Digital Broadcasting Receiver Platform with Functional Expandability by Using Network Connected Devices

Akitsugu BABA† Masaru TAKECHI† Haruo HOSHINO‡ and Yoshiaki SHISHIKUI‡
† NHK Science & Technical Research Laboratories 1-10-11 Kinuta, Setagaya-ku, Tokyo, 157-8510 Japan
E-mail: †{baba.a-ik, takechi.m-fa, hoshino.h-ii, shishikui.y-hw}@nhk.or.jp

Abstract This paper proposes a new digital broadcasting receiver platform that have functional expandability by cooperation between receivers and network connected devices. We developed a Java-based data broadcasting technology that is used for controlling the cooperation and confirmed its feasibility by implementing a prototype system.

Keyword Digital Broadcasting, Reception Environment, Home Network, UPnP, DLNA

1. INTRODUCTION

With the rapid technological advances and diversification of viewer's needs in recent years, expansion of the broadcasting services are expected. However, conventional digital broadcasting receivers aren't equipped with a mechanism that upgrades their functions. Under the circumstances, viewer will be forced to purchase a new receiver whenever broadcasters provide new service which requires additional functions of receiver. So we intend to make the receivers to extend their functions by working together with other devices on a home network.

Digital broadcasting in Japan provides data broadcasting services that offer various multimedia information, such as news, weather forecasts and so on, by using broadcast markup language (BML) [1]. Data broadcasting content is responsible for presenting such information and controlling usage of components, such as video, audio and data, multiplexed in broadcasting content as well. Considering the functional expandability of receivers, computer-program-based data broadcasting content controlling cooperation between receivers and other devices on a network is more capable than that based on markup language. In this paper, we propose a new digital broadcasting receiver platform which utilizes devices on a network under the control of Java-based data broadcasting content and describe development of a prototype system.
2. NEW DIGITAL BROADCASTING RECEIVER PLATFORM

2.1. System Model

We propose a new digital broadcasting receiver platform in which the receiver, controlled by a data broadcasting content, works together with other devices on a home network, as shown in Fig. 1. Fig. 2 shows a structure of the new receiver platform. Typical devices on the home network working with the receiver are;

1. Human Interface Devices (HID)
2. Display Devices (Renderer)
3. Content Conversion Devices (Translator) [2]

HIDs offer input means that are different from the receiver's remote control. For example, an HID with keyboard is more suitable for an entering character content. Renderers enable the viewer to select an appropriate display or use multiple displays for different content. Translators enable a viewer to enjoy content by converting it into a form that fits the viewer's requirements or the capabilities of the display. Some translators may be able to slow down the rate of speech without distortion as a listening aid for elderly people. Thus, by integrating the receiver with functions not equipped in it, broadcasters will be able to provide enriched and flexible broadcasting services.

2.2. Class Design for Network Cooperation

Java-based data broadcasting specifications have been standardized such as ARIB-J [3] and DVB GEM [4]. But, they aren't capable of the functional expandability by network cooperation. We propose new classes that enable the above-mentioned services. Fig. 3 shows an outline of
the UML class diagram of the proposed classes. The APIs of these classes provide the following capabilities:

1. Device Discovery

   DeviceManager discovers devices connected to the network, and provides a method to obtain information about discovered devices to a data broadcasting content. DeviceDiscoveryListener allows an application to respond to DeviceDiscoveryEvent.

2. Device Operation

   The Device Interface represents a discovered device and provides basic methods for all device types, such as the methods to retrieve information about the device and the methods for I/O operations. The HID, RendererDevice and TranslatorDevice Interfaces, which inherit the Device Interface, represent typical specific devices. For example, HID Interface provides methods to operate human interface devices.

3. Device State Detection

   DeviceListener Interface allows an application to receive DeviceEvent that notifies state changes of the devices.

We defined most of the proposed classes as abstract interfaces independent from the hardware and software implementation. Therefore, broadcasters can expect data broadcasting content to have the same behavior on all the receivers. In some receiver implementations, a data broadcasting application can operate devices connected to various network interfaces such as USB, Bluetooth, and Ethernet. We intend to utilize Universal Plug and Play (UPnP) for device discovery of IP networks, because UPnP compliant devices defined by Digital Living Network Alliance (DLNA) are becoming increasingly widespread in Japan and have less incompatibility problem than FireWire.

3. PROTOTYPE IMPLEMENTATION

3.1. Developed Entities

   We carried out a trial implementation to verify the feasibility of the proposed platform. We developed the following entities:

   1) Prototype receiver

      We developed a receiver equipped with a Java virtual machine based on an embedded processor board. We installed libraries for proposed classes that enabled the receiver to communicate with a UPnP device over IP network.

   2) Network connected device (HID)

      We also developed a new UPnP-compliant HID as a network-connected device. This HID has the function of transmitting input data from the user as a UPnP event message.

   3) Data broadcasting content

      Further, we created a data broadcasting content (Java application) that uses user inputs from an HID device equipped with a keyboard.

3.2. Operation Sequence

   Fig. 4 shows an example of the operation sequence when the application uses user inputs from the network-connected HID. First, the application instantiates
Fig. 4. App Using User Input from Network-Connected HID

a DeviceManager and subscribes to the device discovery event. DeviceManager discovers the UPnP HID on the home network and creates an HIDDevice to represent the discovered HID. The application receives a device discovery event that contains information about the HIDDevice and subscribes to the input event from the HID. Finally, the application receives user input data from the network-connected HID and executes an operation using the data. We confirmed that an application program as data broadcasting content can discover the devices on the network and incorporate its functions.

4. CONCLUSION

We proposed a new digital broadcasting receiver platform including its classes for network cooperation. By this platform, viewers can enjoy broadcast content in various ways under the control of Java-based data broadcasting content. Its feasibility is confirmed through a prototype implementation of this platform. We intend to use this technology for advanced digital satellite broadcasting in Japan to be launched after 2011.

REFERENCES


