

An Application Framework for Trend Surfing System based on Multi-aspect, Multi-screen and Multimodal User Interface

TORU KOBAYASHI^{1,a)}

Received: January 8, 2015, Accepted: July 1, 2015

Abstract: Currently, a lot of attractive e-services which can be used for retrieving useful information from the Internet have been available. They have different usage manner, so that supporting system for low information technology literacy users is highly expected. The important issue for such a supporting system is extensibility because the system should be followed by new attractive e-services according to user's preference with minimum revision cost. Therefore, this paper proposes "An Application Framework for Trend Surfing System: TSS Framework" that focuses on an extensible application framework for trend surfing. Trend surfing means that some kinds of related trendy information (multi-aspect) retrieved from different e-services can be displayed on different screen devices (multi-screen) according to user's intuitive operation (multimodal user interface). The proposed the TSS Framework derived from the Model-View-Controller (MVC) pattern is based on the system architecture which enables double mashup not only e-services but also screen devices in front of users. It is implemented in a two-phase information retrieval model which enables multi-aspect trend surfing. We show the TSS Framework through an implementation example. We also demonstrate the effectiveness of the TSS Framework by indicating system modification case study.

Keywords: information retrieval, multi-screen, HTML5, Service-Oriented Architecture, Model-View-Controller, user interface

1. Introduction

Currently, a lot of attractive e-services which can be used for retrieving useful information from the Internet have been already available. For example, a keyword searching engine like Yahoo!, a free encyclopedia like Wikipedia, a video-sharing website like YouTube, a commercial book & product website like Amazon, and a social media like Twitter or Facebook have become popular. These e-services are independent of each other. They have also different usage manner. Therefore, users have to know how to handle them in order to retrieve information across these different e-services.

On the other hand, thanks to the spread of useful, high-performance screen devices like tablet computers or smart phones, multi-screen systems containing advanced interactive user interfaces have gained attention. In particular, research and development is aggressively progressing in the area of IP TV [1], [2], [3] or Smart TV [4] which has merged broadcasting and communication. Although these multi-screen systems relating to TV media have different features, they have one feature in common: an advanced interactive user interface. Through this interface, users can perform more active information retrieval synchronized with user's preferences. The advanced interactive user interface contributes to the realization of a more personal information usage environment.

The active information retrieval across different e-services over multi-screen environment mentioned above causes the problem of unfair information retrieval to users who do not have enough information technology literacy. For example, let us consider two user groups. The first user group consists of users who are accustomed to actively using tablet computers or smart phones to get information from different kinds of e-services every day without problems. The second user group consists of users who normally get information only passively, through broadcast TV. If both groups used a multi-screen system equipped with an advanced interactive user interface, the retrieved information results might not be the same. It is obvious that the first user group would retrieve richer information than the second user group. In brief, the information retrieval capability difference causes the gap [5], [6].

In order to solve these problems, we need to have a system that enables the closing of the information retrieval capability gap. The important issue of this kind of support system is extensibility because the system should be followed by new attractive e-services coming up day by day according to user's preference with minimum revision cost. Therefore, this paper proposes "An Application Framework for Trend Surfing System: TSS Framework" that focuses on an extensible application framework for trend surfing. Trend surfing in this paper means that some kinds of related trendy information retrieved from different e-services can be displayed on different screen devices according to the user's intuitive operation. We call this related trendy information "multi-aspect" information. The appearance of such an action would be like net surfing across different screen devices.

¹ Division of Electrical Engineering and Computer Science, Graduate School of Engineering, Nagasaki University, Nagasaki 852–8521, Japan

^{a)} toru@cis.nagasaki-u.ac.jp

The trendy information of this paper is current trends, such as hot words, news topics, and SNS topics. Hot words are popular searching keywords on search engines like Yahoo!. Hot words reflect the interests of a majority of society. News topic is information distributed from the mass media. SNS topics indicate posted messages among friends of a target user. This trendy information might be new to users at any time, so the information retrieval capability gap could be observed more easily.

The Trend Surfing System consists of several screen devices such as tablet computers. For example, hot words, news and SNS topics will be listed just after launching a browser on a tablet computer. After selecting one of the listed items, not only the detailed information but also the different aspects of related information, such as thumbnails of social movies or private movies, tweets, book & product summaries, will be shown. Related movies can be displayed on different screen devices by a drag and drop action on a tablet computer, by shaking a tablet computer or by speaking words. Detailed information about news topics, SNS topics, and books and products also can be shown on a tablet computer. **Figure 1** shows this usage image. As indicated in Fig. 1, we call such an environment “multi-screen” environment. In this environment, one screen device should be assigned as an operation device by a user like a tablet computer in Fig. 1. We call it a conductor screen device. The other screen devices will be used for showing multi-aspect information. We call them performer screen devices.

In order to enable the multi-aspect information retrieval, we propose a two-phase information retrieval model which contains the first contact and the second contact from users. We propose the system architecture based on Service-Oriented Architecture. This makes it possible the double mashup not only e-services but also screen devices in front of users for the multi-aspect information retrieval and multi-screen usage. We also propose the TSS Framework that is derived from the Model-View-Controller (MVC) pattern to improve system extensibility. In terms of “multimodal user interface,” we applied for advanced Web standardized technologies. We implemented a prototype system based on the TSS Framework as a Web application. We also evaluated the application framework effectiveness through the case study of the prototype system revision experiment covering e-service replacement.



Fig. 1 Multi-screen usage image.

We show the related work in Section 2. We define requirements and two-phase information retrieval model in Section 3 and explain the system architecture and the application framework in Section 4. Then, we show the prototype system outline in Section 5. Finally, we show the TSS Framework effectiveness through the case study and discuss the advantage of the Trend Surfing System based on the TSS Framework in Section 6.

2. Related Work

The purpose of this paper involves “information retrieval” using “multi-screen” environment, so we describe those two issues as related work.

2.1 Information Retrieval

Currently, there are some information retrieval studies that involve giving users several kinds of related information without relying on user’s information retrieval capability. For example, Memorium [7], Discover Apps [8], Goromi-web [9], and InfoSkin [10] are trying to give users some kinds of related information in a passive way.

In Memorium, cards describing keywords move autonomously inside a screen. When one card bumps into another card, a new card describing new keywords will be created. Memorium provides a client application that can be used for browsing detailed information. However, Memorium requires initial keywords. Creation of a new card describing new keywords depends on a random system operation, so that users can not reflect their intentions in the process of new card creation.

Discover Apps can be used for searching iOS applications in a social graph manner. Candidate application icons will emerge after keywords are given. When an icon is tapped, detailed application information appears. Users can also reach similar applications by tracing a social graph. Goromi-web achieves passive web browsing. Categorized web pages appear one at a time, like clouds drifting in the sky. Users can be involved in deciding the drifting direction. This system makes it possible for users to discover fortuitous information through minimum user interaction. These studies enable users to passively retrieve related information by reflecting the user’s intention. However, they need initial keywords.

Infoskin focuses on a window shopping metaphor in the real world. Relevant information items indicated as icons move on a screen. Users can tap or drag and drop these icons to see detailed information or reach different aspects of related information. Infoskin does not compel users to input search keywords because it allows the use of hot words as initial search keywords. However, although Infoskin is a user interface that can be used for passively retrieving related information, it has not been adapted for multi-screen environment nor multimodal user interface mechanism.

The features of the Trend Surfing System are keyword-less searching at the initial stage, reflecting user’s intentions against the information retrieval direction, multi-screen usage, and multimodal user interface. Realizing these four points simultaneously is completely different from existing work. On the other hand, there is no existing work about the application framework aiming for trend information retrieval.

2.2 Multi-screen

Multi-screen systems in the area of TV can be classified into three cases [2]. The first case is when users watch content using screen devices having different specifications. Different specification examples are screen size, CPU power, codec, and network bandwidth. The second case is when users seamlessly watch content over different devices. For example, a user watches content using a smart phone while commuting. After getting home, this user resumes watching the same content using a wide screen TV from the interrupted point. The third case is when users watch main content, like movies, on a main screen device and also simultaneously watch other related contents on different screen devices. For example, when a user is watching a movie scene on a main screen device, advertisements of clothes equivalent to one of an actress are shown on the other screen devices [11].

Our study approach more closely involves the third case. Normally, the third case includes story content, such as movie-oriented content, because the other related content will be modified by the story content's progress. In this case, users can not take part in the story content's progress. On the other hand, in our study, the main screen device's content is trendy information items that can be dynamically selected by users. Then, the other aspects of related content will be displayed on the other screen devices. In this way, we can say that our study approach is the most oriented towards trendy information. In existing studies categorized into the third case, only information related to the story content can be displayed on the other screen devices. However, in our study, users can actively select the desired, trendy information items to see more closely related information, which will be displayed on the other screen devices. That is the difference between existing contributions and our study.

3. Requirements and Information Retrieval Model

In this section, requirements for the Trend Surfing System (Req.1-3) and the application framework (Req.4) are described. Then, the two-phase information retrieval model for satisfying the requirements (Req.1-3) is described.

3.1 Requirements

The research object involves the four requirements listed below.

- Requirement 1: to avoid the “cold start problem,” that concerns the issue of when the system cannot draw any inferences for users about which it has not yet gathered sufficient user information.

For example, users can see trendy information without inputting search keywords.

- Requirement 2: to realize the different aspect information retrieval by reflecting user's intentions.

For example, users can select a desired item from presented trendy information items. Then, different aspects related to the information of the item selected by users are presented. Users can also choose one of such related information items.

- Requirement 3: to show the different aspect information based on a multi-screen environment via multimodal user

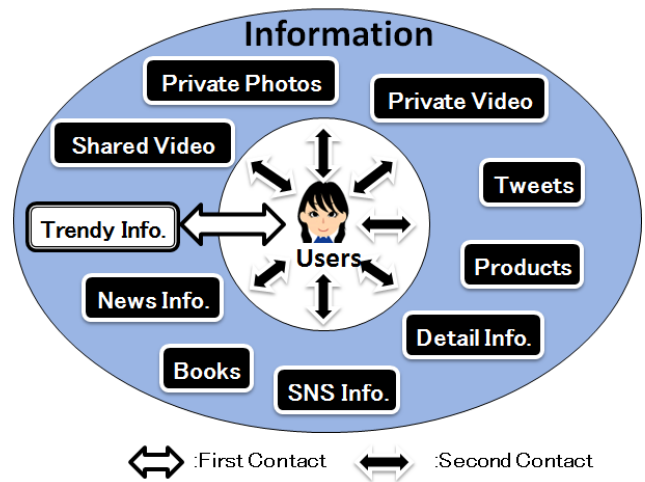


Fig. 2 Two-phase information retrieval model.

interface.

For example, the different aspects related to the information can be placed on multi-screen environment according to user's intuitive operation.

- Requirement 4: to reduce revision cost of the Trend Surfing System in case of changing an e-service.

For example, even if the Trend Surfing System needs to be modified to change an e-service, influenced part should be restricted and total amount of modifications should be slight.

3.2 Two-phase Information Retrieval Model

Figure 2 shows the proposed two-phase information retrieval model. At first, users, indicated in the middle of Fig. 2, can see trendy information items such as hot words, news topics, and SNS topics. In this paper, we call this action the first contact. Users can select a desired trendy item, then different aspect related information candidates will be presented. Users can also choose one of the candidates to retrieve more precise information. We call this action the second contact.

In this model, users do not have to input search keywords to find trendy information at that time. Users only have to view and select one of the presented information items. In this way, users can retrieve details of desired trendy information from several kinds of aspects. Currently, users need to think about search keywords and individually input them into e-services. However, in this model, users do not have to think of search keywords nor input them separately into e-services.

4. Overview of TSS Framework

In this section, we explain the Trend Surfing System architecture behind the TSS Framework and the TSS Framework overview.

4.1 System Architecture

Figure 3 shows the Trend Surfing System architecture behind the TSS Framework. The main feature of this system architecture is that a central control server performs two kinds of combinations (mashup) due to Service-Oriented Architecture. For example, the control server establishes the combination for external e-

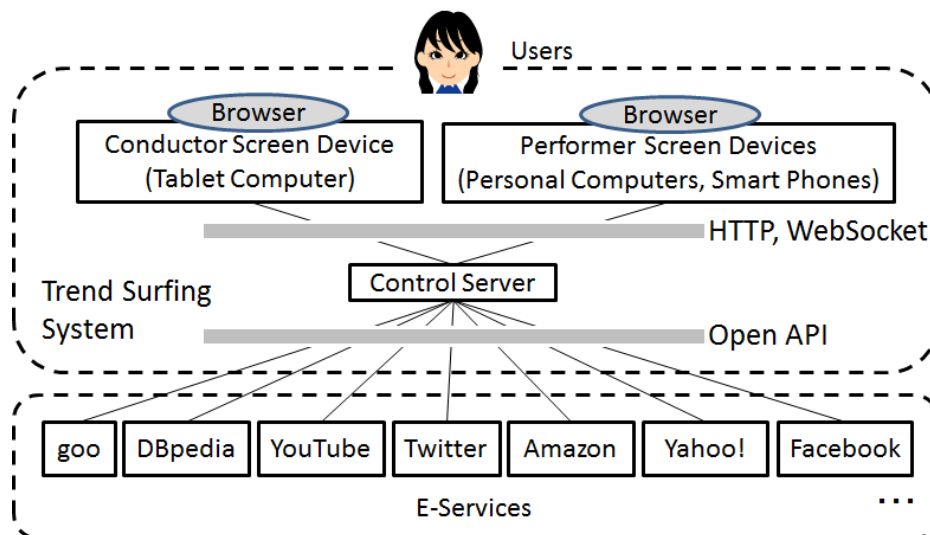


Fig. 3 Trend Surfing System architecture for double mashup.

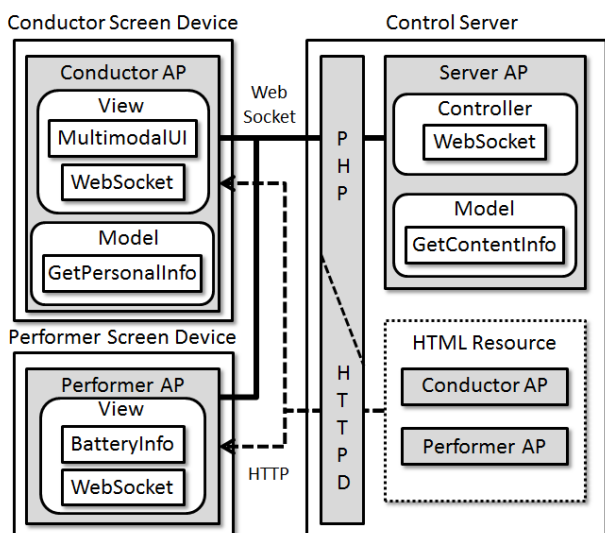


Fig. 4 TSS Framework.

services using open application interfaces (Open API). The control server also combines several screen devices such as a conductor screen device and performer screen devices in front of users. The protocol between the control server and each screen device is HTTP and WebSocket [12], so that heterogeneous screen device combinations can be available just by supposing the browser installing into each device. In this way, the control server allows users to enjoy a multi-screen environment consisting of heterogeneous screen devices surrounding users.

4.2 TSS Framework

Figure 4 shows the TSS Framework. Screen devices operated by users are classified into a conductor screen device and performer screen devices. The conductor screen device is mainly operated by a user. The performer screen devices are used to show many kinds of information. Therefore, on the control server, server application (Server AP) as well as HTML resource including JavaScript for the conductor screen device and the performer screen devices (Conductor AP and Performer AP, respectively)

Table 1 GetContentInfo class method example.

Method Example	Outline
get_hotword	To collect hot words from goo or Yahoo!
get_news	To collect news topics from goo or Yahoo!
get_movie_thumbnail	To prepare movie thumbnails connecting to YouTube.
get_dbpedia	To collect key word outline information from DBpedia.
get_twitter_info	To collect tweets from Twitter.

have been arranged.

The TSS Framework is derived from MVC pattern that can be applied to a wide range of applications on Web relating e-services. A Controller function has been assigned to Server AP. it is implemented by PHP. A Model function has been assigned to Server AP and Conductor AP. They are also implemented by PHP. A View function has been assigned to Conductor AP and Performer AP. They are implemented using JavaScript.

The Controller of Server AP includes WebSocket class library to communicate with Conductor AP or Performer AP by exchanging a message with identification number. In the TSS Framework, we used php-websocket-server 2.1.0 [13] as WebSocket class library for Web servers. The Model of Server AP has GetContentInfo class library to obtain trend relating information from external e-services. We implemented GetContentInfo class library using open API for external e-services. A part of methods of GetContentInfo class is shown on Table 1. The Controller of Server AP calls methods of GetContentInfo class to retrieve information from external e-services and sends the View of Performer AP a message with identification number to display information according to a user operation via the View of Conductor AP. The Model of Conductor AP has GetPersonallInfo module to retrieve personal information from SNS e-services like Facebook. In order to retrieve personal information from SNS, we need to take login process, so that we assigned this function

to Conductor AP as a front end application. We implemented GetPersonalInfo module using SNS open API.

The conductor screen device and the performer screen devices allow HTTP access to the control server to obtain HTML resources with JavaScript, such as Conductor AP and Performer AP, respectively, via each browser. The View of Conductor AP or Performer AP includes WebSocket class prepared by a browser to communicate with the Controller of Server AP by exchanging a message with identification number. The View of Conductor AP has also a multimodal user interface (MultimodalUI) which can be used for intuitive user interfaces against a user like a drag and drop operation by a touch sensor, a motion operation by an acceleration sensor or a voice operation by speech recognition. The motion operation has been implemented using DeviceOrientation Event Specification [14] which is an acceleration sensor API for HTML5. The voice operation has also been implemented using Web Speech API Specification [15] which is a speech recognition API for HTML5. All these multimodal user interface functions have been implemented by JavaScript. The view of Performer AP has also an ability to detect battery information (BatteryInfo) besides information displaying functionality according to message commands of the Controller. The performer screen devices might be a battery-operated device in some cases. Then, we implemented the function to regularly notify Conductor AP about remaining battery quantity via WebSocket based on Battery Status API [16] by JavaScript.

5. Implementation Example Using TSS Framework

We implemented a trend surfing prototype system as an example using the TSS Framework. The prototype system specification is shown in Fig. 5. In terms of the control server, it was constructed in two ways. One was to set up the control server on the local area network of our laboratories. The other was to set up the control server on Amazon Web Services (AWS) [17] cloud system. We confirmed that both ways were applicable for prototype system evaluation.

5.1 Process Flow of the Prototype System

The process flow of the prototype system contains three procedures, described below. The state transition diagram, which describes the process flow, is shown in Fig. 6.

1) *Gathering and displaying first contact items (1st Contact Items)*: After executing Conductor AP on the conductor screen device, the trendy information of first contact items will be displayed. Specifically, the Model of Server AP collects hot words from “goo,” a Japanese search engine portal site, news topics from Yahoo!. The Model of Conductor AP collects SNS topics from Facebook after login process using Facebook API [18]. Goo and Yahoo! are providing the keyword ranking information [19] as RSS (Rich Site Summary) and News Web API [20], respectively. Concerning SNS topics, only top 10 posted messages sorted in reverse order of “likes” were considered, because the number of “likes” might reflect the interest rate among friends of the target user. In terms of the first contact item information image extracted via API of “goo,” “Yahoo!,” and “Facebook,” they

Control Server	Conductor/Performer Screen Devices
<ul style="list-style-type: none"> •Linux CentOS 6.4 •Apache 2.2.15 •PHP 5.3.3 •WebSocketServer 2.1.0 	<ul style="list-style-type: none"> •iPad Mini, iOS 7.0.2, Safari •Windows PC, Win7, 8, Chrome •iPod Touch, iOS 7.0.3, Safari •Windows Tablet, Win8, Chrome •Nexus7, Android 4.3, Chrome

Fig. 5 Prototype system specification.

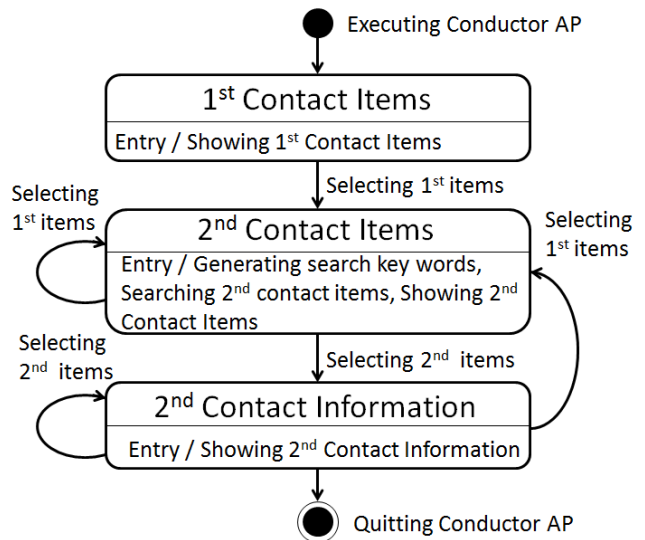


Fig. 6 State transition diagram.

are all text images. For example, hot words from “goo” are just top 16 searching key words. News topics from “Yahoo!” are 8 news titles with time stamp in each category like Topics, Economy, Entertainment, and Sports. SNS topics from “Facebook” are only the beginning part (50 bytes) of each top 10 posted messages, user name, and “likes” number. Then, these information items will be displayed on the conductor screen device as first contact items (at the upper left of Fig. 9).

2) *Gathering and displaying second contact items (2nd Contact Items)*: When one of first contact items is selected by a user, a search keyword will be automatically generated to collect second contact items from external e-services. Then, second contact items will be displayed on the conductor screen device. Specifically, the Model of Server AP extracts a search keyword from the selected first contact information using the key phrase extraction Web API provided by Yahoo! Developer Network [21]. Due to this auto-generated search keyword, the Model of Server AP will collect second contact items from DBpedia, YouTube, Twitter, and Amazon using each open API [22], [23], [24], [25] and display them on the conductor screen device.

3) *Gathering and displaying the selected, second contact item’s detailed information (2nd Contact Information)*: This process collects and displays detailed information about selected, second contact items. Specifically, the Controller of Server AP distributes the second contact item’s detailed information to the View of Conductor AP or Performer AP, which will be shown on the same tab or a new tab of the conductor screen device browser, or different performer screen devices’ browsers. Output destination decision of the detailed information depends on its media

type, like texts, pictures, and movies, because of the viewpoint of the user’s visibility (Fig. 7). Different second contact items can also be redisplayed by selecting different first contact items.

Media Type \ Destination	Conductor Screen Device		Performer Screen Device
	Same Tab	New Tab	
Text (DBpedia, Twitter)	○	—	—
Text, Pictures (Amazon, Yahoo, Facebook)	—	○	—
Movies (YouTube)	—	—	○

Fig. 7 Output destination decision.

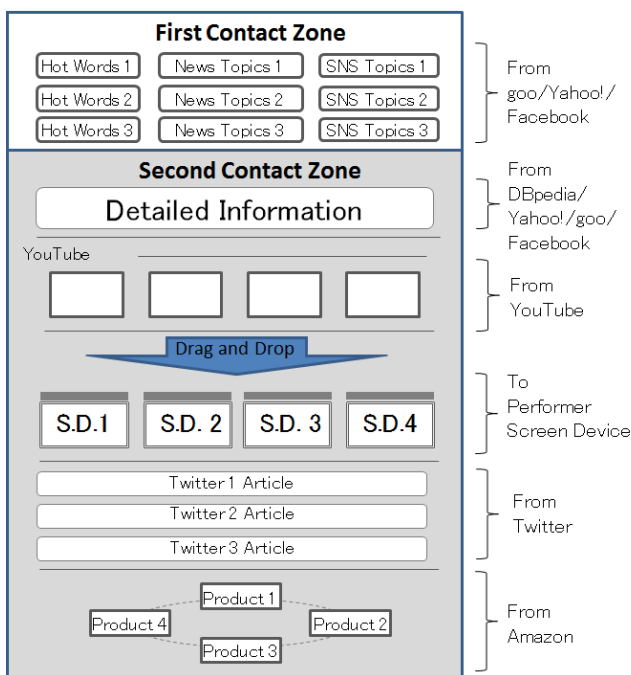


Fig. 8 Design diagram of conductor screen device user interface.

5.2 User Interface

Figure 8 shows a design diagram of the conductor screen device user interface. The proposed user interface is divided into an upper part, called the first contact zone, and a bottom part, called the second contact zone. In the first contact zone, first contact items collected by the Model of Server AP are displayed according to the first procedure described in the previous section. On the other hand, in the second contact zone, second contact items collected by the Model of Server AP are shown according to the second procedure described in the previous section.

Almost all operations can be performed by voice as well as tapping or dragging and dropping using fingers or a mouse. If the conductor screen device contains an acceleration sensor, YouTube movies following the third procedure described in the previous section can be shown on the View of Performer AP on the performer screen devices by shaking the conductor screen device. If there were some performer screen devices, performer screen device selecting turn would be determined according to the order of connecting time from the performer screen device to the control server via WebSocket.

5.3 Practical Experiment

Figure 9 shows the user interface image of the View of Conductor AP, and Fig. 1 shows the multi-screen usage image. We confirmed that the prototype system could extract first contact items without inputting keywords. We also confirmed that second contact items were changed within seconds to new ones just after tapping or speaking the first contact item on the conductor screen device as indicated in Fig. 9. In this way, we could realize a change and educe different aspects related to the information in succession according to the user’s successive tapping or speaking of desired first contact items. We explain three kinds of specific scenarios that focus on a Japanese young singer case, a Japanese popular TV drama case, and an unknown key word embedded in Facebook message case as trendy information.

In terms of a Japanese young singer case, when you look at hot words displayed on the conductor screen device at the up-



Fig. 9 Conductor screen device user interface image.

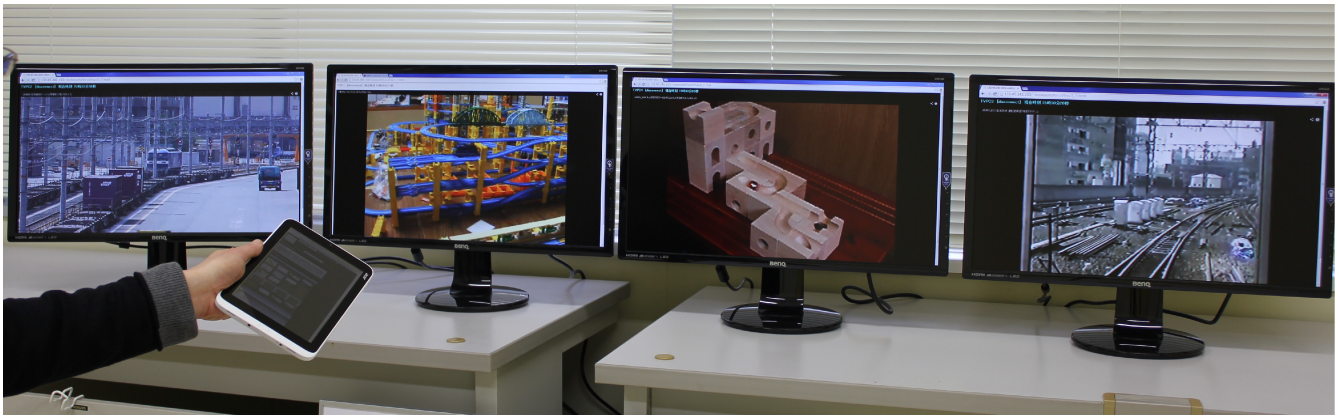


Fig. 10 Trend Surfing example by shaking conductor screen device.

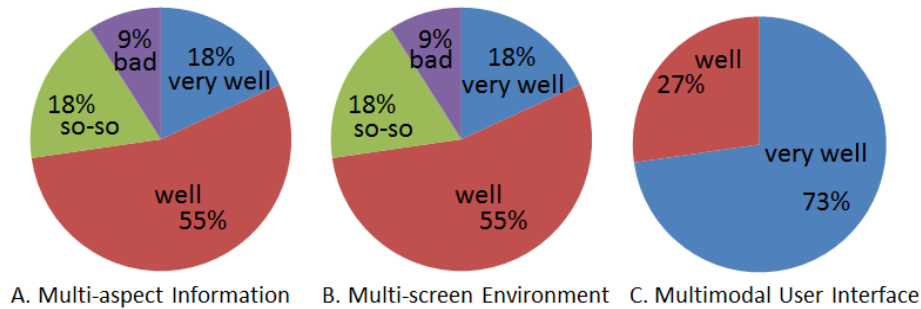


Fig. 11 Subjective evaluation results of Trend Surfing System.

per left of Fig. 9 as the first contact item, you are interested in the hot word of “Tomomi Itano” who is a Japanese young singer. You just tap the hot word of “Tomomi Itano,” then profile information, promotion video thumbnails, tweets, and DVDs or books of “Tomomi Itano” will be displayed on the rest part of the conductor screen device. In this way, you can overlook the relating information about “Tomomi Itano” using the conductor screen device.

Figure 1 presents a Japanese popular TV drama case. When you are interested in the hot word of “Flying Public Relations Office” that is a Japanese popular TV drama title as the first contact item. You just tap it, and then you can see not only text image outline information or tweets on the conductor screen device but also relating videos on the performer screen devices. The different scenes could be shown on a Windows PC, iPod Touch, Nexus 7 as the second contact information. The promotional video of the TV drama theme song could also be shown simultaneously on a different performer screen device (note PC in Fig. 1). These operations could be completed by dragging and dropping or speaking second contact items onto an iPad Mini or a Windows Tablet as the conductor screen device. Due to the performance of this kind of trend surfing by users, it has been confirmed that users could grasp different aspects of the TV drama’s associated information, such as stories, heroes or heroines, and theme songs, even if users have never seen such a TV drama.

Figure 10 shows an unknown key word embedded in Facebook message case. In this case, the auto-generated keyword, “three-dimensional switchback,” has been extracted from the tapped Facebook posted message on the conductor screen device. A few YouTube movies searched by “three-dimensional switchback” could be simultaneously shown just by shaking the conductor screen device. In this manner, concerning user performance of

this kind of trend surfing, not only actual trains could be displayed during live filming, but also toy train movies could be displayed simultaneously. By this experiment, it has been confirmed that users could correctly grasp the meaning of “three-dimensional switchback” rather than seeing only one aspect movie.

5.4 Subjective evaluation

We evaluated the proposed system subjectively from the viewpoint of Multi-aspect, multi-screen and multimodal user interface on 15th, 16th, April, 2014. We asked 11 people (men:8, women:3) who are 4th grade undergraduate or graduate students of Computer and Information Science Program, School of Engineering Nagasaki University to be the volunteer examinees. After using this proposed system to see news topics as trendy information, we asked them the following questions.

- A: Is it useful for you to understand your selected first contact item by watching the multi-aspect information?
- B: Is it useful for you to understand your selected first contact item by using multi-screen environment?
- C: Does the multimodal user interface of this proposed system have any benefit for increasing accessibilities?

We asked the volunteer examinees to answer five-level evaluation, “very well,” “well,” “so-so,” “bad,” and “very bad.” Figure 11 shows the evaluation result.

- A: Against the question about usefulness of the multi-aspect information, 73% examinees answered that it was very well or well.
- B: Against the question about usefulness of the multi-screen environment, 73% examinees answered that it was very well or well.
- C: Against the question of the accessibility improvement due

Table 2 Amount of modifications applied to E-service replacement.

Application	Total lines	Part	Line Re-removed	Line Changed	Line Added	Total Modifications
Server AP	453	Controller	0	0	0	0 lines, 0%
		Model	0	55	24	79 lines, 17.4%
Conductor AP	919	View	9	0	0	9 lines, 1.0%
		Model	0	0	0	0 lines, 0%
Performer AP	108	View	0	0	0	0 lines, 0%
Total	1,480		9	55	24	88 lines, 5.9%

to the multimodal user interface, 100% examinees answered that it was very well or well.

6. Case Study and Discussion

In this section, we study how the TSS framework can be applied to the system modification that is replacing the e-services of the prototype system with new e-services. We also discuss total evaluation results.

6.1 Case Study

In terms of the prototype system, we chose “goo,” a Japanese search engine portal site, to extract hot words and “Yahoo!” to retrieve news topics. Both portal sites provide hot words and news topics by RSS. Therefore, as a case study of replacing e-services based on the TSS framework, we tried to change both e-services each other. Those are to take “Yahoo!” to extract hot words [26] and to take “goo” to retrieve news topics [19]. In this case study, we were leading with a 4th grade undergraduate student of Computer and Information Science Program, School of Engineering Nagasaki University. He had an experience of C language programming, but he was a beginner of PHP or HTML5 programming. Before starting this case study, we explained the TSS Framework (Fig. 4) to him. We also gave him needed information like programming knowledge or the TSS Framework features during modification. Then, after finishing modification, we reviewed changed source codes.

Table 2 shows the modification result applied to this replacement experiment. From Table 2, we can see that modification part has been almost concentrated at the method of “get_hotword” and “get_news” shown on Table 1 in the Model of Server AP. The main modification reason for the Model of Server AP is to handle different RSS formats and providing new data. The total amount of modifications is less than 20% within Server AP, also less than 6% against total system code amount. This result indicates good evidence that the TSS framework works well for reducing the revision cost of the Trend Surfing System in case of changing an e-service (Requirement 4).

6.2 Discussion

We implemented the Trend Surfing System prototype based on the TSS framework. The Trend Surfing System itself focuses on the issue of information retrieval capabilities and aims to create a fair information retrieval system independent of user’s informa-

tion technology literacy. Until recently, the most popular method to retrieve trendy information was the newspaper. The general newspaper approach is listed in the steps below.

- Identifying desired news articles by looking over all headlines.
- Reading identified news article

These steps correspond to the first and second contact described in two-phase information retrieval model in this paper. In short, this prototype system follows the same pattern as information retrieval from newspapers. The newspaper is read all over the world and the problem of information retrieval capability differentials among users has not occurred. It is suggested that the prototype system can solve the problem of information retrieval capability differentials.

Moreover, in the case of newspapers, one headline corresponds to one article. However, the prototype system can simultaneously show different aspects of related contents on different performer screen devices via multimodal user interface. The cases of the popular Japanese TV drama and unknown keyword “three-dimensional switchback,” mentioned in the previous section, present good examples. This means that the prototype system not only follows but also extends the newspaper’s pattern of use. For this reason, the prototype system is expected to not only retrieve trendy information independent of user’s information technology literacy but also understand the meaning of the information itself.

Through the practical experiment, the defined requirements 1-3 in Section 3 have been satisfied from the viewpoint of functionality. We also confirmed that user experience perspective of multi-aspect, multi-screen, and multimodal user interface relating to the requirements 2 and 3 had also good evaluation results. On the other hand, Web technologies and e-services like Web services will be improved day by day. User’s demand will also be changed day by day. In order not to become the system obsolete and to follow user’s demand changing, we need to apply excellent external technologies or services continuously. Therefore, we proposed the TSS Framework derived from MVC pattern to improve system extensibility. According to the system modification case study, we also demonstrated that the TSS framework made influenced parts restricted and made total modification line amount decreased. In this mean, the system based on the TSS Framework would be improved easily mainly by modifying the Model of Server AP to follow new Web service API if external

Web services were improved or emerged as new ones. This could contribute the sustainability of the Trend Surfing System and also satisfy requirement 4.

In these results, we could say that all requirements for the Trend Surfing System and the application framework have been satisfied.

7. Conclusion

This paper focused on the extensible application framework, the TSS Framework, for the Trend Surfing System that closes the information retrieval capability gap. According to the usage of the prototype of the Trend Surfing System, users can deduce different aspects related to the information one after another just after pointing out the desired trendy information item. Users can also distribute the related information to the different screen devices around users through an intuitive user interface like surfing across different screen devices. This prototype system could be expected to not only retrieve trendy information independent of a user's information technology capabilities but also understand the meaning of the retrieved information itself.

We also demonstrated the effectiveness of the TSS Framework in case of modification through the case study. From this result, we could expect that the Trend Surfing System based on the TSS Framework would have sustainability. The Trend Surfing System helps users understand new information. In this sense, The Trend Surfing System is well-suited an education support system. Therefore, we will try to transform the TSS Framework to the framework of an education support system in our future work.

Acknowledgments This work was supported by JSPS KAKENHI Grant Number 25880016.

References

- [1] Peng, T. and Slevinsky, J.: Multi-screen IPTV: Enabling technologies and challenges, *Proc. 2011 IEEE International Conference on Consumer Electronics (ICCE)*, pp.1–2, Las Vegas, USA (Jan. 2011).
- [2] Yoon, C., Um, T. and Lee, H.: Classification of N-Screen Services and its standardization, *Proc. 2012 14th International Conference on Advanced Communication Technology (ICACT)*, pp.597–602, PyeongChang, Korea (Feb. 2012).
- [3] Jang, J., Nam, H. and Kim, Y.: Mobile device-controlled live streaming traffic transfer for multi-screen services, *Proc. 2012 International Conference on Information Networking (ICOIN)*, pp.415–420, Bali, Indonesia (Feb. 2012).
- [4] Ohmata, H., Takechi, M., Mitsuya, S., Otsuki, K., Baba, A., Matsumura, K., Majima, K. and Sunasaki, S.: Hybridcast: A new media experience by integration of broadcasting and broadband, *2013 Proc. ITU Kaleidoscope: Building Sustainable Communities (K-2013)*, pp.1–8, Kyoto, Japan (Apr. 2013).
- [5] Kobayashi, T.: A Proposal of Smart TV System focused on Findability, *2013 IEEE 2nd Global Conference on Consumer Electronics (GCCE)*, pp.507–508 (2013).
- [6] Kobayashi, T.: Information Orchestration System for Closing the Information Retrieval Capability Gap, *Proc. 2014 IEEE 38th Annual International Computers, Software and Applications Conference (COMPSAC2014)*, pp.644–645 (2014).
- [7] Watanabe, K. and Yasumura, M.: Memorium: The concept of persistent interface and its prototype, *Workshop on Interactive Systems and Software, WISS2002*, pp.99–104 (Nov. 2002).
- [8] Discover Apps, available from <http://discovr.info/> (accessed 2013-05-31).
- [9] Otsubo, G.: Goromi-Web browsing for unexpected information on the web, *Proc. 6th ACM SIGCHI Conference on Creativity & Cognition*, pp.267–268 (2007).
- [10] Seko, S., Aoki, R., Ihara, M. and Kobayashi, T.: InfoSkin: User Interface for Passive Information Browsing, *Proc. 2012 Joint Conference of the International Industrial Information Systems Conference & the International Conference on Computers, Communications and Systems*, pp.3–4, Chiang Mai, Thailand (Dec. 2012).
- [11] Lim, T.-B., Kim, K.W., Lee, S.P. and Yoon, K.: Real-time adaptive advertising framework based on MPEG-21 for multi-screen IPTV environment, *2012 IEEE International Conference on Consumer Electronics (ICCE)*, pp.13–16 (Jan. 2012).
- [12] The WebSocket API, available from <http://www.w3.org/TR/2011/WD-websockets-20110929/> (accessed 2014-05-08).
- [13] php-websocket-server 2.1.0, available from <https://github.com/memory-agape/php-websocket-server> (accessed 2014-12-24).
- [14] DeviceOrientation Event Specification, available from <http://dev.w3.org/geo/api/spec-source-orientation.html> (accessed 2014-05-08).
- [15] Web Speech API Specification, available from <https://dvcs.w3.org/hg/speech-api/raw-file/tip/speechapi.html> (accessed 2014-05-08).
- [16] Battery Status API, available from <http://www.w3.org/TR/battery-status/> (accessed 2014-05-08).
- [17] Amazon web services, available from <http://aws.amazon.com/> (accessed 2014-08-27).
- [18] Facebook API, available from <https://developers.facebook.com/docs> (accessed 2014-08-27), Keyword ranking information as RSS provided by goo, available from <http://www.goo.ne.jp/rss/> (accessed 2014-08-27).
- [19] Keyword ranking information as RSS provided by goo, available from <http://www.goo.ne.jp/rss/> (accessed 2014-08-27).
- [20] News Web API provided by Yahoo! JAPAN, available from <http://developer.yahoo.co.jp/webapi/news/> (accessed 2014-08-27).
- [21] Key phrase extraction Web API, available from <http://developer.yahoo.co.jp/webapi/jlp/keyphrase/v1/extract.html> (accessed 2014-08-27).
- [22] DBpedia API, available from <http://dbpedia.org/About> (accessed 2014-08-27).
- [23] Youtube API, available from https://developers.google.com/youtube/getting_started?hl=en (accessed 2014-08-27).
- [24] Twitter API, available from <https://dev.twitter.com/> (accessed 2014-08-27).
- [25] Amazon API, available from <http://docs.aws.amazon.com/AWSECCommerceService/latest/DG/Welcome.html> (accessed 2014-08-27).
- [26] Daily Ranking provide by Yahoo! JAPAN, available from http://searchranking.yahoo.co.jp/burst_ranking/ (accessed 2014-12-24).



Toru Kobayashi is a professor at the Division of Electrical Engineering and Computer Science, Graduate School of Engineering, Nagasaki University. He received Ph.D. (2011) in engineering from the University of Electro-Communications. He has worked for NTT (1987–2013). He is a member of the IEEE Computer Society, and a senior member of IEICE.