USER-CENTRIC COMPANION SCREEN ARCHITECTURE FOR IMPROVING ACCESSIBILITY OF BROADCAST SERVICES

Hisayuki Ohmata and Masaya Ikeo

NHK, Japan

ABSTRACT

Bridging broadcast and internet services can be difficult, because the system architectures are typically different. Though integrated broadcast-broadband systems that can provide companion screen (CS) services have been standardized, existing systems are television-centric; therefore, it can be difficult for users to easily access broadcast services through mobile applications. Therefore, we propose a CS architecture (CSA) that can be used to tune to specific TV channels and launch broadcast-related applications from a CS such as a smartphone, smart speaker, or smart watch. The general versatility and feasibility of the proposed CSA was verified through prototyping of multiple use cases. Moreover, based on user tests, we confirmed that our proposed CSA has a potential to improve broadcast accessibility from mobile applications and increase the number of times the users watch broadcast programs. Thus, our proposed CSA can help bridge broadcast and mobile or Internet-of-Things services.

INTRODUCTION

The diversification of people's lifestyles has complicated the roles of media and associated service consumption. In recent years, several service providers have designed cross-device and mash-up services to provide appropriate content for individual users. In addition, broadcasters now broadcast programs and related content over the internet. From the standpoint of media consumption, the border between broadcast and internet has disappeared. However, bridging broadcast and internet services can be difficult, because the system architectures in these two cases might vary. Hence, standardized integrated broadcast-broadband (IBB) systems such as Hybridcast (1) and HbbTV (2) enable to combine broadcast and internet services on a TV screen. They also provided companion screen (CS) services to extend cross-device services between TVs and smartphones. However, CS services might suffer from certain disadvantages; in particular, the currently employed systems are TV-centric and thus it can be difficult for users to easily access broadcast services through mobile applications.

Therefore, we propose a CS architecture (CSA) that can tune to specific TV channels and launch broadcast-related applications from a CS. Our proposed architecture allows various mobile applications to smoothly access broadcast services through a simple one-tap action. In addition, because the proposed CSA can also be implemented in smart speakers, smart watches, and other devices, it can act as a bridge between broadcast and

Internet of Things (IoT) services. The general versatility of the CSA was verified through prototyping. In particular, we prototyped CS software that supports the CSA for TV, smartphone applications, smart speakers, and smart watches; in particular, we demonstrated various use cases with partnership broadcasters.

RELATIONSHIP BETWEEN BROADCAST AND INTERNET MEDIA

The proliferation of Internet services, smartphones, and IoT-enabled devices has diversified human lifestyle. As previously mentioned, several service providers design cross-device and mash-up services to provide appropriate content for various users; simultaneously, broadcasters now broadcast programs and related content over the internet. Considering media consumption, aside from diversification, the time spent by viewers accessing media has increased in the last decade. In particular, currently, the time spent watching TV is shifting to using smartphones (3). Therefore, one can ask if Internet is a competitor of broadcast services. To address this question, we conducted a survey on user behavior to identify how people knew about TV programs before watching them.

Age	20–59
Gender	Male / Female
Residence	Japan
Other	Users of Both TV and Smartphone.

Table1 – Condition of participants

	20s	30s	40s	50s
Male	125	125	125	125
Female	125	125	125	125

Table2 – Sample size of the questionnaire

We conducted our online questionnaire between Nov. 27–28, 2017. Table 1 shows the condition set for participants. The number of respondents to the questionnaire was 1000. To reduce age and gender bias, we defined segments; the sample size for each segment is listed in Table 2. We provided the questionnaire to participants who met the conditions listed in Table 1. In the questionnaire, via a multiple choice question, we inquired about how the participant got to know about TV programs; the results for this question are shown in Figure 1. From the figure, it is clear that the most popular way viewers get to know about programs is when consuming TV media. However, in our study, we focused on internet media. Though the number of participants selecting an internet service as a source of information for programs, such as through social networking services (SNSs), online news articles, and so on, is not large individually, there are many such services. Consequently, the total number of participants who answered with an internet media option is the second most popular after TV media. Figure 2 shows the type of media used as a source of program information by age. In general, younger participants have a higher rate of obtaining program information through internet media. Based on these results, providing broadcast-related information through various internet services will be effective in increasing the awareness about programs, especially in the case of digital natives. Further, we considered user behavior while viewing TV. From the results presented in (4), 70% of US adults use CS devices such as smartphones while watching TV; it is common for people to browse the internet, check their SNS feeds, or texting while in front of the TV.

Considering this user behavior regarding TV viewing, the Internet seems to be a supporter rather than a competitor of broadcast. Thus, providing information regarding broadcast programs on mobile applications or websites as well as a smooth user interface to access broadcast services has great potential in improving accessibility to broadcast services. Moreover, a common or standard technology allows for diversified service providers, including broadcasters, to bridge the gap between broadcast and mobile services.

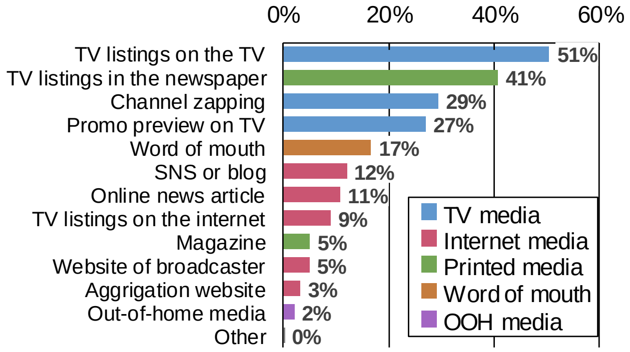

Figure 1 – Source of information regarding TV programs

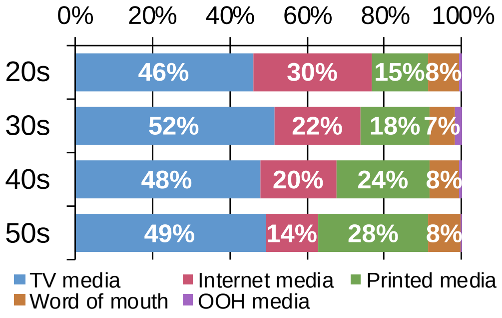

Figure 2 – Type of media as source of TV program information by age

RELATED WORK – COMPANION SCREEN TECHNOLOGIES

The standard IBB systems can provide CS services between TVs and smartphones; these systems enhance user experience by providing broadcast-related content on the smartphone while the user is watching TV. In particular, Hybridcast and HbbTV can execute a web application called broadcast-related application on a TV using a broadcast signal. When providing CS services, the broadcast-related application on a TV sends the command message and URL of a web application that should be launched on the CS. Both applications can then communicate with each other directly via text message on the local network. However, the current architecture is designed using a TV-centric model. Therefore, if a user wants to use broadcast services including the CS service, he/she must first turn on and tune in to the TV using the remote control; in other words, it is impossible for users to access broadcast services by operating only the CS in the current IBB system.

On the other hand, mobile-centric modelled CS technologies are available. DIscovery And Launch (DIAL) (5) is a protocol for device discovery and application launch. Smart TVs pre-installed with YouTube or Netflix applications support DIAL. Using these devices, people can cast streaming videos from a smartphone to the TV. Hybridcast and HbbTV also use this protocol to discover TVs from the CS. In addition, HbbTV 2.0 allows the launch of broadcast-independent applications from the CS, but it cannot handle broadcast channels and data. These solutions are not designed for broadcast services or broadcast-related applications and it is currently not possible for users to access broadcast content using mobile services directly.

COMPANION SCREEN ARCHITECTURE

We propose a CSA that is based on the existing IBB system Hybridcast. We adopt Hybridcast because CS services for it have already been launched in 2013, ahead of HbbTV; in addition, more than 7 million Hybridcast-compliant TVs have been shipped. The existing CSA of Hybridcast (CSAv1) is designed based on a TV-centric model. CSAv1 defines a common protocol that can connect various manufacturers' TVs via a common companion application called "Hybridcast Connect" (HCApp). Figure 3 shows the functional model of the proposed CSA called CSAv2. CSAv2 is primarily composed of three modules: a TV-link module, service-link module on the CS, and CS-link module on the TV. The TV-link module is essential for linking the CS with the TV. In particular, the TV-link and CS-link modules are extensions of CSAv1. In contrast, the service-link module is composed of multiple functions designed to enhance the linking of devices and services; many of these implemented functions might be different depending on the requirement of each service.

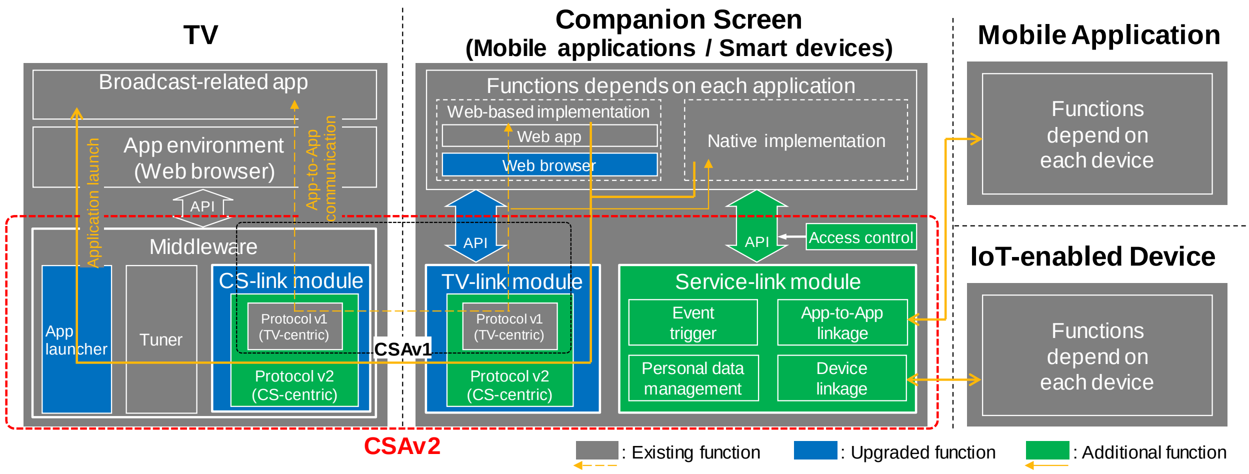

Figure 3 – Functional model of the proposed CSA

TV-Link and CS-Link Modules

The TV-link module simultaneously runs with the CS-link module. CSAv1 provides the device-linkage protocol, including device discovery and handshake mechanisms between the TV and companion applications. The APIs for the applications launched from the TV onto the companion application and app-to-app communications between the TV and CS are also defined by CSAv1. These functionalities are implemented in the TV-link and CS-link modules.

In the CSAv2 which is an extension of CSAv1, we designed a CS-centric protocol, combined it with the channel tuning and application launch functions on the TV, and upgraded the TV-link and CS-link modules. DIAL was adopted to launch new applications because most Hybridcast TVs already support it. In addition, because our aim was to enhance the TV experience, it was decided that the broadcast service applications running on TVs should be broadcast-related rather than broadcast-independent. Therefore, this protocol defines the execution order for applications that are to be launched after tuning into a channel. To allow for this protocol to be used by a Web application on the CS or native functions of mobile applications or smart devices, we also define TV-control APIs for web-based and native implementations.

Service-Link Module

The service-link module is composed of multiple functions designed to enhance linkages between devices and services. Each function is provided by a common API for native applications or Web applications running on the CS. The use of a simple interface ensures that broadcasters and various service providers can provide cross-device services when linking multiple services using these APIs, which is expected to reduce application development costs.

Device-linkage function: This function is used to link CS and IoT-enabled devices and is applied to our proposed architecture to control IoT-enabled devices via broadcasts (6). In particular, this function discovers and authenticates devices near the CS and establishes a connection in order to transmit data. In addition, it abstracts the differences in the device model, type, or communication protocol, if possible, in order to enhance the ease of CS application development.

App-to-app-linkage function: This function enables app-to-app communications on smartphones using the intent function provided by the operating system. It allows a CS application to launch other applications using the appropriate parameters, and vice versa.

Event trigger function: This function is a simple matching engine that matches broadcast-related and user context, such as location, time, or device connection status. Personal data, as explained below, are available as context. Users or service providers can use the data to set conditions that trigger events and actions.

Personal data management function: TV viewing history and life activity history for locations visited and items purchased are managed as personal data. In addition, the device connection and app-to-app communication status are recorded as system logs.

Access Control Mechanism: Web or native applications running on CS can access each function easily using APIs. However, because this mechanism increases security concerns about privacy and device control, a built-in function that authenticates applications when calling APIs based on user and service provider policies is included; this mechanism protects personal data and system functions from incorrect application behavior.

EVALUATION

Prototype

We prototyped application software for smartphones, smart speakers, and smart watches that supports CSAv2 and an upgraded Hybridcast TV. In this study, we described a use case for using the TV while in front of it. Figure 4 shows the architecture of our prototype.

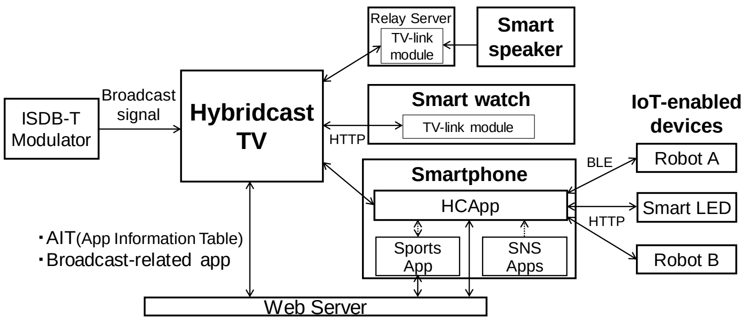

Figure 4 – Architecture of prototype system

The HCApp implements the CSAv2, NHK sports, and SNS applications such as Twitter and LINE were installed on an Android smartphone. Three IoT-enabled devices could be connected with the HCApp. Robot A is an intelligent ball that can rotate and flash; the brightness and color of the smart LEDs in the ball can be controlled. Robot B can move its body and speak. Robot A uses the Bluetooth Low Energy protocol, while Robot B uses HTTP via Wi-Fi. A modulator transmits the broadcast signal of the ISDB-T. The event message (EM) trigger data synchronized with the program are multiplexed on this signal.

We verified feasibility of CSAv2 using the prototype in a scenario-based setting in which the viewer is watching a live soccer program, as shown in Figure 5. A demonstration video can be seen on our website (7).

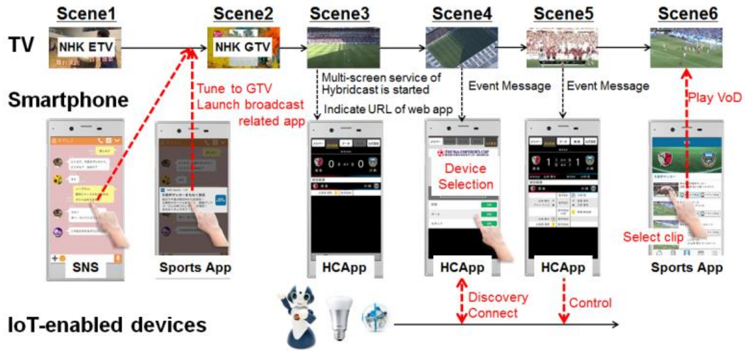

Figure 5 – Prototype scenario

Scene1: While watching NHK ETV, a notification message announces the start of a soccer program on NHK GTV on the sports and SNS applications.

Scene2: When a user taps the notification, the TV is tuned into NHK GTV and a broadcast-related application is launched. The HCApp URL scheme with a control command for broadcast channel information and URL of the application information table (AIT) that indicates the location of the broadcast-related application are embedded in the notification. When the message is tapped, HCApp is launched via the app-to-app communication function and HCApp issues a control command from the TV-link to the CS-link module. Next, the middleware tunes the TV to the specific channel and launches the broadcast-related application.

Scene3: As soon as the soccer program begins, the broadcast-related application on the TV indicates URL of web application to execute on HCApp.

Scene4: A menu asking the user to select IoT-enabled devices that operate in conjunction with the TV program is displayed on HCApp. The device-linkage function discovers and reports available devices that the user can select. Once selected, the device(s) can be connected.

Scene5: When events, such as a goal and kick-off, occur, connected devices move or flash. When the broadcast-related application receives the EM from a broadcast signal, it transmits the EM to HCApp. The control commands are described on the EM as text data and issued to the devices from the device-linkage function. The HCApp sends these commands to appropriate devices.

Scene6: Video clips of the game highlights are provided on the NHK sports application. Both the URL scheme of HCApp and URL of AIT and VoD are embedded as buttons of clips. When a user taps the button, the HCApp is launched, which, in turn, launches the broadcast-related application that works as the MPEG-DASH video player on the TV; this behavior is almost the same as in Scene 2. After this, the user can play any clip on the TV.

Furthermore, we implemented the TV-link module in an Android smart watch application and confirmed that various manufacturers' TVs could be controlled from the smart watch via voice and touch interactions. In addition, we verified connectivity between smart speakers (such as Amazon Echo and Google Home) and TVs using a server-implemented TV-link module. The server is located on the same network as the TV and relays commands from the cloud service of these speakers to the local network.

In 2017, we and our partnership broadcasters demonstrated multiple service use cases using the CSAv2 at our open house and at global exhibitions such as IBC, CEATEC Japan, and Inter BEE. Some use cases involved scenarios not only in front of TV, but also outside the home. For example, program recommendation service by matching metadata of the program and location-based life activity data acquired via GPS and car navigation were considered to enhance the bridge between life activities and TV experiences. Through these prototypes, we confirmed the feasibility and general versatility of CSAv2.

User test

We evaluated the anticipation levels and satisfaction of user operations related to accessing broadcast services from mobile application using CSAv2. We conducted user tests between Dec. 15–17, 2017. Our tests included a total of 103 participants. We defined each consumer segment and its sample size as listed in Table 3. Participants were recruited based on the conditions in Table 1. Before the user test, we explored their anticipation levels regarding the CS-centric TV control. We showed them descriptions and illustrations such as in Figure 6 and enquired about their anticipation level on a scale of one to five. Of these, 60% were positive ("Very positive" or "Positive"), 19% were neutral, and 20% were negative ("Negative" or "Very negative").

	20s-30s	40s-50s
Male	26	26
Female	26	25

Table3 – Sample size of participants in the user test

Figure 6 – Description and illustration of the new function

During the test, they gained actual experience with the new functions of a CSAv2 prototype system in a manner similar to the situation shown in Figure 6. After this, we asked the participants about some questions regarding their experience. In particular, we asked the participants to provide their before and after impression levels regarding this function on a scale of one to five. As shown in Figure 7, the results indicate that 86% of all participants, including 94% of the participants who answered in negative before this test, had better impressions of the function provided by our proposed CSA after the test. Furthermore, we asked the participants who had initially answered positively to describe the advantages of this function via an open-ended question. At 62%, the most common answer was that they would be effective in reducing the number of inadvertently missed programs. The second most popular answer, at 59%, was that this function facilitates easy operation and reduces the effort required to tune into a broadcast program. Finally, we asked the participants whether it was possible that this new function would

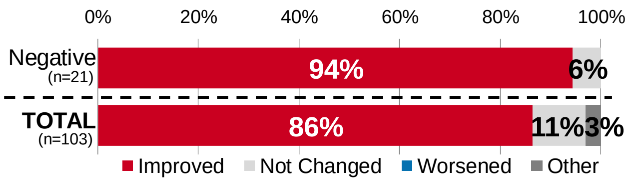

Figure 7 – Change of impression about the new function before and after the experience

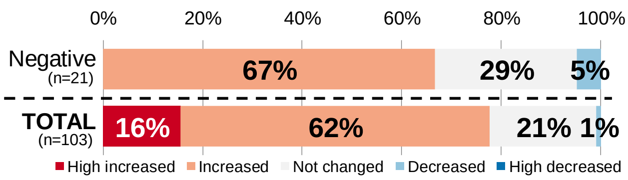

Figure 8 – Possibility of increase or decrease in number of times broadcast programs are viewed via the new function

increase or decrease the number of times that they watched broadcast programs. As shown in Figure 8, the results indicate that 78% of all examinees, including 67% of the examinees who answered in negative earlier, said that using this system would probably increase the amount of time that they watched broadcast programs.

Based on these results, we can conclude that the CS-centric TV control function has the potential to increase user satisfaction by providing smooth inter-operations between mobile to broadcast services and reducing the number of missed broadcast programs. Furthermore, we concluded that the increased user satisfaction might also affect user behavior by increasing the number of times that they watch broadcast services.

CONCLUSION

We developed a CSA that can tune to specific TV channels and launch broadcast-related applications from a CS by redesigning an existing IBB system. The general versatility and feasibility of our proposed CSA was verified through prototyping. In particular, we prototyped CS software that supports our CSA for TV, applications on smartphones, smart speakers, and smart watches, and demonstrated various use cases with partnership broadcasters. Moreover, we surveyed how people got to know about broadcast programs and evaluated broadcast accessibility from a mobile application with our CSA via user tests. Based on results, we conclude that our proposed CSA has a potential to improve broadcast accessibility from mobile applications and increase the number of times that users watch broadcast programs. Furthermore, the standardization procedure of the CSA is nearly complete in Japan; thus, this CSA helps to bridge broadcast and mobile or IoT services and promote the emergence of new broadcasting business ventures.

REFERENCES

1. IPTV FORUM JAPAN, 2014. IPTVFJ STD-0010 Integrated Broadcast-Broadband System Specification Version2.0.
2. ETSI, 2016. ETSI TS 102 796 v1.4.1 Hybrid Broadcast Broadband TV.
3. eMarketer, 2016. Average Time Spent per Day with Major Media by US Adults, 2012-2018, https://www.emarketer.com/Chart/Average-Time-Spent-per-Day-with-Major-Media-by-US-Adults-2012-2018-hrsmins/188929
4. Marketing Dive, 2017. EMarketer: 70% of US adults 'second-screen' while watching TV, https://www.marketingdive.com/news/emarketer-70-of-us-adults-second-screen-while-watching-tv/510341/
5. Netflix. 2017. DIAL DIscovery And Launch protocol specification Version 2.1. http://www.dial-multiscreen.org/
6. H. Ogawa et al. 2018. System Architecture for IoT Services with Broadcast Content, Proceedings of IEEE ICCE 2018, pp.802 to 803.
7. NHK. 2017. Enriching Daily Activity by TV Content Connecting with IoT, https://www.nhk.or.jp/strl/open2017/tenji/2_e.html

ACKNOWLEDGMENTS

The authors would like to thank the partnership broadcasters who considered and prototyped service use cases using CSAv2.