System Architecture for Content-Oriented IoT Services

Hiromu Ogawa, Hisayuki Ohmata, Masaya Ikeo, Arisa Fujii and Hiroshi Fujisawa
Science & Technology Research Laboratories
NHK (Japan Broadcasting Corporation)
Tokyo, Japan
ogawa.h-ju@nhk.or.jp

***Abstract*— We propose a system architecture for content-oriented internet of things (IoT) services. Here, content-oriented implies that a scenario of a content determines the operation of IoT-enabled devices. The proposed system realizes IoT services associated with contents. The proposed system comprises an integrated broadcast-broadband (IBB) system, Web of Things (WoT), and content description (CD), which is a type of structured metadata. Through prototyping, we confirmed the feasibility of our proposed architecture.**

Keywords— Internet of Things (IoT), Web of Things (WoT), Integrated Broadcast-Broadband System, Hybridcast, HbbTV 2.0

I. Introduction

In daily life, different types of contents are consumed. Here, internet streaming content and video-on-demand (VoD) content are called timed media content. Timed media content changes with time and contains not only the video stream but also other elements such as event trigger data, which is metadata indicating some event occurrence in the video stream. Broadcast content transmitted using a broadband system is a type of typical timed media content. The main services provided by a broadcaster are creating and delivering broadcast content. Although many types of information services are available today, broadcast services still assert significant influence on our daily lives.

For a future smart city, smart internet of things (IoT) is considered a key technology [1][2]. There have been several active discussions on how data generated by IoT-enabled devices should be used , such as collecting big data from vast amounts of sensing devices, analyzing the environment or the situation based on these data, and automatically controlling devices to improve their performance. On the contrary, the manner of using timed media content with the devices has not been as actively discussed as data utilization.

However, we are convinced that controlling IoT-enabled devices in conjunction with timed media content can realize new experiences as described below.

a) Realistic broadcasting: A service that provides additional information such as tactile force using a tactile force-sensing device in sync with video played on a TV is used to enhance the feeling of presence.

b) Continuous information transmission using multiple devices: A service that provides a user who has watched broadcast content in his living room and left with continued information about content using IoT-enabled devices around them. Each IoT device provides notifications, in its own capacity, when any event occurs in the content.

c) *Utilization of content viewing footprints:* A service that provides some incentives to a user who has watched a specific scene of particular content. For example, a user who has watched a commercial receives a coupon when he/she is closer to a beacon placed in a store. This bridges the gap between content and IoT services.

In the above-mentioned examples, the scenario of the content determines the operation of IoT-enabled devices; the broadcast content determines a video that is played on TV. Therefore, we propose to call such services as content-oriented IoT services. To realize these services, we designed a system architecture for broadcast content with a standard mechanism to utilize event triggers. In addition, we confirmed the general versatility of our proposed architecture through prototyping.

II. Related Work

Some collaboration systems for broadcast services and physical devices have already been proposed. For example, the method proposed for creating immersion in TV programs in [3] provides a tactile-sense presentation system for smart TVs. Moreover, an interactive robot that enables chat capability with a viewer while watching TV programs was proposed in [4]. These studies presented the effectiveness of new IoT services in conjunction with timed media content. However, these systems are designed for particular content and devices and thus lack versatility.

The World Wide Web Consortium (W3C) is currently considering a web-based IoT architecture named Web of Things (WoT) [5] that enhances the interoperability between IoT-enabled devices conforming to different IoT standards. The Thing Description (TD), which is the primary building block of WoT, is a structured metadata approach to describe device features such as types, functions, and manners. Interpreting and communicating with the device are possible by referring to the TDs. In addition, WoT provides a runtime environment equipped with application program interfaces (APIs) to absorb the difference in a control method for each IoT-enabled device so that these can be controlled in a unified application description manner. Although WoT enables IoT-enabled devices to interoperate automatically, it lacks a function to operate contents.

978-1-5386-9151-9/19/$31.00 ©2019 IEEE

III. Proposed System Architecture

A. System Requirements

To establish a versatile system for content-oriented IoT services, the following issues in the service development must be considered.

- Generally, device providers are unaware of broadcast content before it is broadcasted. Therefore, a mechanism that enables device providers to operate IoT-enabled devices in conjunction with the content is required even if its scenario is unknown beforehand.
- Generally, content providers are unaware of the devices that are present in a user's local environment. Therefore, a mechanism that enables content providers to match the content and suitable devices available in the local environment and control the devices is required even if the devices that are present are unknown beforehand.

The following system requirements are derived based on the abovementioned issues for the newly designed system architecture.

- Ensure that IoT-enabled devices can interpret the scenario of content.
- Ensure that IoT-enabled devices can judge whether it is matched with devices that the content intends to control.
- Ensure that IoT-enabled devices can translate conceptual commands provided by the content into specific instructions for itself and perform them.

B. System Architecture

Our proposed system architecture, which fulfills all above-mentioned requirements, is shown in Figure 1. This is an extension of a previously proposed system architecture [6].

Our system comprises three types of devices: a TV that receives and plays broadcast content, IoT-enabled devices that operate based on command messages, and a companion device such as a smartphone or a smart speaker that works as a hub to exchange messages between devices. Our system also utilizes two types of structured metadata: Content Description (CD) that describes comprehensive information of a broadcast content, and TD.

To operate IoT-enabled devices according to broadcast content, a mechanism to transmit metadata between a TV and IoT-enabled devices is required. Therefore, two systems are applied: an integrated broadcast-broadband (IBB) system and WoT. The IBB system enables a TV and a companion device to exchange metadata. Although standard IBB systems exist such as Hybrid Broadcast Broadband TV (HbbTV 2.0) [7] and Hybridcast [8], Hybridcast is applied because its companion service has more experience than others. Conversely, WoT enables the companion device and IoT-enabled devices to exchange metadata. As both Hybridcast and WoT are web-based systems, a single companion application operated on the companion device can easily control all communications.

To notify an event occurrence, the broadcaster transmits Event Message (EM), an event trigger multiplexed on a broadcast signal to the TV at arbitrary timings in conjunction with the TV program progress. Hybridcast TV is equipped with a web browser to run an HTML5 application that can handle the EM. Meanwhile, the companion application equipped with the Hybridcast module is installed in the companion device so that the companion device can communicate with the Hybridcast TV in the same segment of a local network [9]. Accordingly, the HTML5 application receives the EM and then transfers the event notification to the companion application. Furthermore, a WoT device exposes web APIs to call its functions so that the companion application can call functions when it receives the notification.

Although an EM indicates the occurrence of an event, it neither describes the actions to be performed nor the devices that should perform the actions. The proposed CD and content device-matching module (CDMM) can be used for these functions. The CD represents structured metadata. It contains (1) comprehensive information on the content such as its title, (2) information to be matched with suitable devices, such as types or functions of targeted devices, and (3) information to control devices when an event occurs, such as identifiers of events or commands at the events. Vocabularies such as the ones defined by schema.org [10] are used to describe target types or target functions. The CDMM is a module incorporated in the companion application. It refers and compares a CD and TDs to determine suitable devices depending on the content. In the case of the following, the CDMM regards a device as the suitable one. (1) A targeted device type described in the CD is identical with the type of device described in the TD. (2) A targeted device function described in the CD is identical with a function of the device described in the TD. In addition, the CDMM manages events to control the devices. Given an event trigger, the CDMM judges the commands to be sent and the devices to be controlled according to the CD; it also determines how to send the commands to the device according to the TDs.

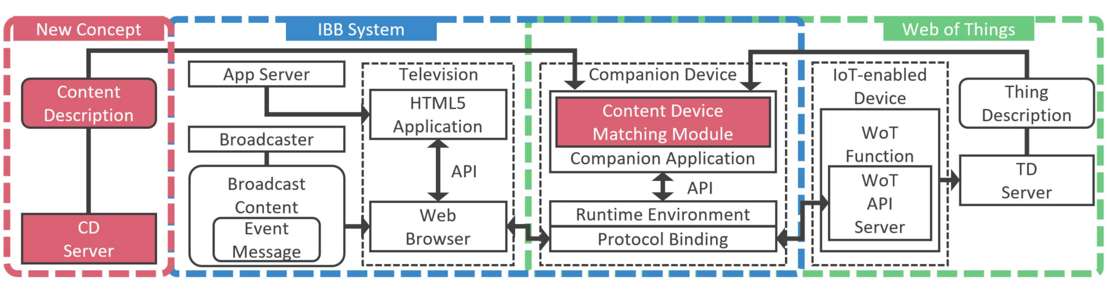


Fig. 1. Proposed system architecture

IV. Implementation and Evaluation

Two services were prototyped to validate the proposed system: a realistic broadcasting described as example (a) and continuous information transmission using multiple devices described as example (b) in section I based on the proposed system. A demonstration movie for realistic broadcasting is available in [11]. The implementation of continuous information transmission using multiple devices, as shown in Figure 2, is outlined below.

Fig. 2. A robot, mirror, and light in the implementation of example (b)

A. Service Scenario

The prototyped service comprises a television, a smartphone as a companion device, a communication robot in a kitchen, a smart mirror in a bathroom, and a smart LED light on the ceiling of a hallway as IoT-enabled devices. A user is watching a soccer game program on his TV in his living room. For some reason, the user may leave and stop watching the game. In this situation, when an event occurs in the game (e.g., a kick off), each device will notify the user about the event in its own capacity. The robot will notify the user about the event through its voice; the mirror will display a text message expressing the game situation; the light shines in the colors of the team, thereby notifying the user about the goal. As a result, the user can notice the occurrence of the event and come back to his living room immediately so that he does not miss watching the game.

B. Service Sequence

The user launches the companion application. Then, the companion application searches for surrounding IoT-enabled devices and fetches these TDs. Next, the TV receives the broadcast content and plays its video. Then, the HTML5 application sends a message containing the source URL of the CD. Subsequently, the companion application fetches the CD from the URL and matches by comparing the CD and TDs (described in the succeeding paragraph in detail). When an event occurs in the game, the broadcaster sends an EM containing the identifier of the event to the TV. Then, the HTML5 application transfers it to the companion application. Next, the companion application sends control commands to suitable devices (described in the succeeding paragraph). The IoT-enabled devices operate through the above process.

The manner in which a CDMM matches broadcast content and IoT-enabled devices and sends control commands to devices based on CD and TDs is described below. Figure 3 shows (1) a CD of the soccer game program, (2) a TD of the communication robot, and (3) a TD of the smart LED light. These are the extracts of CD and TD required for the explanation below.

In (1) CD, a reference to a namespace used in this CD is described (L.1). Then, comprehensive information regarding the broadcast content such as its title (L.2) is described. Next, device features controlled in conjunction with this content are described (L.4-10). Here, two types of devices are targeted: the first object (L.5-6) designates a device with a "*showInfo*" function defined in namespace "*myiot*" (L.5) and the second object (L.8-9) designates a device typed as a "*SmartLight.*" Additionally, identifiers "*informer*" (L.5) and "*smartLight*" (L.8) are provided to each device. Meanwhile, in (2) TD of the device, it is mentioned that this robot has a "*showInfo*" function (L.26). Furthermore, in (3), the light is described as a "*SmartLight*" type (L.34). Based on this information, the CDMM judges whether the devices are suitable for content and allocates identifiers for each device.

The CDMM sends commands to each device when an EM is provided. Here, the CDMM is notified that a "*kickOff*" event occurred. The CDMM verifies whether the CD describes a process for "*kickOff*" events. Here, the first object (L.13-20) of "*event*" (L.12) describes it. "*control*" (L.14) is an array of objects that describes how each device is controlled when the event occurs. It comprises "*@id*" (L.15), the identifier of a controlled device, and "*interaction*"(L.16), the object for interaction. The "*interaction*" object contains "*@type*" (L.17), the function of the device to call and "*schema*" (L.18), an object that describes the value and type of argument passed on to the device. In this example, the "*showInfo*" function of a device named "*informer*" is called and an argument that has a "*String*" type (L.19) and "*kick off*" value (L.20) is passed. Next, the CDMM refers to the TD of the device named "*informer.*" Here, it is (2), and it confirms how the "*showInfo*" function should be called. "*href*" (L.28) described a protocol and an endpoint, and "*mediaType*" (L.29) describes a body type of request. In this example, the CDMM grasps the need to send an HTTP request with "*application/json*" typed body to "*http://robot_addr/say.*" As per the process described above, the CDMM matches CD and TD and grasps how the devices must be controlled.

C. Discussion

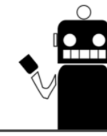
Through the implementation, we confirmed that all devices acted as soon as an event occurred in the program; thus, users can immediately notice the event occurrence. In addition, we obtained the following positive feedback from subjects: “It prevents me from turning off the TV because it provides valuable information even if I’m not in front of the TV;” and “In some situations, it is difficult to use a smartphone to collect information. But, this service provides information through devices that may be suitable in a given situation so that I can receive them easily.” Consequently, we confirmed the validity of our proposed system and indicated the effectiveness of new IoT services with broadcast content.

In the implementation, all requests to each device were sent using HTTP protocols. Each device has a different time period from receiving the messages to starting the actions. For this reason, the order of actions of devices is reversed from the order of commands sent to devices. It may have important meaning in some cases. In response to this issue, we should build a mechanism to control devices in a desired order.

```
"@context": ["myiot": "http://myvocabrary.com/"],
"title": "NHK News",
…
"targetDevice": [{
  "@type": ["Action", "myiot:showInfo"],
  "@id": "informer"
},{
  "@type": ["Thing", "myiot:SmartLight"],
  "@id": "smartLight"
},
…
"event"[{
  "@type": ["event", "kickOff"],
  "control": [{
    "@id": "Informer",
    "interaction": [{
      "@type": ["Action", "myiot:showInfo"],
      "schema": {
        "type": "String",
        "value": "kick off"
        …
```

(1) CD of a Soccer Game Program

```
"@context": ["myiot": "http://myvocabrary.com/"],
"@type": ["Thing", "myiot:CommunicationRobot"],
"name": "myRobot",
"interaction": {
  "@type": ["Action", "myiot:showInfo"],
  "form": {
    "href": "http://roboto_addr/say",
    "mediaType": "application/json
  },
  "schema": { "type": String },
}, …
```



(2) TD of a Communication Robot

```
"@context": ["myiot": "http://myvocabrary.com/"],
"@type": ["Thing", "myiot:SmartLight"],
"name": "myLight",
"interaction": {
  "@type": ["Action", "myiot:PowerOn"],
  "form": {
    "href": "http://light_addr/poweron",
    "mediaType": "application/json
  },
  "schema": { "type": boolean },
}, …
```

(3) TD of a Smart LED Light

Fig. 3. Examples of the CD and TDs

V. Conclusion and Future Work

In this paper, we proposed a system architecture for content-oriented IoT services. We also prototyped cases to validate the system. The proposed system enables IoT-enabled devices to operate in conjunction with the content even if the content is unknown beforehand. In addition, it enables matching of the scenario of content and suitable devices available in a user's local environment and control the devices even if the devices that are present are unknown beforehand. Consequently, the proposed system provides a new expression of content with IoT-enabled devices so that a user can realize new content experiences.

As future work, we plan to extend CD to describe context of a scene such as who is performing the action or what action is being performed. The context enables the companion application to determine the manner in which IoT-enabled devices need to be controlled. To express the context of a scene with structured data, the elements of the scene must be defined. Although this is a difficult task, a sports program is relatively suitable than programs of other genres because events occurring in a sports game are limited. Therefore, we will undertake a study considering this genre. In addition, we will implement a service that does not require synchronization with the broadcast content in real time, and another service that uses an IoT-enabled device located in a user's house. Then, we will extend the proposed architecture in light of the feedback received for these implementations. Furthermore, we will extend the proposed system to not only broadcast content but also other timed media content and general content. In addition, we will assess the user experience of our services.

References

[1] A. Zanella, N. Bui, A. Castellani, L. Vangelista, M. Zorzi, "Internet of Things for Smart Cities," IEEE Internet Things J., vol. 1, no. 1, pp. 22–32, Feb. 2014.

[2] A. Sheth, "Internet of Things to Smart IoT Through Semantic, Cognitive, and Perceptual Computing," IEEE Intell. Syst., vol. 31, no. 2, pp. 108–112, Mar. 2016.

[3] K. Ariyasu, H. Kawakita, T. Handa, and H. Kaneko, "Tactile sensibility presentation service for Smart TV," in Proc. 2014 IEEE 3rd Glob. Conf. Consum. Electron. (GCCE), 2014, pp. 236–237.

[4] S. Nishimura, H. Kawanami, M. Kanbara and N. Hagita. "A TV Chat Robot with Time-Shifting Function for Daily-Use Communication," In International Conference on Social Robotics (ICSR). Springer. Cham. DOI= https://doi.org/10.1007/978-3-319-70022-9_51, pp. 516–525, 2017.

[5] Web of Things at W3C, https://www.w3.org/WoT/, [Nov. 10, 2018]

[6] H. Ogawa, M. Ikeo, H. Ohmata, C. Yamamura, and H. Fujisawa, "System architecture for IoT services with broadcast content," in Proc. 2018 IEEE Int. Conf. Consum. Electron. (ICCE), 2018, pp. 1–2.

[7] ETSI TS 102 796 v1.4.1 Hybrid Broadcast Broadband TV. http://www.etsi.org/deliver/etsi_ts/102700_102799/102 796/01.04.01_60/ts_102796v010401p.pdf, [Nov. 10, 2018]

[8] IPTVFJ STD-0010 Integrated Broadcast-Broadband System Specification Version2.0, https://www.iptvforum.jp/en/download/input.html, [Nov. 10, 2018]

[9] H. Ohmata, M. Ikeo, H. Ogawa, T. Takiguchi, and H. Fujisawa, "Companion Screen Architecture for Bridging TV Experiences and Life Activities," in Proc. 2018 ACM Int. Conf. Interact. Exp. TV Online Video (TVX'18), 2018, pp. 137–147.

[10] schema.org, https://schema.org/, [Nov. 10, 2018]

[11] NHK Science & Technology Research Laboratories OPEN HOUSE 2017, http://www.nhk.or.jp/str/open2017/tenji/14_e.html, [Nov. 10, 2018]