moment of designing receivers. 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 the buttons are different according to the implementation, and that at least two different names, such as one name used in receivers and another name used by data broadcasting, are used for a single button. Also, multiple buttons shall provide functions for the same key types. Implementation of the 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 a screen keyboard. It is implementation dependent on which method the receiver adopts; however, this specification is specified for an implementation method in view of simple user operation that is required in the contents. So configuration of receivers with suitable buttons on a remote controller is a mandatory implementation. On the other hand, configuration of receivers without suitable buttons on a remote controller is a permissible implementation on software, but in this specification, it is a mandatory implementation to configure the receiver with suitable buttons on a remote controller; for example, displaying a list of keys on a popup screen selectable by a single user operation should be implemented to allow direct user operation to the extent possible.

Table 2-8: Keys to be used in X-Profile reception

Key type	Mounting		Guidelines
	Mandatory/ Option	Physical button	
Group 1: BML contents receive key events			
Selection	Δ	Δ	Such as “Selection” and “Execution”
Return	Δ	Δ	Such as “Cancel” and “Return” Cancelation of the user operation. Back space of user input character (Or batch

				delete process) (*) A BML document can be used for “Return” purposes; however, give consideration to the existence of the return destination.
	*	Δ	Δ	Selection (execution) of the user operation
	#	Δ	Δ	Selection (execution) of the user operation
	0 - 9 (numeric keys)	Δ	Δ	Numerical input. However, it is assumed in the contents that numerical keys are not allocated to the receiver specification or user setting, and both options of “Execute by up and down focus movement” and “Select directly by numerical keys” are necessary.
	TVlink	Δ	Δ	Writing of TVlink
Group 2: Data broadcasting browser receives key events (does not receive contents)				
	↑,↓ (up and down keys) Jog dial	O	Δ	Movement of scrolling and selection focus
	←, → (left and right keys)	Δ	Δ	Movement of focus
Group 3: Used for receiver functions				
	Monitoring 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
	ECG key	O	O	ECG On-Off function

◎: Mandatory, O: Preferable, Δ: Optional

(*1) These are basic functions in this specifications, however, receivers not implemented with TVlinks are exceptions.

2.3.2 Key masks

Multimedia contents of the X-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, “*” and “#” may be assigned to one-touch tuning numbered 10 or higher, so it is best for assignment and release of key masks on numerical keys, “*”, and “#” to be done simultaneously.

Some receivers are assumed to have a main function other than that of the X-profile receiver, such as mobile phones. In this case, to serve the main function of the device, it is also allowed adaptively that the keys are acquired by other than multimedia contents.

2.4 Memory that should be installed in receivers

Refer to section 2.1.2 for the definition and location within the receivers for each memory in

basic receivers

2.4.1 RAM

As shown in the reference model, a receiver 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 2-9. Refer to section 2.1.2 for details.

Table: 2-9 RAM

Item	Provisions
Bcontents	512KB
XfrBA	128KB

For the buffer size for caption and superimpose, refer to section 3.3.

To receive storage-type contents, it is necessary to provide a memory buffer that can follow the transmission speed, but it is not specified here.

2.4.2 NVRAM

As primary nonvolatile memory featured in receivers related to data broadcasting, there is BproNV which stores unique information on each receiver user and broadcaster which is indicated in the reference model. Table 2-10 shows the type and capacity of the area usable as BproNV. The BproNV area allows not only the packaging that is always secured but the packaging whose necessary area is secured during writing.

Table 2-10: Type and capacity of BproNV

Type	Capacity
Area exclusively for the X-profile broadcaster	128KB
TVlink area	12.8 kB or more (at least 50 events)

For details of BproNV, refer to section 3.4.2. The area indicated in Table 2-10 should, from the confidentiality perspective, be constructed so that viewers cannot read or write using other functions of receivers not defined in standards of related documents or in provisions of this document or by using devices connected to receivers. In order to satisfy the above restrictions, the corresponding areas should be located in the memory of the NVRAM, etc. receiver. Refer to section 2.1.2 for details.

The memory for storing the stored type contents is not referred to here, because it is a matter of product planning.

2.5 Communication functions

Table 2-11 shows the communication functions assumed in bidirectional communications for X-profile data broadcasting service and data supplements stored by storage-type contents.

The communication functions related to mobile phones are dependent on the communication lines (carriers) used, so it is necessary to individually check with communication carriers.

Table 2-11: Communication functions

Item	Provisions
Non-secured communications	The communication functions can be used by designating a name space beginning with http://. These communication functions should allow communications with http servers on the Internet.
Secured communications	The communication functions can be used by designating a name space beginning with https://. These communication functions should allow communications with https servers on the Internet.

- The receivers should package HTTP/1.1 specified in RFC2616.
- The receivers should package Keep-Alive. The header incorporated is a matter of product planning.
- The receivers should package an encryption communication measure of TLS1.0 or higher and/or SSL3.0 or higher.
- It is assumed that Cache-Control or pragma:no-cache is explicitly instructed from the viewpoint of program configuration. Therefore, when Cache-Control or pragma:no-cache is specified for the receiver, it is best that the receiver interprets it and operates. For details of the cache operation, refer to Annex 11, Guideline concerning the cache function and URI history “1. Cache function in the receiver”.

See Annex 12 in this volume for details of the communication function.

2.6 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. The HTML browser (see Chapter 5 in this volume for details) for playback of storage-type contents is not defined here, because it is assumed that a standard character entry function is packaged. This application supports entry of characters into the input element and textarea element of a BML document.

Functions that this application should be equipped with as standard are described in the following sections.

2.6.1 Function specifications

The functions that this application is equipped with are the matter of product planning of the receiver. Only the character types that can be applied to the BML contents are defined.

2.6.2 User interface

Regardless of the used-key-list characteristic values specified by the BML contents that are currently being presented, the character entry application is allowed to exclusively receive key entries.

2.6.3 Character type

For the characters that can be entered into the input element and textarea element, refer to ARIB STD-B24 Vol. 1, Part 2, 7.3 “Shift-JIS character codes”. However, the codes allocated from Row 90 to Row 94 of ARIB-STD-B5 Kanji set are excluded. Refer to section 3.4.9 for details.

2.6.4 Kana Kanji conversion function

The existence of function and function specifications are a matter of product planning.

2.7 TVlink

2.7.1 Introduction of TVlink

Taking into account the functions and loads in packaging required by mobile phone reception, the X-profile does not use the media cross-sectional “bookmarks” specified in ARIB STD-B24 and used by the A-profile, and defines the unique “television link (TVlink).”

2.7.2 What is a TVlink?

This function records information on a link to a site that offers program-related information that has previously been provided in BML contents, when the user presses a button.

The flow of TVlink service is as follows:

1. When there is a communication website that provides information related to broadcast content and a broadcast station offers this service, the broadcast content will present a TVlink icon on the data broadcast browser, according to the timing designated in advance.
2. When the user is interested in the contents of the program (and when the TVlink icon embedded in advance in the broadcast contents is displayed on the screen), the user presses the button to instruct recording of the TVlink. The contents for recording TVlink will record the URI of the communication site that provides relevant information in the NVRAM.
3. The user later designates a desired TVlink from the TVlink list screen that is provided as a reception function and browses the communication contents on the site that has

relevant information.

It is assumed that the contents for recording TVlink will be broadcast as a part of program-associated contents or CM contents.

An external communication server is assumed for the link destination recorded as a TVlink.

2.7.3 Operation of NVRAM in TVlink

2.7.3.1 Number of TVlink areas

Although the number of recordable TVlinks is a matter of product planning, it is preferable that there are at least 50.

2.7.3.2 Sharing with other media

The NVRAM area that records TVlinks may share with bookmark record services of other media that are compatible with the provisions on the types of TVlink, types of parameter, data length, and need for recording.

2.7.4 TVlink format and operation

2.7.4.1 TVlink format

The parameters used by TVlink and the length of its data, necessity of writing to NVRAM, and necessity of displaying the TVlink list function of receivers are shown in Table 2-12. The maximum required number of bytes for the NVRAM area necessary for one 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 the BML contents.

Table 2-12: Data configuration of TVlinks

Parameter	Length (byte)	XproBMtype (*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
XproBMtype	1	O/Δ	O/Δ	O/Δ	O/Δ	O/Δ	
expire	5(*3)	◇/◇	◇/◇	◇/◇	◇/◇	◇/◇	YYYYMMDDHHmm format
broadcaster_id	2	Δ/ X	O/ X	Δ/ X	Δ/ X	Δ/ X	
service_id	2						
reserved	10						

[Write/Display] O: Mandatory, ◇: Preferable, Δ: Product planning, X: Prohibited

(*1) The existence of support for every XproBMtype depends on the product planning of the receiver.

(*2) The characters used in URI designation are specified in ARIB STD-B24 Vol. 2 section 9.2

(*3) Assumed to be recorded as numerical values.

2.7.4.2 Operation of each data item

Operation of title:

- Record the character string displayed in the title column of the TVlink list.
- Up to 40 bytes (not including length) should be usable for a title. The operation when specified exceeding 40 bytes is a matter of product planning. When 40 bytes are exceeded, it is better for a receiver to record up to 40 bytes, not recording the 41st byte and later. If the 40th byte is the first byte of a 2-byte character, it is preferable not to record the character.
- Recording this information into NVRAM is mandatory.

Operation of the link destination URI (dstURI):

- Record the URI of the communication contents at the link destination.
- Up to 60 bytes (not including length) should be usable for the URI at the link destination. When 60 bytes are exceeded, the receiver does not make a recording, and returns NaN as a return value.
- Specify an empty character string for memo information (XproBMtype=0).
- Recording this information into NVRAM is mandatory.

Operation of TVlink outline (outline):

- Record the outline of the communication contents at the link destination.
- Specify an empty character string if there is no TVlink outline.
- Up to 130 bytes (not including length) should be usable for a TVlink outline. The operation when specified exceeding 130 bytes is a matter of product planning. When 130 bytes are exceeded, it is better for a receiver to record up to 130 bytes, not recording the 131st byte and later. If the 130th byte is the first byte of a 2-byte character, it is better not to record the character.
- Recording of a TVlink outline into NVRAM is mandatory when the link type is memo information (XproBMtype=0) and for a TVlink of X-profile link contents (XproBMtype=1). Recording of a TVlink outline is optional. However, it is better to record a TVlink outline to enhance the operability for users.

Operation of TVlink type (XproBMtype):

- The TVlink type shows the type of contents at the link destination. The receiver can use this information to select the browser that can display the communication contents designated by the URI at the link destination.

- Table 2-13 shows the correspondence of TVlink types.
- The TVlink types are designated by the numerical value of 0-255.
- Recording this information into NVRAM is mandatory. Displaying the information should be optional.

Operation of expiration dates (expire)

- For the expiration date, the last date on which the TVlink is valid is recorded.
- Always designate the year, month, day, hour, and minute for an expiration date.
- The argument of record function (X_TMM_writeXproBM()) can be omitted. When omitted, the TVlink is always valid.
- Recording this information into NVRAM should be optional.

2.7.5 Operation of the TVlink type

Table 2-13 shows the types of TVlink designated by the TVlink type.

Table 2-13: TVlink Types

	TVlink type
0	Memo information (no link destination)
1	X- profile link contents (BML contents)
2	X-profile non-link contents (BML contents)
3	HTML contents
4	Communication contents when assuming that a specific network is used
5 to 255	Reserved

Operation of TVlink types is as follows:

Operation of memo information (XproBMtype: 0)

- This type is used for the application of memo recording of text information related to the broadcast program.
- For this type, the link destination URI is not specified. An empty character string is specified for the argument of the record function (X_TMM_writeXproBM()). If a character other than an empty character string is specified, recording will fail.
- Recording and reading of TVlink of memo information (XproBMtype=0) is mandatory.

Operation of X-profile link contents (XproBMtype: 1)

- This type is specified when recording information of the link to X-profile link contents (BML contents) that are assumed to be presented by the data

broadcast browser.

- The specified X-profile link contents are limited to the BML contents for vertical display.
- Recording and reading of TVlink of X-profile link contents (XproBMtype=1) is mandatory.

Operation of X-profile non-link contents (XproBMtype: 2)

- This type is specified when recording information of the link to X-profile non-link contents (BML contents) that are assumed to be presented by the X-profile communication browser.
- The specified X-profile non-link contents are limited to the BML contents for vertical display.
- Recording and reading of TVlink of X-profile non-link contents (XproBMtype=2) should be optional. It is better that the receiver that is equipped with the X-profile communication browser can record TVlinks of this type.

Operation of HTML contents (XproBMtype: 3)

- This type is specified when recording information of the link to the HTML contents that are assumed to be presented by the HTML browser that is, for example, mounted in PDA, etc.
- Recording and reading of TVlink of HTML contents (XproBMtype=3) should be optional. It is preferable that a receiver that is equipped with the HTML presentable browser can record TVlinks of this type.

Operation of communication contents (XproBMtype: 4) using specific network

- This type is specified when recording information of the link to the communication contents that are assumed to be presented by the HTML browser that is supposed to use a specific network mounted on a mobile phone, etc.
- Recording and reading of TVlink of communication contents (XproBMtype=4) using a specific network should be optional. It is better that a receiver of the mobile phone provider, etc. that is equipped with a unique browser assumed to use a specific network can record TVlinks of this type. On the other hand, it is preferable that a receiver that does not have a unique browser assumed to use a specific network does not record TVlinks of this type.

2.7.6 Guidelines for function operation and receiver operation of every TVlink

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_TMM_writeXproBM()) of each TVlink type.

In case of memo information (XproBMtype=0):

(Function operation during recording)

- When this type is specified for the argument, recording is made unconditionally.
- If a character other than an empty character string is specified for the argument (link destination: URI), recording will fail.
- If the TVlink area of NVRAM that records cannot be secured, -1 will be returned as a return value.

(Receiver operation during calling)

- When this type is selected from the TVlink list by user operation, the TVlink outline is only displayed, and no transitional operation is performed.

In case of X-profile link contents (XproBMtype=1):

(Function operation during recording)

- When this type is specified for the argument, the receiver links the broadcaster_id and service_id of the relevant broadcast station to individual TVlinks when recording TVlinks and stores them.
- If the TVlink area of NVRAM that records cannot be secured, -1 will be returned as a return value.

(Receiver operation during calling)

- When this type is selected from the TVlink list by user operation, the following operation is performed:
  - Acquire the contents designated by URI via communications and present them in the data broadcast browser in link status.
  - Use the stored broadcaster_id to control access to NVRAM.
  - In this case, it is better for the data broadcast browser to be a full-screen display.
  - It is better that the time handled by the Date object of ECMAScript is TOT that the receiver has acquired last or the time calibrated by other measures, but an error may be returned.

In case of X-profile non-link contents (XproBMtype=2):

(Function operation during recording)

- If the receiver does not package a X-profile communication browser, recording is not made, in principle, and -2 will be returned as a return value. However, this is not applied when packaging of the X-profile communication browser cannot be cognized from the data broadcast browser.
- If the TVlink area of NVRAM that records cannot be secured, -1 will be returned as a return value.

(Receiver operation during calling)

- When this type is selected from the TVlink list by user operation, the X-profile communication browser is started, and the X-profile non-link contents are presented. The operation of the receiver that cannot start the X-profile communication browser is a matter of product planning.

In case of HTML contents (XproBMtype=3)

(Function operation during recording)

- If the receiver does not package a browser that can present HTML, recording is not made, in principle, and -2 will be returned as a return value. However, this is not applied when packaging of HTML browser cannot be cognized from the broadcast browser.
- If the TVlink area of NVRAM that records cannot be secured, -1 will be returned as a return value.

(Receiver operation during calling)

- When this type is selected from the TVlink list by user operation, the HTML browser is started, and the HTML contents are presented. The operation of the receiver that cannot start the browser that can present HTML is a matter of product planning.

In case of specific network communication contents (XproBMtype=4)

(Function operation during recording)

- If the receiver is not equipped with a browser that can present the contents that are assumed to use a specific network of mobile phones, recording is not made in principle, and -2 will be returned as a return value. However, this is not applied when packaging of the relevant browser cannot be cognized from the broadcast browser.
- If the TVlink area of NVRAM that records cannot be secured, -1 will be returned as a return value.

(Receiver operation during calling)

- When this type is selected from the TVlink list by user operation, the browser that can present the contents that are assumed to use a specific network is started, and the communication contents are presented. The operation of the receiver that cannot start the relevant browser is a matter of product planning.

2.7.7 Recording TVlinks

- Recording of TVlinks into the NVRAM is performed from the broadcasting contents or X-profile link contents.
- The TVlinks are always recorded based on user instruction. Automatic writing of contents without user instruction is prohibited.
- When TVlinks are recorded by specifying the optional XproBMtype, it is better that the contents are used according to whether the receiver can handle the relevant XproBMtype before displaying the TVlink.
- From the viewpoint of personal information protection, the TVlinks of X-profile link contents (XproBMtype=1) must not package the recording or updating function using the receiver's function. It is better that the TVlinks of other types do not package the recording or updating function either using the receiver's function.
- The following shows the receiver operations when the contents that record TVlinks call the record function (X_TMM_writeXproBM()):
  - The receiver uses the TVlink type specified by the argument and judges whether it can present the communication contents. If the receiver judges that it cannot present them, it is better not to record them. In this case, the return value of the function is -2. Also, it is preferable for the receiver to show the reason that it cannot record the contents to users.
  - If there is no empty TVlink area, the following TVlinks previously recorded can be deleted and recording can be made. In this case, it is better to present the title of TVlink to be deleted to users and obtain the users' permission.

    (1) Expired TVlinks

    (2) Oldest TVlinks

    When recording a TVlink has succeeded after obtaining the user's permission, "1" will be returned as the return value of the function. If user permission cannot be obtained, fail (NaN) will be returned.

    For a receiver that does not perform the deletion process like this when there is no empty TVlink area, "-1" will be returned as the return value.

2.7.8 Guidelines for the TVlink list function

It is mandatory to package a "TVlink list" that shows TVlinks on a list and has the function of transferring to the desired communication contents by the user's selection. The following shows the functions. The items not shown are a matter of product planning.

2.7.8.1 Starting and ending the TVlink list using the receiver's application

- The TVlink list is started with the receiver's application by executing the receiver function of the button, etc. on the remote controller or the function of multimedia contents (X_TMM_startResidentApp()). This function will be described later.
- To start the TVlink list with the receiver's application while the multimedia contents are being presented, the display of the multimedia contents may be interrupted. However, it is preferable to continue the operation of multimedia contents.
- The method of displaying the TVlink list is, for example, superimposed on the data area. At this time, it is desirable to continue the display of the video.
- When producing contents, be careful that the data broadcasting browser does not acquire a key input while the TVlink list is being displayed.
- When the TVlink list is displayed with the presentation of multimedia contents interrupted, and a TVlink is selected by the user and the TVlink list ends without transferring to other contents, it is better that the multimedia contents whose display has been interrupted are displayed again, and the operation continues. The key input is also returned to the multimedia contents.
- If the TVlink list is started while the multimedia contents are being presented, the screen presented when the TVlink list ends and the multimedia contents reappear depends on the operation of the TV broadcasting browser executed during presentation of the TVlink list because the TV broadcasting browser continues operation asynchronously with the TVlink list even while the list is being presented. For example, if the data event changes while the TVlink list is being displayed, the display contents of the data broadcast when the TVlink list is started are different from those when the TVlink list is ended. Even if the data screen is displayed, the data broadcast may not be presented when the TVlink list display is ended.

2.7.8.2 Displaying information of TVlink

- The TVlink list has a function of displaying a list of TVlink titles.
- The display "link destination URI" is optional.
- The "TVlink outline" does not need to be displayed on a list: It is acceptable when the list can be displayed by user button operation.
- The "expiration date" display is optional, but it is better it can be displayed.

- Displaying the information shown by “TVlink type” is optional.
- It is better that the “bookmarks” recorded with a communication browser other than data broadcasting browsers are clearly distinct when viewed by users. It is necessary to exercise special care when the TVlinks and the bookmarks recorded by a communication browser are displayed using one application (especially, the TVlinks of X-profile link contents (XproBMtype=1)).
- It is desirable that the items whose TVlink type is memo information (XproBMtype=0) in a list are displayed so that user can understand that they do not have link destination.
- The other specifications on display are a matter of product planning.

2.7.8.3 Selection of TVlink and transition to the link destination

- A user can select a desired TVlink from a TVlink list, and display the communication contents shown for the link destination URI that is included in the selected TVlink.
- At this time, the user refers to the TVlink type described in the TVlink if necessary, starts the browser that can present the communication contents of the specified type, and displays the relevant communication contents.
- If a TVlink whose expiration date has passed is selected by a user, it is better that the user is clearly shown that the expiration date has passed and that transition to the specified communication contents is not made. It is better at this time that the TVlink is deleted by obtaining the user’s permission.

2.7.8.4 Erasing information of TVlink

- TVlinks are erased using the TVlink list function (receiver function). Because they cannot be erased from the contents, it is mandatory to package the erase function.
- Erasing TVlinks should be based on the user’s will.
- It is better that the TVlinks with an expiration date are automatically erased with the user’s permission. (*) The timing for executing automatic erasure is a matter of product planning.
- In addition, the specifications of the erase function user interface, etc. are a matter of product planning.

* Because recording the expiration date is optional, it is necessary to be careful that the TVlinks whose expiration date has passed are not erased depending on the receiver even if the expiration date is specified during writing. In this case, the TVlinks once recorded are stored in the receiver until the user erases them. Therefore, the broadcaster needs to exercise care even in this case so that the details of the communication coontents at the link destination do not collapse.

2.8 Data broadcasting browser

As for the functions with which the data broadcasting browser is equipped, this chapter has specified the hardware resources and software resources. The transmission, mono-media, and multimedia coding will be specified in the following chapters, and in this section, the specifications that are not included in these items will be defined.

2.8.1 Browser-unique display

The browser must not perform a browser-unique display that may cover over the details that the browser intends to present. This is a restriction that, for example, avoids the occurrence of display loss of contents due to the URI display frame that is generally used by the browser for the computer and the location of buttons for browser operation. However, this restriction does not mean to eliminate the display of channel banners and permissions while the relevant browser is operating.

2.9 Communication browser

It is assumed that the receiver that receives the video, audio, and multimedia data specified in this chapter has an optional communication browser as well as the data broadcasting browser defined in this section, etc. The following specifies the applications of each browser and the relationship between them.

1) Data broadcasting browser

The BML browser is a browser that operates according to the provisions of the X-profile defined in this document. This browser operates in data broadcast reception status and link status, and is used for the application that presents the contents provided when a broadcaster performs services as a broadcasting industry via the broadcasting signal and communication lines.

2) Communication browser

Communication browsers are divided into the X-profile communication browsers and communication carrier-specific browsers as shown below. In this document, a communication browser is positioned as one of the receiver's applications from the viewpoint of the data broadcasting browser. Therefore, the start from a data broadcasting browser uses the same method as the start of other applications (start by X_TMM_startResidentApp()) specified in Chapter 4. This browser is used for the application that presents the contents allocated on the server offered as a service other than broadcasting.

a) X-profile communication browser

This browser has specifications that include the X-profile BML-related provisions (elements, attributes, CSS, DOM, ECMAScript, etc.) specified in this document. However, executing a part of the augmented functions is restricted. Also, reference to and transition of broadcasting contents is prohibited (see Annex 9).

b) Communication carrier-specific browser

A browser packaged based on the specifications set down by communication carriers is assumed. This document specifies only the range of use as the receiver's application of the X-profile receiver, and does not make any internal prescription.

c) HTML browser

A browser packaged to view HTML contents is assumed. This document specifies only the range of use as the receiver's application of the X-profile receiver, and does not make any internal prescription. This browser can be used in common with a browser (see Chapter 6) used to play back storage-type contents (HTML contents).

The figure below shows the relationship among each browser.
The HTML contents can be presented by the HTML browser mounted on the receiver, as well as the X-profile communication browser and communication carrier-specific browser shown in the figure.

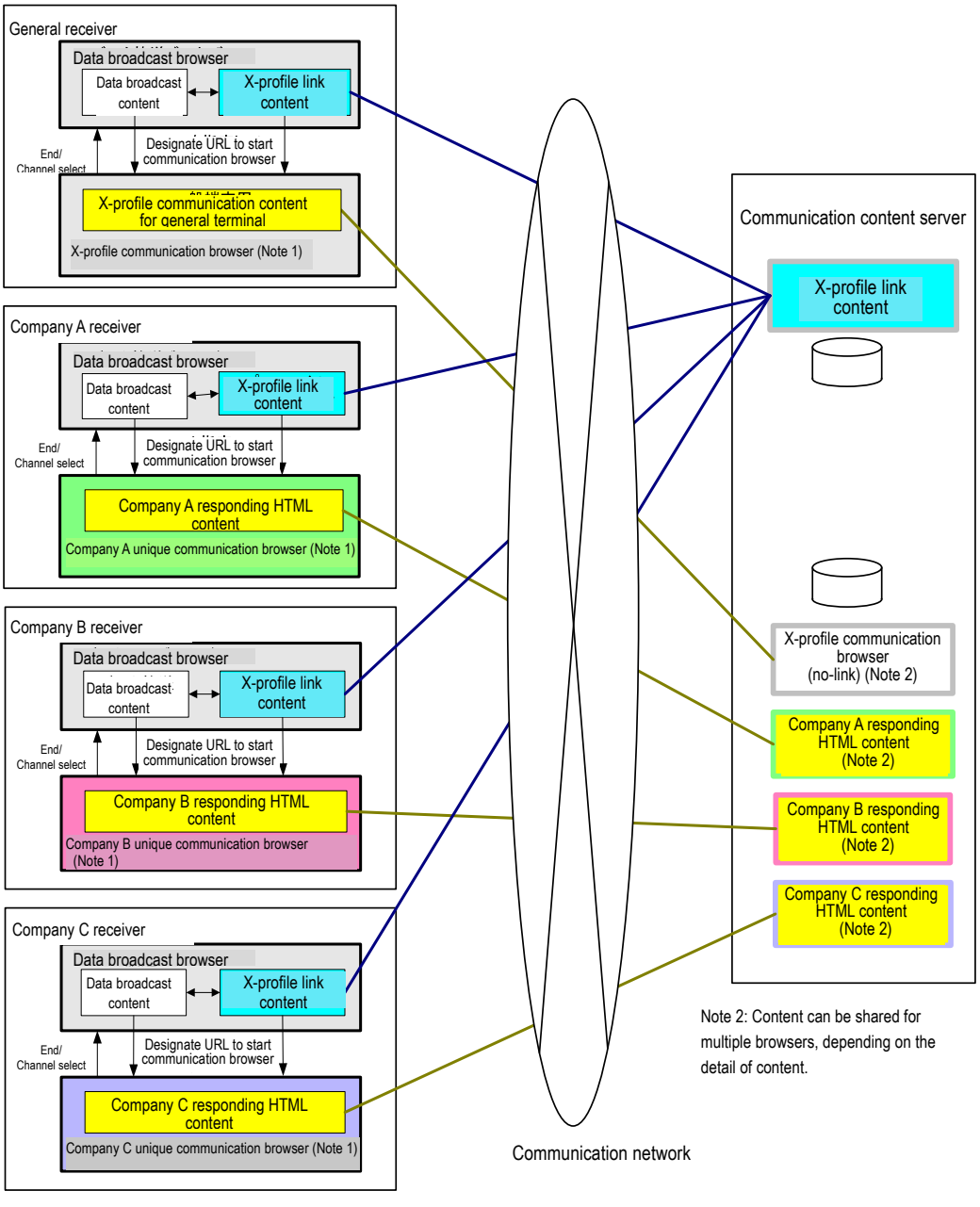


Figure 2-2-5: Relationship between data broadcasting browsers and communication browsers

2.10 One-segment data broadcast coalition function

It is assumed that the receiver can receive not only multimedia broadcasts but also 1-segment broadcasts. However, the coalition functions of one-segment broadcasts and multimedia broadcasts will not be operated for the time being.

Chapter 3 Operation of Rea-time Type Broadcasting Services

3.1 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”

3.1.1 PSI/SI

3.1.1.1 Types of data broadcasting services

3.1.1.1.1 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.

3.1.1.1.2 Types of data broadcasting service programs

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

Table 3-1: 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	Programs with a Data Component Descriptor located in the first PMT 2nd loop component.

3.1.1.1.3 "service_type" of channels that operate data broadcasting programs

1) TV programs with additional data are broadcast by service_type=0xC2 (TV services) channels.
2) Independent data programs are broadcast by service_type=0xC2 (data services) channels.

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

3.1.1.2.1 Contents and local contents

The relationship between events and definitions of contents and local contents are shown in Table 3-2 and Figure 3-3-1.

Table 3-2: 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.

Time

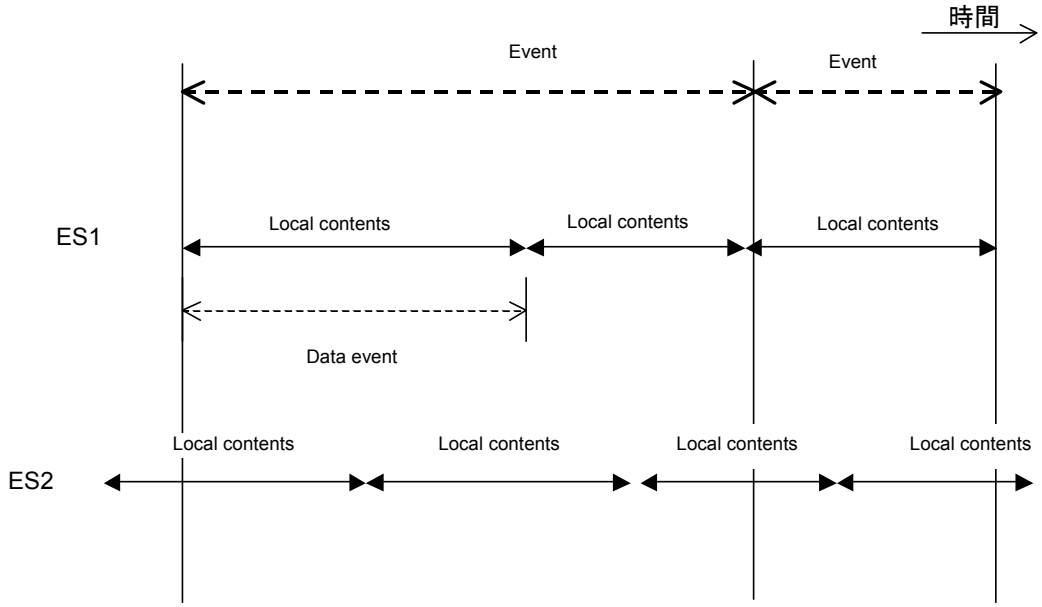


Figure 3-3-1: Local contents and events

3.1.1.2.2 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.

3.1.1.2.3 Operation of component tags

For the operation of the tag values of the components used, follow Vol. 7, section 3.1.2.

3.1.1.2.4 Configuration of ES to be transmitted per service

- Video: 1ES maximum
- Audio: 2ES maximum
- Data carousel: 2ES maximum
- Event message: 2ES maximum (not operated for NPT)
- Closed caption: 1ES maximum
- Superimpose: 1ES maximum
- PCR: 1ES maximum

3.1.1.2.5 Identification of entry components

The default components with tag values of 0x40 are always transmitted by the data broadcasting program. The identification of entry components, which transmit startup documents on tuning, 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. Therefore, the entry module is always configured in the entity format. The <resouceNames> of two startup documents are assumed to be startup.bml (vertical display) and startup_landscape.bml (horizontal display).

3.1.1.2.6 Identification of event message components

When a general-purpose event message is transmitted using the components that transmit the data carousel, the components with tag values 0x40 and 0x41 are used. When a general-purpose event message is transmitted with ES other than the components that transmit the data carousel, the components with tag values 0x50 and 0x51 are always used, and 0x50 and 0x51 are assumed to be ES exclusively for general-purpose event messages.

3.1.1.2.7 Detailed regulation when transmitting section data

When a data carousel or event message is transmitted, multi-section transmission (two or more sections are transmitted with 1 TS packet) is not performed.

The following standards are established for the transmission bit rate when each section data of data carousel and event message is transmitted:

- Six or more TS packets are not transmitted continually with the same PID.
- The total bit rate of the components (4PID maximum) received simultaneously when one content is received is 650 kbits/s or less. The components include DII, DDB and event messages.
- The maximum bit rate per subtable shall also be 650 kbits/s (2 kB±100% per 32 ms) or less.
- It is preferable that transmission is performed without exceeding these standards. If the above standards are exceeded for transmission, it is assumed that the reception efficiency of section will drop, depending on the receiver, and the time required to acquire the necessary section will be longer.

3.1.1.2.8 Default maximum bit rate of a data broadcasting program

Follow Vol. 7.

3.1.1.3 Operation of PMT

3.1.1.3.1 Operation of the PMT specific to data broadcasting services

Refer to section 3.1.1.5.2 for the receiver operation when PMT is updated.

It is described in PMT that the following three components are allowed, but ES does not exist:

Component that transmits the data carousel

Component that transmits closed caption and superimpose

Component that transmits only event messages

However, if ES of the data carousel does not exist, although it is described in PMT, the receiver tries to acquire data, so the display showing that data is being received may continue for a certain period of time. To avoid this state, it is preferable that the transmission side continues to send out an empty carousel in at least 0x40 components even though there is no data to be transmitted by the transmission side. See Annex 5.

3.1.1.3.2 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 section 3.3 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.

Operation of Data Component Descriptor is shown in Table 3-3.

Table 3-3: Operation of the Data Component Descriptor

Flag	Operation
data_component_id	0x1B
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	Always auto_start_flag=1
document_resolution	Always 1111
use_xml	・It is always 1 only for component of component_tag= 0x40
default_version_flag	(component of component_tag= 0x40 that transmit modules including documents that should be started first when data broadcasting
independent_flag	programs are selected.) It is 0 for any other components.
style_for_tv_flag	Not operated. Fixed at 0.

Flag	Operation
bml_major_version, bml_minor_version	Operated as in the standard. If bml_major_version is 14 when these fields are allocated, the mobile phone basic receiver judges that viewing is possible. 0 is always specified for bml_minor_version. See Annex 6 for the receiver operation for this version.
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.

3.1.1.3.3 Target Resion Descriptor in the PMT

The Target Resion Descriptor is not used.

3.1.1.4 Operation of Data Contents Descriptor in W-EIT

Data Contents Descriptor in W-EIT is not operated.

3.1.1.5 Performance of related receiver

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.

The entry component (0x40) may not exist. However, while it does exist, the PID value does not change. The receiver operation when the PID value changes is specified in section 3.1.1.5.2.

3.1.1.5.1 Receiver operation when data broadcasting programs start

1. When the PMT second loop includes the component with component_tag=0x00, the receiver regards it as a video stream and plays it back. When it includes the component with component_tag=0x10 or 0x11, the receiver regards it as an audio stream and plays it back.

2. The receiver identifies the entry component (component with component_tag=0x40) among the components allocated in the PMT second loop.

3. When an entry component (0x40) exists in the PMT second loop, and data_component_id of the Data Component Descriptor is 0x1B, the receiver judges it as a data broadcasting program using the multimedia coding format based on XML, and proceeds with the process shown below. If the receiver does not respond to the data coding format, no data broadcasting service starts.

4. 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.

5. 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.

6. However, if the entry component has an empty carousel after the data broadcasting engine starts, the receiver continues to monitor the data event switching of the entry component. When data event switching occurs and the startup module appears, the receiver acquires and presents the startup document (see section 3.1.2.3 for the empty carousel).

3.1.1.5.2 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.

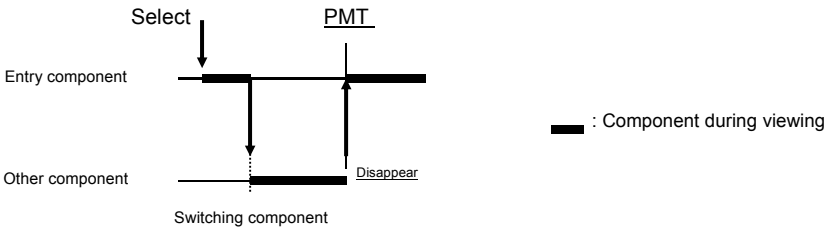


Figure 3-3-2: Behaviour when the components during viewing disappear due to the PMT update

When the entry component (tag value = 0x40) disappears:

→ Since it is no longer data broadcasting program, the BML-engine ends. However, Annex 5 is recommended for the operation when presenting link contents.

When the entry component (tag value = 0x40) appears:

→ If the data broadcasting engine does not start up, the receiver starts the data broadcasting engine, and performs the operation equivalent to during channel selection.

When PID of the component that transmits the data carousel (including an empty carousel) being viewed is changed:

→ The receiver should judge that the content of data broadcast has changed, and perform the process equivalent to re-selection of the channel.

・ Disposes of the document being presented.

・ Erases all the acquired resources from the BContents memory.

・ Reacquires the startup document of the entry component and represents it.

3.1.1.5.3 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 3-4 should be output along with it. (As specified in Vol. 2)

Table 3-4: Descriptor outputting to partial TS

Descriptor name	Written table	Location standard
Data Component Descriptor	PMT 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 multimedia contents are being played with partial transport stream as input depend on models. For example, the following are examples of extended functions for broadcasting.

- Selecting station (epgTune), reservations (epgReserve)

3.1.2 Data carousel transmission operation

3.1.2.1 Data carousel transmission operation

Maximum number of modules sent per 1 data carousel is 64.

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.

3.1.2.1.1 Introduction 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.

Data events are identified by “data_event_id” of DII.

3.1.2.2 Operation of data events

3.1.2.2.1 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 3-3-2)

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 3-3-2)

Manage/update “data_event_id” by each component. “data_event_id” is operated by values other than 15 (0xF).

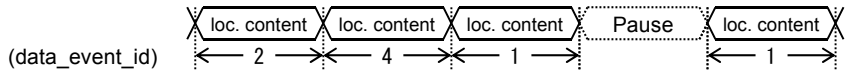


Figure 3-3-2: Local Contents and “data_event_id”

3.1.2.2.2 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 3.1.1.2.2)

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.

3.1.2.2.3 Receiver basic behavior during data broadcasting program presentation

The receiver will observe constantly DII of all the data components transmitted.

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 launchDocument() / reloadActiveDocument() / epgTune() / quitDocument() function is used in the "DataEventChanged" event handler then do not obtain the startup document and follow the specifications of the function.

3.1.2.2.4 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.

3.1.2.3 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 Annex 5.

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 3.1.2.4) is applied.

A newer version of DII may be occurred while transmitting an empty carousel. Moreover, an update of the data event might be occurred, 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.

3.1.2.4 Operation of the DownloadInfoIndication (DII) message

When presenting multimedia contents, the reception of DII of all carousels is necessary.

Refer to section 3.1.2.1.1 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 3-5.

Table 3-5: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 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	

Field	Operation	Remarks
	• 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 dependently.
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 section 3.1.2.3 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 section 3.1.2.5 for details.	
Module information area	The descriptor described later is allocated.	

Field	Operation	Remarks
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	Operated.	
Control Descriptor	Not operated.	
RootCertificate Descriptor	Not operated.	

3.1.2.5 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 Annex 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 3-6.

Table 3-6: Operation of DDB: downloadDataMessage()

Field	Operation	Remarks

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.

3.1.3 Operation of event messages

Only general event messages are operated.

3.1.3.1 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.

3.1.3.2 Transmission of general event messages

The general event message is transmitted by components with component tag values 0x40, 0x41, 0x50, and 0x51. It transmits by components with a component tag value = 0x50 and 0x51 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 Data Component Descriptor 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 specifying the 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 fully that the receiver may not be able to acquire the event message when authoring contents.

3.1.3.2.1 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 3.1.3.2, 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 3-3-4.

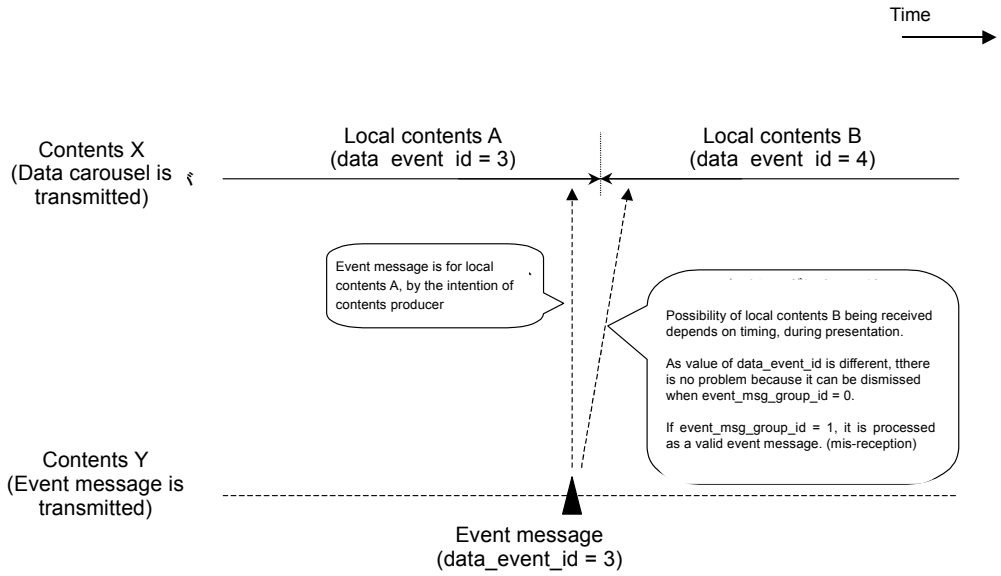


Figure 3-3-3: Mis-reception of general event messages

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 3-3-5).

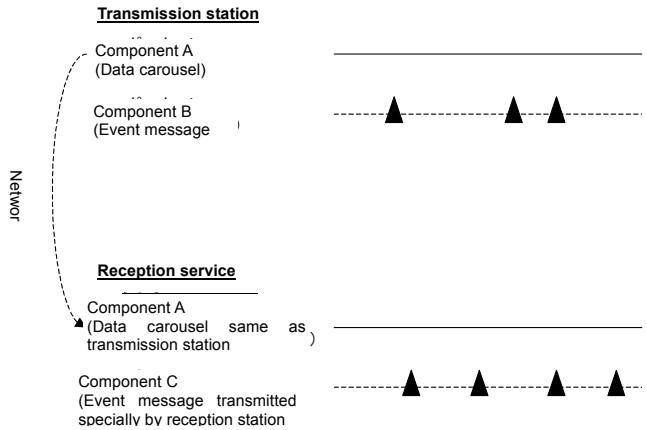


Figure 3-3-4: When the data_event_id values are not unified

3.1.3.2.2 Regarding component specifications of general event messages from the BML document

Among the four components shown in section 3.1.3.2, 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.

3.1.3.2.3 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 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.

3.1.3.3 Operation of DSMCC_section()

Table 3-7: 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.	

3.1.3.4 Operation of the General Event Descriptor

Table 3-8: 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.	

Field	Operation	Remarks
private_data_byte	Operated as specified.	Max. 244 bites.

3.1.4 Receiver performance

3.1.4.1 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. (See "Chapter 5 Guidelines for browser behaviors (operation of motion control function")
- It is enabled the monitoring of the ModuleUpdated for the module of the component that transmits all carousels.

Table 3-9: Filtering resources required for acquiring the data carousel

Object ES	PID filter number	Section filter number	
		DII	DDB
ES of tag value =0x40	1	1	1
ES of tag value =0x41	1	1	1

Filtering resources necessary for general event messages

Required transmission operations

The general event message is transmitted by components with component tag values of 0x40, 0x41, 0x50 and 0x51. When only the general event message is transmitted, transmit them by components with component tag values of 0x50 and 0x51.

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 3-10: Filtering resources required for acquiring the general event message

Object ES	PID filter number	Section filter number
ES of general event message tag value =0x50	1	2
ES of general event message tag value=0x51	1	2

3.1.4.2 Best operation of the receiver in relation to the acquisition of modules

The receiver unconditionally acquires the transmitted modules to the extent possible, precaches them, and can thus improve the response of presentation. 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.

3.1.4.3 Receiver operation concerning updating the version

Refer to Annex 6 for the receiver behavior receiving the major_version excluding 14.

3.2 Operation of mono-media coding

3.2.1 Image coding

Follow Vol. 7, section 2.1.1.

3.2.2 Still picture and bitmap figure coding

3.2.2.1 JPEG

- In conformity with the BaseLine method of ISO/IEC 10918-1(ITU-T T.81).
- For the colorimetry of JPEG, ARIB STD B24 Vol. 1, Part 1, 7.2 "Colorimetry" is applied.

3.2.2.1.1 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.

3.2.2.1.2 Other restrictions

- Do not operate progressive mode.

3.2.2.1.3 Operation markers and marker segments

Markers and marker segments operated by JPEG are shown in Table 3-11.

Table 3-11: 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	General process

	Only SOF0 (FFC0) is the subject of encoding.	
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.

3.2.2.2 GIF

The graphics file format in GIF uses specifications based on “GRAPHICS INTERCHANGE FORMAT Version 89a" specified by Compuserve Inc. America.

3.2.2.2.1 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.

3.2.2.2.2 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

3.2.2.2.3 Trailer

1 will always exist at the end.

Field	Operation	Remarks
GIF Trailer	O	Fixed value 0x3B

3.2.2.2.4 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	

3.2.2.2.5 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.

3.2.2.2.6 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	0-426	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

3.2.2.2.7 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.

3.2.2.2.8 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.

3.2.2.2.9 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

3.2.2.2.10 Comment Extension

Operation is arbitrarily and can be multiple. The syntax is operated according to standards.

3.2.2.2.11 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 X-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.

3.2.3 Audio coding

Follow Vol. 7, section 2.1.2.

3.2.3.1 MPEG-2 AAC

3.2.3.1.1 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).

3.2.3.1.2 Limitations of data carousel transmission

- File size is 128KB or less.
- When start or stop 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.

3.2.3.2 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 3-12.

Table 3-12: Allocation of receiver 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.

3.2.4 Character codes

3.2.4.1.1 8-bit code

Refer to section 3.3.

3.2.4.1.2 Shift JIS

Refer to Chapter 2 in this volume.

3.2.5 Metadata coding

Refer to Vol. 7, section 2.1.5.

3.3 Operation of coding of closed captions and superimpose

For the items not specially described, closed captions and superimpose are treated in the same way.

3.3.1 Definition and range of service

The following services of closed captions and superimpose are performed.

Closed caption : Closed caption service (for example, translated character closed caption, etc.) synchronized with main video, audio and data.

Superimpose: Closed caption service (for example, news flash, program remarks, time signal, earthquake early warning, etc.) that is asynchronous with the main video, audio, and data.

3.3.2 Organization and transmission operation

3.3.2.1 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

The numbers of ES of closed captions and superimpose that can be transmitted at the same time are 1 ES each, a total of 2 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 for closed caption or superimpose.

(6) Display modes that can be used

For closed captions, only “select display during reception, select display during recording/playback” is operated. For superimpose, only “select display during reception, select display during recording/playback” and “auto display during reception, select display during recording/playback” are operated. It is a same when multiple languages are transmitted.

(7) Operation of built-in sound and additional sound

Only the sound built into the receiver can be operated for the warning sound in superimpose. However, playback of the receiver built-in sound is not mandatory for the receiver. For superimpose, the sound built into the receiver is not operated. Neither is additional sound operated for closed captions or superimpose.

(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. After the closed caption management data and closed caption data are received, closed captions or superimpose are basically displayed. For closed captions, however, the operation of the data group is fixed (see section 3.3.2.4), so as soon as the closed caption data is received, it is possible to display the sentences of closed captions 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.

Because it is necessary to always check the display mode for superimpose, receiving the closed caption management data is mandatory.

3.3.2.2 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 3-13 shows the parameters set in the PES packet.

Configuration parameters : Refer to Table 3-13

Maximum number of ESs 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 are 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 1 language and 2 languages)

Apart from this, 256bytes is needed for the DRCS. Refer to section 3.6.6.

Table 3-13: 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 preferable to have movements described in 3.6.6.

3.3.2.3 PES transmission method used in superimpose

The asynchronous PES transmission method is applied. Table 3-4 shows the parameters set in the PES packet.

Configuration parameters : Refer to Table 3-13

Maximum number of ESs 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 are 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 1 language and 2 languages)

Apart from this, 256bytes is needed for the DRCS. Refer to section 3.6.6.

Table 3-14: Configuration parameters of PES packets in superimpose

Field	Operation
Stream_id	0xBF (private_stream_2)
PES_packet_length	Number of bytes in a PES packet that follow later. *1
data_identifier	0x81
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 PES_data_packet_header_length, without fail.

For PES packet delivery, the following restrictions are established.

For the interval of PES packet transmission of the closed caption data group, the interval between the nth-order PES packet and n+1th-order PES packet is larger than Td of the nth-order data. Here, Td shows the time from the completion of reception to the completion of presentation, and a reference is 0.5 seconds for only characters.

At the nth-order PES packet transmission completion time + Td time, the total information amount of PES packets that start to be transmitted in the nth order or later shall not exceed the capacity of the receiving buffer (1280 bytes).

The movement of receiver exceeding the reception buffer is implementation dependent. The movement of the receiver exceeding the reception buffer for the DRCS is implementation dependent; however, it is preferable to have the movements described in 3.6.6.

3.3.2.4 Operation of data groups

Data group transmission shall be operated as shown in Table 3-15. 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 3-15: 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.

3.3.2.5 Operation of closed caption management data

Data units are not allocated in closed caption management data.

3.3.2.6 Closed caption management data used for closed captions

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 3-16. shows the parameters that can be specified for the closed caption management data used in closed caption.

Table 3-16: Parameters of closed caption management data for closed captions

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.

3.3.2.7 Closed caption management data used for superimpose

Considering time signal superimpose, TMD is possible for both free setting and real-time setting to perform time synchronization by STM. Table 3-17 shows the parameters that can be specified to the closed caption management data used for superimpose.

Table 3-17: Parameters of closed caption management data for superimpose

Field	Operation
TMD	‘00’ (free) ‘01’ (real-time) Free and real-time cannot be mixed during presentation.
num_languages	1 - 2
language_tag	0 - 1
DMF	‘0010’ (auto display during reception, select display during recording/playback) ‘1010’ (select display during reception, select display during recording/playback)
ISO_639_language_code	Language code used (fixed at “jpn”)
Format	Not operated. ‘1111’
TCS	‘00’ (8-bit code)
rollup_mode	‘00’ (non-rollup)
data_unit_loop_length	‘0’
data_unit	Data unit is not located.

3.3.2.8 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 3-18.

Table 3-18: Parameters of closed caption text data

Field	Operation
TMD	'00' (free) '01' (real-time): Superimpose only However, set the same value as the closed caption management in the same program)
STM	Operated as defined. However, is valid only when the Data Component Descriptor of PMT is timing='10' (time synchronization).
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.

3.3.2.9 Operation of data units

Parameters that can be configured in data units are shown in Table 3-19.

Table 3-19: 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.

3.3.2.10 Operation of PSI/SI

3.3.2.10.1 Operation of component tags

Component tag value of the closed caption ES is 0x30. The component tag value of superimpose ES is 0x38.

3.3.2.10.2 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.

3.3.2.10.3 Stream format identification

The stream_type of the closed caption ES is 0x06 (independent PES_packet).

3.3.2.10.4 Descriptor operation

Descriptor operation of the PMT and EIT for closed caption is shown in Table 3-20.

Table 3-20: 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.

3.3.2.10.5 Data Component Descriptor

Data_component_id of Data Component Descriptor is 0x1C for both closed captions and superimpose. Table 3-21 shows the parameters to configure the additional information identification.

Table 3-21: Configuration parameters of additional information identification for the Data Component Descriptor

Field	Operation
DMF	Closed caption: '1010' Superimpose: '0010' or '1010'
Timing	Closed caption: '01' (program synchronization) Superimpose: '00' (asynchronous) or 10' (time synchronization)

3.3.2.10.6 Target Region Descriptor

The Target Region Descriptor is not operated.

3.3.2.10.7 Data Contents Descriptor

The Data Contents Descriptor is not operated.

3.3.3 Closed captions and superimpose display format

3.3.3.1 Display format

For producing superimpose data and closed caption data, broadcasting stations shall consider that the receiver display formats for superimpose and closed caption have following differences.

For superimpose, there are 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). Broadcasters shall produce the superimpose text in consideration of displaying all characters of the superimpose text data, even if the receiver provides a display area with the above-

mentioned features.

For closed caption, broadcasters shall produce the closed caption text in consideration of displaying all characters of the closed caption text data, assuming that the recivers have the lateral direction of a normal size of 16 characters or more with 3 lines (both in case of vertical and holizontal screen).

The receiver should displays with a linefeed in the place of the control code of the active position line feed of the closed caption and the superimpose 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 and superimpose text produced according to the above-mentioned assumption, all characters should be able to be displayed.

Overlapping of the closed caption and superimpose text and the image is implementation dependant. Moreover, it is recommended, in implementation dependant, to display the closed caption and superimpose text for receiver that do not meet the display format of the above-mentioned assumption appropriately.

3.3.4 Characters used for closed captions and superimpose

3.3.4.1 Character entity

The character encoding method used in closed captions uses 8-bit character codes for X-profile with a subset of 8-bit character codes for the X-profile. In the 8-bit character code for the X-profile, the GL code area is fixed to the DRCS character set (1-byte code) provisioned by ARIB STD-B24, the GR code area is fixed to the Kanji character set (2-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

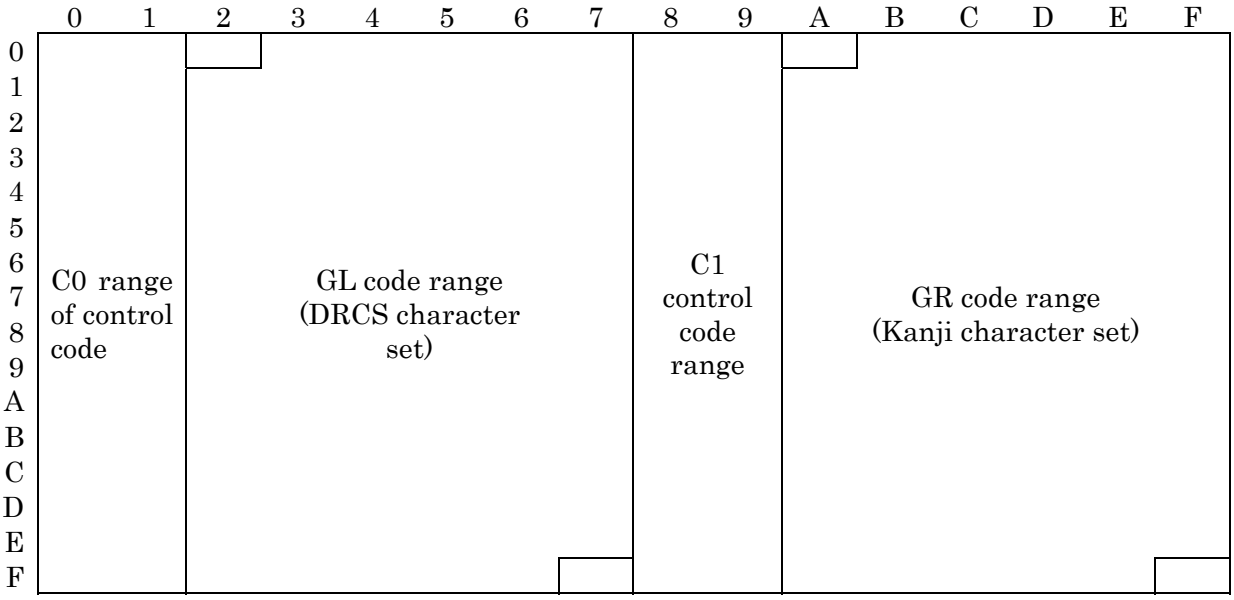


Figure 3-3-5: Table of 8-bit character codes range for X-profile

3.3.4.2 Character fonts

This is implementation dependant of the receiver .

3.3.4.3 Character size

It is assumed that the size of the characters displayed for closed captions and superimpose is medium font. 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 3-22. Moreover, the voiced sound mark specified as medium may be converted into medium size double characters in the receiver .

Table 3-22 shows the restrictions of character display.

Table 3-22: 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

Character	2 byte code Row (Cell)	Standard	Medium
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 .

3.3.4.4 Display partition

As this is a letter space and line space without a specified operation, the definition of display partition is not done.

3.3.4.5 Non-spacing characters

Non-spacing characters are not used.

3.3.5 Control codes used in closed captions and superimpose

3.3.5.1 Control code

The control code used in closed captions and superimpose is compliant with ARIB STD-B24, Vol. 1, Part 2, 7.1.2. However, the operation limitations in Table 3-23, Table 3-24, Table 3-25 are established.

Table 3-23: 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 X-profile.

O: possible use　Δ: possible use with limitations　X: cannot be used

Table 3-24: 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 Table 3-22.
NSZ	Normal size	Δ	Operation complies with Table 3-22.
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 section 3.3.5.2.
CDC	Conceal control	X	
WMM	Writing mode modification	X	

C1 Control code	Control function	Yes or no, of use	Control item and supplement
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	Δ	

O: possible use Δ: possible use with limitation X: cannot be used

3.3.5.1.1 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 3-24. 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 Annex 8 for the CLUT common fixed colors.

3.3.5.2 Operation of flashing

Flashing of 8-bit character codes character string for X-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 section 3.3.7 is performed and the deletion of the display screen is indicated by the CS control code, the flashing of an aforesaid character is terminated.

3.3.5.3 Extension control code

Operation of the extension control code is shown in Table 3-25.

Table 3-25: 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	Control characters used for closed caption and superimpose are restricted by this table. Presentation position indication by content data is not operated, as the behavier of the recievers, which receives these control character, is not guaranteed.
SDP	Display position specification	X	Control characters used for closed caption and superimpose are restricted by this table. Presentation position indication by content data is not operated, as the behavier of the recievers, which receives these control character, is not guaranteed
SSM	Character configuration dot specification	X	
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	X	

Character	Control function	Yes or no, of use	Control item and supplement
	setting		
ORN	Character decorations specification	X	
MDF	Character format at specification	X	
PRA	Built-in sound playback	X	Operated only for superimpose. However, playback of receiver built-in sound is not assumed mandatory for the receiver.
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

3.3.6 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", Attachment "Operational guidelines for closed captions".

Table 3-26: 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

3.3.7 Operation of the initialization operation

The receiver shall perform the initialization operation of closed captions and superimpose shown in 3.3.7.1 to 3.3.7.5.

Table 3-27: 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 area Left top corner of display
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)

3.3.7.1 Initialization by switching classes in the data group

The receiver shall initialize all items prescribed in Table 3-27 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.

3.3.7.2 Initialization by closed caption text

The receiver shall initialize all items prescribed in Table 3-27 when it receives closed caption text data of the same data group class and language as those currently being processed for presentation.

3.3.7.3 Initialization by main text data unit

The receiver shall initialize the items in "Status instruction" in Table 3-27 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.

3.3.7.4 Initialization by character control codes

The receiver shall initialize the items under "Display screen", "Operation instruction" and "Status instruction" in Table 3-27 immediately before executing the clear screen (CS) process.

3.3.7.5 Initialization by receiver operation

The receiver shall initialize all items prescribed in Table 3-27 when a station selection operation or switching of the closed caption language has been performed.

3.3.8 Mono-media used in closed captions and superimpose

3.3.8.1 Operation of geometric

Geometric is not operated.

3.3.8.2 Operation of bitmap data

Bitmap data is not operated for closed captions and superimpose.

3.3.8.3 Operation of built-in sound

The built-in sound is operated for superimpose, but is not operated for closed captions.

The built-in sound for superimpose is allocated as follows. The coding method for the receiver built-in sound is a matter of product planning.

0: Quick chime 1	1: Quick chime 2	2: Quick chime 3	3: Quick chime 4
4: Quick 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:

The above numbers show the designation of built-in sound when using extended control codes PRA of 8-bit codes.

Note that it is different from the built-in sound used from the multimedia codes.

3.3.8.4 Operation of additional sound

Additional sound is not operated.

3.3.9 Ideal receiver operation

The receiver presentation of closed captions and superimpose is controlled independently.

Closed captions and superimpose are operated so that the display areas are not overlapped, in principle, but if the displays are overlapped, a superimpose has precedence and is displayed in front of a closed caption.

It is preferable that the receiver 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.

If, while simultaneous displays are given on the same screen as the other display area, the DMF field of closed caption management data receives a superimpose of auto display upon reception, it is preferable that an appropriately sized superimpose is presented.

3.3.9.1 Initialization and termination of closed caption display

The receiver 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.

3.3.9.2 Initialization and termination of superimpose display

The receiver controls initialization and termination of the superimpose independent of user operation when the DMF field of closed caption management data is set to auto display upon reception. When the DMF field of closed caption management data is set to select display upon reception, initialization and termination of superimpose are controlled by the user operation.

3.4 Operation of multimedia encoding

3.4.1 Introduction

The operation of the multimedia encoding follows,

・" Attachment 1, Operational Guidelines" and

・"Attachment 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 broadcast reception/playback function is implemented in receivers, 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 designers are aware of the above assumption, they may design receivers without the communication function.

3.4.2 Operation of NVRAM

The NVRAM used to store persistent information is provided with the broadcaster-exclusive area and TVlink area shown in Table 3-28.

Table 3-28: NVRAM used

Class	Purpose	Capacity of NVRAM
Broadcaster-exclusive area	Area used by broadcasters	・ Total of 128 KB ・ Capacity per broadcaster: 8 KB (256-byte fixed length block x 32) (*)
TVlink area	Area used for TVlink service	・Maximum of 256 bytes variable block ・Writable block number :50 or more

(*) Note that the capacity per block of fixed length is different from that of TR-B14.

It is preferable that the broadcaster-exclusive area and TVlink area are packaged so that they are fixed in receiver, but packaging without securing an area in a fixed way is also allowed. That is, they can share the physical nonvolatile storage area with other applications in 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 broadcaster-exclusive area for the affiliation shall ideally be allocated in the unit of 1 affiliation (8KB).

3.4.2.1 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 Annex 1.

3.4.2.2 Allocation of NVRAM

3.4.2.2.1 Allocation of broadcaster-exclusive area

The initial value of the secured broadcaster-exclusive area shall be an empty character string.

If the broadcaster-exclusive area has not yet been secured, the readPersistentArray function will return a null.

If the broadcaster-exclusive area has already been secured but access to the broadcaster-exclusive area is limited (*) by user operation or setting on the receiver, the readPersistentArray function will set the array Array[0] to null as a return value.

If the receiver is packaged so that the memory area is secured when writing is first commanded, it is allowed that the area cannot be secured, because the memory area is short.

(*) For example, when the broadcaster-exclusive area is accessed, the receiver function asks the user's permission for access such as "yes/no", and the user selects rejection of access. Or, when the user sets the receiver to reject access to the broadcaster-exclusive area.

3.4.2.2.2 Broadcaster-exclusive area erasing function

Receivers need to have a function to erase the broadcaster-exclusive area by the instruction of the user.

The following shows the guidelines for management of erasure of the receiver's broadcaster-exclusive area:

The initial value of the broadcaster-exclusive area when erasure is executed is an empty character string.

The erasure of the broadcaster-exclusive area is executed by starting the erasure function from the menu provided by the receiver and by the user instruction. At this time, it is preferable that a dialog, etc. is presented to the user to show that erasure will pose a serious problem for the broadcast-related service.

The erasure function and other specifications are the matters of product planning.

It is preferable that the broadcaster-exclusive area erasure function is exclusively provided for the broadcasting application. When the erasure function is used in common with other applications, it is necessary to sufficiently attract the user's attention so that they do not erase it by mistake.

3.4.2.2.3 Identification of broadcaster-exclusive area

When information is read from the MM service and written to the broadcaster-exclusive area, one fixed-length block is regarded as one file and readPersistentArray()/ writePersistentArray() is executed. Information is read from the MM service and written to the broadcaster-exclusive area in units of fixed-length blocks. The following URI is used to identify the fixed-length block:

nvram:// <broadcaster_id>/<block number>

<broadcaster_id> : Broadcaster ID. Described in hexadecimals, using 4 digits.

<Block number>　: 0 - 31

<broadcaster_id> cannot be omitted.

When a broadcaster ID other than the broadcaster, who is broadcasting the contents that are currently being played back, is designated in the broadcaster_id area, access to this area will fail.

3.4.3 Operation of keys from the Multimedia service

3.4.3.1 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 3-30 are virtual names not physical keys. A key to allocate physically in the key code shown in Table 3-30 is a matter of product planning.

However, the receiver should offer viewers the keys virtually (software key or physical key), because contents are produced assuming the key codes shown in Table 3-30 can be sent.

Table 3-29: 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.
(Note 3)　Virtual name; it is not intended to be a physical key.

3.4.3.2 Correspondence of keys, key codes and access keys

Table 3-30 shows the mapping of the characters specified as keys, key codes, and access keys that can be used by the Multimedia service.

Table 3-30: 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

3.4.4 Operation of the BML version

The BML version in the BML document is major_version=14 and minor_version=0. This PI is described following the XML declaration and the DOCTYPE declaration. Refer to Annex 6 for the receiver behavior when the update of major_version and minor_version.

<?bml bml-version="14.0" ?>

3.4.5 Operation of character coding schemes

See ARIB STD-B24, Vol. 2, Attachment 4, “4.1. Character Coding Schemes”.

3.4.6 Operation range of media type and mono-media

It follows ARIB STD-B24, Vol. 2, Attachment 4, "4.2. Media Types and Operational Range of Mono-media ".

3.4.7 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, Attachment 4, are referred. For the area not described here, ARIB STD-B24 is applied .

3.4.7.1 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 14.0//JA"
"http://www.arib.or.jp/B24/DTD/bml_14_0.dtd">

3.4.7.2 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 Annex 3 for DTD of the operation.

Table 3-31: Operation of elements

Module		BML (operation)	Element
Core	Structure (required)	O	body
		O	head
		O	html
		O	title
	Text (required)	-	abbr
		-	acronym

Module		BML (operation)	Element
		-	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

Module		BML (operation)	Element
		-	option
		O	textarea
		-	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&*

Module		BML (operation)	Element
		-	*button&*
Metainformation		O	meta
Scripting		-	noscript
		O	script
Stylesheet		-	style
Style Attribute		O	
Link		O	link
Base		-	base
Name Identification (deprecated)		-	*a&*
			applet&
			form&
			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&*

Module		BML (operation)	Element
	Basic Mobile	O	*bml:bevent*
		O	*bml:beitem*
		O	*object&*

3.4.7.3 Attributes

Table 3-32 shows the operation of the attributes specified as operated in section 3.4.7.2 (div p br pre span a form input textarea img object meta title script link body head html bml:bevent bml:beitem).

Table 3-32: Operations for attributes of elements

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	Δ	In case of a reciever without a hardware key, it is optional.
	onkeyup	Δ	In case of a reciever without a hardware key, it is optional.
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	-	

Element	Attribute	Operation	Restriction on operation
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	
	%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;	-	

Element	Attribute	Operation	Restriction on operation
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	-	
	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).

Element	Attribute	Operation	Restriction on operation
	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	-	
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	-	

Element	Attribute	Operation	Restriction on operation
&			
	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	
	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.

Element	Attribute	Operation	Restriction on operation
			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 refers to the data transmitted in a data carousel
	subscribe	O	
object&	bml:streamstatus	O	
	bml:onfocus	O	
	bml:onblur	O	
	bml:accesskey	O	

Explanatory note:

The attributes with "O" shown in the operation column are described in the BML document, but for those with "Δ", the fixed value is set in the receiver and it is not described in the BML document.

Note 1) Characters exceeding maxlength (textarea is always 240 characters in two-byte code character) in the input and textarea are truncated. 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, Attachment 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.

3.4.7.3.1 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.

3.4.7.3.2 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.

3.4.7.3.3 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.

3.4.7.4 Entity

See ARIB STD-B24, Vol. 2, Attachment 4, “4.4.3. Entity”.

3.4.7.5 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.

At the same time, the maximum number of StoreFinished interrupt events that the subscribe attribute can be set in subscribe is 1. The component that is described as es_ref is only storage-type broadcasting data, and it is in the range included in the TS currently being received.

ModuleUpdated of any data carousel component can be observed regardless of which component is being presented by the contents .

Behavior 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 3-33: 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.

3.4.8 Operation of CSS

Table 3-34 shows the operation of CSS2 properties.

Table 3-34: 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

Property	Operation
padding	-
border-top-width	-
border-right-width	-
border-bottom-width	-
border-left-width	-
border-width	O
border-top-color	O
border-right-color	O
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	-

Property	Operation
quotes	-
counter-reset	-
counter-increment	-
marker-offset	-
list-style-type	-
list-style-image	-
list-style-position	-
list-style	-
Paged media	
"@page"	-
size	-
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	-

Property	Operation
: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	-
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	-

Property	Operation
border-bottom-color-index	-
border-left-color-index	-
outline-color-index	-
resolution	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	-
opacity	O

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.

3.4.8.1 Element applied each properties

Table 3-35 shows the combination of operated properties and applied BML elements

Table 3-35: Element that apply the CSS properties

	body	div	p	pre	br	span,a	form	input,textar	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	-	-

	body	div	p	pre	br	span,a	form	input,textar	object	img
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										
text-align	-	-	O	-	-	-	-	O	-	-
white-space	-	-	Δ	Δ	-	-	-	Δ	-	-
Extended properties										
resolution	O	-	-	-	-	-	-	-	-	-
used-key-list	O	-	-	-	-	-	-	-	-	-
-wap-marquee-style	-	-	O	-	-	-	-	-	-	-
-wap-marquee-loop	-	-	O	-	-	-	-	-	-	-
-wap-marquee-dir	-	-	Δ	-	-	-	-	-	-	-
-wap-marquee-speed	-	-	O	-	-	-	-	-	-	-
-wap-input-format	-	-	-	-	-	-	-	O	-	-
opacity	-	O	-	-	-	-	-	-	-	-

3.4.8.2 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 3-35.

3.4.8.3 Media type

The media type operated is only tv. Specifying the media type is made by @media rule in style sheet. Moreover, the receiver should ignore any specification of the properties within the @media rule that has any other type than tv.

3.4.8.4 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.

Note, for img and object elements, the value is fixed as none in the operation.

3.4.8.5 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 truncated 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 defined 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 : yes

  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.

3.4.8.6 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

3.4.8.7 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, Attachment 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

3.4.8.8 Fonts

Table 3-36 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 3-36 for the fonts of the receiver depend on the receiver. When a font of Gothic system is installed in the receiver, it is preferable to use aforesaid font.

Table 3-36: 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.

3.4.8.9 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)

3.4.8.10 Default style sheets

The default style sheet is defined in Appendix 4. This style sheet is assumed to be loaded virtually in every receiver and it does not need be transmitted.

3.4.8.11 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”.

3.4.9 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 section 3.4.3.1, 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..

  Values : 240x480 | 426x240

  Fixed value : 240x480

  Inheritance : no

This property is used to identify the BML document for vertical display or horizontal display.

- -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.

- -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 (symbols 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 (7F) 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.

● opacity

This property is specified as below in the operation.

Values: 0 | 0.5 | 1

Applicable element: div element

Initial value: 1

Inheritance: no

This property specifies the transmission of background color. 0 shows perfect transmission, and 1 shows imperfect transmission.

3.4.10 Operations for procedural descriptions

3.4.10.1 Operations DOM

3.4.10.1.1 DOM Core Fundamental interfaces

Table 3-37 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 3-37: Operation of 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 3-38 in the operation.

Table 3-38: Operation of 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()	-	
	createAttribute()	-	
	createEntityReference()	-	
	getElementsByTagName()	-	
Node			
	nodeName	-	
	nodeValue	-	
	nodeType	-	
	parentNode	O	R
	childNodes	-	
	firstChild	O	R
	lastChild	O	R
	previousSibling	O	R

Interface	Attribute, method	Operation	Restrictions
	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.

3.4.10.1.2 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 specified in the latter table below according to the situation in which the access to the BMLEvent interface is done.

Table 3-39: Operation of attributes and methods (BML interfaces)

Interface name	Attribute or method name	Operation	Restriction
BMLEvent	type	O	R
	target	O	R
BMLIntrinsicEvent	keyCode	O	R

Interface name	Attribute or method name	Operation	Restriction
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 specified in Table 3-40.

Table 3-40: 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"
Module update is detected.	"ModuleUpdated"
Module is being locked.	"ModuleLocked"
Timer configured in beitem is fired.	"TimerFired"
presentation by mono-media decoder is terminated. (Note 2)	"MediaStopped"
Detected update of data_event_id.	"DataEventChanged"
Storage-type broadcasting data was stored.	"StoreFinished” (Note 4)
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 mono-media 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 dependent. (Excludes input elements with a type attribute of "submit”.)

Note 4) This event occurs only when X_TMM_ecgStoreStart is packaged.

Table 3-41: 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
			StoreFinished		O	X
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 3-42: 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 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 3-43: 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 1)
		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 1)
		focusStyle	-	
		activeStyle	-	
		accessKey	-	
		focus()	-	
		blur()	-	
p	HTMLElement	id	O	R
		className	O	R
		title	-	
		lang	-	
		dir	-	
	HTMLParagraphElement			
	BMLParagraphElement	style	-	
		normalStyle	O	RW (Note 1)
		focusStyle	-	
		activeStyle	-	
		accessKey	-	
		focus()	-	
		blur()	-	
pre	HTMLElement	id	O	R
		className	O	R
		title	-	
		lang	-	
		dir	-	
	HTMLPreElement			
	BMLPreElement	style	-	
		normalStyle	O	RW (Note 1)
		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 1)
		focusStyle	-	
		activeStyle	-	
span	HTMLElement	id	O	R
		className	O	R
		title	-	
		lang	-	
		dir	-	
	BMLSpanElement	style	-	
		normalStyle	O	RW (Note 1)
		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 1)
		focusStyle	-	
		activeStyle	-	
		effect	-	
form	HTMLElement	id	O	R
		className	O	R
		title	-	
		lang	-	
		dir	-	
	HTMLFormElement	elements	-	
		length	-	
		name	-	
		acceptCharset	-	
		action	O	RW
		enctype	-	
		method	O	R
		submit()	O	
		reset()	-	
	BMLFormElement	style	-	
		normalStyle	O	RW (Note 1)
		focusStyle	-	
		activeStyle	-	
		accept	-	

Element by the node	Interface applied	Attribute or method	Operation	Restriction
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	-	
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()	-	

Element by the node	Interface applied	Attribute or method	Operation	Restriction
	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 2)
		useMap	-	
		width	-	
	BMLImageElement	style	-	
		normalStyle	O	RW (Note 1)
		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 2)
		declare	-	
		height	-	
		name	-	
		standby	-	
		tabIndex	-	
		type	O	R
		useMap	-	
		width	-	
	BMLObjectElement	style	-	
		normalStyle	O	RW (Note 1)
		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	-	

(Note1) 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.

(Note 2) For the resulting display of the img/object element in which the updated area displayed by the new resource doesn't match with the original area displayed by the original resource when the resource to be presented is updated, see section 4.1.3.2. In the img element, the media type of the resources to be presented must be same as before when update.

3.4.10.1.3 BMLCSS2Properties interface

The operation of BMLCSS2Properties interface used in Table 3-44 is shown.

Table 3-44: 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	-	

Attribute	Operation	Remarks
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	-	
minHeight	-	
maxWidth	-	
minWidth	-	
whiteSpace	-	
Table related	-	
captionSide	-	
borderCollapse	-	
borderSpacing	-	
tableLayout	-	

Attribute	Operation	Remarks
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	-	
resolution	-	
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	-	
opacity	-	

(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.

3.4.10.2 Operation area of built-in objects

Operation of ECMAScript built-in objects is shown in Table 3-45.

Table 3-45: 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	
	length	O	
	String([value])	O	
	new String([value])	O	
	String.fromCharCode(char0[, char1, . . .])	O	

Embedded object	Method and property	Operation	Remarks
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)	-	
	floor(x)	-	
	log(x)	-	
	max(x, y)	-	
	min(x, y)	-	
	pow(x, y)	-	
	random()	-	
	round(x)	-	

Embedded object	Method and property	Operation	Remarks
	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)
	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, Attachment 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).

3.4.10.3 Operation area of extension objects for broadcasting

Table 3-46 shows the operation of extension objects for broadcasting ECMAScript. Operation of the defined extension object for broadcasting not described in Table 3-46 is assumed to be "-".

Table 3-46: 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	

3.4.10.4 Operation area of Navigator pseudo-objects

Navigator pseudo-objects are not operated.

3.4.10.5 Operation area of browser pseudo-objects

Operated as shown in Table 3-47.

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 by the getBrowserSupport() function.

"O (*2)" Installation of these functions is not necessary only when receivers 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 receivers 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 by the getBrowserSupport() function.

"-" Not a basic function nor an optional function in these provisions. An error will occur in the receiver when the aforementioned function is called.

Table 3-47: Operation area of browser pseudo-objects

	Function	Operation	Observation
Ureg related function			
	Ureg[]	O	
Greg related function			
	Greg[]	O	See "Chapter 5 (Operation of Greg)".
EPG functions			
	epgGetEventStartTime()	O	
	epgGetEventDuration()	O	
	epgTune()	O(*1)	
	epgTuneToComponent()	-	
	epgTuneToDocument()	-	
	epgIsReserved()	-	
	epgReserve()	-	
	epgCancelReservation()	-	
	epgRecIsReserved()	-	
	epgRecReserve()	-	
	epgRecCancelReservation()	-	
Event group index functions			
	grpIsReserved()	-	
	grpReserve()	-	
	grpCancelReservation()	-	
	grpRecIsReserved()	-	
	grpRecReserve()	-	
	grpRecCancelReservation()	-	
	grpGetNodeEventList()	-	
	grpGetERTNodeName()	-	
	grpGetERTNodeDescription()	-	
	epgXTune()	-	
Series reservation functions			
	seriesIsReserved()	-	
	seriesReserve()	-	

	Function	Operation	Observation
	seriesCancelReservation()	-	
	seriesRecIsReserved()	-	
	seriesRecReserve()	-	
	seriesRecCancelReservation()	-	
Non-volatile memory functions			
	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()	-	
	saveFtpServerFileAs()	-	
	saveFtpServerFile()	-	
	sendFtpServerFileAs()	-	
	sendTextMail()	-	
	sendMIMEMail()	-	
	transmitTextDataOverIP()	O(*2)	
	setDelayedTransmissionData()	-	
	getTransmissionStatus()	-	
	getTransmissionResult()	-	
	setCacheResourceoverIP()	-	

	Function	Operation	Observation
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()	-	
	detectComponent()	O	
	lockModuleOnMemoryEx()	O	
	unlockModuleOnMemoryEx()	O	
	unlockAllModulesOnMemory()	O	
	getLockedModuleInfo()	O	
	getBrowserStatus()	O	
	getResidentAppVersion()	O	
	isRootCertificateExisting()	-	

	Function	Operation	Observation
	getRootCertificateInfo()	-	
	startResidentApp()	-	
	getDataDisplayAreaSize()	-	
	setFullDataDisplayArea()	O(*1)	[Reference information] Will not be used in contents for the time being.
Receiver 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()	-	
	getContentSource()	-	
	getStorageInfo()	-	
Data carousel accumulation function			
	saveCarouselAs()	-	
	saveCarousel()	-	

	Function	Operation	Observation
	saveModuleAs()	-	
	saveModule()	O(*1)	
	saveResourceAs()	-	
	saveResource()	O(*1)	
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()	-	
Unique functions of digital terrestrial television broadcasting			
	X_TMM_mailTo()	O (*1)	
	X_TMM _startResidentApp()	O	
	X_TMM _phoneTo()	O (*1)	
	X_TMM _getRcvCond ()	O (*1)	[Reference information] Will not be used in contents for the time being.
	X_TMM _getCurPos ()	O (*1)	
	X_TMM _saveExAppFile ()	O (*1)	[Reference information] Will not be used in contents for the time being.
	X_TMM _startExAV()	O (*1)	[Reference information] Will not be used in contents for the time being.
	X_TMM _stopExAV()	O (*1)	[Reference information] Will not be used in contents for the time being.
	X_TMM _writeSchInfo()	O (*1)	
	X_TMM _getComBrowserUA ()	O (*2)	
	X_TMM _writeAddressBookInfo()	O (*1)	
	X_TMM _launchDocWithLink()	O (*2)	[Reference information] Will not be used in contents for the time being.
	X_TMM _chkAVtype()	O (*1)	
	X_TMM _getIRDID()	O	
	X_TMM _writeXproBM()	O (*2)	[Reference information] Will not be used in contents for the time being.
	X_TMM_epgReserve()	O (*1)	[Reference information] Will not be used in contents for the time being.
	X_TMM_epgCancelReservation()	O (*1)	[Reference information] Will not be used in contents for the time being.
	X_TMM_epgCheckReservation()	-	
	X_TMM_epgSeriesCheckReservation()	-	
	X_TMM_epgSeriesReserve()	O (*1)	[Reference information] Will not be used in contents for the time being.
	X_TMM_epgSeriesCancelReservation()	O (*1)	[Reference information] Will not be used in contents for the time being.
	X_TMM_epgRecCheckReservation()	-	

	Function	Operation	Observation
	X_TMM_epgRecReserve()	O (*1)	[Reference information] Will not be used in contents for the time being.
	X_TMM_epgRecCancelReservation()	O (*1)	[Reference information] Will not be used in contents for the time being.
	X_TMM_epgSeriesRecCheckReservation()	-	
	X_TMM_epgSeriesRecReserve()	O (*1)	[Reference information] Will not be used in contents for the time being.
	X_TMM_epgSeriesRecCancelReservation()	O (*1)	[Reference information] Will not be used in contents for the time being.
	X_TMM_ecgLaunchContent()	O (*1)	
	X_TMM_ecgStoreStart()	O (*1)	[Reference information] Will not be used in contents for the time being.
	X_TMM_ecgStoreTerminate()	O (*1)	[Reference information] Will not be used in contents for the time being.
	X_TMM_ecgCheckStorageStatus()	O (*1)	
	X_TMM_ecgStoreReserve()	O (*1)	
	X_TMM_ecgStoreCancelReservation()	O (*1)	
	X_TMM_ecgStoreCheckReservation()	O (*1)	[Reference information] Will not be used in contents for the time being.
	X_TMM_ecgIsContentStored()	O (*1)	
	X_TMM_lockStoredContent()	O (*1)	[Reference information] Will not be used in contents for the time being.
	X_TMM_unlockStoredContent()	O (*1)	[Reference information] Will not be used in contents for the time being.
	X_TMM_ecgIsLockedStoredContent()	O (*1)	[Reference information] Will not be used in contents for the time being.
	X_TMM_ecgDeleteStoredContent()	O (*1)	[Reference information] Will not be used in contents for the time being.
	X_TMM_ecgGetStorableSpace()	O (*1)	[Reference information] Will not be used in contents for the time being.
	X_TMM_startExFullDisplay()	O (*1)	
	X_TMM_stopExFullDisplay()	O (*1)	

3.4.10.5.1 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_TMM_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.

3.4.10.5.2 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.

3.4.10.6 Extended function uniquely specified in digital terrestrial television broadcasting (1)

- getResidentAppVersion() : Receiver's native application recognition function

For the definition of functions, refer to "ARIB STD-B24 Vol. 2, 7.6.8 Operation control 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, Attachment 4, 4.1.1) provided arbitrarily by each manufacturer. Moreover, refer to Table 3-48 for the value that can be specified for argument appName. However, ComBrowser is not specified for appName.

3.4.10.7 Extended function uniquely specified in digital terrestrial television broadcasting (2)

- X_TMM_mailTo(): Text mail transmission by mail applications of receivers.

Syntax:

```
Number   X_TMM_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_TMM_startResidentApp(): Receiver's native application initialization function

Syntax:

```
Number X_TMM_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 to work according to the specification of this argument, as this argument is only hint information for the receiver.

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 using X_TMM_getComBrowserUA() when a communication browser is started, and by changing to the appropriate URI. The operation follows the specifications in section 3.4.10.8, for whether to end the BML browser by starting the

receiver 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 Annex 7.

Table 3-48: 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 implementation dependent. However, when ComBrowser is not displayed on a full screen, the operation is a matter of product planning. 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_TMM_phoneTo (): Make calls by the specified number. "mobile phone native features"

Syntax:

Number X_TMM_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 (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_TMM_getRcvCond (): Acquires the communication state in the telecommunication lines

Syntax:

Number X_TMM_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_TMM_getCurPos (): Obtain receiver geographical position.

Syntax:

Array X_TMM_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 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_TMM_saveExAppFile(): An external application is stored in the destination according to the receiver setting. (T.B.D.)

Syntax:

Number X_TMM_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 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_ TMM _startExAV (): The audio and video from the communication contents is displayed.

Syntax:

Number X_TMM_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_ TMM_stopExAV (): The audio and video from the communication contents is discontinued.

Syntax:

Number X_TMM_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_TMM_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_TMM_writeSchInfo(): Information is written in the schedule notebook of the receiver features.

Syntax:

```
Number X_TMM_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_TMM_getComBrowserUA (): Information to identify if a communication contents browser is acquired.

Syntax:

Array X_TMM_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.

For the character string of User-Agent Header in communication browser, the character string specified in "4.3.14 Operation of User-Agent for identifying the data broadcasting browser or communication browser" will be returned.

● X_TMM_writeAddressBookInfo (): Information is written in the address book of the receiver features.

Syntax:

```
Number X_TMM_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 receiver start, 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_TMM_launchDocWithLink (): Base URI directory is switched, and the BML document is presented while linked.

Syntax:

Number X_TMM_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 directory name specified by this function is set to the base URI directory (for the definition of base URI directory, see “4.3.7 Mapping of scope of communication contents”). 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_TMM_launchDocWithLink() is not continued.

- Neither load nor the unload event occur when X_TMM_launchDocWithLink() is executed in a global code.

- The continuance of the script execution that continues behind, when X_TMM_launchDocWithLink() fails, is not guaranteed.

- Receiver operation fails, and presents the broadcasting contents according to section 4.3.11.4. Guidelines when receiver operation fails, when X_TMM_launchDocWithLink() is used in the broadcasting reception status.

● X_TMM_chkAVtype (): The type of audio and video from the communication contents is confirmed.

Syntax:

Number X_TMM_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_TMM_getIRDID (): The identifier to specify the receiver or the viewer is acquired.

Syntax:

String X_TMM_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 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_TMM_writeXproBM (): Registration of TVlink.

Syntax:

Number X_TMM_writeXproBM(

input String title,

input String dstURI,

input String outline,

input Number XproBMtype

[,input Date expire]

)

Argument:

title	Title of TVlink
dstURI	URI link destination of TVlink
outline	Explanation of TVlink
XproBMtype	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 XproBMtype, 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 XproBMtype shows the receiver operation when registered information is used. For example, the receiver 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 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 (XproBMtype=0) or the linked contents (XproBMtype=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 XproBMtype does not need to be registered in the receiver, 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.

X_TMM_ epgReserve (): Reserve program viewing.

Syntax:

```
Number   X_TMM_ epgReserve (
                                    input String content_path
                                    [,input Date startTime]
                                )
```

Argument:

content_path	The program Content Reference Identifier is specified
startTime	Program start time

Return value:

1:	Success
NaN:	Failure

Explanation:

A program to start at the time specified by startTime, which is specified by content_path, is reserved for viewing, and success or failure is returned using a return value.

The Content Reference Identifier (= CRID) is specified for content_path. For identification of usage-unit content using CRID, follow the definition of name space in this technical data "Vol. 10, Chapter 8 Operation of identification information of contents and metadata".

If the specified content_path does not exist, reservation will fail.

If startTime is omitted, processing is performed for the event specified by content_path.

If startTime does not match the start time of the program specified by content_path, content_path is prioritized for processing.

- X_TMM_ epgCancelReservation (): Cancel reservation of program viewing.

Syntax:

```
Number    X_TMM_epgRecCancelReservation(
                                    input String content_path
                                )
```

Argument:

content_path	The program Content Reference Identifier is specified

Return value:

1: Success
NaN: Failure

Explanation:

The reservation of viewing a program specified by content_path is canceled, and success or failure is returned using a return value.

The Content Reference Identifier (= CRID) is specified for content_path. For identification of usage-unit content using CRID, follow the definition of name space in this technical data "Vol. 10, Chapter 8 Operation of identification information of contents and metadata".

- X_TMM_epgCheckReservation(): Check the status of program viewing reservation.

Syntax:

```
Number  X_TMM_epgCheckReservation(
                                    input String content_path
                                    [,input Date startTime]
                                )
```

Argument:

content_path	The program Content Reference Identifier is specified
startTime	Start time of the program

Return value:

2: Viewing of program is already reserved
1: Not yet reserved (overlapped with other reservation)
0: Not yet reserved (not overlapped with other reservation)
NaN: Failure

Explanation:

The reservation of viewing a program specified by content_path is surveyed regarding whether it is during viewing reservation and it is overlapped with another reservation, and return is made using a return value.

The Content Reference Identifier (= CRID) is specified for content_path. For identification of usage-unit content using CRID, follow the definition of name space in this technical data "Vol. 10, Chapter 8 Operation of identification information of contents and metadata".

If startTime is omitted, processing is performed for the event specified by content_path.

If startTime does not match the start time of the program specified by content_path, content_path is prioritized for processing.

- X_TMM_epgSeriesCheckReservation(): Check the status of series program viewing reservation.

Syntax:

```
Number  X_TMM_epgSeriesCheckReservation(
                                    input String series_path
                                    ,input Date expire_date
                                  )
```

Argument:

series_path	The series is specified
expire_date	Expiration date of the series

Return value:

2:	Viewing of program is already reserved
1:	Not yet reserved (overlapped with other reservation)
0:	Not yet reserved (not overlapped with other reservation)
NaN:	Failure

Explanation:

The reservation of viewing a series program specified by series_path is surveyed regarding whether it is during viewing reservation and it is overlapped with another reservation, and return is made using a return value. However, if the absolute time when this function was executed has passed the time specified by expire_date, it is judged that the designation of series_path is invalid and NaN is returned.

The groupCRID is specified for series_path. The groupCRID is described in the groupeId of the group information element. For a description of series_path, follow the regulation of name space in this technical data "Vol. 10, Chapter 8 Operation of identification information of contents and metadata".

- X_TMM_epgSeriesReserve(): Reserve viewing of the series program.

Syntax:

```
Number  X_TMM_epgSeriesReserve(
                                    input String series_path
                                    ,input Date expire_date
                                  )
```

Argument:

series_path	The series is specified

expire_date Expiration date of the series

Return value:

1: Success

NaN: Failure

Explanation:

A series program specified by series_path is reserved for viewing, and success or failure is returned using a return value. However, if the absolute time when this function was executed has passed the time specified by expire_date, it is judged that the designation of series_path is invalid and NaN is returned.

The groupCRID is specified for series_path. The groupCRID is described in the groupeId of the group information element. For a description of series_path, follow the regulation of name space in this technical data "Vol. 10, Chapter 8 Operation of identification information of contents and metadata".

- X_TMM_epgSeriesCancelReservation(): Cancel viewing reservation of series program.

Syntax:

```
Number    X_TMM_epgSeriesCancelReservation(
                                    input String series_path
                                    ,input Date expire_date
                                  )
```

Argument:

series_path The series is specified

expire_date Expiration date of the series

Return value:

1: Success

NaN: Failure

Explanation:

The reservation of viewing a series program specified by series_path is canceled, and success or failure is returned using a return value. However, if the absolute time when this function was executed has passed the time specified by expire_date, it is judged that the designation of series_path is invalid and NaN is returned.

The groupCRID is specified for series_path. The groupCRID is described in the groupeId of the group information element. For a description of series_path, follow the regulation of name space in this technical data "Vol. 10, Chapter 8 Operation of identification information of contents and metadata".

- X_TMM_epgRecCheckReservation(): Check the program recording reservation status.

Syntax:

```
Number  X_TMM_epgRecCheckReservation(
                                    input String content_path
                                    [,input Date startTime]
                                  )
```

Argument:

content_path	The program Content Reference Identifier is specified
startTime	Program start time

Return value:

2:	Recording is already reserved
1:	Not yet reserved (overlapped with another reservation)
0:	Not yet reserved (not overlapped with another reservation)
NaN:	Failure

Explanation:

The reservation of recording a program specified by content_path is surveyed regarding whether it is during recording reservation and it is overlapped with another reservation, and return is made using a return value.

The Content Reference Identifier (= CRID) is specified for content_path. For identification of usage-unit content using CRID, follow the definition of name space in this technical data “Vol. 10, Chapter 8 Operation of identification information of contents and metadata”.

If startTime is omitted, processing is performed for the event specified by content_path.

If startTime does not match the start time of the program specified by content_path, event_ref is prioritized for processing.

- X_TMM_epgRecReserve(): Reserve program recording.

Syntax:

```
Number  X_TMM_epgRecReserve(
                                    input String content_path
                                    [,input Date startTime]
                                  )
```

Argument:

content_path The program Content Reference Identifier is specified

startTime Program start time

Return value:

1: Success

NaN: Failure

Explanation:

A program to start at the time specified by startTime, which is specified by content_path, is reserved for recording, and success or failure is returned using a return value.

The Content Reference Identifier (= CRID) is specified for content_path. For identification of usage-unit content using CRID, follow the definition of name space in this technical data "Vol. 10, Chapter 8 Operation of identification information of contents and metadata".

If the specified content_path does not exist, reservation will fail.

If startTime is omitted, processing is performed for the event specified by content_path.

If startTime does not match the start time of the program specified by content_path, content_path is prioritized for processing.

- X_TMM_epgRecCancelReservation(): Cancel recording reservation of specified program.

Syntax:

```
Number  X_TMM_epgRecCancelReservation(
                                input String content_path
  )
```

Argument:

content_path The program Content Reference Identifier is specified

Return value:

1: Success

NaN: Failure

Explanation:

The reservation of recording a program specified by content_path is canceled, and success or failure is returned using a return value.

The Content Reference Identifier (= CRID) is specified for content_path. For identification of usage-unit content using CRID, follow the definition of name space in this technical data "Vol. 10, Chapter 8 Operation of identification information of

contents and metadata".

- X_TMM_epgSeriesStoreCheckReservation(): Check the status of series program recording reservation.

Syntax:

```
Number  X_TMM_epgSeriesStoreCheckReservation(
                        input String series_path
                        ,input Date expire_date
                        )
```

Argument:

series_path	The series is specified
expire_date	Expiration date of the series

Return value:

2:	Recording of program is already reserved
1:	Not yet reserved (overlapped with another reservation)
0:	Not yet reserved (not overlapped with another reservation)
NaN:	Failure

Explanation:

The reservation of recording a series program specified by series_path is surveyed regarding whether it is during recording reservation and it is overlapped with another reservation, and return is made using a return value. However, if the absolute time when this function was executed has passed the time specified by expire_date, it is judged that the designation of series_path is invalid and NaN is returned.

The groupCRID is specified for series_path. The groupCRID is described in the groupeId of the group information element. For a description of series_path, follow the regulation of name space in this technical data "Vol. 10, Chapter 8 Operation of identification information of contents and metadata".

- X_TMM_epgSeriesRecReserve(): Reserve recording of series program.

Syntax:

```
Number  X_TMM_epgSeriesRecReserve(
                        input String series_path
                        ,input Date expire_date
                        )
```

Argument:

series_path	The series is specified

expire_date Expiration date of the series

Return value:

1: Success

NaN: Failure

Explanation:

A series program specified by series_path is reserved for recording, and success or failure is returned using a return value. However, if the absolute time when this function was executed has passed the time specified by expire_date, it is judged that the designation of series_path is invalid and NaN is returned.

The groupCRID is specified for series_path. The groupCRID is described in the groupeId of the group information element. For a description of series_path, follow the regulation of name space in this technical data "Vol. 10, Chapter 8 Operation of identification information of contents and metadata".

- X_TMM_epgSeriesRecCancelReservation(): Cancel series program recording reservation.

Syntax:

```
Number X_TMM_epgSeriesRecCancelReservation(
                    input String series_path
                    ,input Date expire_date
                    )
```

Argument:

series_path The series is specified

expire_date Expiration date of the series

Return value:

1: Success

NaN: Failure

Explanation:

The recording reservation of a series program specified by series_path is canceled, and success or failure is returned using a return value. However, if the absolute time when this function was executed has passed the time specified by expire_date, it is judged that the designation of series_path is invalid and NaN is returned.

The groupCRID is specified for series_path. The groupCRID is described in the groupeId of the group information element. For a description of series_path, follow the regulation of name space in this technical data "Vol. 10, Chapter 8 Operation of identification information of contents and metadata".

- X_TMM_ecgLaunchContent(): Transfer to a specified storage-type broadcast content and launch playback of the content.

Syntax:

```
Number  X_TMM_ecgLaunchContent(
                    input Number disp_mode
                    ,input String content_path
                    )
```

Argument:

disp_mode	Playback presenting position 0: Displayed in the data broadcasting block 1: Displayed in the video presenting block 2: Displayed on a full screen
content_path	URI of usage-unit content (resource) at the transfer destination

Return value:

1:	Success
NaN:	Failure

Explanation:

Playback of a specified usage-unit content or resource is started.

The playback presenting position is specified for disp_mode.

When disp_mode = 0 is specified, playback is performed in the data broadcasting presentation area. In this case, the browser may be stopped. When disp_mode = 1 is specified, playback is performed in the video presentation area. In this case, do not stop the browser. Also, switch the video presentation area to the video being broadcast after playback is finished.

When disp_mode = 2 is specified, playback is performed on a full screen, and the application of the real-time-type broadcast receiving function may be exited.

However, since disp_mode is only reference information for the receiver application, it does not always need to be followed, and the operation with always set to disp_mode ＝2 is acceptable.

The Content Reference Identifier (= CRID) is specified for content_path.

Refer to this technical data “Vol. 10, Chapter 8 Operation of identification information of contents and metadata” for details of identification of usage-unit content and the corresponding relationship with the name space.

- X_TMM_ecgStoreStart(): Start storage of storage-type broadcast. (TBD)

Syntax:

```
Number  X_TMM_ecgStoreStart(
                        input String content_path
                        ,input String es_ref
                        )
```

Argument:

content_path URI of usage-unit content (resource)

es_ref Component through which the usage-unit content (resource) is transmitted

Return value:

1: Success of start of storage

NaN: Failure

Explanation:

It is instructed to start storing the usage-unit content (resource) specified by content_path, and success or failure is returned using a return value.

The Content Reference Identifier (= CRID) is specified for content_path.

Refer to this technical data “Vol. 10, Chapter 8 Operation of identification information of contents and metadata” for details of identification of usage-unit content and the corresponding relationship with the name space.

The component through which the usage-unit content (resource) is transmitted is specified as es_ref. For the name space that identifies the component ES, follow ARIB STD-B24, Vol. 2, “9.2.11 Identification of component ES”, and no abbreviation shall be used.

If another storing process is executed for the usage-unit content (resource) of the storage destination, the function behavior will fail.

- X_TMM_ecgStoreTerminate(): Terminate the storage process of storage-type broadcast. (TBD)

Syntax:

```
Number  X_TMM_ecgStoreTerminate(
                        input String content_path
 )
```

Argument:

content_path URI of usage-unit content (resource)

Return value:

1: Success

NaN: Failure

Explanation:

Storing the usage-unit content (resource) specified by content_path is terminated, and success or failure is returned using a return value.

The Content Reference Identifier (= CRID) is specified for content_path.

Refer to this technical data "Vol. 10, Chapter 8 Operation of identification information of contents and metadata" for details of identification of usage-unit content and the corresponding relationship with the name space.

- X_TMM_ecgCheckStorageStatus(): Inquires about the storage status of storage-type broadcast.

Syntax:

```
Number  X_TMM_ecgCheckStorageStatus(
                                input String content_path
 )
```

Argument:

content_path URI of usage-unit content (resource)

Return value:

1: During storage (It is already instructed to start storage.)

0: Not during storage

NaN: Failure

Explanation:

The storage process status of the usage-unit content (resource) specified by content_path returned using a return value.

The Content Reference Identifier (= CRID) is specified for content_path.

Refer to this technical data "Vol. 10, Chapter 8 Operation of identification information of contents and metadata" for details of identification of usage-unit content and the corresponding relationship with the name space.

- X_TMM_ecgStoreReserve(): Reserve the storage of storage-type broadcast.

Syntax:

```
Number  X_TMM_ecgStoreReserve(
```

input String content_path

)

Argument:

content_path URI of usage-unit content (resource)

Return value:

1: Success

NaN: Failure

Explanation:

Storing a usage-unit content (resource) specified by content_path is reserved, and success or failure is returned using a return value.

The Content Reference Identifier (= CRID) is specified for content_path.

Refer to this technical data "Vol. 10, Chapter 8 Operation of identification information of contents and metadata" for details of identification of usage-unit content and the corresponding relationship with the name space.

The data necessary for storage is acquired from the transmission control metadata based on CRID and stored.

Refer to this technical data "Vol. 11, 2.1.5.4 Transmission control metadata" for the transmission control metadata.

- X_TMM_ecgStoreCancelReservation(): Cancel storage reservation of storage-type broadcast.

Syntax:

Number X_TMM_ecgStoreCancelReservation(

input String content_path

)

Argument:

content_path URI of usage-unit content (resource)

Return value:

1: Success

NaN: Failure

Explanation:

The storage reservation of the usage-unit content (resource) specified by content_path is canceled, and success or failure is returned using a return value.

The Content Reference Identifier (= CRID) is specified for content_path.

Refer to this technical data "Vol. 10, Chapter 8 Operation of identification information of contents and metadata" for details of identification of usage-unit content

and the corresponding relationship with the name space.

- X_TMM_ecgStoreCheckReservation(): Check the status of storage reservation of storage-type broadcast. (T BD)

Syntax:

```
Number    X_TMM_ecgStoreCheckReservation(
                                    input String content_path
          )
```

Argument:

content_path URI of usage-unit content (resource)

Return value:

2: Storage is already reserved
1: Not yet reserved (overlapped with another reservation)
0: Not yet reserved (not overlapped with another reservation)
NaN: Failure

Explanation:

The storage reservation of the usage-unit content (resource) specified by content_path is checked whether storage is being reserved and whether it is overlapped with another reservation, and return is made using a return value.

The Content Reference Identifier (= CRID) is specified for content_path.

Refer to this technical data “Vol. 10, Chapter 8 Operation of identification information of contents and metadata” for details of identification of usage-unit content and the corresponding relationship with the name space.

- X_TMM_ecgIsContentStored(): Verify whether the usage-unit content has already been stored.

Syntax:

```
Boolean   X_TMM_ecgIsContentStored(
                                    input String content_path
          )
```

Argument

content_path URI of usage-unit content (resource)

Return value:

true: The usage-unit content is stored
false: The usage-unit content is not stored

Explanation:

The storage status of the usage-unit content specified by content_path is acquired.

The Content Reference Identifier (= CRID) is specified for content_path.

Refer to this technical data "Vol. 10, Chapter 8 Operation of identification information of contents and metadata" for details of identification of usage-unit content and the corresponding relationship with the name space.

If storage of the usage-unit content has been complete, true is returned.

- X_TMM_lockStoredContent (): Lock usage-unit content. (TBD)

Syntax:

```
Number  X_TMM_lockStoredContent (
                                    input String content_path
)
```

Argument:

content_path	URI of usage-unit content (resource)

Return value:

1:	Success
-1:	Error of parameter
-820:	No specified usage-unit content
-821:	The maximum number of lockable usage-unit contents has been exceeded.
-822:	Lock is not possible due to other processes (during storage of relevant usage-unit content).
-823:	The specified usage-unit content has already been locked.
Nan	Other failure

Explanation:

A specified usage-unit content is locked so that it is not deleted.

The Content Reference Identifier (= CRID) is specified for content_path.

Refer to this technical data "Vol. 10, Chapter 8 Operation of identification information of contents and metadata" for details of identification of contents in units of use and the corresponding relationship with the name space.

"Lock" of contents in units of use is intended to protect contents in units of use from being deleted, and does not inhibit any playback (transfer) operation or updating process.

The lock is automatically released when quitDocument() in executed in contents in

units of use at the lock source, or when transfer is made from contents in units of use at the lock source to other contents in units of use. The maximum number of contents in units of use that can be locked at the same time is five. If a deleting instruction is given to locked contents in units of use from the contents, the receiver operation will fail. The operation of deletion by a resident application is a matter of product planning.

- X_TMM_unlockStoredContent (): Release the lock of usage-unit content. (TBD)

Syntax:

```
Number  X_TMM_unlockStoredContent (
                                    input String content_path
 )
```

Argument:

content_path URI of usage-unit content (resource)

Return value:

1: Success

-824: The specified usage-unit content is not locked.

Explanation:

The lock of usage-unit content is released.

The Content Reference Identifier (= CRID) is specified for content_path.

Refer to this technical data “Vol. 10, Chapter 8 Operation of identification information of contents and metadata” for details of identification of usage-unit content and the corresponding relationship with the name space.

- X_TMM_ecgIsLockedStoredContent(): Judge whether usage-unit content is locked. (TBD)

Syntax:

```
Boolean  X_TMM_ecgIsLockedStoredContent(
                                    input String content_path
 )
```

Argument:

content_path URI of usage-unit content (resource)

Return value:

true: Locked

false: Not locked

Explanation:

It is judged whether the usage-unit content specified by content_path is locked.

The contents referral identifier (= CRID) is specified for content_path.

Refer to this technical data "Vol. 10, Chapter 8 Operation of identification information of contents and metadata" for details of identification of usage-unit content and the corresponding relationship with the name space.

If the relevant usage-unit content does not exist, false is returned.

- X_TMM_ecgDeleteStoredContent(): Delete usage-unit content. (TBD)

Syntax:

```
Number  X_TMM_ecgDeleteStoredContent(
                                input String content_path
)
```

Argument:

content_path	URI of usage-unit content (resource)

Return value:

1:	Success
-1:	Error of parameter
-820:	No specified usage-unit content
-823:	Specified usage-unit content is already being locked.
-825:	Specified usage-unit content is being played back and cannot be deleted.
-826:	Specified usage-unit content is being stored and cannot be deleted.
NaN:	Other failures

Explanation:

The usage-unit content specified by content_path is deleted.

The Content Reference Identifier (= CRID) is specified for series_path.

Refer to this technical data "Vol. 10, Chapter 8 Operation of identification information of contents and metadata" for details of identification of usage-unit content and the corresponding relationship with the name space.

If the relevant contents in units of use do not exist, false is returned.

When this function is used, it is preferable that the function is executed after user instruction or approval is given.

- X_TMM_ecgGetStorableSpace(): Acquire the storable free space of the receiver's storage device. (TBD)

Syntax:

Number X_TMM_ecgGetStorableSpace(
input Number drive_type
)

Argument:

drive_type The drive type is specified
0: Receiver body
1: External memory of the receiver

Return value:

Value of free space (in units of megabytes): Success
NaN: Failure

Explanation:

The currently storable free space in the receiver's storage device is returned using a return value.

The space is in units of megabytes (MB).

1 MB = 1,000,000 bytes. If the storage space is less than 1 MB, "0" is returned.

The return value is only for reference and does not guarantee an exact free space.

- X_TMM_startExFullDisplay(): Setting the data display area to full screen.

Syntax:

Number X_TMM_startExFullDisplay (
input Array setting_info
)

Argument:

setting_info Data display area full-screen setting information

Array[0][0] (Number): Absolute X-coordinate position on the data broadcast virtual plane using the top left of the video being presented as a reference
Array[0][1] (Number): Absolute Y-coordinate position on the data broadcast virtual plane using the top left of the video being presented as a reference
Array[1][0] (Number): X-coordinate system size of the video being presented
Array[1][1] (Number): Y-coordinate system size of the video being presented

Return value:

1: Success

NaN: Failure

Explanation:

Data is displayed on a full screen, and video is presented at a specified position. Audio should be in consecutive playback status.

After this function is executed, the data display full screen continues the full-screen display until X_TMM_stopExFullDisplay() is executed.

However, if switching of the direction of presentation or switching of data event is detected during channel selection, the full-screen display is finished, the BML document being displayed will be destructed, and the startup document of entry module will be acquired. However, while a link content is being displayed, it is not necessary to detect switching of a data event.

Array[0][0] and Array[0][1] show the positions of the X and Y coordinate system using the top left of the data broadcast virtual plane of the video being presented as a standard. Array[1][0] and Array[1][1] show the video size of the video ES presented in the data display block. In this case, the receiver will enlarge or reduce the video to a specified size. A value of 1 or higher is set to Array[1][0] and Array[1][1]. The expression that no video is displayed during full-screen data display is feasible by setting 0 to Array[0][0] and Array[0][1] and setting 1 to Array[1][0] and Array[1][1].

See the following when executing functions in the vertical display:

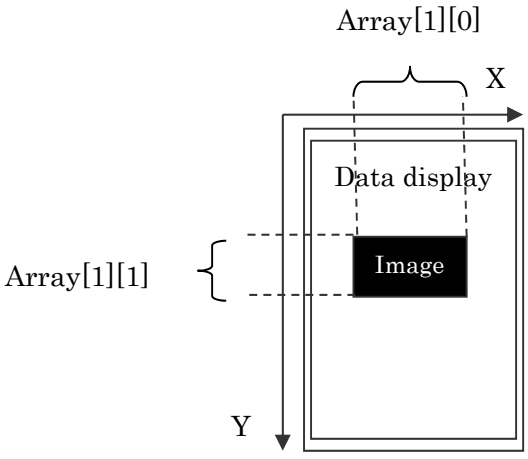


Figure 3-3-6: Example of display during vertical display

See the following when executing functions in the horizontal display:

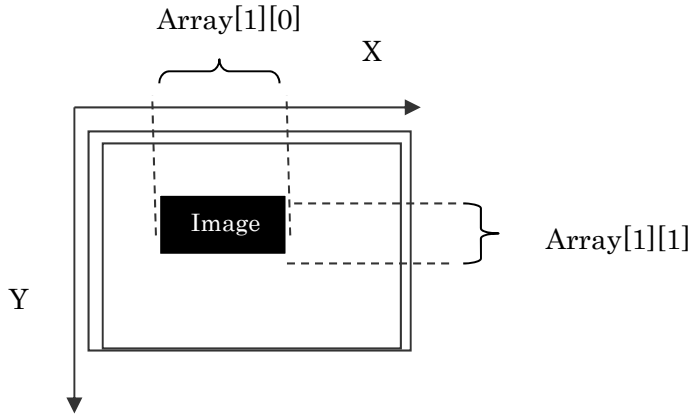


Figure 3-3-7: Example of display during　horizontal display

The processing on the receiver side for the receiver that supports this function is assumed to be achieved by the description of content as follows: The video display area is located against the background of the colored body element and the video display area part of body element is hollowed by the div element of opacity0, or the video display area is located against the background of the transparent body element, and parts other than the video display area are covered by the div element as needed. It is also assumed that no data broadcast is presented in the video display area, but it is preferable from the viewpoint of content production that nothing other than the body element or the above-mentioned div element (opacity0) is located in the video display area specified when this function is executed. If any other element is located, it is preferable for the operation to follow the restrictions presented on the screen plane in Table 2-2. The operation when the specified coordinates exceed the resolution of the broadcast virtual plane is a matter of product planning.

Handling with the scroll function when the data broadcast full-screen is displayed by this function during vertical display is a matter of product planning, and the scroll function may be invalid.

- X_TMM_stopExFullDisplay(): Exit the full screen of data display area.

Syntax:

Number　X_TMM_stopExFullDisplay ()

Argument:

None

Return value:

1: Success

NaN: Failure

Explanation:

The full screen of the data display that was executed by X_TMM_startExFullDisplay() is exited. When exiting with this function, be careful that X_TMM_startExFullDisplay() is not executed again after it is exited.

3.4.10.8 Execution operation of the extended function

It is assumed that the processes of X_TMM_startResidentApp(), X_TMM_mailTo(), X_TMM_phoneTo(), X_TMM_startExAV(), X_TMM_writeSchInfo(), X_TMM_writeAddressBookInfo () are asynchronously processed among the extended functions specified by 3.4.10.7 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.

3.4.11 Restrictions in BML document descriptions

3.4.11.1 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)

3.4.11.2 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 follow:

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)

abcdefghij

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.

3.4.11.3 Vertical writing

The specification of the direction of writing by the BML document is not operated. Vertical writing is not operated.

3.4.11.4 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 3.2.2.11.

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 3-49: 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

3.4.11.5 Operation related to object elements

Note) The presentation behaviour of individual mono-media presented by object elements is specified in section 3.4.12.3.

3.4.11.5.1 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 section 3.4.12.6.2.

- 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.

3.4.11.6 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.

3.4.11.7 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 for 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.

3.4.12 Operational guidelines related to presentation

3.4.12.1 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 3-3-9. 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 3-3-9.

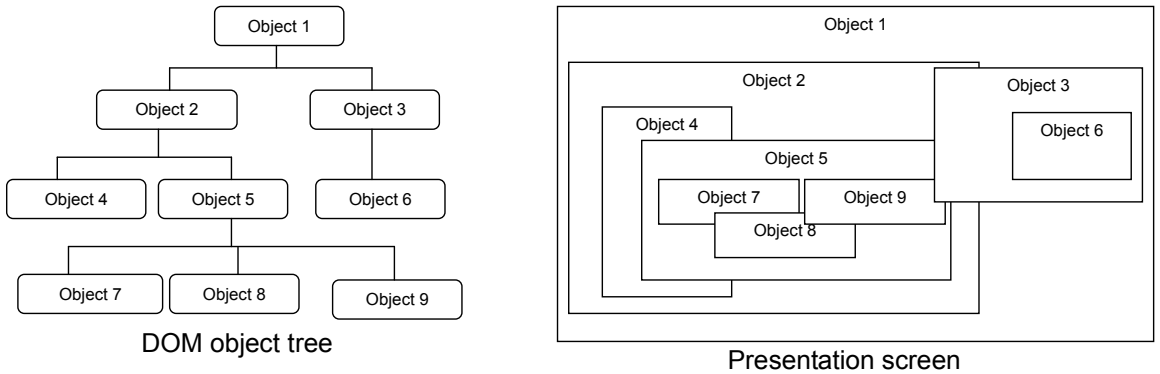


Figure 3-3-9: Structure and presentation of DOM objects

3.4.12.2 Presentation plane

Rendering of the objects described in section 3.12.1 is done in a virtual plane for a BML browser shown in section 2.2.1. 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 relevant virtual plane has 240 pixels horizontal x 480 pixels vertical during vertical display, and 426 pixels horizontal x 240 pixels vertical during horizontal display.

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 2.2.2.

3.4.12.3 Operation of mono-media presentation

The following shows operation of mono-media presentation using an object elements and img elements.

3.4.12.3.1 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 3-50 .

Table 3-50: 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 3-51. 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 3-51: 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 .

■ Vertical and horizontal pixel numbers

W (Image horizontal pixel number), H (Image vertical pixel number) will take the values of Table 3-52.

Table 3-52: Values applicable to W and H

resolution property	240 x 480		426 x 240	
type attribute (Media type)	W	H	W	H
image/jpeg	16-240	16-320	16-426	16-240
image/gif (case of GIF)	2-240	2-320	2-426	2-240
image/gif (case of animation GIF)	2-240	2-240	2-240	2-240

3.4.12.4 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 3-3-10. 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.

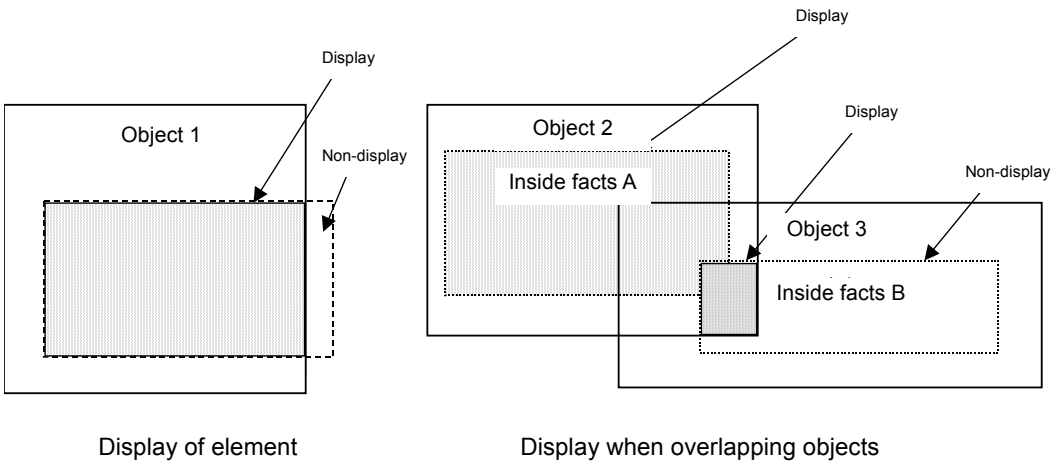


Figure 3-3-10: Example of a clipping behaviour

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 mono-media 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.

3.4.12.5 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 3-53 .

Table 3-53: 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

3.4.12.6 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 3-3-11. Moreover, the arbitrary boxed elements shown by section 3.4.11.2 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.

In case of vertical display:

$x0 \geq -240$, $y0 \geq -480$, $x0 + w \leq 480$, $y0 + h \leq 960$, $0 \leq w \leq 240$, $0 \leq h \leq 480$

In case of horizontal display:

$x0 \geq -426$, $y0 \geq -240$, $x0 + w \leq 852$, $y0 + h \leq 480$, $0 \leq w \leq 426$ $0 \leq h \leq 240$

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.

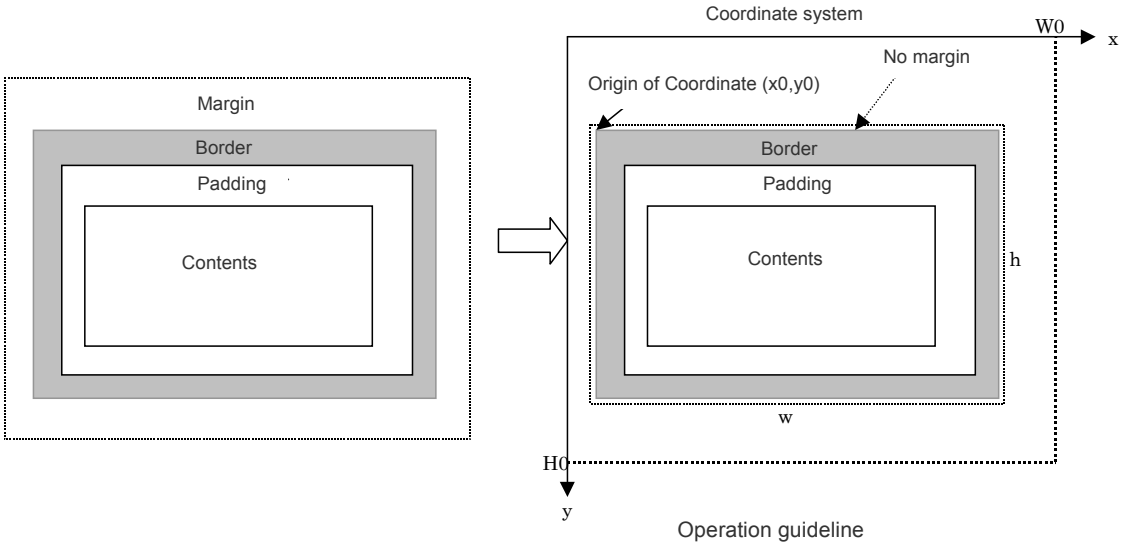


Figure 3-3-11: Presentation of the box model

3.4.12.6.1 Attribute operation related to stream presentation behaviour

Table 3-54 shows the operation of the streamstatus attribute of object elements

Table 3-54: 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.

3.4.12.6.2 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 3-55.

Table 3-55: 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 3-56.

Table 3-56: 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.	-

Chapter 4 Guidelines for BML Browser's Behaviors

4.1 Guidelines for presentation

The guidelines for the presentation function of X-profile receivers are shown below.

In addition to the description in this chapter, refer to the service guidelines of each broadcaster.

4.1.1 Presentation of video and audio

4.1.1.1 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 720x480 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 720x480 or higher.

4.1.1.2 Optimizing video display size

With receivers whose display device pixel size is not 720 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.

4.1.2 Specifications regarding communication contents presentation

4.1.2.1 Contents presented in BML browsers

BML browsers are browsers implemented based on these specifications, and present data broadcasting contents and X-profile linked contents

4.1.2.2 Activation of communication content other than X-profile linked content

If unlinked communication content will be activated from data broadcasting or X-profile linked content, the content shall be presented by activating another receiver-specific communication browser using X_TMM_startResidentApp().

4.1.3 Presentation of multimedia data

4.1.3.1 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.

4.1.3.2 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.

4.1.3.3 Character fonts and possible number of characters for display

For medium-size characters, fonts of 20 dots or less are used. For example, it is assumed that at least 12 characters can be displayed horizontally in a vertical-display BML document.

For large-size characters, fonts of 30 dots or less are used. For example, it is assumed that at least 8 characters can be displayed horizontally in a vertical-display BML document.

For small-size characters, fonts of 16 dots or less are used. For example, it is assumed that at least 15 characters can be displayed horizontally. Also, small-size characters may not be able to be implemented due to the display device pixel size, and implementation may not be possible in such case.

4.1.3.4 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.

4.1.4 Guidelines for contents creation

Below are the guidelines to be followed when broadcasters, etc. create contents etc.

4.1.4.1 Pixel size assumed in contents creation

The resolution of the virtual plane for data broadcasting browser assumed by contents creators is 240 pixels horizontal x 480 pixels vertical (vertical display), and 426 pixels horizontal x 240 pixels vertical (horizontal display).

4.1.4.2 Character size controls and layout to be assumed in contents

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.

4.1.4.3 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.

4.1.4.4 CSS

Always perform positioning specification by the CSS. Display is not performed accordingly if the CSS is not written.

4.1.4.5 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.

4.1.4.6 BML documents for vertical display and for horizontal display

The BML documents for vertical display and for horizontal display acquire the startup

document identified by <resourceName>, which is contained in the entry module and matches the direction of presentation according to the direction of presentation, and displays it in the data broadcast area in accordance with the presentation area.

However, since documents other than the startup document cannot be identified by <resourceName>, the receiver presents the document transferred from each startup document, and if the direction of presentation does not match the resolution property, the presentation could collapse. To prevent this, when contents are created, the BML document for vertical display must be definitely distinguished from the BML document for horizontal display, and sufficient care must be taken when contents are created so that no transfer occurs between the documents.

The receiver operation when a transfer between the BML documents for vertical display and for horizontal display occurs is a matter of product planning. If transfer to a document with a different display direction occurs, it is preferable that the relevant document is regarded as an error and the transfer source document is presented again. However, if the presentation is not possible, re-reading of the startup document is performed.

4.1.4.7 BML document for horizontal display when superimpose is operated

When superimpose is operated during horizontal display, it is forecast that the display will overlap the data broadcasting display during horizontal display. Therefore, consider the position where the receiver presents superimpose when creating the contents, taking into account the superimpose display area shown in 4.1.5.

However, when superimpose is not operated or when it is not necessary to consider the above in actual operation, the above description be ignored.

4.1.5 Display position of closed caption/superimpose

For receivers that can start up closed caption/superimpose and the data broadcasting browser, consider the following when packaging the receiver.

For vertical display: When the receiver cannot take the closed caption/superimpose display area in an area other than the video display area or data broadcasting display area, the closed caption/superimpose can be displayed in the data broadcasting display area by user operation. Also, if the visibility of a video is not degraded even if the closed caption/superimpose is displayed, overlapping the video display area, the closed caption/superimpose may be displayed over the video display area.

For horizontal display: For the recievers, which does not have a separate closed caption/superimpose display area not overlapping with video and data broadcasting area, it is preferable that the presentation position of closed caption/superimpose is set for the receiver, by assuming that the upper half of the screen (120x426) is transparent. Also, it is assumed

that information of higher urgency is operated for superimpose, so packaging is different from that of closed caption, so that data broadcast is not presented. However, this does not need to be especially considered for the receivers that secure the display area of closed caption/superimpose in an area other than the video display area.

4.1.6 Guidelines for DOM operation

Refer to ARIB STD-B24 Vol. 2 Attachment 4 “5.3. Guidelines on behaviors of DOM”. However, ignore the comments and do not generate the comment node.

4.1.7 Guidelines for external characters operation

External characters are not operated in X-profile.

4.1.8 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 Attachment 4 “5.4.1. Operation of script working environment”
- ARIB STD-B24 Vol. 2 Attachment 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”

4.1.9 Guidelines for operation of extended objects for broadcasting

4.1.9.1 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.

4.1.10 Guidelines for browser pseudo-object operations

Operations of browser pseudo-object are provided below. Follow,

“Attachment 1 Operational guidelines”

“Attachment 2 Operation guidelines for implementing basic services”

in ARIB STD-B24 Vol. 2 “XML-based multimedia coding scheme” for those not specifically recorded.

4.1.10.1 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 Attachment 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.

4.1.10.2 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 X-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.

4.1.10.3 Operation of EPG functions

In order to correspond to EPG functions, L-EIT reception is needed by the receiver. However, functions operated in X-profile are epgGetEventStartTime() and epgGetEventDuration(), so compatibility with 8-bit character codes is not necessary.

(1) epgGetEventStartTime()

The events that can be specified by event_ref are those that are currently being presented; acquire the start_time value, referring to the W-EIT or EPG/ECG metadata.

(2) epgGetEventDuration()

The events that can be specified by event_ref are those that are currently being presented; acquire the duration value, referring to the W-EIT or EPG/ECG metadata.

(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.

(4) epgTune()

The channel selection services by this function are the real-time-type broadcasting services of multimedia broadcasting. The operation to be performed when other media are specified is a matter of product planning.

4.1.10.4 Interaction channel communication – Operation of TCP/IP

Refer to section 4.3.6.3.

4.1.10.5 Operation of the operational control function

Follow STD-B24 for operations of functions not especially relevant to this clause.

Refer to section 4.3.6.4 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 (0x40 0x41) modules that transmit Xprofile data carousels, independent from whether it is being presented or not, are always lockable using lockModuleOnMemoryEx(). 0x40 0x41 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 Attachment 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 Attachment 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 Attachment 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 section 4.3.7 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 4-1 is a summary of conditions to release lockModuleOnMemoryEx.

Table 4-1: 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 component viewing)	Unlock
Transition to link state	No change
When another application was started up by using X_TMM_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 4-4-1. 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().

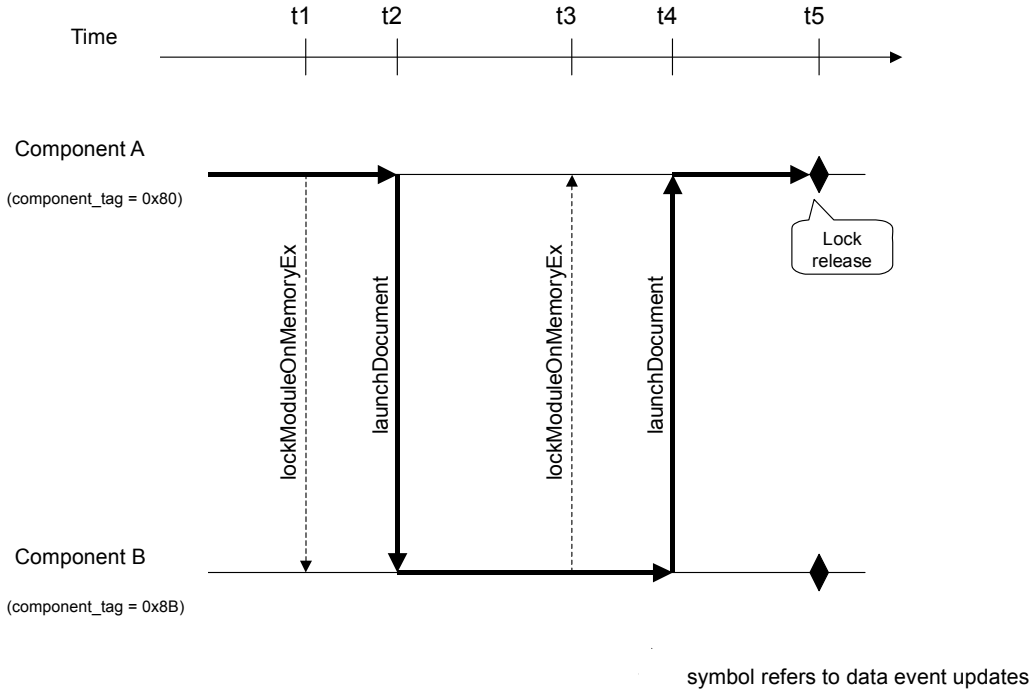


Figure 4-4-1: Example 1 of lockModuleOnMemoryEx operations

Operation example 2

A complicated operation example of lockModuleOnMemoryEx() is shown in Figure 4-4-2. 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().

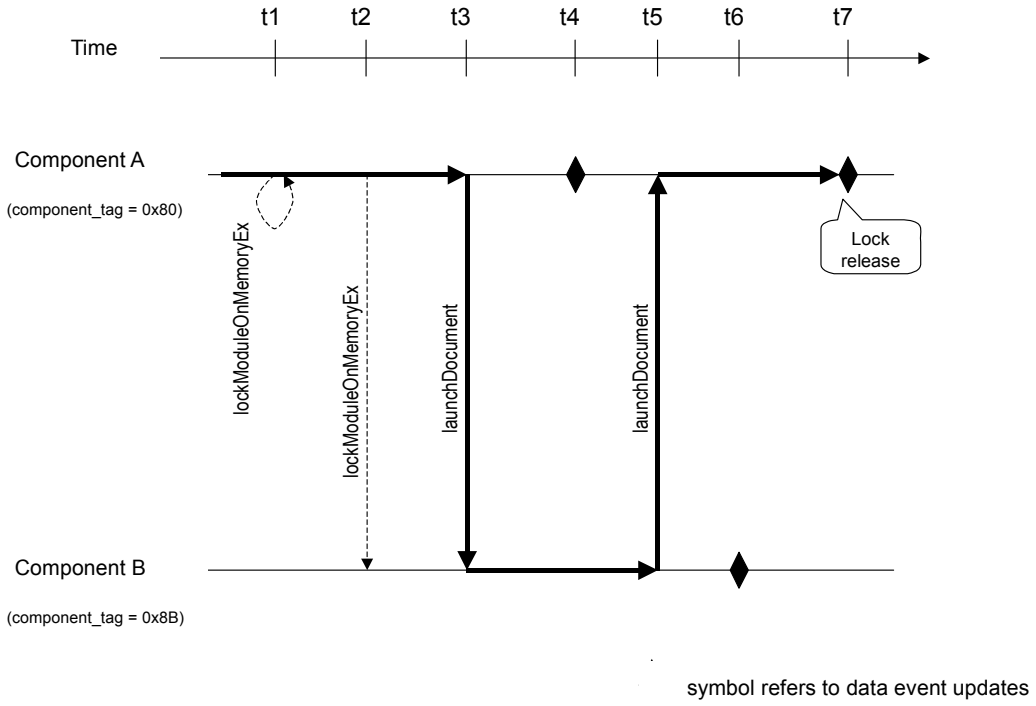


Figure 4-4-2: Example 2 of lockModuleOnMemoryEx operations

(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 X-profile.

(4) Operation of quitDocument()

Refer to "4.3.6.4 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 section 4.2.3 to use the broadcasting contents, and use abbreviations starting with "/" for URI's that specify components. Follow the specifications in section 4.3.10.4 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()

The program start time (start_time) described in the W-EIT or EPG/ECG data and the current time acquired with the receiver will be returned in units of seconds.

(8) Operation of getActiveDocument()

Return abbreviated URI values (relative path starting with "/").

(9) Operation of getBrowserStatus()

Refer to Annex 14 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 X-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.

- "TMMXpro" 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 4-2: Value of additional info

	Function	additionalinfo
Ureg related functions		
	Ureg[]	Misc.Basic
Greg related functions		
	Greg[]	Misc.Basic
EPG functions		
	epgGetEventStartTime()	EPG.Basic
	epgGetEventDuration()	EPG.Basic
	epgTune()	EPG.Basic
	epgTuneToComponent()	
	epgTuneToDocument()	
	epgIsReserved()	
	epgReserve()	
	epgCancelReservation()	

	Function	additionalinfo
	epgRecIsReserved()	
	epgRecReserve()	
	epgRecCancelReservation()	
Event group index functions		
	grpIsReserved()	
	grpReserve()	
	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		

	Function	additionalinfo
	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()	
	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		

	Function	additionalinfo
	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
	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

	Function	additionalinfo
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()	
	getContentSource()	
	getStorageInfo()	
Data carousel storage functions		
	saveCarouselAs()	
	saveCarousel()	
	saveModuleAs()	
	saveModule()	
	saveResourceAs()	
	saveResource()	
Bookmark control functions		
	writeBookmarkArray()	

	Function	additionalinfo
	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()	
Multimedia broadcasting-specific functions		
	X_TMM_mailTo()	Xtmm.mailTo
	X_TMM_startResidentApp()	Xtmm.RAStart
	X_TMM_phoneTo()	Xtmm.phoneTo
	X_TMM_getRcvCond ()	Xtmm.RcvCond
	X_TMM_getCurPos ()	Xtmm.CurPos
	X_TMM_saveExAppFile()	Xtmm.saveExApp
	X_TMM_startExAV()	Xtmm.startExAv
	X_TMM_stopExAV()	Xtmm.stopExAv
	X_TMM_writeSchInfo()	Xtmm.SchInfo
	X_TMM_getComBrowserUA ()	Xtmm.ComBrowserUA
	X_TMM_writeAddressBookInfo()	Xtmm.AddressBook
	X_TMM_launchDocWithLink()	Xtmm.launchWithL
	X_TMM_chkAVtype()	Xtmm.chkAV
	X_TMM_getIRDID()	Xtmm.getIRDID
	X_TMM_writeXproBM()	Xtmm.XproBM
	X_TMM_epgReserve()	Xtmm. epgRsv
	X_TMM_epgCancelReservation()	Xtmm. epgCancelRsv
	X_TMM_epgCheckReservation()	
	X_TMM_epgSeriesCheckReservation()	
	X_TMM_epgSeriesReserve()	Xtmm.epgSeriRsv
	X_TMM_epgSeriesCancelReservation()	Xtmm.epgSeriCancelRsv
	X_TMM_epgRecCheckReservation()	
	X_TMM_epgRecReserve()	Xtmm.epgRecRsv
	X_TMM_epgRecCancelReservation()	Xtmm.epgRecCancelRsv
	X_TMM_epgSeriesRecCheckReservation()	
	X_TMM_epgSeriesRecReserve()	Xtmm.epgSeriRecRsv
	X_TMM_epgSeriesRecCancelReservation()	Xtmm.epgSeriRecCancellRsv
	X_TMM_ecgLaunchContent()	Xtmm.ecgLaunchCnt
	X_TMM_ecgStoreStart()	Xtmm.ecgStoreStart
	X_TMM_ecgStoreTerminate()	Xtmm.ecgStoreTerminate

	Function	additionalinfo
	X_TMM_ecgCheckStorageStatus()	Xtmm.ecgChkStorageSts
	X_TMM_ecgStoreReserve()	Xtmm.ecgStoreRsv
	X_TMM_ecgStoreCancelReservation()	Xtmm.ecgStoreCancellRsv
	X_TMM_ecgStoreCheckReservation()	Xtmm.ecgStoreChkRsv
	X_TMM_ecgIsContentStored()	Xtmm.ecgIsCntStored
	X_TMM_lockStoredContent()	Xtmm.lockStoredContent
	X_TMM_unlockStoredContent()	Xtmm.unlockStoredContent
	X_TMM_ecgIsLockedStoredContent()	Xtmm.ecgIsLockedStoredCnt
	X_TMM_ecgDeleteStoredContent()	Xtmm.ecgDelStoredCnt
	X_TMM_ecgGetStorableSpace()	Xtmm.ecgGetStorableSp
	X_TMM_startExFullDisplay()	Xtmm.startExFullDisp
	X_TMM_stopExFullDisplay()	Xtmm.stopExFullDisp

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
WriteXproBM	"BMtype02"	If the receiver can write TVlink of XproBMtype=2, return 1
	"BMtype03"	If the receiver can write TVlink of XproBMtype=3, return 1
	"BMtype04"	If the receiver can write TVlink of XproBMtype=4, return 1
OSDResolution	"240x480"	If a communication browser is installed in the receiver as a receiver's native application, return 1
	"426x240"	If the receiver has a 426x240 coordinate area as the virtual plane for the data broadcasting browser, 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	If it is possible for the browser to implement all of

	specification Refer to Table 6-3 for correspondence to extended function for each broadcaster and extended function group specification.	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.

4.1.10.6 Operation of non-volatile memory functions

(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 Attachment 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().

4.1.10.7 Operation of functions related to TVlinks

When recording X-profile contents (XproBMtype=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.
5. Event StoreFinished does not occur.

4.1.10.8 Operation of special function for terrestrials

(1) Operation of X_TMM_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_TMM_phoneTo () (optional)

This calls the telephone number specified by the argument. The calling operation and permission procedure are implementation dependent.

(3) Operation of X_TMM_ 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_TMM_ 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_TMM_saveExAppFile() (T.B.D.)

(6) Operation of X_TMM_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.

(7) Operation of X_TMM_stopExAV () (optional)

This stops the presentation of video sound streamed via communication presented by the (X_TMM_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_TMM_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.

(9) Operation of X_TMM_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.

(10) Operation of X_TMM_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.

(11) Operation of X_TMM_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.

(12) Operation of X_TMM_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.

(13) Operation of X_TMM_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_TMM_startExAV ()) function implementation.

(14) Operation of X_TMM_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.

(15) Operation of X_TMM_epgReserve() (optional)

This function reserves viewing of a specified real-time-type broadcasting program.

(16) Operation of X_TMM_epgCancelReservation () (optional)

This function cancels the specified viewing reservation of a real-time-type broadcasting program.

(17) Operation of X_TMM_epgSeriesReserve () (optional)

This function reserves viewing of a specified real-time-type broadcasting series. The series described in groupCRID of EPG/ECG metadata is a target.

(18) Operation of X_TMM_epgSeriesCancelReservation () (optional)

This function cancels the specified viewing reservation of a real-time-type broadcasting series.

(19) Operation of X_TMM_epgRecReserve () (optional)

This function reserves recording of a specified real-time-type broadcasting program.

(20) Operation of X_TMM_epgRecCancelReservation () (optional)

This function cancels the specified reservation of recording of a real-time-type broadcasting program.

(21) Operation of X_TMM_epgSeriesRecReserve () (optional)

This function reserves recording of a specified real-time-type broadcasting series. The series described in groupCRID of EPG/ECG metadata is a target.

(22) Operation of X_TMM_epgSeriesRecCancelReservation () (optional)

This function cancels the specified reservation of recording of a real-time-type broadcasting series.

(23) Operation of X_TMM_ecgLaunchContent () (optional)

This function executes playback of specified storage-type broadcasting contents. The presentation status, including the playback position, is a matter of product planning.

(24) Operation of X_TMM_ecgStoreStart () (optional)

This function executes storage of specified storage-type broadcasting contents. The operation after storage starts is a matter of product planning.

(25) Operation of X_TMM_ecgStoreTerminate () (optional)

This function stops the storage process of a storage-type broadcast whose storage is started by X_TMM_ecgStoreStart ()

(26) Operation of X_TMM_ecgStoreReserve () (optional)

This function reserves storage of specified storage-type broadcasting contents. The operation when reservation of storage is overlapped is a matter of product planning.

(27) Operation of X_TMM_ecgStoreCancelReservation () (optional)

This function cancels the reservation of storage of specified storage-type broadcasting contents.

(28) Operation of X_TMM_ecgIsContentStored () (optional)

This function checks the storage reservation status of specified storage-type broadcasting contents.

(29) Operation of X_TMM_lockStoredContent () (optional)

This function locks specified storage-type broadcasting contents.

(30) Operation of X_TMM_unlockStoredContent () (optional)

This function releases the lock of specified storage-type broadcasting contents.

(31) Operation of X_TMM_ecgIsLockedStoredContent () (optional)

This function checks the lock status of specified storage-type broadcasting contents.

(32) Operation of X_TMM_ecgDeleteStoredContent () (optional)

This function deletes specified storage-type broadcasting contents.

(33) Operation of X_TMM_ecgGetStorableSpace () (optional)

This function acquires the storable free space in the receiver's storage device. However, since the value acquired as free space is information only for reference, use the value acquired by this function as a reference value when creating contents.

(34) Operation of X_TMM_startExFullDisplay () (optional)

This function displays a full screen of the data display area, and locates the reduced video in an arbitrary position. Assume that when data broadcasting is displayed on a full screen by this function, the scroll operation may be invalid when creating contents.

(35) Operation of X_TMM_stopExFullDisplay () (optional)

This function releases the full screen of data display area executed by X_TMM_startExFullDisplay().

4.1.10.9 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 Attachment 1 "Guidelines 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 X-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 4-2 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 section 3.2.2.1
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.

4.1.11 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.

4.1.12 Other restrictions

Refer to ARIB STD-B24 Vol. 2 Attachment 4 "5.7. Other restrictions"

4.2 Transmission, reference and name space of content

4.2.1 Scope mapping to the transmission system

Follow ARIB STD-B24 Vol. 2 Attachment 4 “6.1. Scope mapping to the transmission system”. Refer to 4.3.7 for communication contents.

4.2.2 Restrictions when a mono-media is referred across a different media

In X-profile, reference of mono-media, etc. is not performed other than in the following cases. Refer also to section 4.3.9 for communication contents.

- Tuning by functions for tuning
- Event message reference when the reference origin is X-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 X-profile linked contents (refer to Table 3-14)

4.2.3 Operations of name space

Follow ARIB STD-B24 Vol. 2 Attachment 4 “6.3 Namespace”. Refer to section 4.3.10 for X-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.

epgTune()

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 X-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.

However, when a usage-unit content or series program is referred to, CRID and groupCRID that exist in separate services can be referred to, and for operation of the name space, follow the provisions in this technical data “Vol. 10, Chapter 8 Operation of identification information of contents and metadata”.

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 X-profile linked contents placed in a server can be specified. For the “action” attribute for “form” element, X-profile linked contents placed in servers is specifiable. For X_TMM_launchDocWithLink(), X-profile linked contents placed in servers is specifiable.

Operation of #fragment is available, according to ARIB STD-B24 Vol. 2 Attachment 4 “6.3 Namespace”, but restrictions specified in 4.2.3.4 are established.

4.2.3.1 Restriction for the number of resources which are managed by receiver

The maximum number of resources (including both broadcasting contents and X-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

4.2.3.2 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().

4.2.3.3 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 X-profile within the same service.

4.2.3.4 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_TMM_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.

4.2.4 Reference guidelines of contents transmitted by components different from the BML document being presented

In X-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.

4.3 Operation of X-profile communicating content

4.3.1 Guidelines regarding presentation for X-profile communicating contents

Same as section 4.1 ”Guidelines for presentation”

4.3.2 Guidelines for operations of external fonts in X-profile communicating contents

Same as section 4.1.7 “Guidelines for external characters operation”

4.3.3 Operation of DOM in X-profile communicating contents

Same as section 4.1.6 “Guidelines for DOM operation”

4.3.4 Operation of ECMAScript implementation script in X-profile communicating contents

Same as section 4.1.8 “Guidelines for ECMA script implementation”

4.3.5 External objects for broadcasting in X-profile communicating contents

Same as section 4.1.9 “Guidelines for operation of extended objects for broadcasting”

4.3.6 Operation of browser pseudo-objects in X-profile communicating contents

The behaviourof the extended function for broadcasting which is in browser pseudo-objects in X-profile communicating contents differs depending on the status (data broadcasting reception status and link status) of the receiver. Refer to 4.3.7 for the basic concept of data broadcasting reception status and link status. Actual function operation is specified from Table 4-3 to Table 13. 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 X-profile contents on the internets. If the functi1on is called, receiver performs a failure behaviour. Refer to 4.3.11.4 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.

4.3.6.1 "Ureg" function

Operations of "Ureg" functions in X-profile linked contents are specified in Table 4-3. Refer to section 4.1.10.1 and section 4.1.10.2 for operations in data broadcasting reception status.

Table 4-3: Behaviours of the Ureg function in X-profile linked contents

	Link status
Ureg[ ]	O
Greg[ ]	O

4.3.6.2 EPG functions

Operations of EPG functions in X-profile linked contents are specified in Table 4-4. Refer to section 4.1.10.3 for behaviours in data broadcasting reception status. epgReserve() and epgCancelReservation() are options.

Table 4-4: Behaviour of EPG functions in X-profile links

	Link status
epgGetEventStartTime()	O
epgGetEventDuration()	O
epgReserve()	-
epgCancelReservation()	-

4.3.6.3 Interaction channel function -TCP/IP

Operations of TCP/IP are specified in Table 4-5. Details regarding behaviour of each function are described below.

Table 4-5 Behaviours of TCP/IP interaction channel communication in X-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 4-4-3 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 4-4-3: 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 4-4-4 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 4-4-4: 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 4-6 specifies detailed contents regarding return values.

Table 4-6: 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.

4.3.6.4 Operational control functions

Table specified the operation of function for operational control function in X-profile linked contents. Then, details for each function operation is provided. Refer to section 4.1.10.5 for function operations of the data broadcasting reception status.

Table 4-7: Operations of X-profile linked contents operational control function

	Link status
reloadActiveDocument()	O
getProgramRelativeTime()	O
getBrowserVersion()	O
getProgramID()	O
getActiveDocument()	O
lockScreen()	O
unlockScreen()	O

	Link status
getBrowserSupport()	O
launchDocument()	O
quitDocument()	O
detectComponent()	O
lockModuleOnMemoryEx()	X
unlockModuleOnMemoryEx()	O
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 X-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 X-profile linked content by re-transmitting the URI that specified the concerned BML document to the server.

  For example, the operation when X-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 X-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 X-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 section 4.3.10 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 section 4.3.11.3 for details.

- quitDocument() function behaviour

  Behaviour of the quitDocument() function differ depending on the data broadcasting reception status and link status. Refer to Table 4-8. Refer to section 4.3.11.3 for status transition.

Table 4-8: behavior of 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 section 3.1.1.5.1, 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 X-profile linked contents ends, and it is executed step 3 or later for "Receiver operation at the beginning of data broadcasting" described in section 3.1.1.5.1, 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 section 4.3.12.2. For example, the receiver 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 section 4.1.10.

- 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 section 4.1.10. 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 section 4.1.10. 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

section 4.1.10. 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 section 4.1.10.9 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 section 4.1.10.9 for details.

4.3.6.5 Receiver sound control

Table 4-9 specified functions behaviour of receiver sound control in X-profile linked contents.

Table 4-9: Behaviour of receiver sound control in X-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 section 4.1.10 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.

4.3.6.6 Timer functions

Table 4-10 specifies the behaviour for timer functions in X-profile linked contents. Refer to section 4.1.10 for behaviours in the data broadcasting reception status.

Table 4-10: Behaviour of timer functions in X-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.

4.3.6.7 Other functions

Table 4-11 specified function behaviour of other functions in X-profile linked contents. Refer to section 4.1.10 for the data broadcasting reception status.

Table 4-11: Behaviourof other functions in X-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.

4.3.6.8 Multimedia broadcasting specific functions

Table 4-12 shows the function behavior in the communication contents of multimedia broadcasting-specific functions.

Table 4-12: Operations of multimedia broadcasting specific functions

	Link status
X_TMM_mailTo()	O(*1)
X_TMM_startResidentApp()	O
X_TMM_phoneTo()	O(*1)
X_TMM_getRcvCond ()	O(*1)
X_TMM_getCurPos ()	O(*1)
X_TMM_saveExAppFile()	O(*1)
X_TMM_startExAV()	O(*1)
X_TMM_stopExAV()	O(*1)
X_TMM_writeSchInfo()	O(*1)
X_TMM_writeAddressBookInfo()	X
X_TMM_getComBrowserUA ()	O(*2)
X_TMM_launchDocWithLink()	O(*2)
X_TMM_chkAVtype()	O(*1)
X_TMM_getIRDID()	O
X_TMM_writeXproBM()	O(*2)
X_TMM_epgReserve()	O(*1)
X_TMM_epgCancelReservation()	O(*1)
X_TMM_epgCheckReservation()	X
X_TMM_epgSeriesCheckReservation()	X
X_TMM_epgSeriesReserve()	O(*1)
X_TMM_epgSeriesCancelReservation()	O(*1)
X_TMM_epgRecCheckReservation()	X
X_TMM_epgRecReserve()	O(*1)
X_TMM_epgRecCancelReservation()	O(*1)
X_TMM_epgSeriesRecCheckReservation()	X
X_TMM_epgSeriesRecReserve()	O(*1)
X_TMM_epgSeriesRecCancelReservation()	O(*1)
X_TMM_ecgLaunchContent()	O(*1)
X_TMM_ecgStoreStart()	O(*1)
X_TMM_ecgStoreTerminate()	O(*1)
X_TMM_ecgCheckStorageStatus()	O(*1)
X_TMM_ecgStoreReserve()	O(*1)
X_TMM_ecgStoreCancelReservation()	O(*1)
X_TMM_ecgStoreCheckReservation()	O(*1)

	Link status
X_TMM_ecgIsContentStored()	O(*1)
X_TMM_lockStoredContent()	O(*1)
X_TMM_unlockStoredContent()	O(*1)
X_TMM_ecgIsLockedStoredContent()	O(*1)
X_TMM_ecgDeleteStoredContent()	O(*1)
X_TMM_ecgGetStorableSpace()	O(*1)
X_TMM_startExFullDisplay	O(*1)
X_TMM_stopExFullDisplay	O(*1)

- Behaviour of X_TMM_mailTo()

  X_TMM_mailTo() is an option. Behaviour in link status is the same as the data broadcasting status. Refer to section 4.1.10.

- Behaviour of X_TMM_startResidentApp()

  Behaviour in link status is the same as the data broadcasting status. Refer to section 4.1.10.

- Behaviour of X_TMM_phoneTo ()

  X_TMM_phoneTo () is an option. Behaviour in link status is the same as the data broadcasting status. Refer to section 4.1.10.

- Behaviour of X_TMM_getRcvCond ()

  X_TMM_getRcvCond () is an option. Behaviour in link status is the same as the data broadcasting status. Refer to section 4.1.10.

- Behaviour of X_TMM_getCurPos ()

  X_TMM_getCurPos () is an option. Behaviour in link status is the same as the data broadcasting status. Refer to section 4.1.10.

- Behaviour of X_TMM_saveExAppFile()

  X_TMM_saveExAppFile() is an option. Behaviour in link status shall be the same as data broadcasting status. Refer to section 4.1.10.

- Behaviour of X_TMM_startExAV ()

  X_TMM_startExAV () is an option. Behaviour in link status is the same as the data broadcasting status. Refer to section 4.1.10.

- Behaviour of X_TMM_stopExAV ()

X_TMM_stopExAV () is an option. Behaviour in link status is the same as the data broadcasting status. Refer to section 4.1.10.

- Behaviour of X_TMM_tuneWithRF()

  Behaviour in link status is the same as the data broadcasting status. Refer to section 4.1.10.

- Behaviour of X_TMM_ writeSchInfo()

  X_TMM_ writeSchInfo() is an option. Behaviour in link status is the same as the data broadcasting status. Refer to section 4.1.10.

- Behaviour of X_TMM_writeAddressBookInfo()

  This does not operate in link status.

- Behaviour of X_TMM_getComBrowserUA()

  Behaviour in link status is the same as the data broadcasting status. Refer to section 4.1.10.

- Behaviour of X_TMM_getIRDID()

  Behaviour in link status is the same as the data broadcasting status.

- Behaviour of X_TMM_launchDocWithLink()

  This is mandate for receivers that possess communication functions. However, this function is allowed to use only in link status. If this function would be used in broadcasting status, the receiver shall fail and shall present broadcasting contents following section 4.3.11.4.

- Behaviour of X_TMM_chkAVtype()

  Behaviour in link status is the same as the data broadcasting status.

- Behaviour of X_TMM_writeXproBM()

  Behaviour in link status is the same as the data broadcasting status.

- Behavior of X_TMM_epgReserve()

  X_TMM_epgReserve () is optional. The behavior in link status is the same as in data broadcasting status.

- Behavior of X_TMM_epgCancelReservation ()

  X_TMM_ epgCancelReservation () is optional. The behavior in link status is the same as in

data broadcasting status.

- Behavior of X_TMM_epgSeriesReserve()

  X_TMM_epgSeriesReserve() is optional. The behavior in link status is the same as in data broadcasting status.

- Behavior of X_TMM_epgSeriesCancelReservation()

  X_TMM_epgSeriesCancelReservation() is optional. The behavior in link status is the same as in data broadcasting status.

- Behavior of X_TMM_epgRecReserve()

  X_TMM_epgRecReserve() is optional. The behavior in link status is the same as in data broadcasting status.

- Behavior of X_TMM_epgRecCancelReservation()

  X_TMM_epgRecCancelReservation() is optional. The behavior in link status is the same as in data broadcasting status.

- Behavior of X_TMM_epgSeriesRecReserve()

  X_TMM_epgSeriesRecReserve() is optional. The behavior in link status is the same as in data broadcasting status.

- Behavior of X_TMM_epgSeriesRecCancelReservation()

  X_TMM_epgSeriesRecCancelReservation() is optional. The behavior in link status is the same as in data broadcasting status.

- Behavior of X_TMM_ecgLaunchContent()

  X_TMM_ecgLaunchContent() is optional. The behavior in link status is the same as in data broadcasting status.

- Behavior of X_TMM_ecgStoreStart()

  _TMM_ecgStoreStart() is optional. The behavior in link status is the same as in data broadcasting status.

- Behavior of X_TMM_ecgStoreTerminate()

  X_TMM_ecgStoreTerminate() is optional. The behavior in link status is the same as in data broadcasting status.

- Behavior of X_TMM_ecgCheckStorageStatus()

X_TMM_ecgCheckStorageStatus() is optional. The behavior in link status is the same as the data broadcasting status.

- Behavior of X_TMM_ecgStoreReserve()

  X_TMM_ecgStoreReserve() is optional. The behavior in link status is the same as in data broadcasting status.

- Behavior of X_TMM_ecgStoreCancelReservation()

  X_TMM_ecgStoreCancelReservation() is optional. The behavior in link status is the same as in data broadcasting status.

- Behavior of X_TMM_ecgStoreCheckReservation()

  X_TMM_ecgStoreCheckReservation() is optional. The behavior in link status is the same as in data broadcasting status.

- Behavior of X_TMM_ecgIsContentStored()

  X_TMM_ecgIsContentStored() is optional. The behavior in link status is the same as the data broadcasting status.

- Behavior of X_TMM_unlockStoredContent()

  X_TMM_unlockStoredContent() is optional. The behavior in link status is the same as in data broadcasting status.

- Behavior of X_TMM_ecgIsLockedStoredContent()

  X_TMM_ecgIsLockedStoredContent() is optional. The behavior in link status is the same as in data broadcasting status.

- Behavior of X_TMM_ecgDeleteStoredContent()

  X_TMM_ecgDeleteStoredContent() is optional. The behavior in link status is the same as in data broadcasting status.

- Behavior of X_TMM_ecgGetStorableSpace()

  X_TMM_ecgGetStorableSpace() is optional. The behavior in link status is the same as in data broadcasting status.

- Behavior of X_TMM_startExFullDisplay

  X_TMM_startExFullDisplay is optional. The behavior in link status is the same as in data broadcasting status.

- Behavior of X_TMM_stopExFullDisplay

  X_TMM_stopExFullDisplay is optional. The behavior in link status is the same as in data broadcasting status.

4.3.6.9 Non-volatile memory functions

Table 4-13 specified behaviour of non-volatile memory functions in communication contents. Refer to section 4.1.10 for operations in data broadcasting reception status.

Table 4-13: Behaviour of non-volatile memory functionss in communication contents

	Link status
readPersisitentArray()	O
wirtePersisitentArray()	X

4.3.7 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 “X-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_TMM_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 section 4.3.9 for references.

  X-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.

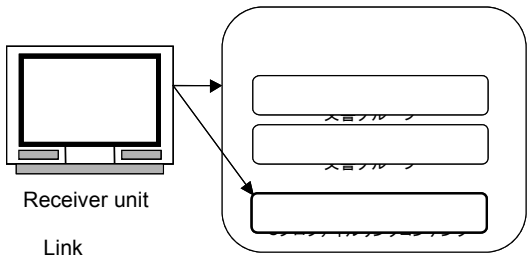


Figure 4-4-5: Link status

● Restrictions of X-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 X-profile linked contents have different name spaces. Scheme of “arib:” or “arib-dc:” should always be specified in order to specify broadcasting contents from X-profile linked contents. Scheme of "http: " or "https: " should be specified in order to specify X-profile linked contents from broadcasting contents. Be aware of the difference in abbreviation handling as well. Refer to section 4.3.10 for operation of name spaces in X-profile linked contents.

In X-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 X-profile linked contents directly. In that case, X-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 X-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 section 3.1.3 for operations of event_msg_group_id.

If transitioned to X-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 section 4.3.11.4. Refer to "Description of the base URI directory" in this section for the concept of the base URI directory.

After transition from X-profile linked contents to link status by X_TMM_launchDocWithLink(), if URI under the base URI directory which was specified

by X_TMM_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 will fail and will present broadcasting contents following section 4.3.11.4.

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 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 X-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_TMM_launchDocWithLink() from X-profile linked contents. For example, the identifier that refers to the X-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 X-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 X-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_TMM_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 X-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 section 4.3.11.4. Refer to section 4.3.12 for status transition details.

- Link status due to X_TMM_launchDocWithLink() transition

If transitioned using the X_TMM_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.

4.3.8 Guidelines for X-profile linked contents transmission

The following shows the operational guidelines for the transmission of X-profile linked contents. Refer to section 2.5 for the communication function of the receiver. Details of the communication protocol related to mobile phone shall follow the separately defined prescriptions.

X-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.

See the guidelines concerning the cache function and URI history for 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

4.3.9 Resource references between X-profile linked contents and broadcasting contents

4.3.9.1 Resource references from broadcasting contents to X-profile linked contents

For resource references from broadcasting contents to X-profile linked contents, follow ARIB STD-B24 Vol. 2 Attachment 4　6.2 “Guidelines on reference across media types” and do not perform reference. Only BML document transition is possible from broadcasting contents to X-profile linked contents.

4.3.9.2 Reference from X-profile linked contents to broadcasting contents

In cases where referring from X-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…/>)

4.3.10 Name space in X-profile linked contents

4.3.10.1 Restriction for URI

For operations of name space, follow ARIB STD-B24 Vol. 2 Attachment 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_TMM_launchDocWithLink () described in BML documents in broadcasting reception status and link status.

4.3.10.2 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 X-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 used, 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_TMM_launchDocWithLink(), use an absolute URI for its argument.

4.3.10.3 Operation of name spaces upon transition from broadcasting contents to X-profile linked contents

Reference from broadcasting contents to X-profile linked contents shall not be performed, according to specifications in section 4.3.9.1. When transitioning from broadcasting contents to X-profile linked contents, use absolute URI that start with "http: " or "https: ".

4.3.10.4 Operation of name spaces upon transition from X-profile link content to broadcasting content

When transitioning from X-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");)

4.3.11 Guidelines for operations of X-profile linked contents

4.3.11.1 Note on operations of X-profile linked contents

Service provider(including broadcaster) that supplies X-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 X-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 X-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. .

4.3.11.2 Guidelines for description of X-profile linked contents

Notes of description of X-profile linked contents are below.

X-profile linked contents are described in profiles according to operations in section 4.1.

Note that there is difference between broadcasting contents and X-profile linked contents regarding the name space. Refer to section 4.3.10 for details.

It is recommended not to specify the port number for X-profile linked contents which are assumed to operate in link status.

Always add "/" at the end when specifying X-profile linked contents by directory specifications. (ex: "http://localhost" ×: incorrect, "http://localhost/"O: correct)

#fragment can be added to the URI that specifies X-profile linked contents. Refer to section 4.2.3.4 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 X-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 X-profile linked contents is "14.0", as broadcasting contents. (ex: <?bml bml-version ="14.0" ?>)

Extension of BML documents that compose X-profile linked contents should be bml. (ex: index.bml)

By putting Mono-media in one page under the same directory, it can reduce receiver's pass search process

BML document should be specified for "href" attributes of the "a "element, arguments of launchDocument() and arguments of X_TMM_launchDocWithLink ()described in X-profile linked contents.

Within X-profile linked contents, links to other BML documents or a procedure to quit it should always be described. This is because receivers 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 X-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 X-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_TMM_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.

4.3.11.3 Guidelines for receiver 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 X-profile linked contents, video and audio are being received and played together.

Refer to section 4.3.7 for link status. Receiver transitions between these 2 statuses by the specification of contents or the operation of users. Figure 4-4-6 specified a situation of status transition.

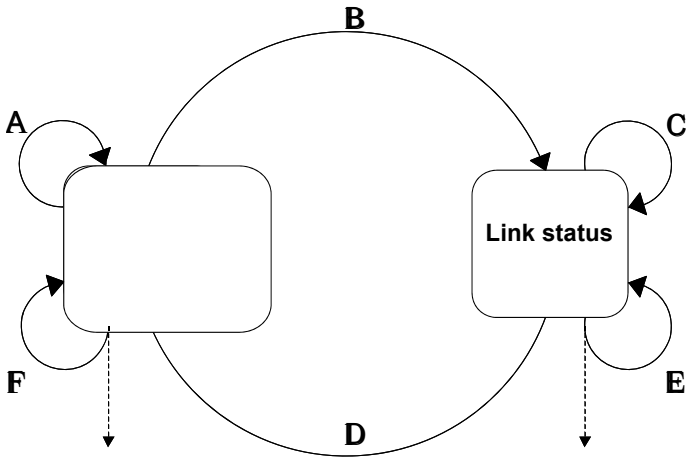


Figure 4-4-6: Status transition of receivers in X-profile linked contents reception

● 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 X-profile linked contents: "http: " and "https: ". Refer to section 4.3.10 for details of name spaces in X-profile linked contents.

  (ex: browser.launchDocument("http://localhost/hoge/index.bml", "cut");)

  X-profile linked contents can specify the directory without specifying the resources. When the directory is specified, a server returns the X-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 section 4.3.11.4, 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 X-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_TMM_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 section 4.3.11.4. 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 section 4.3.11.4.

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_TMM_launchDocWithLink(), but the base URI directory will be set as "//localhost/hoge/foo/".

It will be in link status when transitioned by X_TMM_launchDocWithLink(), even in cases where the host address is changed as in the transition from "http://localhost/hoge/index.bml" to “http://localhosttmm/tmm/foo/test.bml”, for instance. The base URI directory in that case is set as ”//localhosttmm/tmm/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_TMM_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_TMM_launchDocWithLink(), BML browser will fail and will present broadcasting contents following section 4.3.11.4.

● Transition E (transition from link status to broadcasting reception status and other non-link status)

If X_TMM_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 X-profile linked contents will continue. Refer to section 4.1.10.8 for details of X_TMM_startResidentApp().

● Transition F (broadcasting reception status to broadcasting reception status and non-link status)

If X_TMM_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 section 4.1.10.8 for details of X_TMM_startResidentApp().

4.3.11.4 Guidelines for failure behaviour of receivers

If the restricted function in status specified in section 4.3.11.3 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 section 4.3.6 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 section 3.1.1.5.1 "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 X-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.

4.3.12 Guidelines for connection and disconnection

The specification of this section do not apply to receivers which are always on Internet access. .

4.3.12.1 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_TMM_launchDocWithLink())

If transmitTextDataOverIP() is used

4.3.12.2 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_TMM_tuneWithRF() function, the receiver shall completely disconnect the connection before tuning is complete. Within the period between tuning or calling of X_TMM_tuneWithRF() function and completion of tuning, the timing of disconnection is implementation dependent.

4.3.13 Operation of certificates

In X-profile, the root certificate is not transmitted from the broadcasting station to the receiver.

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.

4.3.14 Operation of User-Agent for identification of BML browsers and a browser for the X-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 X-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. .

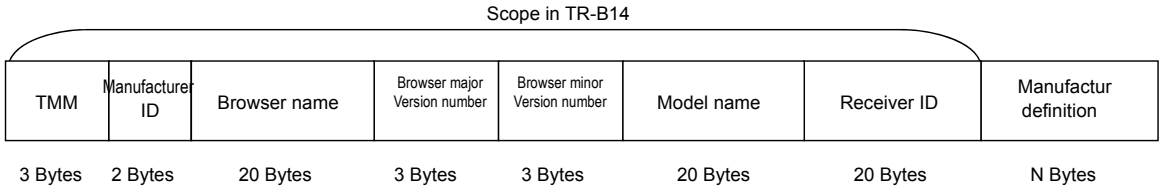


Figure 4-4-7: User-Agent format specified in TR-B14

TMM stands for a request from a browser specified by this standard. String that is a fixed length and “TMM”. When using User-Agent in BML browsers and browser for the X-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 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 : TMM/1C/AAABBBCCC7 /023/007/TMM-H90001 /receiver ID/Manufacturer definition area…

4.3.15 Operation of X_TMM_getComBrowserUA()

By using information acquired by X_TMM_getComBrowserUA() for judgment of browser's models, X_TMM_startResidentApp() can be used in changing the transition URI.

4.3.16 Content transition and browser startup

Figure 4-4-8 specified the contents transition and browser startup.

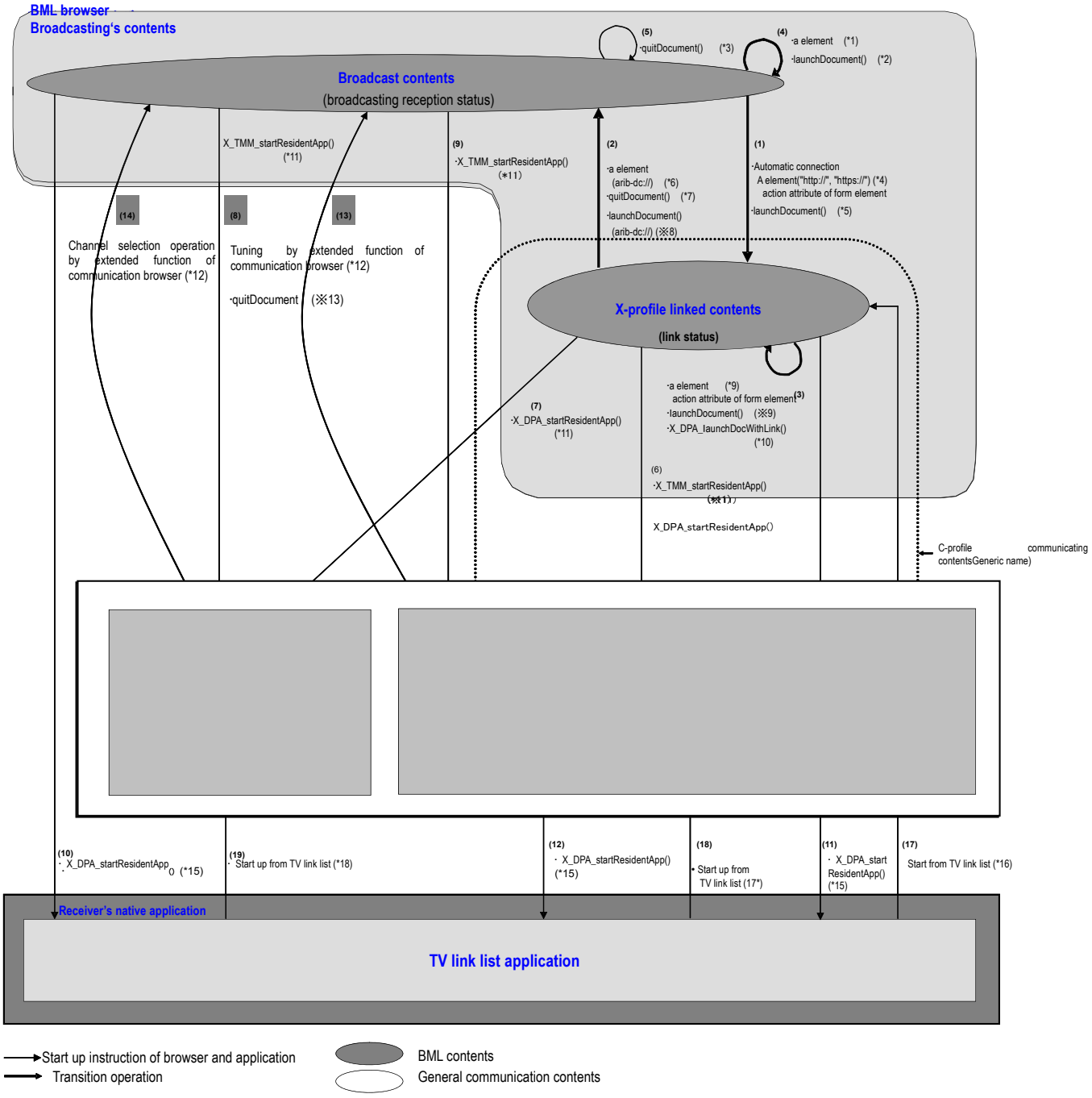


Figure 4-4-8: Transition and startup of each browser

*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 4.3.6.4 for quitDocument(). In this case, however, finish BML browser once and present entry component after restarting.

*4: Specify the URI of X-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 X-profile linked contents. It will be in link status if transitioned by this function. 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 section 4.3.6.4 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 X-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 X-profile linked contents. If transitioned by X_TMM_launchDocWithLink(), the host name or directory name specified by X_TMM_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_TMM_startResidentApp() is executed in broadcasting reception status and link

status, and after that a browser for the contents on the internet is started up.

*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 X-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 X-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, X-profile linked contents and X-profile unlinked contents ,when X_TMM_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 XproBMtype=1 is selected by users in the TVlink list application as one of the receiver’s native applications, present X-profile linked contents according to section 2.7.6.

*17: If TVlink set as XproBMtype=2 is selected by users in the TVlink list application as one of receiver’s native applications, present X-profile unlinked contents according to section 2.7.6

*18: If TVlink set as XproBMtype=3 or XproBMtype=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 section 2.7.6

<Blank Page>

Chapter 5 Operation of Storage Type Broadcasting Services

5.1 Operation of data transmission system

Refer to "Vol. 11, 2.1 Operation of data transmission system" for the operation of the data transmission system.

5.2 Operation of media coding

5.2.1 Video coding

For video coding, the system specified in ARIB-STD B32, Part 1, section 5.2.2 "System compliant with MPEG-4 AVC standard" and Annex 3 "Guidelines for the operation of MPEG-4 AVC standard in low-resolution video services" is used. However, as shown in Table 5-2 below, the high profile is added and the level is expanded to Level 4.2.

Table 5-1 shows the restrictions of coding parameters. For parameters such as buffer size, not included as restrictions here, follow the provisions in ITU-T Rec. H.264|ISO/IEC 14496-10.

Table 5-1: Restrictions of Coding parameters

Item	Restriction
Signal format	YC_BC_R 4:2:0
Quantization bit rate	8 bits
Scan format	Progressive scanning or interlace scanning
Maximum screen size	See Table 5-2
Maximum bit rate	See Table 5-2
Time interval of pictures	Within 0.7 seconds
Color description	Compliant with Rec. ITU-R BT.1361 (Rec. ITU-R BT.709)

The receiving terminal enables playback of the video-coded data of all formats specified in this chapter, but may perform processes such as image size reduction and display frame rate reduction depending on the playback display capability of the terminal.

Table 5-2: Maximum Screen Size and Maximum Bit Rate

Profile	Level	Maximum screen size in operation [number of macro blocks] (corresponding typical number of horizontal pixels x number line numbers)	Maximum bit rate in operation
High	Level 3.1	3600(1280×720)	10 Mbps

	Level 4	8192(1920×1080)	10 Mbps
	Level 4.2	8192(1920×1080)	20 Mbps

5.2.1.1 Video format

ITU-T Rec. H.264|ISO/IEC 14496-10 specifies the maximum screen size and frame rate (number of macroblocks per unit time), according to the. This chapter prescribes the level for operation and the coded video format, taking into account the format of resources, receiving display device, its processes, etc.

Table 5-3 shows the video format specified in this chapter and the corresponding syntax. The 16:9 screen in SQVGA and QVGA are the same as the 4:3 screen, and the screen size has a reduced number of vertical pixels.

Table 5-3: Video Format Used

Format	Screen size	Aspect ratio	seq_ parameter_ set_ rbsp ( )		vui_ parameters( )	
			pic_width _in_mbs_ minus1	pic_height_in_ma p_units_minus1	aspect_ ratio_ info_ present_ flag	aspect_ ratio_ idc
QVGA	320×240	4:3	19	14	1	1
QVGA	320×180	16:9	19	11※		1
VGA	640×480	4:3	39	29		1
525 SD	720×480	4:3	44	29		3
525 SD	720×480	16:9	44	29		5
720p HD	1280x720	16:9	79	44		1
1080 HD	1440x1080	16:9	89	33		255
1080 HD	1920x1080	16:9	119	67*		1

* If the screen width or height cannot be divided by 16, dummy video data is added to the right-hand side of the effective sample or the bottom of the effective line, and in fact, the data is subject to a coding process with the number of samples and number of lines that are multiples of 16. The decoder processes the data and outputs the effective sample and effective line video signals with the dummy data excluded.

5.2.1.2 Frame rate

The frame rate is calculated using the variable of VUI parameters, and is an integral multiple of 1000/1001. See the H.264 standard for details. However, the maximum frame rate [Hz] at each level for the video format to be operated is shown in Table 5-4.

Table 5-4: Maximum frame rate at each level [1/1.001Hz]

	Scan format	Level		
		3.1	4	4.2
QVGA(4:3)	P	15		
	P	30		
QVGA(16:9)	P	15		
	P	30		
VGA	P	30		
525SD(4:3)	P	30		
	P	60		
525SD(16:9)	P	30		
	P	60		
720p HD	P	30		
	P	-	60	
1080 HD	I	-	30	
	P	-	30	
	P	-	-	60

5.2.1.3 Color description

The color description complies with Rec. ITU-R BT. 1361 (Rec. ITU-R BT. 709). In case of video_ signal_ type_ present_ flag=0 or color_ description_ present_ flag=0 in VUI parameters, all values of color_ primaries, transfer_ characteristics, and matrix_ coefficients will be 2 (Unspecified), but all values shall be interpreted as 1 (Rec. ITU-R BT. 709) on the decoder side.

5.2.1.4 Handling 3D video

A 3D video is handled by the identification method using the frame packing arrangement SEI described in ARIB TR-B15 Vol. 7, section 10.1 “3D identification signal in MPEG-4”. The playback function of a 3D video on a terminal is optional. The video format, which is an object of 3D video, is applicable to all video formats handled by this TR.

5.2.2 Still picture and bitmap graphic coding

5.2.2.1 JPEG

Still pictures must comply with the JPEG (Joint Picture Experts Group) BaseLine format of ISO/IEC 10918-1(ITU-T T.81).

Table 5-5: Content Type

Extension	.jpg, .jpeg
MIME Type	image/jpeg

5.2.2.2 GIF

The graphics file format based on GIF uses the specifications based on “GRAPHICS INTERCHANGE FORMAT Version 89a” specified by W3C.

Table 5-6: Content Type

Extension	.gif
MIME Type	image/gif

5.2.2.3 PNG

PG complies with PNG (Portable Network Graphics) GIF of ISO/IEC 15948:2004.

Content Type

Extension	.png
MIME Type	image/png

5.2.2.4 WBMP

WBMP complies with WAP WAE Specification Version 1.1 specified by the WAP Forum.

Content Type

Extension	.wbmp
MIME Type	image/vnd.wap.wbmp

5.2.3 Audio coding

5.2.3.1 Audio input format

(1) Input sampling frequencies

The input sampling frequencies are 32 kHz, 44.1 kHz, and 48 kHz.

(2) Input quantization bit rate

The input quantization bit rate is at least 16 bits.

(3) Number of input channels

The maximum number of input audio channels is five channels plus one channel (channel for boosting the low-frequency band).

5.2.3.2 MPEG Audio coding format

The audio coding format based on the MPEG-4 audio (ISO/IEC 14496-3) is used. And HE-AAC v1 (ISO/IEC 14496-3:2003:Amd.1), HE-AAC v2 (ISO/IEC 14496-3 : 2005:Amd.2), and MPEG Surround (ISO/IEC 23003-1) are added.

For the operation of MPEG Surround, the following decoder tool is used:

MPEG Surround Decoder Tool

MPEG Surround tool	Decoder tool used in these operational guidelines	(Reference) MPEG Baseline Profile
Upmix (M1, M2, M3 as defined by OTT and TTT modules, including decorrelators)	O	O
Residual Coding	O	O
Temporal Shaping	O	O
Parameterized External Downmix compensation	O	O
Residual coding-based External Downmix compensation	X	X
Binaural decoding (parametric)	O	O
Binaural decoding (filtering)	X	O
Mono downmix (5151 and 5152 tree configuration)	X	O
Support of stereo output for 515	X	O
Enhanced Matrix Mode decoding	X	O
Matrix compatibility	X	O
3D Audio decoding (Binauralization using MPEG Surround)	X	O

The receivers that can handle playback of MPEG Surround should support at least one of the following operation modes (levels):

MPEG Surround Decoder Level

Level	Tree Configurations	Maximum number of output channels	Maximum sampling rate [kHz]	Maximum bandwidth of residual coding [QMF bands]	Anticipated utilization scenario
A	525	2.0 (Binaural)	48	0	Binaural playback on decoder side using stereo headphones
B(LP)	525	5.1	48	7	Low-power (LP) consumption playback using 5.1 ch speakers
B(HQ)	525	5.1	48	64	High-quality (HQ) playback using 5.1 ch speakers

Although the receivers must be able to receive the audio data with surround, decoding of only the AAC base part is acceptable.

The coding sampling frequencies are 16 kHz, 22.05 kHz, and 24 kHz in addition to the input sampling frequencies (32 kHz, 44.1 kHz, and 48 kHz).

The receiving terminals enable playback of a part of abilities, according to the playable number of channels, etc. of the terminals.

5.2.4 File format

Video, audio, closed caption, etc. are stored in accordance with the MP4 file format (ISO/IEC 14496-14), AVC file format (ISO/ICE 14496-15), and 3GPP Timed Text (3GPP TS 26.245).

Table 5-7: Content Type

Extension	.mp4
MIME Type	audio/mp4: Audio only video/mp4: Miscellaneous

An MP4 file used for storage-type broadcasting in this technical data can store the video-coded data specified in section 5.2.1, audio-coded data specified in section 5.2.3, and 3GPP Timed Text.

- Compact sample size box “stz2” is not used.
- File-type box “ftyp”: “isom”, “mp42”, “iso2”, or “avc1” is described in major_brand or compatible_brands.
- As for files for which the pseudo-streaming operation by manual completion is expected:
  - Allocate the “moov” box immediately after “ftyp”.
  - When multiple media tracks are contained, allocate each track interleaved with an interleaving depth of one second or less (for a definition of the interleaving depth, follow section 5.4.5 of 3GPP TS26.244-910).
- For encoding of Timed Text, only UTF-8 or UTF-16 is used; UTF-16 (byte-reversed) is not used.

The following shows the relationships between the MP4 file coding parameters and Spec:

Table 5-8: Relationships between MP4 File Coding Parameters and Spec

Profile	Level	Image size	Aspect ratio	Frame rate	Scan system	Maximum bit rate	Spec1	Spec2	Spec3	Spec4
High	3.1	320×240	4:3	15	P	10Mbps	O	O	O	O
				30			O	O	O	O
		320×180	16:9	15	P		O	O	O	O
				30			O	O	O	O
		640×480	4:3	30	P		O	O	O	O
		720×480	4:3	30	P		O	O	O	O
				60			X	X	X	O
			16:9	30	P		O	O	O	O
				60			X	X	X	O
		1280×720	16:9	30	P		X	O	O	O
	4.0	1280×720	16:9	60	P		X	X	X	O
		1440×1080	16:9	30	I		X	X	X	O
		1920×1080	16:9	30	I		X	X	X	O
					P		X	X	O	O
	4.2	1920X1080	16:9	60	P	20Mbps	X	X	X	O

5.2.5 Other coding

5.2.5.1 Character coding

See section 5.4.1. For details, follow the guidelines for “Chapter 6 Playback operation of storage-type contents”.

Table 5-9: Content Type

HTML	Extension	.html, .htm
	MIME Type	text/html
XHTML	Extension	.xml, .xhtml, .html
	MIME Type	application/xhtml+xml
CSS	Extension	.css
	MIME Type	text/css
JavaScript	Extension	.js
	MIME Type	application/javascript
XML	Extension	.xml
	MIME Type	application/xml

5.2.6 Media type

The media type for storage-type broadcasting handled by these operational guidelines is as follows:

(1) Media type of contents

Table 5-10: Media Type of Contents

Media type	Meaning
text/html	HTML documents
application/xhtml+xml	XHTML documents
application/javascript	JavaScript document compliant with ECMA-262
text/css	CSS documents compliant with CSS3, CSS2.1, and CSS1
application/xml	XML documents
image/jpeg	JPEG
image/gif	GIF
audio/mp4, video/mp4	MP4 files
application/X-arib-meta+xml, application/X-arib-meta+bim, multipart/mixed	Metadata

5.2.7 Classification of contents in storage-type broadcasting

In storage-type broadcasting, contents (including program transmission B) are classified into the following three types:

- Common contents
- Consigned broadcaster individual contents
- Receiver corporation individual contents

The classification of contents is specified in the metadata. See Vol. 10 for details.
The following explains the above three types of contents:

5.2.7.1 Common contents

Storage-type broadcasting contents that can be commonly handled by all broadcasters of multimedia broadcasting and all compatible receivers

5.2.7.2 Consigned broadcaster individual contents

Storage-type broadcasting contents specified by each consigned broadcaster of multimedia broadcasting to offer unique services

The specifications of consigned broadcaster individual contents are defined by each broadcaster, and are be managed in this technical data.

5.2.7.3 Receiver corporation individual contents

Contents specific to the receiver of each receiver corporation of multimedia broadcasting These are assumed to be used by the receiver corporations when various types of contents acquired by receiver corporations through communications are obtained by broadcasting.

Therefore, the specifications of receiver corporation individual contents are defined by each receiver corporation, and are not managed in this technical data.

■ Version control of common contents

<Prerequisites>

- Thumbnails are outside version control, and can be identified by ContentType CS of ECG metadata included in Vol. 10.
- The manifest file must always exist, and it must include the contents to be displayed first.

<Thinking of versioning>

- As shown in this technical data, Vol. 10, the version information of common contents is provided as follows, according to each version element. Also, the relationship between the version values and version elements is shown in Table 5-11.

[Version information description format]

Base: <version value: x>.<version value: y>

[Example of description of version information]

Base: 3.1 * In case of x=3, y=1

Table 5-11: Relationship between Version Values and Version Elements

Version value	Version element	Definition
x	Spec of MP4 file	Spec1 to Spec4 (see Table 5-8) respectively correspond to x=1 to 4
y	Configuration of contents	y=0: Consists of one MP4 file and does not respond to scenario control of the manifest file. y=1: Consists of at least one resource and does not respond to scenario control of the manifest file. y=2: Consists of at least one resource and responds to scenario control of the manifest file.

5.3 Operation of the manifest file

5.3.1 Overview of the manifest file

The following shows an overview of the manifest file:

The manifest file is one of the configuration files of storage-type contents, and information on the configuration file management and scenario management is provided in XML document format. Configuration file management specifies the conditions for use as files, such as the encryption status of each file and the relevance with RMPI, together with the list of files configuring content. Scenario management specifies the conditions for the scenario of contents, such as file playback and display order.

The manifest file is a file that the renderer refers to first among the multiple files

configuring content when it plays back and displays the stored contents. One manifest file is always defined for each of the contents in units of use, and the file name shall be "TmmManifest.xml". However, the manifest file is not defined for the EPG/ECG metadata, which is transmitted in program transmission A and program transmission B and is not included in the configuration files of EPG/ECG metadata.

The manifest file enables use of contents regardless of the type of contents, and makes it possible to use even consigned broadcaster individual contents and receiver corporation individual contents. The manifest files used in consigned broadcaster individual content and receiver corporation individual content can be extended respectively to consigned broadcasters and receiver corporations.

5.3.2 XML schema of manifest file

The following defines the XML schema of the manifest file:

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema
  xmlns="http://www.arib.or.jp/tmm/manifest/2011/03"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://www.arib.or.jp/tmm/manifest/2011/03"
  elementFormDefault="qualified">

  <xsd:element name="manifests">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="manifest" type="manifest-type" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="manifest-type">
    <xsd:sequence>
      <xsd:element name="assets" type="assets-type"/>
      <xsd:element name="scenario" type="scenario-type"/>
    </xsd:sequence>
    <xsd:attribute name="version" type="version-type" use="required"/>
  </xsd:complexType>

  <xsd:complexType name="assets-type">
    <xsd:sequence>
      <xsd:element name="asset" type="asset-type" maxOccurs="9999"/>
    </xsd:sequence>
    <xsd:attribute name="count-asset-id" type="asset-id-type"/>
  </xsd:complexType>

  <xsd:complexType name="asset-type">
    <xsd:sequence>
      <xsd:element name="scene" type="scene-type" minOccurs="0" maxOccurs="127"/>
    </xsd:sequence>
    <xsd:attribute name="id" type="asset-id-type" use="required"/>
    <xsd:attribute name="src" type="file-type" use="required"/>
    <xsd:attribute name="encrypted" type="xsd:boolean" default="true"/>
```

```
    <xsd:attribute name="content-type" type="xsd:string"/>
    <xsd:attribute name="value" type="xsd:string"/>
  </xsd:complexType>

  <xsd:complexType name="scene-type">
    <xsd:attribute name="scene-id" type="scene-id-type" use="required"/>
    <xsd:attribute name="start" type="sci-type"/>
    <xsd:attribute name="end" type="sci-type"/>
  </xsd:complexType>

  <xsd:complexType name="scenario-type">
    <xsd:sequence>
      <xsd:element name="sequence" type="sequence-type" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="sequence-type">
    <xsd:attribute name="label" type="label-type" use="required"/>
    <xsd:attribute name="asset-idref" type="asset-id-type" use="required"/>
    <xsd:attribute name="prev" type="label-type"/>
    <xsd:attribute name="next" type="label-type"/>
    <xsd:attribute name="time" type="time-type"/>
  </xsd:complexType>

  <xsd:simpleType name="sci-type">
    <xsd:restriction base="xsd:string">
      <xsd:pattern value="[0-9]{2}:[0-9]{2}:[0-9]{2}.[0-9]{3}"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:simpleType name="asset-id-type">
    <xsd:restriction base="xsd:unsignedShort">
      <xsd:minInclusive value="1"/>
      <xsd:maxInclusive value="9999"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:simpleType name="scene-id-type">
    <xsd:restriction base="xsd:byte">
      <xsd:minInclusive value="1"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:simpleType name="version-type">
    <xsd:restriction base="xsd:string">
      <xsd:pattern value="¥d+¥.¥d+"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:simpleType name="file-type">
    <xsd:restriction base="xsd:string">
      <xsd:maxLength value="255"/>
    </xsd:restriction>
  </xsd:simpleType>
```

```
  <xsd:simpleType name="label-type">
    <xsd:restriction base="xsd:string">
      <xsd:maxLength value="60"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:simpleType name="time-type">
    <xsd:restriction base="xsd:unsignedInt">
      <xsd:minInclusive value="1"/>
    </xsd:restriction>
  </xsd:simpleType>

</xsd:schema>
```

5.3.2.1 <manifests> element

The <manifests> element is a root element of the manifest file, and a container element for containing manifest definitions of multiple versions.

5.3.2.2 <manifest> element

The <manifest> element is a manifest definition element of each version. A manifest-defined version is described in the version attribute.

5.3.2.3 <assets> element

The <assets> element is a container element for containing various media configuring the content.

The count-asset-id attribute counts the number of viewings of a whole content. It is troublesome to count the number of viewings of all media (defined by the <asset> element) in order to count the number of viewings of the content. Therefore, one of the defined media is counted as representative and used as the number of viewings of the entire content.

5.3.2.4 <asset> element

The <asset> element defines various resources that configure content. The id attribute is a unique value in the <manifest> element used to discriminate the asset. The src attribute is used to describe the object resource file, including the logical position (directory). The object resource described in the src attribute is unique in the <manifest> element. The encrypted attribute can be used to specify whether the file is encrypted (the default is “true” that shows encryption).

The elements of all resources (files) must exist in the <asset> element. Therefore, if the resource list of the content created from the FDT instance does not match the <asset> element, the content must not be played as is. In this case, it is preferable that the latest transmission control metadata is acquired via communications and the storage is completed.

The content-type attribute is used to select the application suitable for processing a file. The value attribute is used to make it possible to hand over the parameter to the application. The

specific value of the value attribute is not specified in this technical data.

5.3.2.5 <scene> element

The <scene> element is used to divide a video into several scenes.

The start and end attributes mean the start time and end time in the video file, and the user can use them as the chapters of a designated video file.

The scene-id attribute is used to identify scenes, and is combined with scene-id for trick play inhibition restriction set in RMPI, which is specified in "Vol. 5, 2.3.6.3 Conditions for use of content RMPI)", to perform trick play inhibition control. This attribute is unique in the <manifest> element.

5.3.2.6 <scenario> element

The <scenario> element is a container element used to define the content playback order.

5.3.2.7 <sequence> element

The <sequence> element is used to define the playback order of the resource defined by <asset>.

The label attribute is a name that defines this <sequence>, and is unique in <scenario>.

The asset-idref attribute designates the resource, and is specified by the id of the <asset> element defined as above. If an undefined id is specified, it is regarded as an invalid value, and its <sequence> element is not interpreted.

The prev and next attributes regard the entire scenario as a digraph and define its playback order. The next attribute value describes the label of <sequence>, which connects to the following scene. "_last_" is defined as a special attribute value that can be used for the next attribute. This shows that the content is complete here. When the next attribute is omitted, the following <sequence> element is regarded as the next node. If the following <sequence> cannot be found, it is interpreted as _last_.

The prev attribute is also used to define the playback order in the same way as the next attribute. "_first_" is defined as a special prev attribute that shows the first node of the content. If it is omitted, the <sequence> element just before is regarded as the front node. If "_first_" is not specified, the first <sequence> element is interpreted as the start node of the content.

The time attribute has a meaning if the resource is stationary (HTML file, still picture, etc.). After the time (ms) specified by the time attribute has elapsed, playback of the next content node will start.

5.3.3 Semantics definition of the manifest file

The following defines the semantics of the manifest file:

Table 5-11: Semantics of the Manifest File

Element/Attribute name	Appear able no.	Restriction of character number limiting value	Remarks
manifests	1		
manifest	1..∞		
@version	1	¥d+¥.¥d+	For example, a version such as 1.10 is assumed.
assets	1		
@count-asset-id	0..1	Integer value (1 - 9999)	
asset	1..9999		
@id	1	Integer value (1 - 9999)	
@src	1	String 255 bytes max	
@encrypted	0..1	“true” | “false”	Omission is regarded as “true”.
@value	0..1	String	Not specified in this technical data.
@content-type	0..1	String	See RFC2045 5.Content-type.
scene	0..127		Can be specified for videos.
@scene-id	1	Integer value (1 - 127)	
@start	0..1	String	Shown in hh:mm:ss.ms format. Omission is regarded as the start position of a video.
@end	0..1	String	Shown in hh:mm:ss.ms format. Omission is regarded as the end position of a video.
scenario	1		
sequence	1..∞		
@label	1	String 60 bytes max	
@asset-idref	1	Integer value (1 - 9999)	
@prev	0..1	String Same as the above label attribute.	Omission is interpreted that the <sequence> element just before is interpreted as the front node of content. “_first_” can be used as a reserved word.
@next	0..1	String Same as the above label attribute.	Omission is interpreted that the <sequence> element just after is interpreted as the next node of content. “_last_” can be used as a reserved word.
@time	0..1	Integer value	In units of milliseconds

						(1 or more)	

5.3.4 XML instance of the manifest file

The following shows the XML instance of the manifest file.

When a model is shown, there is a typical scenario pattern of storage-type contents as follows:

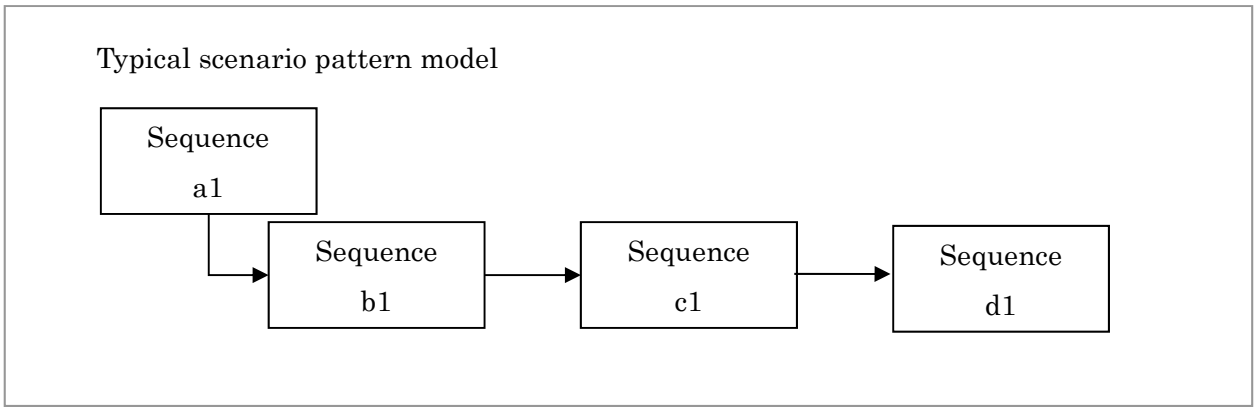


Figure 5-5-1: Typical Scenario Pattern Model

The following shows the XML instance by defining the above pattern model as an XML instance.

```
<?xml version="1.0" encoding="UTF-8"?>
<manifests xmlns="http://www.arib.or.jp/tmm/manifest/2011/03">
  <manifest version="1.0">
    <assets count-asset-id="1">
      <asset id="1" src="a1.html" encrypted="true" content-type="text/html"/>
      <asset id="2" src="b1.mp4" encrypted="true" content-type="video/mp4">
        <scene scene-id="1" start="00:02:00.000" end="00:02:30.000"/>
        <scene scene-id="2" start="00:05:30.000" end="00:06:00.000"/>
        <scene scene-id="3" start="00:09:30.000" end="00:10:00.000"/>
      </asset>
      <asset id="3" src="c1.jpg" encrypted="false" content-type="image/jpeg"/>
      <asset id="4" src="d1.mp4" encrypted="true" content-type="video/mp4">
        <scene scene-id="6" start="00:16:00.000" end="00:17:00.000"/>
      </asset>
    </assets>
    <scenario>
      <sequence label="a1" asset-idref="1" prev="_first_" next="b1"/>
      <sequence label="b1" asset-idref="2" prev="a1" next="c1"/>
      <sequence label="c1" asset-idref="3" prev="b1" next="d1" time="20000"/>
      <sequence label="d1" asset-idref="4" prev="c1" next="_last_"/>
    </scenario>
  </manifest>
</manifests>
```

5.4 Operation of multimedia coding

This section shows the operation of multimedia coding. This section is associated with 5.2.5.1.

5.4.1 Operation of the HTML version

The HTML version is 4.0 or higher, and the detailed operations are shown in the operations of character encoding, HTML element, CSS, DOM, and built-in objects described in 5.4.2 and later. For the specifications of the elements, CSS, DOM, built-in objects, mono-media, and others, packaging in the receiver is not limited.

5.4.2 Operation of character encoding

Responds to ASCII, Shift-JIS, JIS, EUC, and UTF-8.

Table 5-12: Correspondence with Charset

Character encoding	Charset parameter
ASCII	US-ASCII
Shift-JIS	Shift_JIS
JIS	ISO-2022-JP
EUC	EUC-JP
UTF-8	UTF-8

5.4.3 Operation range of the media type and mono-media

See section 5.1.

5.4.4 Operation of HTML elements

The following shows the operation of HTML elements. Table 5-15 shows the elements specified by HTML5. For elements other than the video and audio elements and that are not shown in Table 5-14, check the status handled by the receiver before use. It is preferable to package a navigator object as a measure to survey the status of handling by receiver.
Do not use elements not shown in Table 5-14 to Table 5-15 for contents. The elements of HTML specified here comply with “WD-html5-20110525” of W3C. However, any future change in the HTML specifications of W3C will not be followed up.

The meanings in the “Operation” columns are as follows:
“O”: Basic functions in this provision
“Δ”: Optional functions in this provision
“-”: Neither a basic nor optional function in this provision
When the relevant function is called, an error will occur in the receiver.

Table 5-13: Operation of Elements

Element	Attribute	Operation	Remarks
a		O	
	accesskey	Δ	
	class	O	
	href	O	
	id	O	
	name	O	
	onclick	O	
	style	O	
	target	O	
base		O	
	href	O	
	target	Δ	
blockquote		O	
	class	O	
	id	O	
	style	O	
body		O	
	alink	O	Specifying by style sheet is recommended.
	background	O	Specifying by style sheet is recommended
	bgcolor	O	Specifying by style sheet is recommended
	class	O	
	id	O	
	link	O	Specifying by style sheet is recommended
	onload	O	
	onunload	O	
	style	O	
	text	O	Specifying by style sheet is recommended
	vlink	O	Specifying by style sheet is recommended
br		O	
	class	O	
	clear	O	
	id	O	
	style	O	
button		O	

	accesskey	Δ	
	class	O	
	id	O	
	name	O	
	style	O	
	type	O	
	value	O	
caption		O	
	align	O	
	class	O	
	id	O	
	style	O	
dd		O	
	class	O	
	id	O	
	style	O	
div		O	
	align	O	
	class	O	
	id	O	
	style	O	
dl		O	
	class	O	
	id	O	
	style	O	
dt		O	
	class	O	
	id	O	
	style	O	
form		O	
	action	O	
	class	O	
	enctype	O	
	id	O	
	method	O	
	style	O	
frame		O	
	class	O	
	id	O	
	name	O	
	src	O	
	style	O	
frameset		O	
	class	O	
	cols	O	

	id	O	
	rows	O	
	style	O	
h1		O	
	align	O	
	class	O	
	id	O	
	style	O	
h2		O	
	align	O	
	class	O	
	id	O	
	style	O	
h3		O	
	align	O	
	class	O	
	id	O	
	style	O	
h4		O	
	align	O	
	class	O	
	id	O	
	style	O	
h5		O	
	align	O	
	class	O	
	id	O	
	style	O	
h6		O	
	align	O	
	class	O	
	id	O	
	style	O	
head		O	lang attribute and dir attribute are not operated
hr		O	
	align	O	
	class	O	
	id	O	
	noshade	O	
	size	O	
	style	O	
	width	O	
html		O	
	version	Δ	
iframe		O	
	class	O	
	height	O	

	id	O	
	name	O	
	src	O	
	style	O	
	width	O	
img		O	
	align	O	
	alt	Δ	
	border	O	
	class	O	
	height	O	
	hspace	O	
	id	O	
	src	O	
	style	O	
	vspace	O	
	width	O	
input		O	
	accesskey	Δ	
	class	O	
	Id	O	
	style	O	
	name	O	
	type	O	The following keywords can be specified: Checkbox, password, radio, text, reset, submit, hidden, button
	value	O	Can be specified when the type attribute is as follows: Checkbox, password, radio, text, reset, submit, hidden, button
	checked	O	Can be specified when the type attribute is checkbox or radio.
	maxlength	O	Can be specified when the type attribute is password or text.
	size	O	Can be specified when the type attribute is as follows: Password, text
	accept	Δ	Shall be non-compliant, because this attribute can be specified only when the type attribute is file: Even if it is specified, ignore it.
li		O	
	class	O	
	id	O	
	style	O	
	type	O	
	value	O	
link		O	
	class	O	
	href	O	
	id	O	

	media	O	
	rel	O	
	style	O	
meta		O	
	content	O	
	http-equiv	O	
	name	O	
noscript		O	
	class	O	
	id	O	
	style	O	
object		O	
	class	O	
	data	O	
	declare	Δ	
	height	O	
	id	O	
	style	O	
	type	O	
	width	O	
	classid	Δ	
	align	O	
ol		O	
	class	O	
	id	O	
	start	O	
	style	O	
	type	O	
option		O	
	class	O	
	id	O	
	selected	O	The control of form by button is not operated.
	style	O	
	value	O	The control of form by button is not operated.
p		O	
	align	O	
	class	O	
	id	O	
	style	O	
param		O	
	id	O	
	name	Δ	
	value	Δ	
	valuetype	O	
pre		O	
	class	O	

	id	O	
	style	O	
script		O	
	src	O	
select		O	
	class	O	
	id	O	
	multiple	O	
	name	O	
	size	Δ	
	style	O	
span		O	
	class	O	
	id	O	
	style	O	
style		O	
	media	O	
	type	O	
table		O	
	align	O	
	bgcolor	O	
	border	O	
	class	O	
	id	O	
	style	O	
	width	O	
td		O	
	align	O	
	bgcolor	O	
	class	O	
	colspan	O	
	height	O	
	id	O	
	rowspan	O	
	style	O	
	valign	O	
	width	O	
textarea		O	
	accesskey	Δ	
	class	O	
	cols	O	
	id	O	
	name	O	
	rows	O	
	style	O	
th		O	
	align	O	

	bgcolor	O	
	class	O	
	colspan	O	
	height	O	
	id	O	
	rowspan	O	
	style	O	
	valign	O	
	width	O	
title		O	
tr		O	
	align	O	
	bgcolor	O	
	class	O	
	id	O	
	style	O	
ul		O	
	class	O	
	id	O	
	style	O	
	type	O	

Table 5-14: Operation of Elements (HTML5.0)

Element	Attribute	Operation	Remarks
(Global)		O	The event and handler attributes are included in the global attribute, but they are not shown.
	accesskey	Δ	
	aria-*	Δ	See the WAI-ARIA specifications.
	class	O	
	contenteditable	Δ	
	contextmenu	Δ	
	data-*	O	
	dir	O	
	draggable	Δ	
	hidden	Δ	
	id	O	
	lang	O	
	role	Δ	See the WAI-ARIA specifications.
	spellcheck	Δ	
	style	O	
	tabindex	Δ	
	title	Δ	
a		O	
	href	O	
	hreflang	O	
	media	Δ	

	ping	Δ	
	rel	O	
	target	Δ	
	type	O	
abbr		O	
address		O	
area		O	
	alt	Δ	
	coords	O	
	shape	O	
	href	O	
	target	Δ	
	ping	Δ	
	rel	O	
	media	Δ	
	hreflang	O	
	type	Δ	
article		O	
aside		O	
audio		O	
	src	O	
	preload	Δ	
	autoplay	O	
	loop	O	
	controls	O	
b		O	
base		O	
	href	O	
	target	Δ	
bdi		Δ	
bdo		O	
blockquote		O	
	cite	O	
body		O	
br		O	
button		O	
	autofocus	O	
	disabled	O	
	form	O	
	formaction	O	
	formenctype	O	
	formmethod	O	
	formnovalidate	O	
	formtarget	O	
	name	O	
	type	O	
	value	O	

canvas		O	
	width	O	
	height	O	
caption		O	
cite		O	
code		O	
colgroup		O	
	span	O	
col		O	
	span	O	
command		Δ	
	type	Δ	
	label	Δ	
	icon	Δ	
	disabled	Δ	
	checked	Δ	
	radiogroup	Δ	
datalist		Δ	
dl		O	
dt		O	
dd		O	
del		O	
	cite	O	
	datetime	O	
details		O	
	open	Δ	
dfn		O	
div		O	
em		O	
embed		Δ	
	src	Δ	
	type	Δ	
	width	Δ	
	height	Δ	
fieldset		O	
	disabled	O	
	form	O	
	name	O	
figcaption		O	
figure		Δ	
footer		O	
form		O	
	accept-charset	O	
	action	O	
	autocomplete	Δ	
	enctype	Δ	
	method	O	

	name	O	
	novalidate	O	
	target	Δ	
html		O	
	manifest	O	The browser that can handle the cache control function of offline content can interpret this attribute.
head		O	
header		O	
hgroup		O	
hr		O	Be careful that operation is different from HTML4.
h1		O	
h2		O	
h3		O	
h4		O	
h5		O	
h6		O	
i		O	
iframe		O	
	src	O	
	srcdoc	Δ	
	name	O	
	sandbox	O	
	seamless	Δ	
	width	O	
	height	O	
img		O	
	alt	O	
	src	O	
	usemap	O	
	ismap	O	
	width	O	
	height	O	
input		O	The applicable attribute differs, depending on the keyword specified for type. See the HTML5 specifications for details.
	accept	Δ	
	alt	Δ	
	autocomplete	Δ	
	autofocus	Δ	
	checked	O	
	disabled	O	
	form	Δ	
	formaction	Δ	
	formenctype	Δ	
	formmethod	Δ	
	formnovalidate	O	
	formtarget	Δ	
	height	O	

	list	Δ	
	max	Δ	
	maxlength	O	
	min	Δ	
	multiple	O	
	name	O	However, non-compliant during type=url, datatime.
	pattern	O	
	placeholder	O	
	readonly	O	
	required	O	
	size	O	
	src	O	
	step	Δ	
	type	Δ	The following keywords can be specified: hidden, text, serch, telephone, url, email, password, number, checkbox, radio, submit, image, reset, button
	value	O	
	width	O	
ins		O	
	cite	O	
	datetime	O	
kbd		O	
keygen		Δ	
	autofocus	Δ	
	challenge	Δ	
	disabled	O	
	form	O	
	keytype	O	
	name	O	
label		O	
legend		O	
li		O	
	value	O	
link		O	
	href	O	
	rel	O	
	media	O	
	hreflang	O	
	type	O	
	sizes	Δ	
map		O	
	name	O	
MathML		Δ	Necessary when the MathML element is supported. See the MathML specifications and related specifications for details.
mark		Δ	
menu		O	
	type	Δ	

	label	Δ	
meta		O	
	name	Δ	
	http-equiv	Δ	
	content	O	
	charset	O	A document has only one meta attribute whose charset attribute is specified.
meter		O	
	value	O	
	min	O	
	max	O	
	low	O	
	high	O	
	optimum	O	
	form	O	
nav		O	
noscript		O	
object		O	
	data	O	
	type	O	
	name	O	
	usemap	Δ	
	form	O	
	width	O	
	height	O	
ol		O	
	start	O	
	reversed	Δ	
optgroup		O	
	disable	O	
	label	O	
option		O	
	disabled	O	
	label	O	
	selected	O	
	value	O	
output		O	
	for	O	
	form	O	
	name	O	
p		O	
param		Δ	
	name	Δ	
	value	Δ	
pre		O	
progress		Δ	
	value	Δ	
	max	Δ	

	form	Δ	
q		O	
	cite	O	
rp		O	
rt		O	
ruby		O	
samp		O	
script		O	
	src	O	
	async	O	
	defer	O	
	type	O	
	charset	O	
section		O	
select		O	
	autofocus	O	
	disabled	O	
	form	O	
	multiple	O	
	name	O	
	size	Δ	
small		O	
source		O	
	src	O	
	type	O	
	media	O	
span		O	
strong		O	
style		O	
	type	O	
	media	O	
	scoped	Δ	
sub		O	
summary		Δ	
sup		O	
SVG		Δ	Necessary when the SVG element is supported. See the SVG specifications for details.
table		O	
	summary	Δ	
tbody		O	
td		O	
	colspan	O	
	rowspan	O	
	headers	Δ	
textarea		O	
	autofocus	O	
	cols	O	
	disabled	O	

	form	O	
	maxlength	O	
	name	O	
	placeholder	O	
	readonly	O	
	required	Δ	
	rows	O	
	wrap	Δ	
tfoot		O	
th		O	
	colspan	O	
	rowspan	O	
	headers	Δ	
	scope	Δ	
thead		O	
title		O	
time		Δ	
	dateime	Δ	
	pubdate	Δ	
tr		O	
track		Δ	
	kind	Δ	
	src	Δ	
	srclang	Δ	
	label	Δ	
	default	Δ	
ul		O	
var		O	
video		O	
	src	O	
	poster	O	
	preload	O	
	autoplay	O	
	loop	O	
	controls	O	
	width	O	
	height	O	
wbr		O	

5.4.5 Operation of CSS

The following shows the operation of CSS.

Specifications to be referred to: http://www.w3.org/TR/2011/REC-css3-selectors-20110929/.

Table 5-15: Operation of CSS

Property	Value	Operation
(Selector Pattern) –	E[foo^="bar"]	O
	E[foo$="bar"]	O
	E[foo*="bar"]	O
	E:root	O
	E:nth-child(n)	O
	E:nth-last-child(n)	O
	E:nth-of-type(n)	O
	E:nth-last-of-type(n)	O
	E:last-child	O
	E:first-of-type	O
	E:last-of-type	O
	E:only-child	O
	E:only-of-type	O
	E:empty	O
	E:target	O
	E:enabled	O
	E:disabled	O
	E:checked	O
	E:not(s)	O
	E ~ F	O
background	'background-color' || 'background-image' || 'background-repeat' || 'background-position'	O
	Inherit	O
background-attachment	Fixed	Δ
	Inherit	Δ
	Scroll	O
background-color	<color>	O
	Inherit	O
	transparent	O
background-image	<uri>	O
	inherit	O
	none	O
background-position	[top | center | bottom] || [left | center | right]	O
	<length>{1,2}	O
	<percentage>{1,2}	O
	inherit	O
background-repeat	inherit	O
	no-repeat	O
	repeat	O
	repeat-x	O
	repeat-y	O
border	border-width' || 'border-style' || ' <color>'	O

border-bottom	border-bottom-width' || 'border-bottom-style' || 'border-bottom-color'	O
border-bottom-color	<color>	O
	inherit	O
border-bottom-style	hidden	O
	none	O
	solid	O
border-bottom-width	<length>	O
	inherit	O
	medium	O
	thick	O
	thin	O
border-collapse	collapse	O
	inherit	O
	separate	O
border-color	<color>{1,4}	O
	inherit	O
border-left	border-left-width' || 'border-left-style' || 'border-left-color'	O
border-left-color	<color>	O
	inherit	O
border-left-style	hidden	O
	none	O
	solid	O
border-left-width	<length>	O
	inherit	O
	medium	O
	thick	O
	thin	O
border-right	border-right-width' || 'border-right-style' || 'border-right-color'	O
border-right-color	<color>	O
	inherit	O
border-right-style	hidden	O
	none	O
	solid	O
border-right-width	<length>	O
	inherit	O
	medium	O
	thick	O
	thin	O
border-style	<border-style>{1,4}	O
border-top	border-top-width' || 'border-top-style' || 'border-top-color'	O
border-top-color	<color>	O
	inherit	O
border-top-style	hidden	O
	none	O

	solid	O
border-top-width	<length>	O
	inherit	O
	medium	O
	thick	O
	thin	O
border-width	<border-width>{1,4}	O
	inherit	O
bottom	<length>	O
	<percentage>	O
	auto	O
	inherit	O
clear	both	O
	inherit	O
	left	O
	none	O
	right	O
color	<color>	O
	<Foreground color: the ‘color’ property>	O
	<Transparency: the 'opacity' property>	O
	<HTML4 color keywords>	O
	<RGB color values>	O
	<RGBA color values>	O
	<'transparent' color keyword>	O
	<HSL color values>	O
	<HSLA color values>	O
	<SVG color keywords>	O
	< 'currentColor' color keyword>	O
	inherit	O
display	block	O
	inherit	O
	inline	O
	none	O
	-wap-marquee	Δ
float	inherit	O
	left	O
	none	O
	right	O
font-size	<length>	O
	<percentage>	O
	inherit	O
	large	O
	larger	O
	medium	O
	small	O
	smaller	O
	x-large	O

	x-small	O
	xx-large	O
	xx-small	O
font-style	inherit	O
	italic	O
	normal	O
font-weight	bold	O
	inherit	O
	normal	O
height	<length>	O
	<percentage>	O
	auto	O
	inherit	O
left	<length>	O
	<percentage>	O
	auto	O
	inherit	O
line-height	<integer>	O
	<length>	O
	<percentage>	O
	inherit	O
	normal	O
list-style	inherit	O
	'list-style-type' || 'list-style-position' || 'list-style-image'	O
list-style-image	<uri>	O
	inherit	O
	none	O
list-style-position	inherit	O
	inside	O
	outside	O
list-style-type	circle	O
	decimal	O
	disc	O
	lower-alpha	O
	none	O
	square	O
	upper-alpha	O
margin	<margin-width>{1,4}	O
	inherit	O
margin-bottom	<length>	O
	<percentage>	O
	auto	O
	inherit	O
margin-left	<length>	O
	<percentage>	O
	auto	O
	inherit	O

margin-right	<length>	O
	<percentage>	O
	auto	O
	inherit	O
margin-top	<length>	O
	<percentage>	O
	auto	O
	inherit	O
overflow	hidden	O
	inherit	O
	visible	O
padding	<padding-width>{1,4}	O
	inherit	O
padding-bottom	<length>	O
	<percentage>	O
	inherit	O
padding-left	<length>	O
	<percentage>	O
	inherit	O
padding-right	<length>	O
	<percentage>	O
	inherit	O
padding-top	<length>	O
	<percentage>	O
	inherit	O
position	absolute	O
	fixed	Δ
	inherit	O
	relative	O
	static	O
right	<length>	O
	<percentage>	O
	auto	O
	inherit	O
text-align	center	O
	inherit	O
	left	O
	right	O
text- decoration	inherit	O
	none	O
	underline || overline || line-through	O
text-indent	<length>	O
	<percentage>	O
	inherit	O
top	<length>	O
	<percentage>	O
	auto	O

	inherit	O
vertical-align	<length>	O
	baseline	O
	bottom	O
	inherit	O
	middle	O
	sub	O
	super	O
	top	O
visibility	hidden	O
	inherit	O
	visible	O
white-space	inherit	O
	normal	O
	nowrap	O
width	<length>	O
	<percentage>	O
	auto	O
	inherit	O
z-index	<integer>	O
	auto	O
	inherit	O
-wap-marquee-style	alternate	Δ
	scroll	Δ
	slide	Δ
-wap-marquee-dir	rtl	Δ
	ltr	Δ
-wap-marquee-loop	<integer>	Δ

Genre	Property	Operation
(elements and modules)	Rounded corners	O
	Multiple background images	O
	Shadows and box shadows	O
	Text strokes	O
	Border images	O
	Multi-column layout	O
	2D Transforms	O
	Transitions	O
	Media Queries	O

5.4.6 Scope of operation of DOM

5.4.6.1 Basic interface of the DOM core

The following shows the operation of the basic interface of the DOM core:

Table 5-16: Operation of Basic Interface of the DOM Core

Interface	Operation
DOMException	O
DOMImplementation	O
DocumentFragment	O
Document	O
Node	O
NodeList	O
CharacterData	O
Attr	O
Element	O
Text	O
Comment	O

The following shows the operation of the attributes of the basic interface of the DOM core.

Table 5-17: Operation of Attributes and Methods

Interface	Attribute/Method	Operation
DOMException		
	code	O
DOMImplementation		
	create Document	O
Document		
	documentElement	O
	implementation	O
	createAttribute(in DOMString name)	O
	createComment(in DOMString data)	O
	createDocumentFragment()	O
	createElement(in DOMString tagName)	O
	createTextNode(in DOMString data)	O
	getElementById(in DOMString elementId)	O
	getElementsByTagName (in DOMString tagname)	O
Node		
	attributes	O
	childNodes	O
	firstChild	O
	lastChild	O
	nextSibling	O
	nodeName	O
	nodeType	O
	nodeValue	O
	ownerDocument	O

	parentNode	O
	prefix	O
	previousSibling	O
	appendChild(in Node newChild)	O
	cloneNode(in boolean deep)	O
	hasChildNodes()	O
	insertBefore(in Node newChild, in Node refChild)	O
	removeChild(in Node oldChild)	O
	replaceChild(in Node newChild, in Node oldChild)	O
NodeList		
	length	O
	item(in unsigned long index)	O
CharacterData		
	data	O
	length	O
	appendData(in DOMString arg)	O
	deleteData(in unsigned long offset, in unsigned long count)	O
	insertData(in unsigned long offset, in DOMString arg)	O
	replaceData(in unsigned long offset, in unsigned long count, in DOMString arg)	O
	substringData(in unsigned long offset, in unsigned long count)	O
Attr		
	name	O
	specified	O
	value	O
Element		
	tagName	O
	getAttribute(in DOMString name)	O

	getAttributeNode(in DOMString name)	O
	getElementsByTagName(in DOMString name)	O
	removeAttribute(in DOMString name)	O
	removeAttributeNode(in Attr oldAttr)	O
	setAttribute(in DOMString name, in DOMString value)	O
	setAttributeNode(in Attr newAttr)	O
Text		
	splitText(in unsigned long offset)	O

5.4.6.2 DOM HTML interface

The following shows the operation of the DOM HTML interface:

Table 5-18: Operation of Attributes and Methods

Interface	Attribute/Method/Event handler	Operation
HTMLDocument		O
	anchors	O
	applets	Δ
	body	O
	cookie	O
	domain	O
	forms	O
	images	O
	lastModified	O
	links	O
	referrer	O
	title	O
	URL	O
	close()	O
	getElementsByName(in DOMString elementName)	O
	open()	O
	write(in DOMString text)	O
	writeln(in DOMString text)	O
HTMLElement		O
	className	O
	id	O
	innerHTML	O
	lang	O
	offsetParent	O

	title	O
HTMLHtmlElement		O
	version	O
HTMLHeadElement		O
HTMLLinkElement		O
	href	O
	media	O
	rel	O
HTMLTitleElement		O
	text	O
HTMLMetaElement		O
	content	O
	httpEquiv(http-equiv)	O
	name	O
HTMLBaseElement		O
	href	O
	target	O
HTMLStyleElement		O
	media	O
	type	O
HTMLFormElement		O
	action	O
	elements	O
	enctype	O
	length	O
	method	O
	name	O
	reset()	O
	submit()	O
HTMLSelectElement		O
	form	O
	length	O
	multiple	O
	name	O
	options	O
	selectedIndex	O
	size	O
	type	O
	value	O
	blur()	Δ
	focus()	Δ
HTMLOptionElement		O
	defaultSelected	O
	selected	O
	text	O
	value	O
HTMLInputElement		O

	accept	O
	accessKey	O
	checked	O
	defaultChecked	O
	defaultValue	O
	form	O
	maxLength	O
	name	O
	size	O
	type	O
	value	O
	blur()	Δ
	focus()	Δ
HTMLTextAreaElement		O
	accessKey	O
	cols	O
	defaultValue	O
	form	O
	name	O
	rows	O
	type	O
	value	O
	blur()	Δ
	focus()	Δ
HTMLButtonElement		O
	accessKey	O
	form	O
	name	O
	type	O
	value	O
	blur()	Δ
	focus()	Δ
HTMLUListElement		O
	type	O
HTMLOListElement		O
	start	O
	type	O
HTMLDListElement		O
HTMLDirectoryElement		O
HTMLMenuElement		O
HTMLLIElement		O
	type	O
	value	O
HTMLBlockquoteElement		O

	cite	O
HTMLDivElement		O
	align	O
HTMLParagraphElement		O
	align	O
HTMLHeadingElement		O
	align	O
HTMLPreElement		O
HTMLBRElement		O
	clear	O
HTMLFontElement		O
	color	O
	size	O
HTMLHRElement		O
	align	O
	noShade	O
	size	O
	width	O
HTMLAnchorElement		O
	accessKey	O
	href	O
	name	O
	target	O
	blur()	Δ
	focus()	Δ
HTMLImageElement		O
	align	O
	alt	O
	complete	O
	height	O
	hspace	O
	src	O
	vspace	O
	width	O
	onload	O
HTMLObjectElement		O
	data	O
	declare	O
	height	O
	type	O
	width	O
HTMLParamElement		O
	name	O
	value	O
	valueType	O
HTMLScriptElement		O
	src	O

HTMLTableElement		O
	align	O
	bgColor	O
	border	O
	rows	O
	width	O
HTMLTableCaptionElement		O
	align	O
HTMLTableRowElement		O
	align	O
	bgColor	O
	cells	O
	vAlign	O
HTMLTableCellElement		O
	align	O
	bgColor	O
	colSpan	O
	height	O
	rowSpan	O
	vAlign	O
	width	O
HTMLFrameSetElement		O
	cols	O
	rows	O
HTMLFrameElement		O
	src	O
HTMLIFrameElement		O
	height	O
	name	O
	src	O
	width	O
CSS2Properties		O
	background	O
	backgroundAttachment	O
	backgroundColor	O
	backgroundImage	O
	backgroundPosition	O
	backgroundRepeat	O
	border	O
	borderBottom	O
	borderBottomColor	O
	borderBottomStyle	O
	borderBottomWidth	O

borderCollapse	O
borderColor	O
borderLeft	O
borderLeftColor	O
borderLeftStyle	O
borderLeftWidth	O
borderRight	O
borderRightColor	O
borderRightStyle	O
borderRightWidth	O
borderStyle	O
borderTop	O
borderTopColor	O
borderTopStyle	O
borderTopWidth	O
borderWidth	O
bottom	O
clear	O
color	O
cssFloat	O
display	O
fontSize	O
height	O
left	O
lineHeight	O
listStyle	O
listStyleImage	O
listStylePosition	O
listStyleType	O
margin	O
marginBottom	O
marginLeft	O
marginRight	O
marginTop	O
overflow	Δ*1)
padding	O
paddingBottom	O
paddingLeft	O
paddingRight	O
paddingTop	O
position	O
right	O
textAlign	O
textDecoration	O
textIndent	O
top	O
verticalAlign	O

	visibility	O
	whiteSpace	O
	width	O
	zIndex	O

*1) Non-compliant for only “scroll”.

5.4.7 DOM interface in HTML5 specifications

Since the HTML5 specifications are not mandatory, their details for the DOM interface are not included in these provisions. Packaging is performed according to the conforming status of elements and attributes with HTML5 specifications. However, it is preferable to comply with the DOM interface related to the video and audio elements (HTMLVideoElement, HTMLAudioElement, and HTMLMediaElement).

5.4.8 Scope of operation of built-in objects

The following shows the operation of JavaScript built-in objects:

Table 5-19: Operation of JavaScript

Object	Method/Property	Operation
Global		
	NaN	O
	Infinity	O
	Undefined	O
	eval(x)	O
	parseInt(string [, radix])	O
	parseFloat(string)	O
	isNaN(number)	O
	isFinite(number)	O
	toString()	O
	decodeURI(encodedURI)	O
	decodeURIComponent(encodedURIComponent)	O
	encodeURI(uri)	O
	encodeURIComponent(uriComponent)	O
	escape(string)	O
	unescape(string)	O
Objects		
	Object([value])	O
	new Object([value])	O
	Length	O
	Prototype	O
	Constructor	O
	toString()	O
	valueOf()	O

	toSource()	-
Function		
	Function([p1,p2, ... , pN,] body)	O
	new Function([p1,p2, ... , pN,] body)	O
	Length	O
	Prototype	O
	Constructor	O
	apply(thisArg [, argArray])	O
	call(thisArg [, arg1, arg2, ... , argN])	O
	toSource()	-
	Length	O
	Prototype	O
Arguments		
	Callee	O
	Length	O
Array		
	Array([item1, item2, ... , itemN])	O
	new Array([item0, item1, ... , itemN])	O
	new Array(len)	O
	Length	O
	Prototype	O
	Constructor	O
	toString()	O
	concat([item1, item2, ... , itemN])	O
	join(separator)	O
	pop()	O
	push([item1, item2, ... , itemN])	O
	reverse()	O
	shift()	O
	slice(start [, end])	O
	sort(comparefn)	O
	splice(start, deleteCount [, item1, item2, ... , itemN])	O
	unshift([item1, item2, ... , itemN])	O
	toSource()	-
	Length	O
String		
	String([value])	O
	new String([value])	O
	Length	O
	Prototype	O
	fromCharCode([code0, code1, ... , codeN])	O
	constructor	O

	toString()	O
	valueOf()	O
	charAt(pos)	O
	charCodeAt(pos)	O
	concat([item1, item2, ... , itemN])	O
	indexOf(pattern [, pos])	O
	lastIndexOf(pattern [, pos])	O
	localeCompare(that)	O
	match(regexp)	O
	replace(searchValue, replaceValue)	O
	search(regexp)	O
	slice(start [, end])	O
	split(separator [, lim])	O
	substring(start, end)	O
	substr(start [, length])	O
	toLowerCase()	O
	toLocaleLowerCase()	O
	toUpperCase()	O
	toLocaleUpperCase()	O
	toSource()	-
Boolean		
	Boolean(value)	O
	new Boolean(value)	O
	prototype	O
	constructor	O
	toString()	O
	valueOf()	O
	toSource()	-
Number		
	Number([value])	O
	new Number([value])	O
	prototype	O
	MAX_VALUE	O
	MIN_VALUE	O
	NaN	O
	NEGATIVE_INFINITY	O
	POSITIVE_INFINITY	O
	constructor	O
	toString([radix])	O
	valueOf()	O
	toFixed(fractionDigits)	-
	toExponential(fractionDigits)	O
	toPrecision(precision)	-
	toSource()	-
Math		
	E	O

	LN10	O
	LN2	O
	LOG2E	O
	LOG10E	O
	PI	O
	SQRT1_2	O
	SQRT2	O
	abs(x)	O
	acos(x)	O
	asin(x)	O
	atan(x)	O
	atan2(y, x)	O
	ceil(x)	O
	cos(x)	O
	exp(x)	O
	floor(x)	O
	log(x)	O
	max([value1, value2, ... , valueN])	O
	min([value1, value2, ... , valueN])	O
	pow(x, y)	O
	random()	O
	round(x)	O
	sin(x)	O
	sqrt(x)	O
	tan(x)	O
Date		
	Date ([year, month, date, hours, minutes, seconds, ms])	O
	new Date(year, month [, date, hours, minutes, seconds, ms])	O
	new Date()	O
	new Date(milliseconds)	O
	new Date(dateString)	O
	Length	O
	prototype	O
	parse(string)	O
	UTC(year, month [, date, hours, minutes, seconds, ms])	O
	constructor	O
	toString()	-
	toDateString()	O
	toTimeString()	-
	toLocaleString()	-
	toLocaleDateString()	O
	toLocaleTimeString()	O

	valueOf()	O
	getTime()	O
	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)	O
	setMilliseconds(ms)	O
	setUTCMilliseconds(ms)	O
	setSeconds(sec [, ms])	O
	setUTCSeconds(sec [, ms])	O
	setMinutes(minute [, sec, ms])	O
	setUTCMinutes(minute [, sec, ms])	O
	setHours(hour [, minute, sec, ms])	O
	setUTCHours(hour [, minute, sec, ms])	O
	setDate(date)	O
	setUTCDate(date)	O
	setMonth(month [, date])	O
	setUTCMonth(month [, date])	O
	setFullYear(year [, month, date])	O
	setUTCFullYear(year [, month, date])	O
	toUTCString()	O
	toGMTString()	O
	getYear()	O
	setYear(year)	O
	toSource()	-
RegExp		
	RegExp(pattern [, flags])	O
	new RegExp(pattern [, flags])	O
	prototype	O
	constructor	O
	exec([string])	O
	test([string])	O
	toString()	O

	toSource()	-
	valueOf()	O
	source	O
	global	O
	ignoreCase	O
	multiline	O
	lastIndex	O
Error		
	Error(message)	O
	new Error(message)	O
	prototype	O
	constructor	O
	name	O
	message	O
	toString()	O
EvalError	Similar to "Error Object"	O
RangeError	Similar to "Error Object"	O
ReferenceError	Similar to "Error Object"	O
SyntaxError	Similar to "Error Object"	O
TypeError	Similar to "Error Object"	O
URIError	Similar to "Error Object"	O

<Blank Page>

Chapter 6 Guidelines for Playback Operation of Storage Type Contents

6.1 Guidelines for presentation

The following shows the guidelines for the presentation function:

Refer to not only the contents shown in this chapter but also the service guidelines of each broadcaster.

6.1.1 Presenting video/audio contents by storage-type broadcasting browsers

6.1.1.1 Presentation and control of video/audio contents

Some storage-type broadcasting browsers may present different files (audio, video, etc.) in content. In this case, when an object file is referred to from HTML, the use of video and audio contents is assumed.

The common contents that can be referred to by using the video and audio elements are MP4 files. However, the media type of MP4 file that can be described in the src attribute of the video element is video/mp4, and the media type of MP4 file that can be described in the src attribute of the audio element is audio/mp4. The receiver operation when an unsupported media type is specified is a matter of product planning.

A browser that handles the source element uses the src attribute on the source element to specify a file other than an MP4 file with the MP4 file, thereby enabling presentation of a file other than the MP4 file.

When video/audio contents are presented, the description in the HTML document is followed, except when a full screen is displayed depending on the performance of receiver, but the aspect ratio of the content is presented without being changed.

6.1.1.2 HTML attribute when video/audio contents are presented

The attributes shown in Table 5-15 are applied to the video/audio/embed elements. However, since the global element specified by HTML5 differs from that for HTML4, if not all attributes are supported, the class attributes, id attributes, and style attributes are at least supported.

6.1.1.3 URL dispensing function

It is preferable to install the function that, during playback of content, any playback position of the content is dispensed as URL, and the function that the dispensed URL is interpreted and played back from a specified position. For the format of the URL to be dispensed, follow the format of media scheme in Annex 13. For example, the format is as follows:

media:DTVTR-X.jp? crid=xxxx&cnt_index=0001&ptime=13:54:00

6.1.2 Playback operation using the manifest file

6.1.3 Content transfer operation with the <sequence> element

The receiver performs the content transfer operation in accordance with the description of the <sequence> element. The transfer operation includes automatic transfer whereby contents are automatically transferred according to the attribute and manual transfer by user operation. The following describes each transfer operation:

6.1.3.1 Automatic transfer

When the time attribute is specified with an HTML file or still picture, transfer to the content described in the next attribute occurs in accordance with the time described in the time attribute. With an HTML file, the a element causes transfer to an HTML acquired by the HTML document or communication server, and the content presented may be different from the file specified by the <sequence> element, but transfer occurs regardless of the current presentation status. When the time attribute is specified with an MP4 file, it should be ignored.

6.1.3.1.1 Manual transfer

Manual transfer occurs when a tool of operation enabling transfer to a file described in the prev attribute or next attribute based on the manifest file is used on the receiver. The tool of transfer is a matter of product planning: For example, transfer using arrow keys may be considered. In this case, it is preferable that the tool of operation can be distinguished from that for the playback function of individual media (HTML, MP4, etc.) and from the browser-specific function (cache function, tab browse function, etc.).

It is preferable that resume playback is possible when manual transfer is performed during playback of an MP4 file and then transfer to the MP4 file is executed again. With manual transfer, it is also possible to transfer to another document in the content of the HTML document where the a element is described or to an HTML document on the communication server, in the same way as automatic transfer. Transfer occurs to the destination described in the prev or next attribute.

6.1.3.1.2 Operation during interruption of transfer

It is preferable to perform the suspend operation when the same content is played back during interruption of transfer caused by the operation, not described as a content, other than operations of transfer to another document in the content of HTML document where the a element is described or to an HTML document on the communication server and transfer caused by <sequence> in the manifest. Also, resuming playback is preferable when restart of application is necessary at the end of the application, etc.

Refer to sections 6.3.1 and 6.3.3 for transfer from a storage-type broadcasting browser to HTML browser content or to browser content for mobile phone, and for transfer to storage-type

content with a Content Reference Identifier and a storage-type broadcasting browser when content information is presented by the content management function.

6.1.3.1.3 Presentation of storage-type contents

For the presentation of an MP4 file incorporated in an HTML document, basically follow the description of contents in the HTML document. From the viewpoint of receiver performance, however, when an MP4 file is played back or presented on another screen, user operability must not be degraded in operability of content, transfer operation of content, etc.

The presentation method when playing back an MP4 file not incorporated in an HTML document is a matter of product planning. However, the video image follows the size of the presentation screen and the resolution, and appropriate scaling shall be performed without changing the video aspect ratio. It is preferable to consider the presentation of different media types (video/mp4, audio/mp4) for packaging.

6.1.3.2 Operation due to content use conditions

For playback of storage-type content, the content use conditions (RMPI) specified in "Vol. 5, 2.3.6.3 Content use conditions (RMPI)" are applied to the entire content. When playing back a storage-type content using a manifest file, the following content described in the manifest file is also supported:

- Resource file described in the count-asset-id attribute of the <assets> element and is a control object of counting the number of viewings
- Trick play acceptable information described in the scene-id attribute located under the <assets> element.

See Vol. 5 for the processes of the receiver due to content use conditions.

6.1.4 Provisions for presentation of communication contents

6.1.4.1 Content presented by a storage-type broadcasting browser

The storage-type broadcasting browser specified in this standard is a browser packaged based on this standard to present storage-type content, and is specified as a different browser from HTML browsers.

6.1.4.2 Storage-type broadcasting browser and communication HTML browser

A storage-type broadcasting browser and an HTML browser show logical ideas and do not limit the receiver packaging method. The storage-type broadcasting browser and HTML browser can be packaged in one receiver, and used by switching the operation logically.

6.1.5 Guidelines for creation of content

The following shows the guidelines to be observed by broadcasters when they create content:

6.1.5.1 Layout assumed for creation of content

When content is specified from a broadcasting content, note during creation that the content

may be displayed in the data broadcast content presentation area.

6.1.5.2 Other items related to content

The content fonts, sizes, and graphic sizes used are not specified in this operation. When creating content, it is preferable to consider the operational guidelines of each broadcaster as well as the version and functions of the compatible storage-type broadcasting browser of each receiver manufacturer.

6.1.6 Guidelines for operation of DOM

6.1.6.1 Handling of control codes

In texts other than literal texts (pre element, etc.), the control code is interpreted as one halfwidth space.

Ignore the line feed immediately after the start or immediately before the end of the element specified in this volume. Interpret the line feed immediately after the start or immediately before the end of the invalid, nonsupported, or vacant element as one space.

6.2 Transmission, reference, and name space of content

Refer to section 6.3.4 for the transmission of a content to be referred from a storage-type broadcasting browser. The name space of a content consisting of multiple contents (files) is in the same layer. Therefore, when reference is made from a storage-type broadcasting browser, access to it is possible by only specifying a file name.

6.3 Transfer of content and startup of browser

From a content on a storage-type broadcasting browser, it is possible to transfer to HTML content on communications, browser content for mobile phone presented by a browser other than this browser, or HTML browser content. The specific transfer method is shown in the following:

6.3.1 Transfer to HTML browser content

From a storage-type broadcasting browser, transfer to HTML content or browser content for mobile phone is possible by describing the URI of the transfer destination in the href attribute of the a element of the HTML document. However, it is difficult for the storage-type broadcasting browser to display HTML content of the specified URI character string as is. In this case, a browser other than one of the storage type (HTML browser or browser for mobile phone) is started up to acquire the content and display it. To achieve the above operation, when designating HTML browser content or browser content for mobile phone, use the URI schema described in the href attribute to specify the object type of browser to start up. As a

specific example, when starting up an HTML browser and presenting HTML content, describe the URI character string as follows:

brw://example.com/sample.htm

Two URI schemas can be specified for a browser for mobile phone and an HTML browser, as shown in Table 6.1: URI Schemas.

Table 6-1: URI Schemas

Type of browser	Schema
Browser for mobile phone	mbrw
HTML browser	brw

Specifying the type of browser only shows the recommended browser, and the operation to select a presented browser is receiver-dependent. The operation of the storage-type broadcasting browser after an HTML browser or a browser for mobile phone is started up is also receiver-dependent.

When the HTML browser or the browser for mobile phone called up by this measure shows a full-screen display. However, when the following is described, a full screen is not displayed, and the storage-type broadcasting browser, HTML browser, and browser for mobile phone are displayed at the same time. The position of screens displayed at the same time is a matter of product planning.

brw://example.com/sample.htm?fcbrowser=remain

Here, fcbrowser is a character string specifying the storage-type broadcasting browser, and remain shows the securement of presentation areas after the HTML browser and the browser for mobile phone startup. If a video has been displayed by the storage-type broadcasting browser, it is preferable to only display the video. At that time, the browser display is restored after the display of the video is finished.

The operation for simultaneously displaying the storage-type broadcasting browser, HTML browser, and browser for mobile phone shown above is a matter of product planning, with the acceptability of packaging as a function included. Therefore, when fcbrowser is described as a query character string, ignore it.

6.3.2 Transfer between storage-type contents

Transfer of HTML browser contents with the same crid occurs when the URI of the transfer

destination is described in the href attribute of the a element. In this case, transfer is possible by specifying the absolute URI starting with "crid://". The description format of the Content Reference Identifier is as follows, as shown in Vol. 10, 8.1.1.

crid://<authority>/<data >

A relative URI can also be described in the same way as for HTML browser.

6.3.3 Transfer to storage-type content with a different Content Reference Identifier

When specifying a different Content Reference Identifier, transfer is possible only by describing the Content Reference Identifier in the href attribute of the a element. In this case, it is assumed that the storage-type content identified by the specified Content Reference Identifier exists in the terminal, and that the content has already acquired a license and has a format that can be presented by a storage-type broadcasting browser. The operation when these conditions are not satisfied is receiver-dependent.

It is also possible to present content information using the content management function by specifying ECG with the method shown in 6.3.1　The specific description is as follows:

crid://<authority>/<data >?mmbiapp=ecg

Here, ecg indicates ECG. The application type can only be specified when the Content Reference Identifier has been specified. The operation of the storage-type broadcasting browser after ECG starts up is receiver-dependent.

Table 6-2: Application Type

Application type	Character string
ECG	ecg

6.3.4 Transfer of contents in multiple browsers

Transfer of contents between the storage-type broadcasting browser, browser for mobile phone, and HTML browser is shown in Figure 6-6-1.

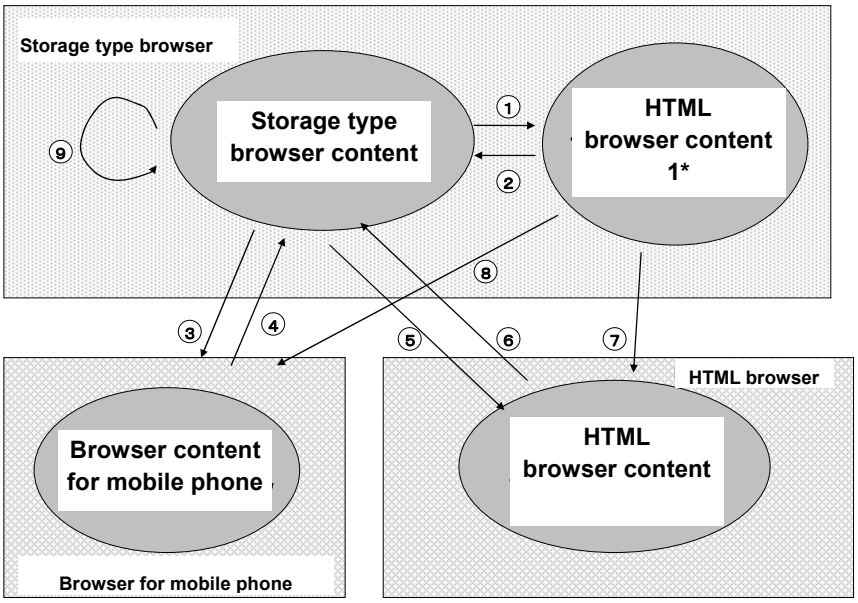


Figure 6-6-1: Relationship of transfer between browsers

The following explains each transfer:

Transfer ①: Use the href attribute of the a element to specify the URI of HTML browser content. For the URI scheme, specify “http:” or “https:”.

Transfer ②: Use the href attribute of the a element to specify the URI of storage-type broadcasting browser content. For the URI scheme, specify “crid:” or “media:”.

Transfer ③: Use the href attribute of the a element to specify the URI of mobile phone browser content. For the URI scheme, specify “mbrw:”.

Transfer ④: Use the href attribute of the a element to specify the URI of storage-type broadcasting browser content. For the URI scheme, “media:” can be specified. Refer to Media Scheme for details.

Transfer ⑤: Use the href attribute of the a element to specify the URI of HTML browser content. For the URI scheme, specify “brw:”.

Transfer ⑥: Use the href attribute of the a element to specify the URI of storage-type broadcasting browser content. For the URI scheme, “media:” can be specified. Refer to Media Scheme for details.

Transfer ⑦: Use the href attribute of the a element to specify the URI of HTML browser content. For the URI scheme, specify “brw:”.

Transfer ⑧: Use the href attribute of the a element to specify the URI of mobile phone browser content. For the URI scheme, specify “mbrw:”.

Transfer ⑨: Use the href attribute of the a element to specify the URI of storage-type broadcasting browser content. For the URI scheme, specify “crid:” or “media:”. However, if the transfer destination and source have the same crid, a relative URI can be described.

Annex 1 Considerations for NVRAM access

- Life of NVRAM

NVRAM is assumed to be packaged using a semiconductor memory device called a flash memory. This device has a limited number of rewrite iterations when reaching its life. The general upper-limit number of rewrite iterations is approximately 100,000 as of the year 2000.

To store information with an elapsed content presentation time, the use of global variables and Greg is preferable. For temporary data storage such as data transfer between multiple services, using the Greg function is preferable if it is supported. It is not recommended to use NVRAM for temporary data storage.

<Blank Page>

Annex 2 **Module compression format**

To compress a module, use the same compression format as PNG, based on RFC-1950 (ZLIB Compressed Data Format Specification version 3.3). See Table S2-1.

<Blank Page>

Table S2-1: Detailed Operation of zlib Compression Format

Field	Operation
Compression Method (4 bits)	8 ("deflate") only
Compression Info (4 bits)	7 or less (window 32 KB or less)
Flags	
FCHECK (5 bits)	(Value specified by RFC-1950)
Preset Dictionary (1 bit)	0 (no preset dictionary) only
Compression Level (2 bits)	(Any: Ignored during decoding)

<Blank Page>

Annex 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|StoreFinished)"
>
<!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 -->
```

<Blank Page>

Annex 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: "丸ゴシック" !important }
/* text-align */
p, input, textarea { text-align: left }
/* white-space */
p, input { white-space: normal !important }
pre, textarea { white-space: pre !important }
/* resolution */
body { resolution: 240x480 }
/* marquee */
p { -wap-marquee-loop: 1; -wap-marquee-dir: rtl !important }
}
/* opacity */
div { opacity: 1 }
```

<Blank Page>

Annex 5 Arrangement of elementary stream and empty carousel operations in the PMT

As an operation example of receivers, the specifications shown in Table S5-1 are assumed. In this example, the receiver controls the presence or absence of the display area based on the existence of description of the default component (component_tag=0x40) to 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 S5-1: 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	
	PMT: 0x40	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

Item	Reception condition			Presentation status			
				Presentation of broadcasting contents		Presentation of link contents	
	PMT: 0x40	Carousel status	Other causes	Presentation area	Presentation contents	Presentation area	Presentation contents
	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 existence of 0x40)	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 0x40 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.

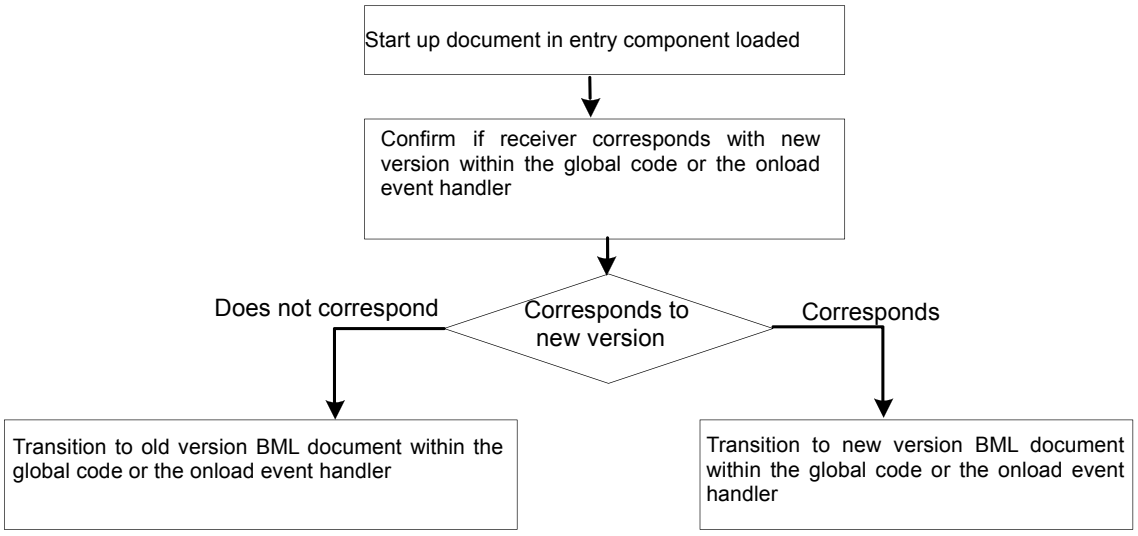
Annex 6 Version update of the BML version

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 14:

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 14 and the other number:

C. When transmitting startup documents in another component

- Specify a new version entry component by component tag value other than 0x40.

- 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 = 14, 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=14 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 Annex 7.

<Blank Page>

Annex 7 **Governing organization (T.B.D)**

<Blank Page>

Annex 8 Receiver common fixed colors

Receiver common fixed colors are indicated in Table:S8-1. 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:S8-1: Receiver 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

Index value	R	G	B	Alpha	Name/Comments
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	
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	

Index value	R	G	B	Alpha	Name/Comments
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
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	

Index value	R	G	B	Alpha	Name/Comments
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	
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	

Annex 9 Operation scope for browser pseudo-object in Browser for the X-profile contents on the internets

Operated as Table S9-1.

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 X-profile contents on the internets.
If the function is called, failure is returned as the return value. Refer to 4.3.11.4 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 S9-1 Operation scope of browser pseudo-objects

Function	Operations in Browser for the X-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					

Function	Operations in Browser for the X-profile contents on the internet	Notes			
		Others	Operations in broadcasting status	Operations in link status	(Note in original table)
Greg[]	O		O	O	Greg lifetime shall be the time period while reception function is operating
EPG functions					
epgGetEventStartTime()	O		O	O	
epgGetEventDuration()	O		O	O	
epgTune()	-	The browser shall be exited after execution.	O(*1)	O(*1)	
epgTuneToComponent()	-		-	-	
epgTuneToDocument()	-		-	-	
epgIsReserved()	-		-	-	
epgReserve()	-		-	-	
epgCancelReservation()	-		-	-	
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 functions					
readPersistentString()	-		-	-	
readPersistentNumber()	-		-	-	
readPersistentArray()	X		O	O	
writePersistentString()	-		-	-	
writePersistentNumber()	-		-	-	
writePersistentArray()	X		O	O	
copyPersistent()	-		-	-	
getPersistentInfoList()	-		-	-	

Function	Operations in Browser for the X-profile contents on the internet	Notes			
		Others	Operations in broadcasting status	Operations in link status	(Note in original table)
deletePersistent()	-		-	-	
getFreeSpace()	-		-	-	
Functions for controlling access-controlled areas					
isSupportedPersistentType()	-		-	-	
setAccessInfoOfPersistentArray()	-		-	-	
checkAccessInfoOfPersistentArray()	-		-	-	
writePersistentArrayWithAccessCheck()	-		-	-	
readPersistentArrayWithAccessCheck()	-		-	-	
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()	-		-	-	

Function	Operations in Browser for the X-profile contents on the internet	Notes			
		Others	Operations in broadcasting status	Operations in link status	(Note in original table)
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()	-		-	-	
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()	-		-	-	

	Function	Operations in Browser for the X-profile contents on the internet	Notes			
			Others	Operations in broadcasting status	Operations in link status	(Note in original table)
	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	
	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						

Function	Operations in Browser for the X-profile contents on the internet	Notes			
		Others	Operations in broadcasting status	Operations in link status	(Note in original table)
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()	-		-	-	
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		-	-	
saveResourceAs()	-		-	-	
saveResource()	X		-	-	
Functions for controlling bookmark area					
writeBookmarkArray()	-		-	-	
readBookmarkArray()	-		-	-	
deleteBookmark()	-		-	-	
lockBookmark()	-		-	-	
unlockBookmark()	-		-	-	

Function	Operations in Browser for the X-profile contents on the internet	Notes			
		Others	Operations in broadcasting status	Operations in link status	(Note in original table)
getBookmarkInfo()	-		-	-	
getBookmarkInfo2()	-		-	-	
startResidentBookmarkList()	-		-	-	
Printing functions API - printing basic functions					
getPrinterStatus()	-		-	-	
printFile()	-		-	-	
printTemplate()	-		-	-	
printUri()	-		-	-	
printStaticScreen()	-		-	-	
Printing functions API - memory card functions					
saveImageToMemoryCard()	X		O(*1)	O(*1)	
saveHttpServerImageToMemory Card()	X		O(*1)	O(*1)	
saveStaticScreenToMemoryCard( )	-		-	-	
Multimedia broadcasting-specific functions					
X_TMM_mailTo()	X		O(*1)	O(*1)	
X_TMM_startResidentApp()	O		O	O	
X_TMM_phoneTo()	X		O(*1)	O(*1)	
X_TMM_getRcvCond ()	X		O(*1)	O(*1)	
X_TMM_getCurPos ()	X		O(*1)	O(*1)	
X_TMM_saveExAppFile()	X		O(*1)	O(*1)	
X_TMM_startExAV()	X		O(*1)	O(*1)	
X_TMM_stopExAV()	X		O(*1)	O(*1)	
X_TMM_writeSchInfo()	X		O(*1)	O(*1)	
X_TMM_getComBrowserUA ()	O(*2)		O(*2)	O(*2)	
X_TMM_writeAddressBookInfo()	X		O(*1)	X	
X_TMM_launchDocWithLink()	X		X	O(*2)	
X_TMM_chkAVtype()	X		O(*1)	O(*1)	
X_TMM_getIRDID()	X		O	O	
X_TMM_writeXproBM()	X		O(*2)	O(*2)	
X_TMM_epgReserve()	X		O(*1)	O(*1)	
X_TMM_epgCancelReservation()	X		O(*1)	O(*1)	
X_TMM_epgCheckReservation()	-		-	-	
X_TMM_epgSeriesCheckReserva tion()	-		-	-	
X_TMM_epgSeriesReserve()	X		O(*1))	O(*1)	
X_TMM_epgSeriesCancelReserv ation()	X		O(*1)	O(*1)	
X_TMM_epgRecCheckReservatio n()	X		-	-	
X_TMM_epgRecReserve()	X		O(*1)	O(*1)	
X_TMM_epgRecCancelReservati on()	X		O(*1)	O(*1)	
X_TMM_epgSeriesRecCheckRese rvation()	-		-	-	
X_TMM_epgSeriesRecReserve()	X		O(*1)	O(*1)	
X_TMM_epgSeriesRecCancelRes ervation()	X		O(*1)	O(*1)	

	Function	Operations in Browser for the X-profile contents on the internet	Notes			
			Others	Operations in broadcasting status	Operations in link status	(Note in original table)
	X_TMM_ecgLaunchContent()	X		O(*1)	O(*1)	
	X_TMM_ecgStoreStart()	X		O(*1)	O(*1)	
	X_TMM_ecgStoreTerminate()	X		O(*1)	O(*1)	
	X_TMM_ecgCheckStorageStatus()	X		O(*1)	O(*1)	
	X_TMM_ecgStoreReserve()	X		O(*1)	O(*1)	
	X_TMM_ecgStoreCancelReservation()	X		O(*1)	O(*1)	
	X_TMM_ecgStoreCheckReservation()	X		O(*1)	O(*1)	
	X_TMM_ecgIsContentStored()	X		O(*1)	O(*1)	
	X_TMM_ecgIsLockedStoredContent()	X		O(*1)	O(*1)	
	X_TMM_ecgDeleteStoredContent()	X		O(*1)	O(*1)	
	X_TMM_ecgGetStorableSpace()	X		O(*1)	O(*1)	
	X_TMM_startExFullDisplay	X		O(*1)	O(*1)	
	X_TMM_stopExFullDisplay	X		O(*1)	O(*1)	

Annex 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 or higher / SSL3.0 or higher shall be used when personal information is handled in interactive services and X-profile link 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.

Annex 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

If the receiver will implement a function for cacheing X-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 X-profile linked content to broadcasting content

* However, a receiver supporting "If modified since" shall not clear the cache if a transition is made from X-profile linked content to broadcasting content.

Annex 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 higher / SSL3.0 or higher 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

For protection of personal information, etc., the receiver shall be equipped with the encrypted communication measure of TLS1.0 or higher / SSL3.0 or higher. To prevent “impersonation” in services, the receiver supports server authentication.

Annex 13 **Media Scheme**

When the anchor, etc. specified by the media scheme is selected from a storage-type broadcasting browser or the receiver's browser function and mail function, the media scheme function enables viewing, viewing reservation, and recording reservation of the real-time-type broadcast receiving functions as well as storage, storage reservation, playback of store content, etc. of the storage-type broadcast receiving functions. The method of processing each function in a receiver with restriction on these operations is a matter of product planning.

1. Syntax of the media scheme

The following shows the syntax of the media scheme:

Media Scheme	:	"media:" Broadcaster [Options]
Broadcaster	:	[(Callsign | "unknown") "."] MediaType "." CountryCode
Callsign	:	Callsign established by ITU and the organizations of each country
MediaType	:	"DTVTR-X" ; "DTVTR-X" multimedia broadcast mobile phone receiver
CountryCode	:	2ALPHA ; ISO-3166 2-letter country code
Options	:	"?" Option *("&" Option)
Option	:	("p_channel" "=" PCHANNEL) | ("consign_id" "=" CONDIGNID) | ("service_id" "=" SERVICEID) | ("crid" "=" CRID) | ("ptime" "=" PLAYTIME) | ("pdate" "=" PLAYDATE) | ("ecg" "=" VALUE") | ("cnt_index" "=" CNTINDEX)
PCHANNEL	:	4DIGIT ; Shall be recognizable even in case of 1DIGIT～3DIGIT.
CONSIGNID	:	4HEX ; Consigned broadcaster identified
SERVICEID		4HEX ; Service ID
CRID	:	See Vol. 10, 8.1 "Operation of Content Reference Identifier (CRID)". ; <authority>/<data> is described
PLAYTIME	:	2DIGIT ":" 2DIGIT ":" 2DIGIT ;hh:mm:ss
PLAYDATE	:	4DIGIT "-" 2DIGIT "-" 2DIGIT "T" 2DIGIT ":" 2DIGIT ":" 2DIGIT ; DATE format: YYYY –MM-DDThh:mm:ss
CNTINDEX	:	Character string value to be described in the label attribute on the sequence element of the manifest file
ALPHA	:	LOWALPHA | UPALPHA
LOW ALPHA	:	"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"
UPALPHA	:	"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"
DIGIT	:	"0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9"

HEX	:	DIGIT | “A” | “B” | “C” | “D” | “E” | “F” | “a” | “b” | “c” | “d” | “e” | “f”
PCT-ENCODED	:	“%” HEX HEX
MARK	:	“!” | “#” | “$” | “ ’ ” | “(” | “)” | “*” | “+” | “,” | “-” | “.” | “/” | “:” | “;” | “<” | “>” | “@” | “[” | “¥” | “]” | “^” | “_” | “‘” | “{” | “|” | “}” | “~”
VALUE	:	“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”

2. Media scheme description method

The following shows an example of describing the media scheme:

media:DTVTR-X.jp?p_channel=0001&crid=example.jp/20101027/rt/0001/12345

The method for describing each optional parameter is shown below. When an optional parameter is omitted, it is preferable to display the operation screen or menu, which can call up all the functions of the multimedia broadcasting receiver.

3. Supported optional parameters

The following shows the supported optional parameters:

Name	Optional parameter	Object function
Physical CH	p_channel*1)	Real-time-type broadcast receiving function
Service ID	service_id	
Consigned broadcaster ID	consign_id	Common
Content ID	crid	
ECG startup	ecg	
Playback position (time)	ptime	Storage-type broadcast receiving function
Playback date/time	pdate	
Content index	cnt_index*2)	

*1) For the relationship between p_channel and the center frequency, see “Vol. 7, 7.2.1 TS_id list”.

*2) It is optional whether cnt_index is supported on the receiver.

The consigned broadcaster ID is used to judge on the receiver to which application linkage is made during linkage when the receiver has several applications with the same functions. See the service agreement of the consigned broadcaster for details.

However, the physical CH or the consigned broadcaster ID is not operated for the time being.

4. Conditions for anchor clipping and text clipping

Follow the conditions below for clipping HTML anchors and text anchors:

- Specifying MediaType is mandatory (capital letters and lower case are ignored).
- Specifying CountryCode is mandatory (capital letters and lower case are ignored).
- Any order of use of optional parameters is acceptable.
- Recursive clipping is not performed in text clipping. (Clipping is performed in the range of valid character strings, and judgment regarding whether it is valid or invalid is made in that range. When judged invalid, all is handled as invalid.)

5. HTML (XTTML) object tag

The object tag shall be tag A.

6. Transfer to ECG

To start up ECG using the receiver's browser function, mail function, etc., specify "ECG" for optional parameter ecg. It is assumed that parameter ecg is specified with crid: When ecg is specified with another optional parameter, it is preferable that ECG operates with the optional parameter ignored. The operation of the storage-type broadcasting browser after ECG starts up is a matter of product planning.

7. Operation during selection

The following specifies the operation when the media scheme is selected. Link to the appropriate application, following the description below:

(1) Real-time-type broadcast receiving function

As for real-time-type broadcast receiving functions: When viewing of a program, viewing of a reservation, or recording of a reservation is executed, describe optional parameters p_channel, serivce_id, and crid as appropriate.

When the program specified by crid is being broadcast, viewing is started. When the program specified by crid is specified before the start time, the operation is performed to have a viewer reserve viewing or recording.

When service_id is specified, viewing the program being broadcast with the specified ID is started. When service_id is specified, other optional parameters are ignored.

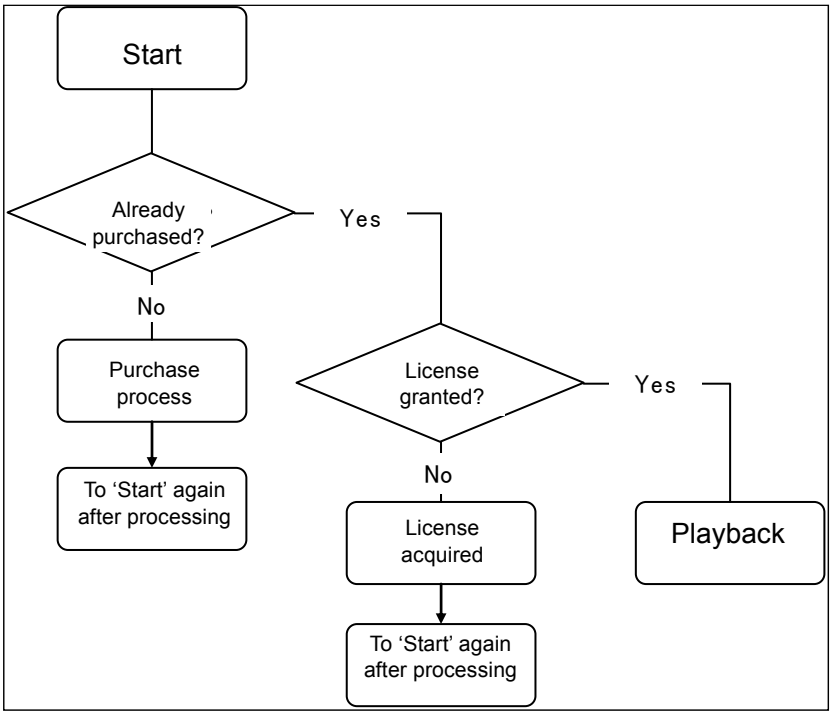
The operation when any change in the program guide occurs due to extension or interruption of a program is a matter of product planning.

(2) Storage-type broadcast receiving function

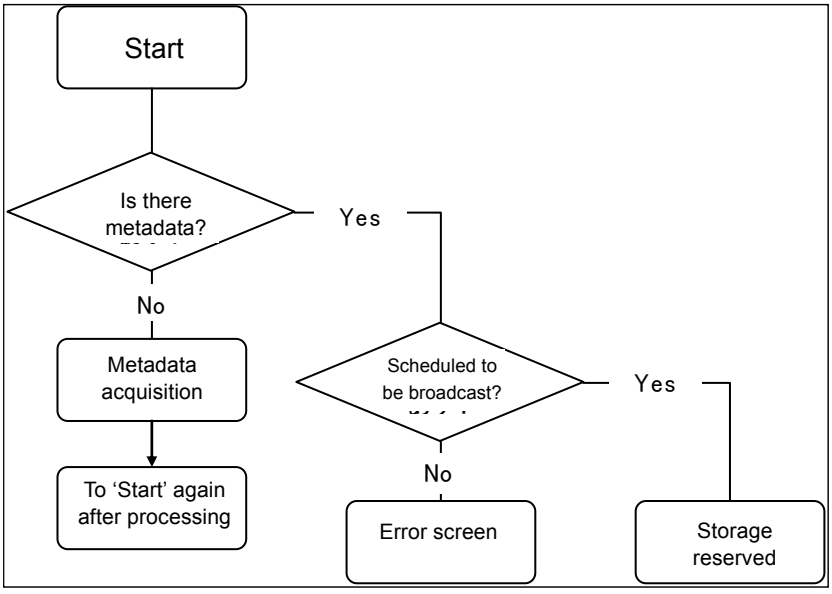
As for storage-type broadcast receiving functions, content playback reservation and

playback execution are possible. The following shows the operations when an object content specified by crid is stored or not as well as the operations leading to playback reservation or playback. The operation when metadata is not yet acquired is a matter of product planning.

(A) When an object content is already stored



(B) When an object content is not yet stored



(C) When object content is already stored and there is unacquired data (Completion of storage content is necessary.)

Complete the storage content and perform transfer in (A) again. If completion of storage content has failed, an error screen will appear.

(D) When pdate is specified

It is assumed that pdate is specified with crid. When pdate is specified, perform playback reservation on the specified date and time. When pdate is specified, restrict trick play (fast forward, fast back, etc.). If the relevant content is not stored, perform storage reservation following the operation "(B) When object content is not yet stored". The operation if content is not stored and a license is not acquired on the date/time when playback reservation is performed is a matter of product planning.

If pdate is past the current time, perform reservation or confirm the user.

In this case, playback reservation is not necessary.

(E) When ptime is specified

It is assumed that ptime is specified with crid and cnt_index. When ptime is specified, perform playback from the playback time position at the start of a video specified by cnt_index.

For example, when 10 seconds is specified for a video content, playback starts from the position of 10 seconds past the video. However, play back a content with no time axis (still picture, etc.) from the start. If the relevant content is not stored, perform storage reservation following the operation "(B) When object content is not yet stored".

Because cnt_index is valid only for a content consisting of multiple files, specifying it in a content of a single file except the manifest file will be invalid. Therefore, a playback operation with ptime specified is requested with a single file except the manifest file, the ptime value is invalid, and playback will start from the start of the content.

If cnt_index is omitted, the file to be first played back will also be played back in the same way as with a single file.

A receiver that cannot interpret cnt_index will perform the same operation as when cnt_index is omitted.

8. Caution

If the options of the real-time-type broadcast receiving function and storage-type broadcast receiving function are specified in the media scheme at the same time, an error will result.

Annex 14 Operation of the getBrowserStatus() argument

The table below shows functions available for inspections in combination of “statusname” and “additionalinfo” when “sProvider” is "TMMXpro".

Table S14—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

OPERATIONAL GUIDELINES FOR
TERRESTRIAL MOBILE MULTIMEDIA BROADCASTING BY
TRANSMISSION SYSTEM BASED ON CONNECTED SEGMENTS
FOR VHF-HIGH BAND

ARIB TECHNOCAL REPORT

ARIB TR-B33 VERSION 2.3-E1
(Fascicle 1)
(September 30th, 2015)

This Document is based on the ARIB technical report of “Operational Guidlines for Terrestrial Multimedia Broadcasting by Transmission System Based on Connected Segments for VHF-HIGH Band” in Japanese edition and translated into English in October 2015.

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