International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

H.703

(07/2016)

SERIES H: AUDIOVISUAL AND MULTIMEDIA SYSTEMS

IPTV multimedia services and applications for IPTV – General aspects

Enhanced user interface framework for IPTV terminal devices

Recommendation ITU-T H.703

ITU-T H-SERIES RECOMMENDATIONS

AUDIOVISUAL AND MULTIMEDIA SYSTEMS

CHARACTERISTICS OF VISUAL TELEPHONE SYSTEMS	H.100–H.199
INFRASTRUCTURE OF AUDIOVISUAL SERVICES	
General	H.200–H.219
Transmission multiplexing and synchronization	H.220–H.229
Systems aspects	H.230–H.239
Communication procedures	H.240–H.259
Coding of moving video	H.260–H.279
Related systems aspects	H.280–H.299
Systems and terminal equipment for audiovisual services	H.300–H.349
Directory services architecture for audiovisual and multimedia services	H.350–H.359
Quality of service architecture for audiovisual and multimedia services	H.360–H.369
Telepresence	H.420–H.429
Supplementary services for multimedia	H.450–H.499
MOBILITY AND COLLABORATION PROCEDURES	
Overview of Mobility and Collaboration, definitions, protocols and procedures	H.500–H.509
Mobility for H-Series multimedia systems and services	H.510–H.519
Mobile multimedia collaboration applications and services	H.520–H.529
Security for mobile multimedia systems and services	H.530–H.539
Security for mobile multimedia collaboration applications and services	H.540–H.549
Mobility interworking procedures	H.550–H.559
Mobile multimedia collaboration inter-working procedures	H.560–H.569
BROADBAND, TRIPLE-PLAY AND ADVANCED MULTIMEDIA SERVICES	
Broadband multimedia services over VDSL	H.610–H.619
Advanced multimedia services and applications	H.620–H.629
Ubiquitous sensor network applications and Internet of Things	H.640–H.649
IPTV MULTIMEDIA SERVICES AND APPLICATIONS FOR IPTV	
General aspects	**H.700–H.719**
IPTV terminal devices	H.720–H.729
IPTV middleware	H.730–H.739
IPTV application event handling	H.740–H.749
IPTV metadata	H.750–H.759
IPTV multimedia application frameworks	H.760–H.769
IPTV service discovery up to consumption	H.770–H.779
Digital Signage	H.780–H.789
E-HEALTH MULTIMEDIA SERVICES AND APPLICATIONS	
Personal health systems	H.810–H.819
Interoperability compliance testing of personal health systems (HRN, PAN, LAN, TAN and WAN)	H.820–H.859
Multimedia e-health data exchange services	H.860–H.869

For further details, please refer to the list of ITU-T Recommendations.

Recommendation ITU-T H.703

Enhanced user interface framework for IPTV terminal devices

Summary

The enhanced user interface (UI) framework defines the functional elements supporting enhanced capability of user interaction over an IPTV terminal device. The enhanced UI functions are located in the end user functions of terminal device and coordinate together with IPTV terminal functions. The enhanced UI functions include user interface of touchscreen and speech.

This Recommendation also describes the event features, general requirements and functionalities of the framework to support enhanced user interface for IPTV terminal device, which are recommended on the basis of IPTV architecture ITU-T Y.1910.

History

Edition	Recommendation	Approval	Study Group	Unique ID*
1.0	ITU-T H.703	2016-07-14	16	11.1002/1000/12908

Keywords

IPTV, speech interface, touch interface, user interface.

* To access the Recommendation, type the URL http://handle.itu.int/ in the address field of your web browser, followed by the Recommendation's unique ID. For example, http://handle.itu.int/11.1002/1000/11830-en.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at http://www.itu.int/ITU-T/ipr/.

Table of Contents

Recommendation ITU-T H.703

Enhanced user interface framework for IPTV terminal devices

1 Scope

This Recommendation describes the event features, general requirements and functionalities of the framework to support enhanced user interface for IPTV terminal device. This Recommendation focuses on touchscreen and speech at a high-level.

The purpose of enhanced UI framework is to define the functional element supporting enhanced capability of user interaction over an IPTV terminal device regarding touchscreen and speech interface.

2 References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

[ITU-T H.760] Recommendation ITU-T H.760 (2009), *Overview of multimedia application frameworks for IPTV services*.

[ITU-T Y.1901] Recommendation ITU-T Y.1901 (2009), *Requirements for the support of IPTV services.*

[ITU-T Y.1910] Recommendation ITU-T Y.1910 (2008), *IPTV functional architecture.*

3 Definitions

3.1 Terms defined elsewhere

This Recommendation uses the following terms defined elsewhere:

3.1.1 application [b-ITU-T Y.101]: A structured set of capabilities, which provide value-added functionality supported by one or more services.

3.1.2 application programming interface (API) [b-ITU-T Y.110]: This is an implementation interface between equipment and a software module and does not have any physical realization as it is internal to the equipment.

3.1.3 content [b-ITU-T H.780]: A combination of audio, still image, graphic, video, or data.

NOTE – Variety of formats is classified as the "data" (e.g., text, encoded values, multimedia description language introduced by [ITU-T H.760]).

3.1.4 terminal device (TD) [ITU-T Y.1901]: An end-user device which typically presents and/or processes the content, such as a personal computer, a computer peripheral, a mobile device, a TV set, a monitor, a VoIP terminal or an audio-visual media player.

3.1.5 user agent [b-W3C WebArch]: One type of web agent; a piece of software acting on behalf of a person.

3.2 Terms defined in this Recommendation

This Recommendation defines the following terms:

3.2.1 approaching tool: A tool to contact or to point touchscreen, e.g., a finger or a stylus pen.

3.2.2 event packet: A packetized stream of touch input event.

3.2.3 speech: Speech is the vocalized form of human communication.

3.2.4 speech recognition: A kind of user interface for translation of spoken words into text.

3.2.5 touchscreen: A touchscreen is an electronic visual display that the user can control through simple or multi-touch gestures by touching the screen.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

API	Application Programming Interface
DB	Database
DNN	Deep Neural Network
DOM	Document Object Model
EPG	Electronic Programme Guide
ICT	Information and Communication Technology
JPEG	Joint Photographic Experts Group
NLP	Natural Language Processing
PNG	Portable Network Graphics
QA	Question Answering
TD	Terminal Device
TV	Television
UI	User Interface
VOD	Video on Demand

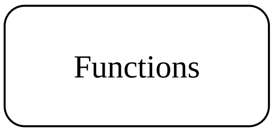
5 Conventions

The following conventions are used in this Recommendation:

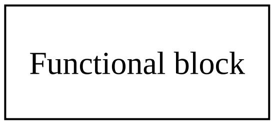
- The keywords "is required to" indicate a requirement which must be strictly followed and from which no deviation is permitted, if conformance to this Recommendation is to be claimed.
- The keywords "is prohibited from" indicate a requirement which must be strictly followed and from which no deviation is permitted, if conformance to this Recommendation is to be claimed.
- The keywords "is recommended" indicate a requirement which is recommended but which is not absolutely required. Thus, this requirement need not be present to claim conformance.
- The keywords "is not recommended" indicate a requirement which is not recommended but which is not specifically prohibited. Thus, conformance with this Recommendation can still be claimed even if this requirement is present.
- The keywords "can optionally" indicate an optional requirement which is permissible, without implying any sense of being recommended. This term is not intended to imply that

the vendor's implementation must provide the option and the feature can be optionally enabled by the network operator/service provider. Rather, it means the vendor may optionally provide the feature and still claim conformance with this Recommendation.

– The keyword "functions" is defined as a collection of functionalities. It is represented by the following symbol in the context of IPTV architecture:



– The keyword "functional block" is defined as a group of functionalities that have not been further subdivided at the level of detail described in this Recommendation. It is represented by the following symbol in the context of IPTV architecture:



6 Introduction

The enhanced UI is considered as a functional element supporting enhanced capability of user interaction in an IPTV terminal device. Figure 1 shows the recommended conceptual diagram for presenting relation of enhanced UI functions in the case of IPTV terminal device, which is based on the IPTV architecture in [ITU-T Y.1910]. The enhanced UI functions element is located in the end user functions of terminal device, and coordinates together with IPTV terminal functions. The enhanced UI functions include user interface of touchscreen and speech.

A touchscreen is an input device that allows users to operate a device simply by touching the display screen. Touchscreen technology is one of the ways for IPTV users to interact with IPTV services. However, most IPTV sets still do not have touchscreen capabilities for direct touching on an IPTV screen. However, there are alternative ways for IPTV viewers to interact with IPTV services, such as using handheld devices with touch screens (e.g., smartphones and tablet computers).

Handheld devices can be used as "screen-mirroring remote controllers". These handheld devices have touchscreen capabilities that, coupled with screen-mirroring remote control service capabilities, can be used to control IPTV functions and applications of the set-top box as if the user directly touched the IPTV screen.

With screen-mirroring remote control service, users can watch IPTV content with an intuitive and convenient way to manipulate IPTV functions and applications.

Speech interface is used in IPTV services to control and interact with the IPTV terminal device. Users can, for example, change channels or control volumes with their own voice. Users can also ask questions in natural language on the IPTV programmes such as the title of a specific TV drama using advanced speech interface functions. Speech interface is much used in the current diverse information and communication technology (ICT) systems including mobile environment that provides convenient user experiences to control various devices and tools due to the advancement of speech recognition and speech synthesis technology. Some examples would include car navigation systems with speech, voice search system, interactive intelligent assistant service, English educational services, and intelligent chatting robots. It also provides the core function for providing accessibility to ICT systems including IPTV for users with disabilities.

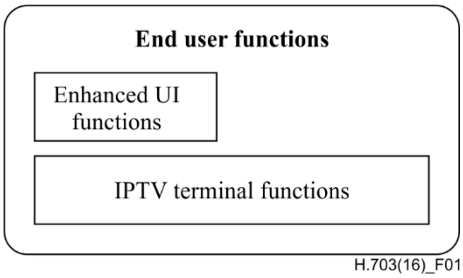


Figure 1 – Enhanced UI functions in case of IPTV terminal device

7 Requirements

General UI requirements for IPTV service are as follows:

- The UI is recommended to support events from input devices like keyboard, mouse, etc. in IPTV terminal device.
- The UI is recommended to support events from input interfaces for joysticks, microphone, etc. in IPTV terminal device.
- The UI is recommended to provide multiple modes of input and output in IPTV terminal device, e.g., touch and speech.
- The UI is required to support an application interacting with a user in IPTV terminal device.
- The UI is required to support MAFR [ITU-T H.760] application, especially web-based application, and hybrid web application interacting with a user in IPTV terminal device, in which a hybrid web application is written by a MAFR standard language together with a system application programming interface (API) call.
- The UI is required to be comprised of three distinct layers for MAFR-based interface between user and IPTV terminal device: MAFR user interface layer, hybrid application interface layer and system service interface layer.
- The UI is recommended to have the MAFR user interface layer that requests a process according to the system API call from a hybrid application interface layer, which performs a process according to the MAFR standard language.
- The UI is recommended to have a hybrid application interface layer that requests the process corresponding to the system API call from system service interface layer, which includes a system API processing unit that processes the system API in response to the request of the MAFR user interface layer.
- The UI is recommended to have a system service interface layer that performs a system-dependent operation corresponding to the system API that is based on the request of the hybrid application interface layer. The UI is recommended to notify a user about what kinds of UI modes are supported in IPTV terminal device.
- The UI is recommended to support user experience of interaction in different mode of IPTV terminal devices, including mobile and pad type device.
- The UI is recommended to support events from a remote controller in IPTV terminal device.
- The UI is recommended to provide an interface for end-user's preference setting in IPTV terminal device.
- The UI is recommended to provide an interface for switching between different display modes in IPTV terminal device.

7.1 Touchscreen requirement

The following are requirements regarding touchscreen in IPTV terminal device:

- The touchscreen component is required to support touch capability to a user.
- The touchscreen component is required to support the touch-based interaction between the user and touchscreen.
- The touchscreen component is required to support touch-sensitive screen to capture interactions between the user and touchscreen.
- The touchscreen component is recommended to support touchscreen remote control functions such as screen-mirroring remote control functions.
- The touchscreen component is recommended to support instantaneous response to user by reaction.
- The touchscreen component, including touch controller, is recommended to consume as little power as possible.
- The touchscreen component is recommended to support other types of touch methods, e.g., pen or mouse.
- The touchscreen component is recommended to support multi-touch input methods.

The requirements for the screen-mirroring remote control service are as follows:

- IPTV terminal device is required to support frame buffer access interface for capturing IPTV screen.
- IPTV terminal device is required to support event adaptation capabilities for adapting events such as touch input event and key input event.
- IPTV terminal device is recommended to support event drivers for analysing events such as touch input event and key input event.
- IPTV terminal device is recommended to support image compression function e.g., joint photographic experts group (JPEG), portable network graphics (PNG), etc.
- IPTV terminal device is recommended to support moving picture encoding function, e.g., ITU-T H.264, etc.
- Remote controller is required to support display capability.
- Remote controller is required to support touchscreen capability.
- Remote controller is required to support event processing capabilities for aggregating events such as touch input event and key input event.
- IPTV terminal device and remote controller is recommended to support a communication protocol and message format for a data communication.

7.2 Speech requirement

The following are the requirements regarding speech interface in IPTV terminal device:

- The speech component is required to support speech capability to a user.
- The speech component is required to support voice commands to control IPTV services.
- The speech component is recommended to be adaptable to the different user devices for speech interface.
- The speech component is recommended to support instantaneous response to user by reaction.
- The speech component is recommended to support the user's preferences.

- The speech component, including speech controller, is recommended to consume as little power as possible.
- The speech component is recommended to support electronic programme guide (EPG) function in natural languages for convenient programme search.
- The speech component is recommended to support information search function in natural languages for convenient information search.
- The speech component is recommended to support question answering function for convenient user interaction with natural language.
- The speech component is recommended to support speech recognition and speech synthesis to make the natural languages dialogue possible between the user and the system.
- The speech component is required to support text input in case speech input is not available.
- The speech component is recommended to support pre-recorded audio output in case speech synthesis is not available.
- The speech component is recommended to support natural language processing (NLP) function to make the natural languages dialogue possible between the user and the system.
- The speech component is recommended to support context recognizing function to make the natural language dialogue possible using the context information such as location and current date and time.
- The speech component is recommended to support dialogue managing function to make the natural language dialogue possible between the user and the system.
- The speech component is recommended to support multimodality in combination with other means of interface when needed.

  NOTE – Multimodality for user interface is a function to provide different modes of interface such as textual, audio, and visual modes in combination with media to create input and output.
- The speech component is recommended to support natural language generating function for the speech output in a dialogue.
- The speech component is recommended to formally define the whole process of natural language processing for the speech interface.

8 UI functionality and interfaces

UI functional components are composed essentially of touchscreen and speech functional block as follows (see Figure 2):

- **Touchscreen functional block**: supports touch functionality between user and IPTV terminal device;
- **Speech functional block**: supports speech functionality between user and IPTV terminal device.

High-level interfaces are described for supporting enhanced UI in the next clauses.

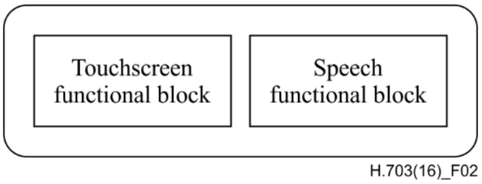


Figure 2 – Logical functional block diagram

8.1 Touchscreen

The touchscreen function is required to support touch-sensitive screen to capture interactions between user and touchscreen. This is a required basic touchscreen function necessary for primitive operations.

The touch-based user interface needs to provide quality support and offer the ability to interpret single-point activity on touchscreen. The interface needs to support not only single-point input, but also continuous-point input as well multi-point input method in an interactive environment.

To support touch capability, relevant product components usually consist of display, touch sensor, touch controller and software driver. Among these, touch sensor is usually composed of a transparent panel with touch-sensitive surface. Touch controller is the interface between the sensor and the display. The controller takes information from the touchscreen and translates it into information a computer can understand, such as left and right clicks, etc. Software driver is the controller to communicate and helps the controller to recognize the touch input.

8.1.1 General touch input functions

The touch input functional entity is composed of hardware abstraction functions and service adaptation functions.

8.1.1.1 Hardware abstraction functions

The hardware abstraction functional entity is composed of input device access module, input core module, and input event handling module. The input device access module communicates with the touch controller of the touch device and report touch input events to the input core module. The touch input events are passed on to the input event handling module by the input core module. The input event handling module distributes the touch input events to the service adaptation functional entity.

The input device access module performs:

- Initializing and configuring the touch device.
- Interpreting the touch event information generated by the touch controller of the touch device and converting the touch event information to the touch input event data that the input event handling module understands and can handle.
- Packetizing the touch input event data and reporting the packetized data to the input core module.

The input core module performs:

- Registering the input device access module to access the touch device.
- Registering the input event handling module.
- Mapping the input device access module to the input event handling module.
- Dispatching the touch input event data with the information of the input device access module to the input event handling module.

The input event handling module performs:

- Abstraction of the touch input events for hardware-agnostic handling of touch input events.
- Structuring and storing the touch input event data.
- Notifying the occurrence of the touch input events to the service adaptation functional entity.

8.1.1.2 Service adaptation functions

The service adaptation functional entity:

- notifies the occurrence of the touch input events to the applications;

- acquires the touch input event data from the hardware abstraction functional entity;
- forwards the touch input event data to the applications.

8.1.2 General touch input interfaces

The touch input interfaces are described in Figure 3.

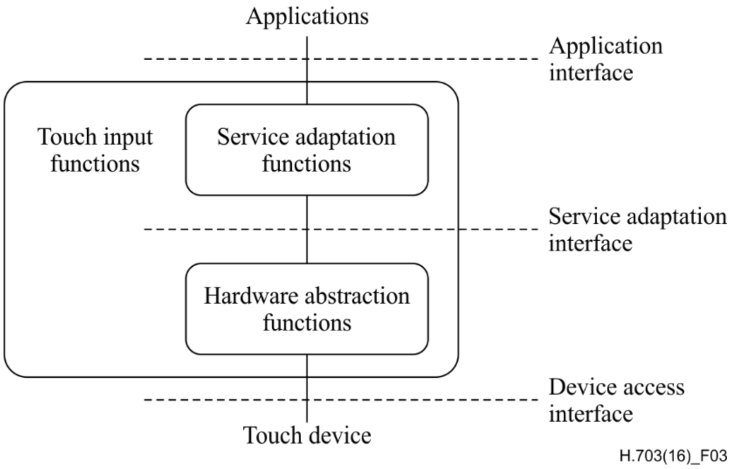


Figure 3 – Touch input functions and interfaces

8.1.2.1 Device access interface

This interface is between the hardware abstraction functional entity and the touch controller of the touch device. This interface facilitates the transfer of the touch event information from the touch controller to the hardware abstraction functional entity. This interface facilitates the transfer of initialization or configuration data from the hardware abstraction functional entity to the touch controller.

8.1.2.2 Service adaptation interface

This interface is between the service adaptation functional entity and the hardware abstraction functional entity. It facilitates the transfer of the touch input event occurrence notification and the touch input event data from the hardware abstraction functional entity to the service adaptation functional entity. The interface allows the service adaptation functional entity to get touch input event data from the hardware abstraction functional entity (e.g., coordination, duration, touch event handler).

8.1.2.3 Application interface

This interface is between applications and the service adaptation functional entity. The application interface facilitates the transfer of the touch input event occurrence notification and the touch input event data from the service adaptation functional entity to applications. The interface allows an application to get touch input event data from the touch input functional entity.

8.1.3 Touch input events

The following touch input event attributes are required for handling touch input events:

- The X coordinate of the centre of the touching area on touchscreen surface, which is perceived by the touch controller when the touch panel is pressed by the approaching tool, e.g., a finger or a stylus pen.
- The Y coordinate of the centre of the touching area on touchscreen surface, which is perceived by the touch controller when the touch panel is pressed by the approaching tool.
- The indication whether the approaching tool is touching the touchscreen surface.

The following touch input event attributes are recommended for handling touch input events:

– The length of the longer dimension of the touching area. Generally, the touching area is expressed as an elliptical shape. This attribute means the largest distance between antipodal points of the touching ellipse and the axis related to this largest distance is called the major axis.
– The length of the shorter dimension of the touching area. Generally, the touching area is expressed as an elliptical shape. This attribute means the smallest distance across the touching ellipse and the axis related to this smallest distance is called the minor axis.
– The length of the longer dimension of the approaching tool.
– The length of the shorter dimension of the approaching tool.
– The physical pressure on the touching area. This attribute may be reported when the touchscreen hardware supports the pressure-sensitive touchscreen technology. This attribute may be approximated by the ratio of the length of the longer dimension of the approaching tool versus the length of the longer dimension of the touching area.
– The distance between the approaching tool and the touchscreen surface. Zero distance means the approaching tool is touching the touchscreen surface.
– The orientation of the touching area or the approaching tool. Generally, the touching area is expressed as an elliptical shape. This attribute represents the rotation degree of the major axis of the touching ellipse in reference to the Y-axis of the touchscreen surface.
– The type of approaching tool, such as a finger or a stylus pen. The tracking identifier of the approaching tool. This attribute is an arbitrary non-negative integer that is used to identify and track each tool independently when multiple tools are active. For example, when multiple fingers are touching the touchscreen surface, each finger should be assigned a distinct identifier that is used as long as the finger remains in contact with the screen. The tracking identifiers may be reused when their associated tools move out of traceable range.

The touch input event data are generated by the input device access module of the hardware abstraction functional entity. The touch input event data are grouped as event packet to be delivered to the input event handling module. Moreover, the touch input event data can also be reported as a packet to applications.

Each touch input event data is composed of three elements: event type, event code, and value, as shown in Table 1.

Table 1 – Elements for touch input event data

Element	Description
Event type	This indicates the type of the input event. For handling touch input events, two types of input events are recommended. One is for indicating the touch input event, another is for separating input events. For example, the event type can be expressed with enumerators as follows: – EV_SYN: This indicates that the input event is the marker or the delimiter for separating input events. – EV_ABS: This indicates that the input event is the touch input event.
Event code	When the event type indicates the marker or the delimiter, this element denotes the marker to indicate the end of input events or the delimiter to separate sets of touch input events. For example, the event code for the marker or the delimiter can be expressed with enumerators as follows:

Table 1 – Elements for touch input event data

Element	Description
	– SYN_REPORT: This means that the input event is the marker to indicate the end of input events. – SYN_MT_REPORT: This means that the input event is the delimiter to separate sets of touch input events
	When the event type indicates the touch input event, this element denotes the attribute of the touch input event. For example, the event code for the touch input event attribute can be expressed with enumerators as follows: – ABS_MT_COR_X: This means that the touch input event attribute is for the X coordinate of the centre of the touching area. – ABS_MT_COR_Y: This means that the touch input event attribute is for the Y coordinate of the centre of the touching area. – ABS_TOUCH: This means that the touch input event attribute is for the indication whether the approaching tool is touching the touchscreen surface or not. If the value for this event attribute is 1, it means that the approaching tool is touching the touchscreen surface. If the value for this event attribute is 0, it means that the approaching tool is released from the touchscreen surface. – ABS_MT_TOUCH_MAJOR: This means that the touch input event attribute is for the length of the longer dimension of the touching area. – ABS_MT_TOUCH_MINOR: This means that the touch input event attribute is for the length of the shorter dimension of the touching area. – ABS_MT_WIDTH_MAJOR: This means that the touch input event attribute is for the length of the longer dimension of the approaching tool. – ABS_MT_WIDTH_MINOR: This means that the touch input event attribute is for the length of the shorter dimension of the approaching tool. – ABS_MT_PRESSURE: This means that the touch input event attribute is for the physical pressure on the touching area. – ABS_MT_DISTANCE: This means that the touch input event attribute is for the distance between the approaching tool and the touchscreen surface. – ABS_MT_ORIENTATION: This means that the touch input event attribute is for the orientation of the touching area or the approaching tool. – ABS_MT_TOOL_TYPE: This means that the touch input event attribute is for the type of approaching tool. – ABS_MT_TRACKING_ID: This means that the touch input event attribute is for the tracking identifier of the approaching tool. The non-negative value for this event attribute is interpreted as an approaching tool that contacts the touchscreen surface. Whenever an approaching tool contacts the touchscreen surface, the tracking identifier for the approaching tool is associated with a slot. If the value for this event attribute is −1, it means that an approaching tool is released from the touchscreen surface and the associated slot becomes unused. – ABS_MT_SLOT: This means that the touch input event attribute is for the slot associated with the tracking identifier of the approaching tool. This event is used for updating the status of each touch contact. The value for this event attribute is a non-negative integer. Whenever an approaching tool contacts the touchscreen surface, the tracking identifier for the approaching tool is associated with a slot. If an approaching tool is released from the touchscreen surface, the associated slot becomes unused.

Table 1 – Elements for touch input event data

Element	Description
	This event code is supported when the pair of the touch device and the input device access module has the capability for tracking multiple touch contacts.
Value	When the event type indicates the touch input event, this element denotes the numerical value for the touch input event attribute. For example, if the event code is ABS_MT_TOOL_TYPE, the value for the touch input event attribute can be expressed with enumerators as follows: – TOOL_TYPE_FINGER: This indicates that the type of approaching tool is a finger. – TOOL_TYPE_PEN: This indicates that the type of approaching tool is a stylus pen.

8.1.4 Screen-mirroring remote control functions

Figure 4 shows a framework for the screen-mirroring remote control service. The service framework consists of two functional activities in the IPTV device and in remote controller device: the screen-mirroring function and the event processing function.

Generally, the first is the IPTV screen that is mirrored onto the display of the handheld device. The second is when the IPTV user touches the screen of the handheld device, the user's touch event input is reflected in the IPTV functions and applications as if the user directly touches the IPTV screen.

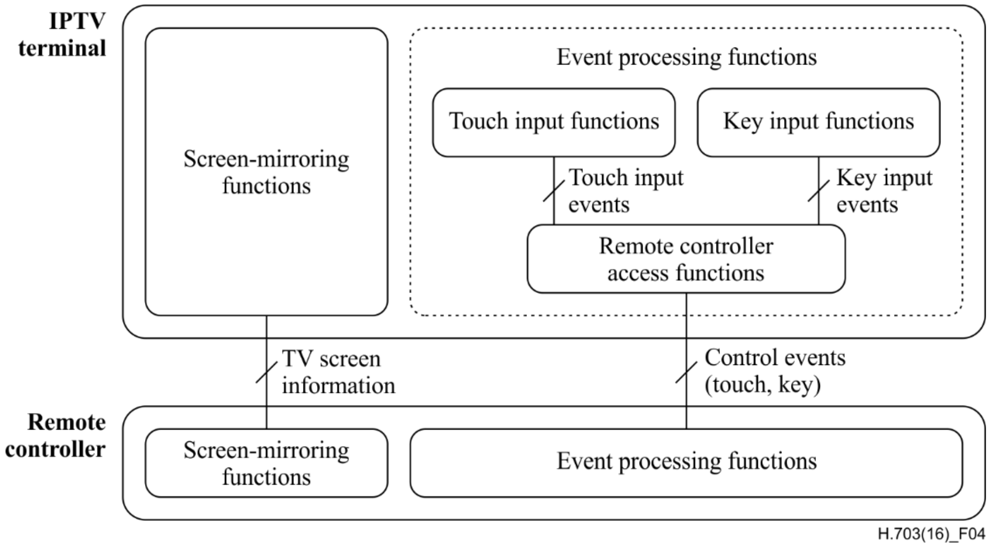

IPTV terminal
Screen-mirroring functions
Event processing functions
Touch input functions
Key input functions
Touch input events
Key input events
Remote controller access functions
TV screen information
Control events (touch, key)
Remote controller
Screen-mirroring functions
Event processing functions
H.703(16)_F04

Figure 4 – Screen-mirroring remote control service framework

As shown in Figure 4, the IPTV terminal has the following functions:

- Screen-mirroring functions: This functional entity is to provide frame-buffer access, IPTV screen capturing and encoding/compressing for image or moving picture. IPTV screen data are accessed and copied from the frame-buffer to another memory by this function. Each frame of the captured IPTV screen data is compressed and stored in an image file format such as JPEG, PNG, etc. Alternatively, the captured IPTV screen data can be encoded as a moving picture stream by a video encoder such as H.264 encoder.
- Event processing functions: This functional entity is for event data analysis and event handling. Event data, which were packaged by the remote controller, are classified for delivery to a right event driver. Generally, an event driver ported on system framework of

IPTV is triggered by an input/output or peripheral device. In the case of this remote control service, the event driver is fed with the event data delivered from the remote controller. By the event driver, the event data are analysed and adjusted to be adapted to the system framework of IPTV. After that, by an event handler of the system framework, the event is applied to IPTV functions and applications.

This functional entity is composed of remote controller access functions, touch input functions, and key input functions. The remote controller access functions parse the event types of the control event packets received from the remote controller. According to the event type, the control event packet is delivered to the touch input functions or the key input functions. The touch input functions receive and handle the control event packets for the touch input events. The key input functions receive and handle the control event packets for the key input events. The input device access module of the touch input functions of the IPTV terminal: parses the touch input event packets, tuning the touch input event data so it can be decoded by the system framework of the IPTV terminal; re-packetizes the tuned touch input event data; and reports the packetized data to the input core module of the touch input functions of the IPTV terminal.

A remote controller device has the following functions:

– Screen-mirroring functions: This functional entity is to decode and render for compressed image or moving picture. The image files, which were stored by the screen-mirroring function of IPTV, are opened and decompressed by an image processing module. These decompressed images are rendered on to the screen of the remote controller. If the IPTV screen is delivered as a moving picture stream, the moving picture stream is decoded and rasterized by a video decoder.

– Event processing functions: This functional entity is for control event aggregation, event classification and event data packaging. Event signals, which were generated by input/output or peripheral devices such as touchscreen panels, are delivered to device drivers. Event signals are translated into event data to be used in operating system ported on the remote controller. Event data are aggregated through application programming interfaces (APIs) supported by the operating system. After that, event data are classified and packaged for delivery to the IPTV terminal.

The control events can be generated by the remote controller as follows:

– Touch input event: If the user touches any point of the touchscreen surface of the remote controller, the touch input event is generated. The touch input event includes the coordinates of the touch point and the state changes of the touchscreen surface.

– Key input event: To support the function of QWERTY and/or legacy remote controller keyboard, a virtual (or software) keyboard can be used. The virtual keyboard is displayed on the touch screen of the remote controller. If the user touches any virtual key, the key input event is generated. The key input event includes the key code and the state changes of the key. Using this virtual keyboard, the user can turn up/down IPTV volume, zap IPTV channels, or call menu items for controlling IPTV applications.

Each control event data delivered from the remote controller to the IPTV terminal is composed of three elements: event type, event code, and value, as shown in Table 2.

Table 2 – Elements for control event data

Element	Description
Event type	This element indicates the type of the control event. The control events can be divided into three types. One is for indicating the touch input event, another is for indicating the key input event, and the other is for separating input events. For example, the event type can be expressed with enumerators as follows: – EV_SYN: This indicates that the input event is the marker or the delimiter for separating input events. – EV_ABS: This indicates that the input event is the touch input event. – EV_KEY: This indicates that the input event is the key input event.
Event code	When the event type indicates the marker or the delimiter, this element denotes the marker to indicate the end of input events or the delimiter to separate sets of touch input events. The usage or expression examples are the same as those described in the previous table of elements for touch input event data.
	When the event type indicates the touch input event, this element denotes the attribute of the touch input event. The usage or expression examples are the same as those described in the previous table of elements for touch input event data.
	When the event type indicates the key input event, this element denotes the key scan code or the virtual-key code. For example, the event code for the key scan code or the virtual-key code can be expressed with enumerators, such as KEY_0, KEY_A, KEY_ENTER, etc.
Value	When the event type indicates the touch input event, this element denotes the numerical value for the touch input event attribute. The usage or expression examples are the same as those described in the previous table of elements for touch input event data.
	When the event type indicates the key input event, this element denotes the status of the key. For example, the value for the status of the key can be expressed with enumerators as follows: – KEY_ACTION_UP: This indicates that the key has been released. – KEY_ACTION_DOWN: This indicates that the key has been pressed down. – KEY_ACTION_MULTIPLE: This indicates that the multiple duplicate key events have occurred in a row. This means that a key stays in a pressed state.

8.2 Speech

The speech function is required to support the interaction between the user and the IPTV system through naturally spoken languages. For the speech function, speech recognition, speech synthesis and natural language processing including language understanding and language generation are the core functionalities.

Speech interface in IPTV systems enables the user to control various IPTV functions, to change channels, to find information on the people or items on IPTV, or to use EPG, conveniently with speech commands or spoken sentences. One example would be an IPTV system equipped with speech interface that enables users to connect to video on demand (VOD) server to find movies of their interest using speech commands or spoken sentences in their own language. With the speech interface, users of IPTV may also ask questions and receive answers conveniently using spoken natural languages from the question answering (QA) server.

The speech interface has the following functions, which are represented in Figure 5:

- Client functions
  - ○ Smart remote controller/smartphone with microphone
    - Transmission of the speech signal streaming to IPTV terminal or speech recognition server.
    - Transmission of information for search and IPTV control functions to IPTV terminal.
- Terminal functions and server functions
  - ○ Common speech interface functions
  - ○ Advanced speech interface functions
    - Commands/Controls: The user can control the device by speech commands.
    - EPG Search: The user can search the EPG to find a programme he/she wants to watch.
    - Information search: The user can search for the information he/she wants to find.
    - Question answering: The user can ask questions through speech interface in natural languages and receive the answers from the QA server with speech synthesis and natural language processing functions.

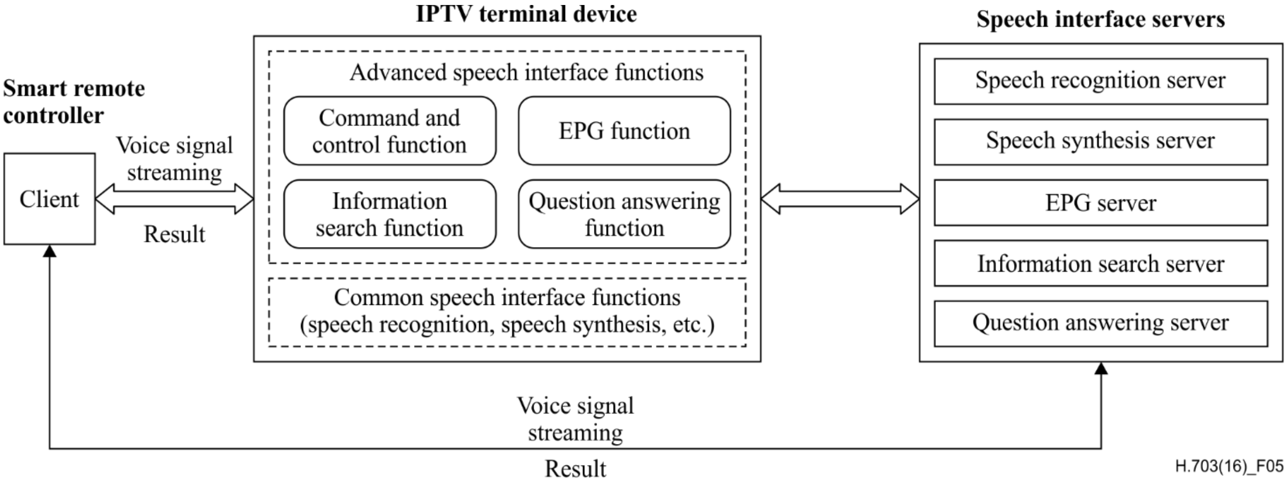

IPTV terminal device
Speech interface servers
Smart remote controller
Advanced speech interface functions
Command and control function
EPG function
Information search function
Question answering function
Common speech interface functions (speech recognition, speech synthesis, etc.)
Client
Voice signal streaming
Result
Speech recognition server
Speech synthesis server
EPG server
Information search server
Question answering server
Voice signal streaming
Result
H.703(16)_F05

Figure 5 – Speech interface

The speech interface functional entity is composed of client functions, terminal functions and server functions.

8.2.1 Client functions

The client functions of speech interface are mainly performed by the smart remote controller equipped with a microphone for speech recognition, which sends the speech signal streaming of the user input to the IPTV terminal device or speech interface servers depending on the type of sentences to be processed. The following are functions of the smart remote controller:

- To record the user's utterance and convert the speech input to speech signal.
- To send the speech signal streaming to IPTV terminal device embedded with speech recognition function to convert the speech for simple sentences into text.
- To send the speech signal streaming to speech recognition server for speech of complex sentences to be converted into text.

- To provide the selection function to choose between the terminal and the server for speech recognition.
- To send command control or search information to IPTV terminal device based on the speech recognition result using infrared signal of smart remote controller.
- To provide the integrated function for search, IPTV control, natural language search for advanced speech interface function as described in Figure 6, which shows the information flow between the client IPTV terminal device and severs.

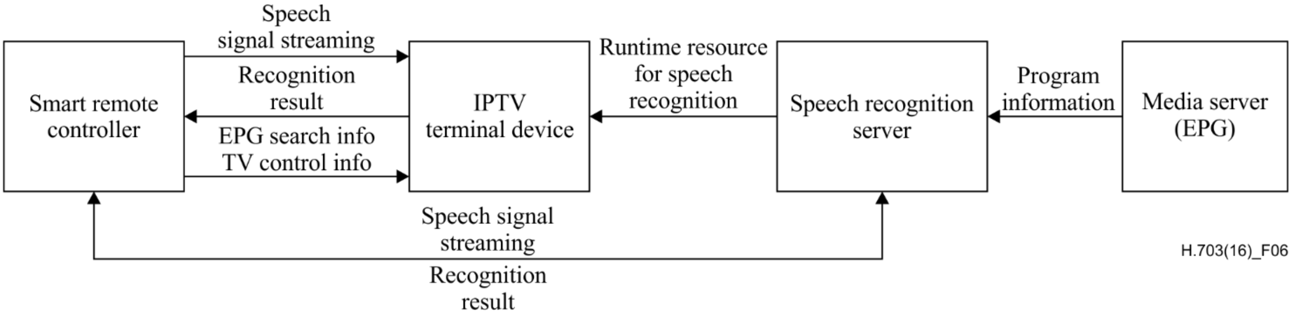


Figure 6 – Function of the client: smart remote controller

8.2.2 Speech interface functions at the terminal and the server

The speech interface functions consist of common speech interface functions and advanced speech interface functions. As described in Figure 6, the speech recognition is performed either at the IPTV terminal device or at the speech recognition server depending on the length and complexity of the spoken sentences to be recognized. The speech recognition server requests and receives programme information from the EPG/media server and composes runtime resources to use for speech recognition. The IPTV terminal device also uses the runtime resource to recognize new words such as TV programme titles or person names that are correctly contained in the EPG information.

8.2.2.1 Common speech interface functions

The common speech interface functional entity is composed of the following functions whose main purpose is to perform speech recognition. Figure 7 shows the functions in the processing order for speech recognition. While the speech recognition function on the terminal device deals with the short sentences only, long and complex sentences are usually processed by the speech recognition functions on the server. Speech synthesis function works the same way with the speech recognition function.

- Input/output processing function

  The Input/output processing function provides the audio input interface function which processes the speech data input from the microphone and the output interface function which recognizes speech data and sends back the recognition result to the user.

- Pre-processing function

  Pre-processing function performs the function of extracting a feature vector, an n-dimensional vector of numerical features that represent speech signal, and removing noise from the nosy speech input. In this module, distributed noises with a high energy level and active noises with low energy level are removed to improve the quality of sound.

- Data communication function

  Data communication function is to send the speech signal and receive the recognized result.

- Speaker adaptation function

  What the speaker adaptation function does is to first collect a small amount of speaker adaptation data from the user and reflect it to the speaker independent acoustic model and then change it to the speaker adapted acoustic model. It is also needed to compose the acoustic model that is best for the user environment and conditions to provide information on various user environments.

– Parallel processing recognition function

Parallel processing recognition function performs speech recognition in parallel mode. The speech recognition function can be based on the deep neural network (DNN)-based signal processing, classification algorithm, and acoustic model. In this functional module, the conventional signal processing algorithm can be replaced with a signal processing DNN to compute feature vectors, the acoustic model parameters are learned by a fused framework of the signal processing and the classification DNNs, and finally a runtime image is built by using the acoustic model and some predefined grammar structure.

– Acoustic model training function

The acoustic model training functional entity produces speaker independent acoustic model from a large size speech database to recognize general user's speech. It also includes the process of collecting speech database and adaptation training for the specific speech environment to make the speaker independent acoustic model optimal in the application environments. The acoustic model database storing a sound model is used to convert the extracted feature vector into a text with a decoder. For the acoustic model training, also known as model parameter learning, a signal processing algorithm for extracting a feature vector from a speech input signal of a time domain can be converted into deep neural network. The model parameter learning method is performed through the deep learning model in which the signal processing DNN and the classification DNN are fused.

– Language processing function

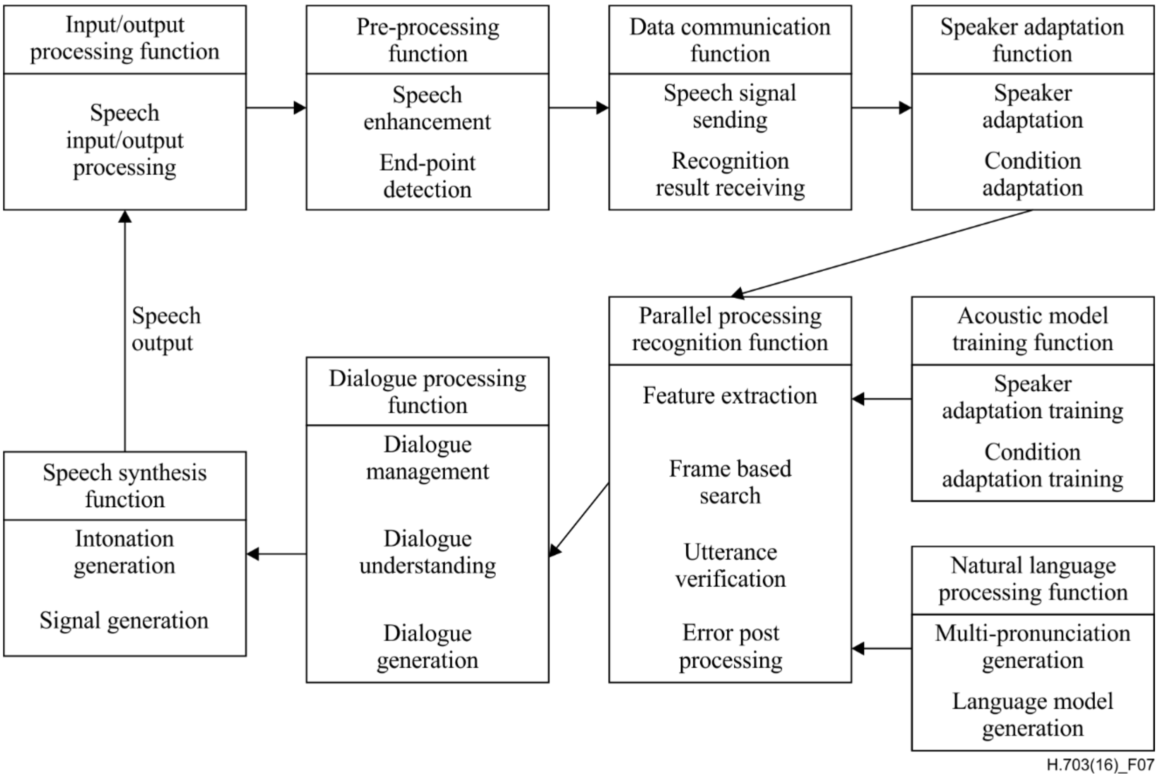
Language processing functional entity performs training of word N-gram language model that defines search network for continuous speech recognition. For language model training, text corpus of the recognition target should be prepared. The text corpus is going through cleaning and refinement process first and then language model training is performed on the clean text corpus. The language model is trained on the limited word list and the utterance dictionary is produced for the target word list. The language model database storing a language model is used to convert the extracted feature vector into a text with a decoder.

– Dialogue processing function

The dialogue processing functional entity performs dialogue act understanding function to express the dialogue intention/act of the utterances by analysing user's utterance strings and understanding their meaning. Dialogue act training function to automatically train dialogue patterns and statistical classification training information based on the domain dialogue corpus annotated with the dialogue acts to understand the participating user's intention. Dialogue management function to analyse n-best dialogue acts and meaning expressions and decide the best dialogue act to generate relevant system dialogue and manage the dialogue situations and dialogue generation function to select the correct system response templet for the user's intention and decide meaningful information values to fill in the slots of the selected templet. Finally, it generates the appropriate system dialogue sentences.

– Speech synthesis function

The speech synthesis functional entity generates speech signals from the text input by text normalization and morphological analysis, intonation target (i.e., pitch contour) generation and speech signal generation using language analysis and intonation prediction information for the input sentences.



Input/output processing function
Speech input/output processing
Pre-processing function
Speech enhancement
End-point detection
Data communication function
Speech signal sending
Recognition result receiving
Speaker adaptation function
Speaker adaptation
Condition adaptation
Speech output
Parallel processing recognition function
Feature extraction
Frame based search
Utterance verification
Error post processing
Acoustic model training function
Speaker adaptation training
Condition adaptation training
Dialogue processing function
Dialogue management
Dialogue understanding
Dialogue generation
Speech synthesis function
Intonation generation
Signal generation
Natural language processing function
Multi-pronunciation generation
Language model generation
H.703(16)_F07

Figure 7 – The common speech interface functions

8.2.2.2 Advanced speech interface functions

The advanced speech interface functions are composed of the following functions: command/control, EPG search, information search, and question answering.

– Command/control function

The command/control function enables the user to control IPTV by speech commands. After the speech commands are recognized by the common speech function, they are processed by command/control function to be mapped with the related commands/control operations at the IPTV terminal device. Commands given in natural languages, from simple to complex ones, are mapped into the operations through the natural language and dialog processing functions which are provided by the common speech interface function.

– EPG search function

The EPG search function enables the user to search the EPG to find content such as a programme he/she wants to watch. Keywords (from simple search to complex natural language ones) are processed through the natural language and dialog processing functions that are provided by the common speech interface function. The search keywords are composed to a query format and sent to the EPG server to find matched information. When the query results of the searched contents return from the EPG server, they are presented to the user screen at the terminal device in the relevant format such as natural sentences or some simple words after natural language generation and speech synthesis. EPG/Media server plays a role of providing contents and maintains database (DB) that contains information on TV programmes and media. When the queries on the contents of the EPG/Media server arrive from the IPTV terminal function, the server function processes the queries and finds the relevant information that matches with the queries in the DB. Then the output contents from the queries are sent back to the EPG function on IPTV terminal device.

– Information search function

The information search function enables the user to search for some wanted information. As in the EPG search function, keywords or long sentences are changed to query formats and sent to the information search server to find the target information in the information database. The results are presented to the user as a text or speech after natural language generation and speech synthesis.

– Information search server provides contents and maintains DBs that contain information on various topics, such as weather or news material. When the queries on the contents of the information server arrive from the IPTV terminal function, the server function processes the queries and finds the relevant information that matches with the queries in the databases. Then the outputs from the queries are sent back to the information search function on IPTV terminal device.

– Question answering function

Question answering function is an advanced function to provide answers for the user's question in a natural language. More systems in the future are expected to be equipped with QA functions for advanced user experience. The user's question in natural spoken language is recognized by the speech recognition function and sent to the server in a text form to receive the answer. The major question answering function is performed at the QA server and the answer to the question is transmitted back to the QA terminal function. The answer in the text form is presented as a speech form after processed by speech synthesis function.

Figure 8 has a general QA architecture, which takes the user question as the input and produces answer as the output after following each step of the QA process.

Question answering functions are basically composed of question analysis, natural language processing, candidate answer generation, answer inference and answer generation functional block as follows:

○ question analysis functional block: supports question analysis process on the QA server;

○ natural language processing functional block: supports natural language processing on the QA server;

○ candidate answer generation functional block: supports information search from the various knowledge databases to generate candidate answers for the question; public information database is also searched when the user's query does not fall within a range of the expected queries;

○ answer inference functional block: supports answer inference based on multiple information sources and confidence calculation to extract short answers from the candidates in relation to the user's query;

○ answer generation functional block: supports best answer generation on the terminal.

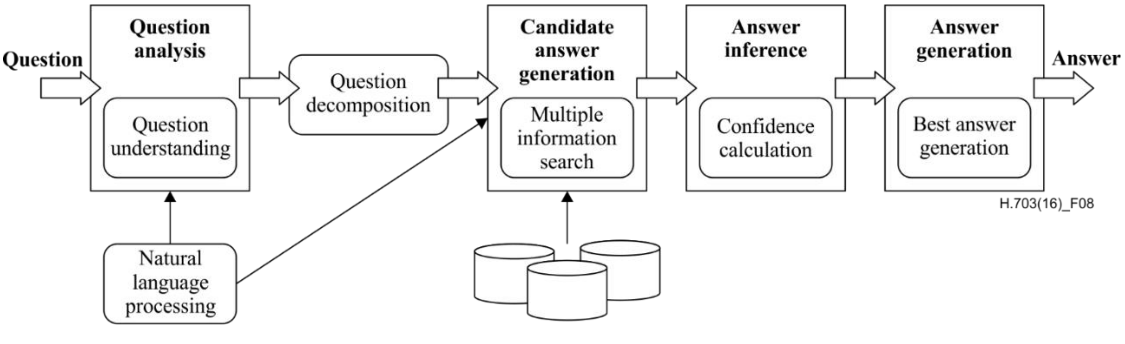


Figure 8 – Question answering function

Appendix I

Examples of functional flow for mirroring remote controller

(This appendix does not form an integral part of this Recommendation.)

This appendix shows some example of procedure of functional flow for understanding of mirroring remote controller.

The screen-mirroring functions between the IPTV and the remote controller are performed through the following steps. Figure I.1 shows the functional flow of the screen-mirroring procedure.

- The IPTV accesses frame-buffer and captures IPTV screen.
- The IPTV compresses and stores each frame of the captured IPTV screen data as an image file, or the IPTV encodes the captured IPTV screen data as a moving picture stream.
- The IPTV transmits image files for frames of the captured IPTV screen to the remote controller, or the IPTV transmits the moving picture stream of the encoded IPTV screen.
- The remote controller decompresses image files of the captured IPTV screen, or the remote controller decodes the moving picture stream of the encoded IPTV screen.
- The remote controller renders and displays image files on the remote controller screen, or the remote controller plays back the moving picture stream.

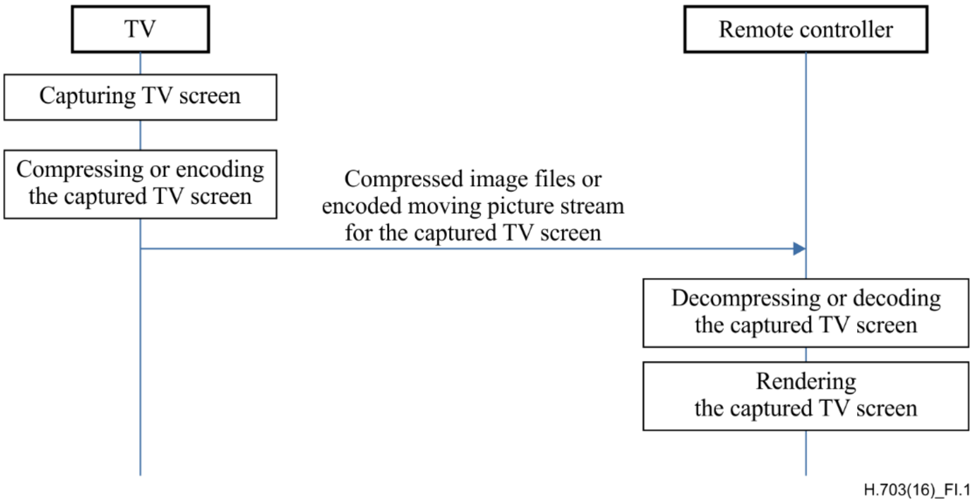

TV
Remote controller
Capturing TV screen
Compressing or encoding the captured TV screen
Compressed image files or encoded moving picture stream for the captured TV screen
Decompressing or decoding the captured TV screen
Rendering the captured TV screen
H.703(16)_FI.1

Figure I.1 – Screen-mirroring procedure

The control event processing functions, for touch input event, are performed through the following steps. Figure I.2 shows the functional flow of the control event processing procedure when the input event is the touch input event.

- The user touches any portion of the image displayed on the remote controller screen.
- The remote controller perceives the user's touch input event signal generated by touchscreen panel.
- The remote controller translates the touch input event signal into a structured touch input event data that an operating system can handle.
- The remote controller classifies the event type and packages the touch input event data including the event type identifier that indicates the event is a touch input event.

- The remote controller transmits the packaged touch input event data.
- The IPTV classifies the packaged touch input event data and, according to the event type identifier, delivers the touch input event data to a touchscreen driver.
- The IPTV analyses and adjusts the touch input event data so that an operating system can handle it.
- The IPTV applies the touch input event to IPTV functions and applications.

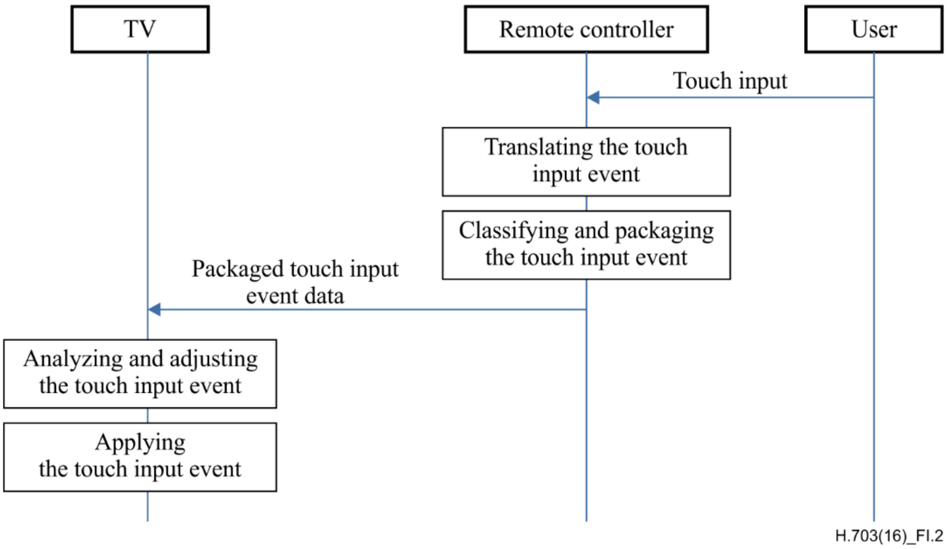

TV
Remote controller
User
Touch input
Translating the touch input event
Classifying and packaging the touch input event
Packaged touch input event data
Analyzing and adjusting the touch input event
Applying the touch input event
H.703(16)_FI.2

Figure I.2 – Event processing procedure for the touch input event

Finally, if the input event type is the key input event, some procedures are changed as follows: Figure I.3 shows the functional flow of the control event processing procedure when the input event is the key input event.

- The user presses a hardware button of the remote controller or a virtual, or software, button displayed on the remote controller screen to trigger or active a virtual, or software, keyboard.
- The remote controller displays the virtual keyboard on the remote controller screen.
- The user touches, or presses, any virtual key of the virtual keyboard.
- The remote controller perceives the user's touch input event signal generated by touchscreen panel.
- The remote controller translates the touch input event signal into a structured touch input event data that an operating system can handle.
- The remote controller analyses the touch input event data and generates the key input event data corresponding to the pressed virtual key of the virtual keyboard.
- The remote controller classifies the event type and packages the key input event data including the event type identifier that indicates the event is a key input event.
- The remote controller transmits the packaged key input event data.
- The IPTV classifies the packaged key input event data and, according to the event type identifier, delivers the key input event data to a keyboard driver.
- The IPTV analyses and adjusts the key input event data so that an operating system can handle it.
- The IPTV applies the key input event to IPTV functions and applications.

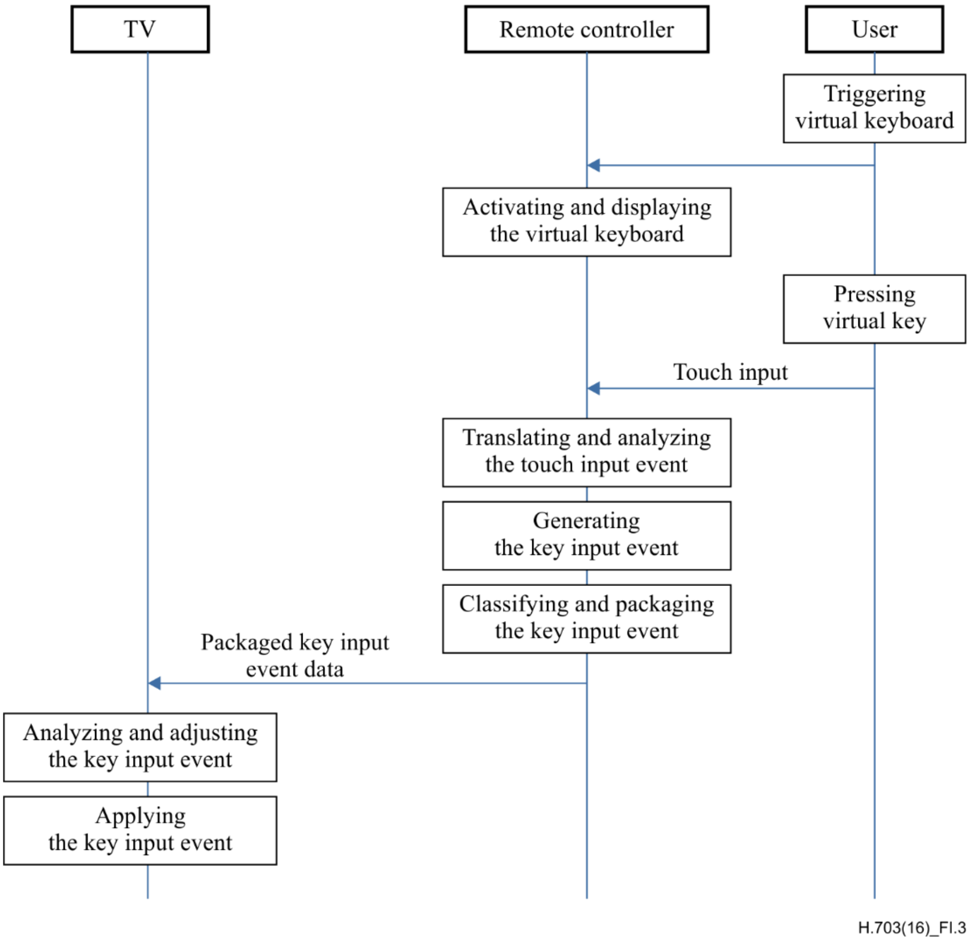

TV
Remote controller
User
Triggering virtual keyboard
Activating and displaying the virtual keyboard
Pressing virtual key
Touch input
Translating and analyzing the touch input event
Generating the key input event
Classifying and packaging the key input event
Packaged key input event data
Analyzing and adjusting the key input event
Applying the key input event
H.703(16)_FI.3

Figure I.3 – Event processing procedure for the key input event

Appendix II

Examples for generating touch/key input event data

(This appendix does not form an integral part of this Recommendation.)

This appendix shows some examples for generating touch input event data.

Each touch input event attribute is reported as the touch input event individually. For example, the X coordinate is reported as a touch input event and the Y coordinate is also reported as another touch input event.

For each touch point, a set of touch input events is reported. For example, touch input events for the X coordinate, the Y coordinate, and the type of approaching tool can be sequentially reported as one set. In this case, the tracking identifier of the approaching tool can be used to distinguish each set of touch input events. If the tracking mechanism for the approaching tool is not supported, another kind of input event is required to distinguish each set of touch input events.

The touch input event data are grouped as a packet to be delivered between functional blocks. Moreover, the touch input event data can also be reported as a packet to applications.

Touch device can be categorized into two types, depending on whether it has the capability for tracking multiple touch contacts or not. If the touch device does not have the capability, applications may have to implement any mechanism for tracking multiple touch contacts. So, the touch device and the input device access module are recommended to have the capability for tracking multiple touch contacts.

The touch input event data can be generated somewhat differently according to whether the touch device has the capability for tracking multiple touch contacts or not.

II.1 Case I: Multi-touch tracking not supported

If the touch device does not support the mechanism for tracking multiple touch contacts and touch input events are caused by two different approaching tools, a group of touch input event data can be generated sequentially as follows (Figure II.1 illustrates the sequence of events):

1. {EV_ABS, ABS_MT_COR_X, 800}
2. {EV_ABS, ABS_MT_COR_Y, 400}
3. {EV_ABS, ABS_TOUCH, 1}
4. {EV_SYN, SYN_MT_REPORT}
5. {EV_ABS, ABS_MT_COR_X, 900}
6. {EV_ABS, ABS_MT_COR_Y, 500}
7. {EV_ABS, ABS_TOUCH, 1}
8. {EV_SYN, SYN_MT_REPORT}
9. {EV_SYN, SYN_REPORT}

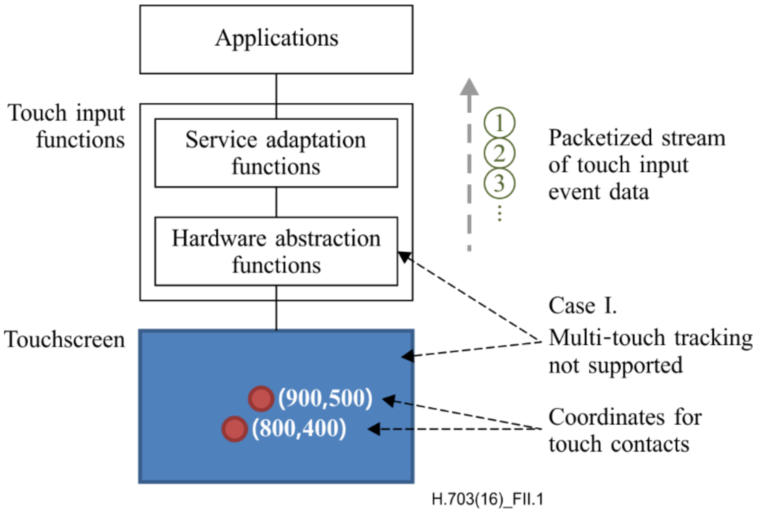

Applications
Touch input functions
Service adaptation functions
Hardware abstraction functions
Touchscreen
(900,500)
(800,400)
1
2
3
Packetized stream of touch input event data
Case I.
Multi-touch tracking not supported
Coordinates for touch contacts
H.703(16)_FII.1

Figure II.1 – Multi-touch tracking not supported

The details of the data format for the touch input event data is described in the table of elements for touch input event data.

In the above state, if the approaching tool for the second touch contact is released from the touchscreen surface, a set of touch input event data can be generated sequentially as follows:

1. {EV_ABS, ABS_MT_COR_X, 900}
2. {EV_ABS, ABS_MT_COR_Y, 500}
3. {EV_ABS, ABS_TOUCH, 0}
4. {EV_SYN, SYN_MT_REPORT}
5. {EV_SYN, SYN_REPORT}

II.2 Case II: Multi-touch tracking supported

If the touch device supports the mechanism for tracking multiple touch contacts and touch input events are caused by two different approaching tools, a group of touch input event data can be generated sequentially as follows (Figure II.2 illustrates the sequence of events):

1. {EV_ABS, ABS_MT_SLOT, 0}
2. {EV_ABS, ABS_MT_TRACKING_ID, 1}
3. {EV_ABS, ABS_MT_COR_X, 800}
4. {EV_ABS, ABS_MT_COR_Y, 400}
5. {EV_SYN, SYN_MT_REPORT}
6. {EV_ABS, ABS_MT_SLOT, 1}
7. {EV_ABS, ABS_MT_TRACKING_ID, 2}
8. {EV_ABS, ABS_MT_COR_X, 900}
9. {EV_ABS, ABS_MT_COR_Y, 500}
10. {EV_SYN, SYN_MT_REPORT}
11. {EV_SYN, SYN_REPORT}

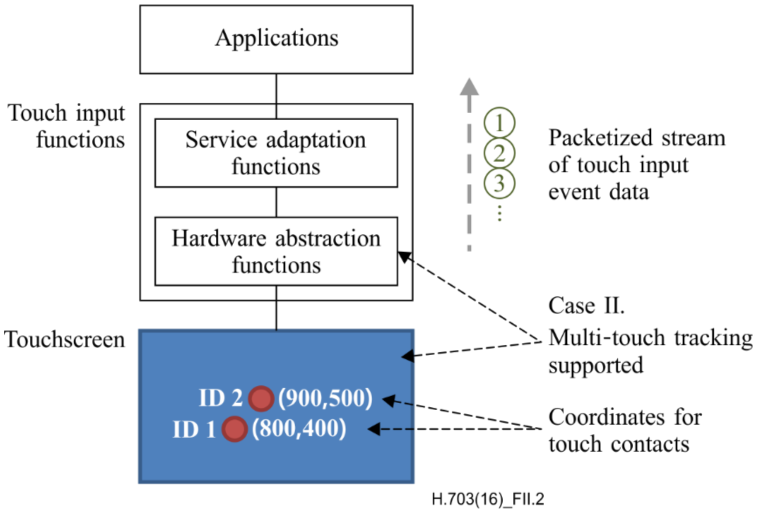

Applications
Touch input functions
Service adaptation functions
Hardware abstraction functions
Touchscreen
ID 2 (900,500)
ID 1 (800,400)
1
2
3
Packetized stream of touch input event data
Case II.
Multi-touch tracking supported
Coordinates for touch contacts
H.703(16)_FII.2

Figure II.2 – Multi-touch tracking supported

The details of the data format for the touch input event data is described in the table of elements for touch input event data.

In the above state, if the approaching tool for the tracking ID 2 is released from the touchscreen surface, a set of touch input event data can be generated sequentially as follows:

1. {EV_ABS, ABS_MT_SLOT, 1}
2. {EV_ABS, ABS_MT_TRACKING_ID, -1}
3. {EV_SYN, SYN_MT_REPORT}
4. {EV_SYN, SYN_REPORT}

Slot 1 associated with the tracking identifier of the approaching tool is not used. In this state, if another approaching tool with the tracking ID 3 contacts the touchscreen surface, a set of touch input event data can be generated sequentially as follows:

1. {EV_ABS, ABS_MT_SLOT, 1}
2. {EV_ABS, ABS_MT_TRACKING_ID, 3}
3. {EV_ABS, ABS_MT_COR_X, 1000}
4. {EV_ABS, ABS_MT_COR_Y, 600}
5. {EV_SYN, SYN_MT_REPORT}
6. {EV_SYN, SYN_REPORT}

II.3 Case III: Example of key button "A" input

The key input event data can be generated sequentially as follows, in case of user's push and release for "A" key (see Figure II.3).

1. {EV_KEY, KEY_A, KEY_ACTION_DOWN}
2. {EV_SYN, SYN_MT_REPORT}
3. {EV_KEY, KEY_A, KEY_ACTION_UP}
4. {EV_SYN, SYN_MT_REPORT}
5. {EV_SYN, SYN_REPORT}

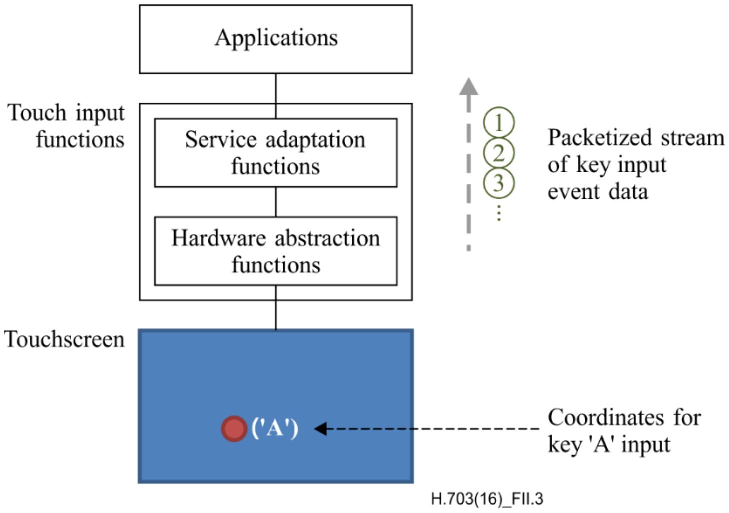

Applications
Touch input functions
Service adaptation functions
Hardware abstraction functions
Touchscreen
('A')
1
2
3
Packetized stream of key input event data
Coordinates for key 'A' input
H.703(16)_FII.3

Figure II.3 – An example of the key input event data generation

Appendix III

Summary of relevant standardization activity

(This appendix does not form an integral part of this Recommendation.)

In this appendix, other SDO's relevant activity is briefly summarized on relevant specification title and key scope, especially on W3C, ITU-T and ISO/IEC JTC1 SC35.

I.1 W3C

Specification for touch events

The touch events specification [b-W3C TE] defines a set of low-level events that represent one or more points of contact with a touch-sensitive surface, and changes of those points with respect to the surface and any document object model (DOM) elements displayed upon it (e.g., for touchscreen) or associated with it (e.g., for drawing tablets without displays).

Pointer events specification

[b-W3C PE] defines events and related interfaces for handling hardware agnostic pointer input from devices like a mouse, pen, or touchscreen.

Multimodal interaction framework

W3C Note [b-W3C MMIF] introduces the W3C Multimodal Interaction Framework, and identifies the major components for multimodal systems. Refer also to http://www.w3.org/2002/mmi/.

I.2 ISO/IEC JTC1 SC35

ISO/IEC JTC 1 SC35 is entitled *User Interfaces* and targets standardization in the field of user-system interfaces in information and communication technology environments, and support for these interfaces to serve all users, including persons with disabilities or other specific needs, with a priority of meeting the JTC 1 requirements for cultural and linguistic adaptability.

It developed several standards related with Gesture Interface, in particular

- ISO/IEC 30113-1 (2015), Information technology – User interface – Gesture-based interfaces across devices and methods – Part 1: Framework <http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=53233>
- ISO/IEC 30113-11 (ongoing), Information technology – User interface – Gesture-based interfaces across devices and methods – Part 11: Single-point gestures for common system actions <http://www.iso.org/iso/catalogue_detail.htm?csnumber=62863>

ISO/IEC JTC1 SC35 WG5 "Cultural and linguistic adaptability" developed the following standard:

- ISO/IEC PRF 30122-1 (2016), Information technology – User interfaces – Voice commands – Part 1: Framework and general guidance. <http://www.iso.org/iso/catalogue_detail.htm?csnumber=53242>

In this document, various voice commands are defined to be used for speech recognition to control various ICT devices as shown in Figure 1 of that document.

I.3 ITU-T

The speech-related standards of interest to this Recommendation are:

- ITU-T F.745 *Functional requirements for network-based speech to speech translation services*. In particular, see Figure 2 "Definition of network-based S2ST".
- ITU-T H.625, *Architectural requirements for network-based speech translation services*.

Bibliography

[b-ITU-T H.264] Recommendation ITU-T H.264 (2016), *Advanced video coding for generic audiovisual services*.

[b-ITU-T H.625] Recommendation ITU-T H.625 (2010), *Architecture for network-based speech-to-speech translation services*.

[b-ITU-T H.780] Recommendation ITU-T H.780 (2012), *Digital signage: Service requirements and IPTV-based architecture*.

[b-ITU-T Y.101] Recommendation ITU-T Y.101 (2000), *Global Information Infrastructure terminology: Terms and definitions*.

[b-ITU-T Y.110] Recommendation ITU-T Y.110 (1998), *Global Information Infrastructure principles and framework architecture*.

[b-W3C MMIF] W3C Note (2003), *W3C Multimodal Interaction Framework.*
<https://www.w3.org/TR/mmi-framework/>

[b-W3C PE] W3C Recommendation (2015), *Pointer Events*.
<https://www.w3.org/TR/pointerevents/>

[b-W3C WebArch] W3C Recommendation (2004) *Architecture of the World Wide Web, Volume One*
<https://www.w3.org/TR/webarch/>

[b-W3C TE] W3C Recommendation (2013), *Touch Events*.
<https://www.w3.org/TR/touch-events/>

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	General tariff principles
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	**Audiovisual and multimedia systems**
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Environment and ICTs, climate change, e-waste, energy efficiency; construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Terminals and subjective and objective assessment methods
Series Q	Switching and signalling
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks, open system communications and security
Series Y	Global information infrastructure, Internet protocol aspects and next-generation networks, Internet of Things and smart cities
Series Z	Languages and general software aspects for telecommunication systems

Printed in Switzerland
Geneva, 2016