

A DIGITAL TELEVISION NAVIGATOR

CHENGYUAN PENG AND PETRI VUORIMAA

Telecommunications Software and Multimedia Laboratory,
Helsinki University of Technology,
P.O. Box 5400, FI-02015 HUT, Finland.
Email: pcy@tml.hut.fi and petri.vuorimaa@hut.fi

ABSTRACT

Digital television is a new, interesting, and rich platform for developing next generation multimedia services. Navigator is the most important Multimedia service of digital television. It acts as the main index of all the services available in set-top box. In this paper, we described the definition of a Navigator, its functionality, and its information source such as transport stream (TS) and Service Information (SI). The processes of developing a Navigator in Java are presented. A software model including navigation model and state model of a remote control is introduced. The conclusions part contains development issues and further research topics.

Keywords: Navigator, digital television, SI, multimedia service, TS, Java.

1 INTRODUCTION

The Multimedia Home Platform (MHP) is specified by the Digital Video Broadcasting (DVB). It is a common platform for user to transparently access a range of multimedia services [1]. It includes software architecture and hardware devices. Its hardware devices consist of the home terminal (set-top box, TV, PC), its peripherals, and the in-home digital network [1].

The DVB-MHP solution covers the whole set of technologies that are necessary to implement digital interactive multimedia services in the home. These technologies include protocols, Application Programming Interfaces (APIs), interfaces and recommendations [1]. The multimedia services are categorized as enhanced broadcasting, interactive broadcasting, and the Internet access services in the DVB-MHP specifications.

Navigator is a resident application in set-top box for enhanced broadcasting (without a return channel). It is a part of the system software [2]. The Navigator is typically provided by the set-top box manufacturer. Its functionality and look & feel are also determined by the manufacturer.

The viewers of tomorrow will receive a multitude of channels with their set-top box. They need help. The Navigator is the guide of the viewer to select services and applications, initiate interoperable applications, boot loading, and store user profiles.

The DVB-SI broadcast specifications provide the information structures necessary for the development of a basic Navigator [1]. The Navigator uses the DVB-SI data supplied by the network operator or the broadcaster.

The DVB-SI data is multiplexed together with MPEG-2 compressed video and audio data streams to form TSs. The multiplexed TS must be demultiplexed in the set-top box before the Navigator can use it. The DVB-SI adds information that enables set-top box to tune automatically to particular services and allows services to be grouped into categories with relevant schedule information.

There is a button labeled “Navigator” in the remote control. The viewer can activate the Navigator at any time with a remote control. When the viewer presses the “Navigator” button, the main graphical user interface will be displayed, through which the viewer then can navigate the services provided by set-top box manufacturer, broadcaster, and service provider.

2 TRANSPORT STREAM AND DVB-SI

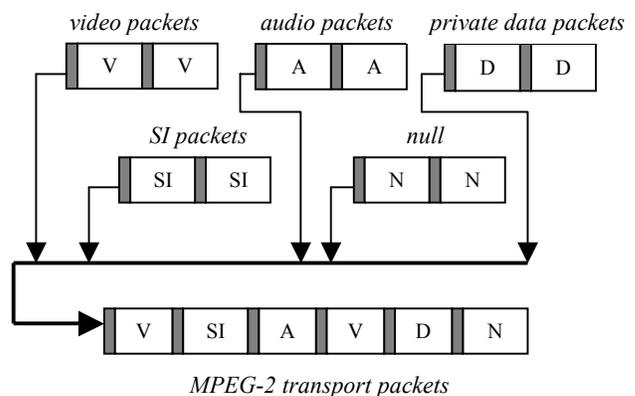


Fig. 1. MPEG-2 Transport Stream.

TS packets are derived from elementary streams, service information, private data, and conditional access control (cf. Fig. 1). All the packetized elementary streams, including video and audio, that are multiplexed together are converted into transport packets. Similarly, the DVB-SI data, private data, and conditional access control data are converted into transport packets. The resulting transport packets are then output sequentially to form an

MPEG-2 transport packet stream. Null transport packets are used to soak-up any spare multiplex capacity, which can also appear in the TS [3].

MPEG-2 allows a separate Service Information system to be used to complement its Program Specific Information (PSI). The DVB-SI with accompanied DVB signals can be used by set-top box. Therefore the viewer may navigate through the array of offered services.

The DVB-SI is based on four service information tables (PSI) [4], plus a series of optional tables. Each table contains ordinary fields and descriptors fields outlining the characteristics of the services/events being described. Each descriptor can have nested descriptors.

The PSI data is structured in the following four tables: *Program Association Table* (PAT) indicates the location (i.e., the Packet Identifier (PID) values of the TS packets) of the corresponding Program Map Table (PMT). It also gives the location of the Network Information Table (NIT) [5]. *Conditional Access Table* (CAT) provides information on the Conditional Access (CA) system used in the multiplex [5]. *Program Map Table* (PMT) identifies and indicates the locations of the streams that make up each service, and the location of the Program Clock Reference fields for a service [5]. *Network Information Table* (NIT) contains information relating to the physical organization of the multiplexes carried via a given network and the relevant tuning information [5]. It groups services together belonging to a particular network provider. It is also used to signal a change in the tuning information.

In addition to the PSI, the additional data is needed to provide identification of services and events carried by different multiplexes, and even on other networks. This data is structured in the following tables:

Bouquet Association Table (BAT) provides information regarding bouquets. It provides a list of services for each bouquet as well as the name of the bouquet [5]. A particular service can belong to one or more bouquets. *Service Description Table* (SDT) contains data describing the services in the system, e.g., the names of services, the service provider, etc [5]. It lists the names and other parameters associated with each service in a multiplex. *Event Information Table* (EIT) (including present/following and schedule tables) contains data concerning events such as event name, start time, duration, etc [5]. The use of different descriptors in EIT allows the transmission of different kinds of event information, e.g., different service types. It is also used to transmit the information relating to all the events that occur or will occur in the MPEG multiplex.

The DVB-SI tables are transmitted in sections. Each table must be segmented into one or more sections with the same *table_id* as the table before being inserted into TS packets. A section is in fact a syntactic structure. Sections may be variable in length. The sections within each table are limited to 1024 bytes in length, except for

sections within EIT, which are limited to 4096 bytes [5]. Each section is uniquely identified by the combination of the five fields in a section, such as *table_id*, *section_number*, etc.

More detailed fields and descriptors of a section for different table can be found in [5].

3 SENDING TS PACKETS

We simulated the DVB-SI transmission through the Internet by using User Datagram Protocol (UDP) as a test. At the server side, the Datagram packets carrying DVB-SI tables were sent recursively. These Datagram packets were decoded at the client side where the Navigator was running. When the DVB-SI data is changed on the server, the client is signaled, and the changed part is renewed in the Navigator (i.e., it reads the tables again).

For SI specified in ETS [5] the minimum time interval between the arrival of the last byte of a section and the first byte of the next transmitted section, which belongs to the same table should be 25 milliseconds. This limit applies for transport streams with a total data rate of up to 100Mbit/s. The method of sending UDP Datagrams can serve the purpose.

Applications that communicate via UDP Datagrams send and receive completely independent packets of information [6]. These clients and servers do not have and do not need a dedicated point-to-point channel. A Datagram is an independent, self-contained message sent over the network [6]. The delivery of Datagrams to their destinations is not guaranteed. Nor is the order of their arrival.

A TS packet is always 188 bytes long [3]. It comprises a 4-byte header followed by a payload (or an adaptation field, or both). The payload is 184 bytes in length (cf. Fig. 2).

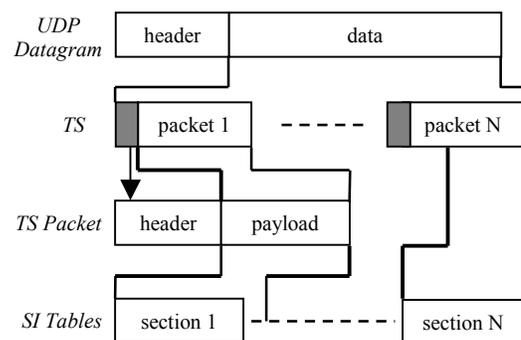


Fig. 2. A UDP Datagram Packet.

A TS packet begins with 4-byte header [3]. The first byte is a sync-byte having the value 0x47 (hex), which is used for the decoder to identify the start of each new transport packet. In the following two bytes, the first bit is *transport_error_indicator*, the second bit is *payload_unit_start_indicator*, the third bit indicates *transport_priority*, and the last 13-bit is PID. The first two

bits of the fourth byte are used for *transport_scrambling_control*. The second two bits are used for *adaptation_control*. The last 4-bit is *continuity_count* field.

Fig. 2 illustrates the method of inserting the DVB-SI tables into a Datagram packet.

4 NAVIGATOR FUNCTIONALITY

This section gives a description of the main functions of the Navigator implemented in the Future TV project. Also a detailed description of the fields of the tables, which contributed to the functionality of the Navigator, is presented.

There are five main functions in the Navigator (cf. Fig. 3): *Channel Info Bar*, *Channel Guide*, *Program Guide* (i.e., EPG), *Services*, and *Configuration*. All these views can be accessed via the main user interface. Pushing the “Navigator” button on the remote control displays the main menu of the Navigator. Next, the viewer presses the “up” or “down” arrow and “OK” button to select a particular function of the Navigator (cf. Fig. 3). A conceptual panel of the PC remote control is shown in Fig. 9.



Fig. 3. The Main Menu of the Navigator.

Channel Info Bar

The purpose of the *Channel Info Bar* is to show the information of the current program/event immediately. Thus the viewer does not need to browse the newspaper or TextTV. When the viewer presses a “TV” button, the *Channel Info Bar* is displayed on the screen (cf. Fig. 4).

Pressing “TV” again hides the display. The *Channel Info Bar* displays current channel and program/event information (i.e., service/channel name, channel logo, event name, detailed information about the current event, and the date and time of the set-top box).

The information comes from the EIT present/following table (*table_id* = 0x4E) and SDT table (*table_id* = 0x42). Two descriptors are used in the *Channel Info Bar* (i.e., *Short Event descriptor* and *Extended Event Descriptor*). The *Short Event Descriptor* transmits the name and a short text description for the event. The language code indicates in which language the text was written. The

Extended Event Descriptor provides a detailed text description of the event.



Fig. 4. The Channel Info Bar.

In the *Channel Info Bar*, the channel/service name is required. The EIT table does not include *Service Descriptor* field, but it can be uniquely identified by *service_id* field, *event_id*, and *transport_stream_id* field (cf. Fig. 7).

Channel Guide

The *Channel Guide* is used to help the viewer to browse the present and following program/event of all the channels/services available in the broadcasting network.



Fig. 5. The Channel Guide User Interface.

Fig. 5 shows the *Channel Guide* user interface. The current date and real time clock is displayed in the upper right corner. The right panel shows the present (upper part) and following (lower part) program information about current selected channel. It has the same data information carried in the EIT present/following table as the *Channel Info Bar*. The left upper part is a MPEG-1 video stream, which was used to show the selected channel video. The left lower panel shows channel information (i.e., channel logos and channel names).

The “up” or “down” arrow and “OK” button on the remote control can be used to select a channel. When “OK” button is pressed, the video and text on the right side change immediately. The channel information is read

from the SDT table. Its *Service Descriptor* field contains service name and service provider name.

The “Types” button at the bottom of the Fig.5 can be activated with the “yellow” button on the remote control (cf. Fig. 5). When the “yellow” button is pressed, a submenu will popup. It displays all the available channel types of the broadcasting network (i.e., the classification of the channel, e.g., sports, news, etc.). The type information is in the BAT table. Its *Bouquet Name Descriptor* field transmits the name of the type. The *Service List Descriptor* contains a list of services and service types, e.g., digital television service, FM radio service, NVOD, Teletext, mosaic service, etc.

The viewer can select or deselect part or all of the types by switching “Space” button on the remote control and confirm by pressing “OK” button. When the viewer browses those unsubscribed channels, only text information will be displayed, the video will not be shown.

The “Channels” button corresponds to “green” button on the remote control. This submenu lists all the channel names. When pressing “left” or “right” button, one of the two submenus can also be activated (cf. Fig. 10). The viewer can return to watch the current selected channel by pressing “TV” button.

Program Guide

The *Program Guide* is a simple Electronic Program Guide (EPG). It has seven days program schedule information for the channels. Fig. 6 shows the *Program Guide* user interface.



Fig. 6. The Program Guide User Interface.

The right upper part displays one day’s program schedule of the selected channel. The “up” or “down” arrows can be used to browse the program schedule information. At the same time, the detailed text information of the program is displayed on the lower part of the Fig. 6. The video will not change, thus the viewer can still watch the current program when browsing the schedule information of the programs. When the viewer presses the “TV” button, the Navigator returns to full screen video from the *Program Guide*.

The “Date” button (cf. Fig. 6) corresponds to “Red” button on the remote control. This submenu is used to select the date. The “Channels” and “Types” submenu are the same as the *Channel Guide*’s “Channels” and “Types” submenus. The selection method is also the same.

Services

The *Services* provides a list of multimedia services, e.g., www, Email, Home-shopping, and Screen Information service. In our previous work, we have already implemented some of them [7].

Configuration

The *Configuration* function is used to initialize and set up the set-top box. Initialization includes caching network information in the NIT table from transmission and downloading drivers, operating system, and applications from transmission or modem [2]. Setting contains setting up network connections, look & feel of the user interface, and user preferences, such as aspect ratio, subtitle language, etc.

The stored information includes the name of a physical network (in *Network Name Descriptor* field), the information about transport streams of the delivery system, a list of all services carried in all the transport streams (from *Service List Descriptor* field), and the physical parameters for each transport stream in the network (cf. Fig. 7). These physical parameters come from delivery system descriptors (i.e., *Satellite, Cable, and Terrestrial Delivery System Descriptor*). They are frequency, orbital-position, polarization, modulation, symbol rate, etc.

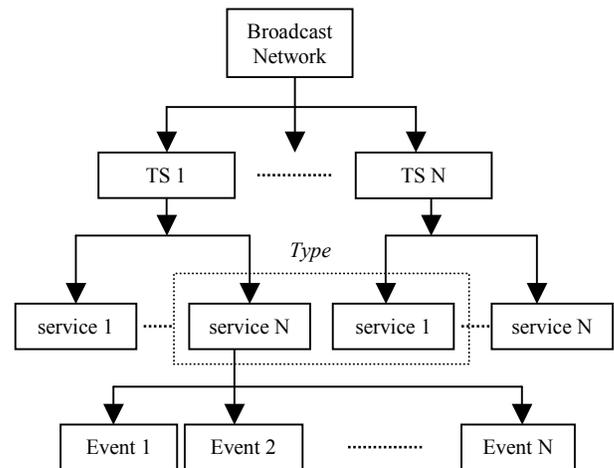


Fig. 7. Service Information Organization.

When using the tuning information, a service can be found through the path *original_network_id*->*transport_stream_id* -> *service_id* (cf. Fig. 7).

After initiation or reconfiguration of the Navigator, all the services and their bouquets in the network are available. The Navigator also provides means for the selection of bouquets and services.

5 IMPLEMENTATION

The main data structures of the Navigator were simulation of the Service Information Organization shown in Fig. 7. It is a satellite broadcasting network (in practice, there may be several other networks) which transmits many transport streams. Each transport stream carries many services/channels provided by broadcaster or service provider. Each service has a series of events/programs. The services with the same type can cross different transport streams.

Initialization of the Navigator

The Navigator initialization is done when the set-top box is initialized, or a new version of the Navigator is downloaded. There is no need for the viewer to initiate it often, because searching the network takes quite a lot of time.

The initialization includes scanning all accessible networks (cf. Fig. 7), storing all the services and types transmitted in multiplexed transport streams. Finally, the Navigator initializes the components of Java user interface (i.e., the look & feel).

Navigation Process Model

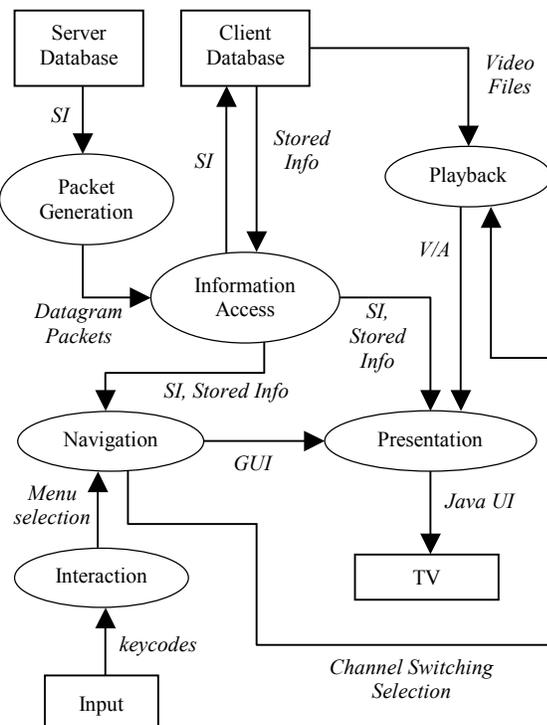


Fig. 8. Data Flow Diagram of the Navigator.

Fig. 8 illustrates the first level data flow diagram of the Navigator. It was used to model information flow of the navigation process.

The DVB-SI data including the NIT, BAT, SDT, EIT present/following tables, and EPG text files were stored in the server database, which was in server PC. The process

of *Packet Generation* was encoding and sending UDP Datagram packets to the client all the time.

The client database stores data containing the binary file for channel tuning, EPG text files, the icons of channel logos, video files, and image files for keeping look & feel of the Navigator user interface [7].

Information Access process included decoding Datagram packets coming from the *Packet Generation* process, getting data from the local client database, and sending data needed for *Navigation* process. This process also stores the SI transmitted in the Datagram packets in the local client database and sends SI data to *Presentation* process.

The JMF video/Audio player is used in the *Playback* process. The video and audio stream (V/A) is output to the *Presentation* process. The *Interaction* process gets input from the remote control, e.g., activates the Navigator main menu. The *Presentation* process displays the Navigator user interfaces.

The *Navigation* process is responsible for getting input from the *Interaction* process and the decoded SI data from *Information Access* process, controlling the player in *Playback* process, and preparing all user interface components for the *Presentation* process.

State Model of The Navigator Control

The interaction tool was a PC remote control. Only the main buttons are drawn on the panel (cf. Fig. 9). The state model of the Navigator by PC remote control is illustrated in Fig. 10. The purpose is to model the Navigator user interface display via PC remote control.

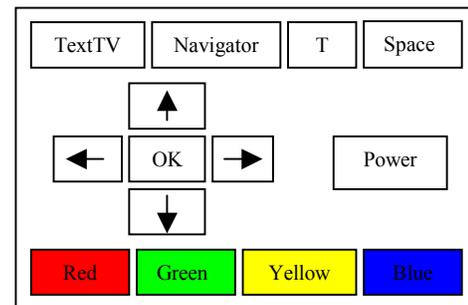


Fig. 9. A Conceptual PC Remote Control Panel.

6 CONCLUSIONS

In this paper, we have implemented a basic set-top box Navigator in resident subset of Java classes. In particular, the Navigator's features related to the DVB-SI tables was discussed, simulation of broadcasting transmission of transport streams via the Internet was designed and implemented.

This work demonstrated that parsing transport streams and decoding the DVB_SI data in the set-top box are the key functions of the Navigator for retrieving and updating its data information. Furthermore, Java is a good platform

for coding the Navigator and accessing data from files and network. It is especially suitable for coding remote control events, the graphical user interface, etc.

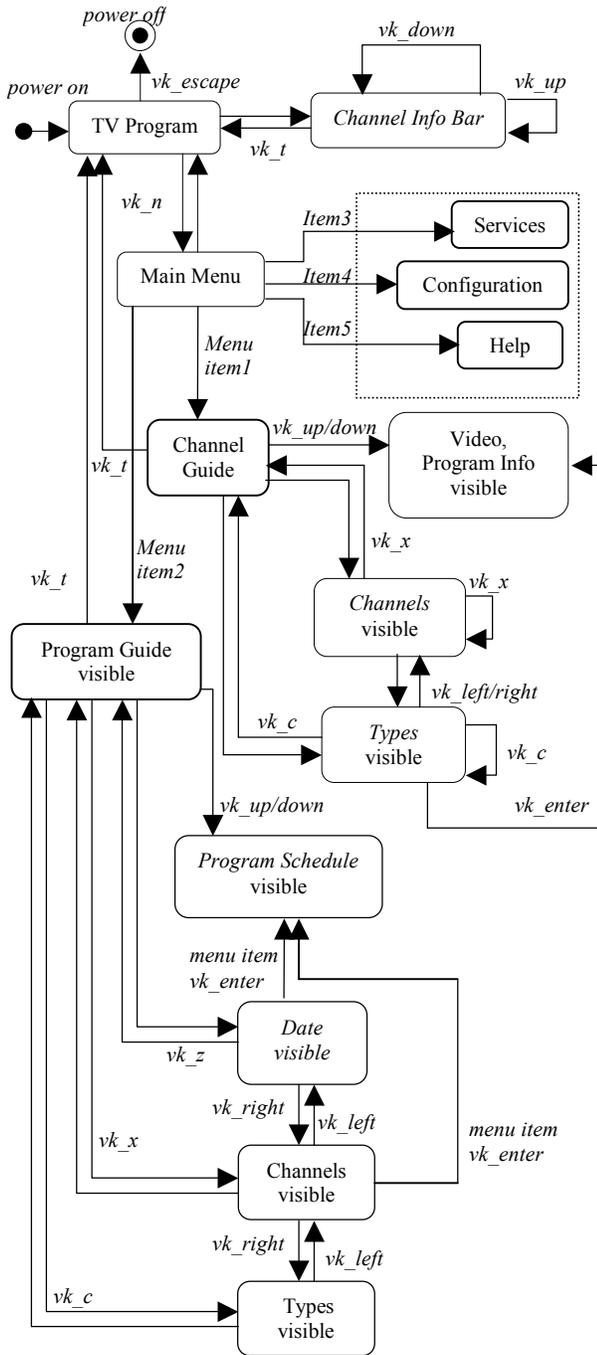


Fig. 10. State Model of the Navigator.

Our work constitutes a basis for developing other multimedia services in the set-top box (e.g., Near Video-on-Demand, mosaic services, enhanced Teletext service, etc), designing the user interface of these services, decoding, displaying and controlling subtitles in the set-top box, etc.

We also gathered statistics about the consumption of memory. Java bytecodes of the Navigator consumes about 107 kB. The SI data files consisting of binary and text files need about 350 kB. The look & feel images of the user interface including channel logos need about 202 kB. This statistics indicate that memory space of caching the SI and the user interface initialization data is greater than embedding the Java bytecodes of the Navigator. Therefore, balancing accessing time and display performance with memory consumption will be one of the design issues in the set-top box with limited resources.

We plan to embed the Navigator bytecodes in our system and make it work with other system functions and services in real broadcasting environment. Another goal is to develop other multimedia services as mentioned above in conjunction with DSM_CC (Digital Storage Media Command and Control) data carousels and object carousels.

7 ACKNOWLEDGMENTS

We thank Mr. Juha Vierinen for helping with the graphical design of the Images in the Future TV project. The Future TV project is funded by the National Technology Agency of Finland together with major Finnish television, telecommunications, and digital media companies. The author Chengyuan Peng would also like to thank Nokia Oyj Foundation for providing the scholarship and support during the research work.

8 REFERENCES

- [1] J. - P. Evain, The Multimedia Home Platform – an overview, *EBU Technical Review*, spring 1998.
- [2] NorDig II and III, Digital Integrated Receiver Decoder Draft Specification, for use in cable, satellite and terrestrial networks, January 2000.
- [3] P.A. Sarginson, MPEG-2: Overview of the systems layer, *Research and Development Report*, No. 1996/2.
- [4] ISO/IEC 13818-1, Information Technology - Generic Coding of Moving Pictures and Associated Audio Recommendation H.222.0 (systems).
- [5] ETS 300 468, Digital Video Broadcasting (DVB) Specification for Service Information (SI) in DVB systems, *European Telecommunications Standards Institute*, January 1997.
- [6] Justin Couch, *Java™ 2 Networking*, McGraw-Hill, 1999.
- [7] C. Peng and P. Vuorimaa, Developing Java User Interface for Digital Television. *The 8th International Conference in Central Europe on Computer Graphics, Visualization and Interactive Digital Media' 2000*, Czech Republic, February 2000.