Development of User Interface for Multiple Devices Connecting Path Planning System for Bus Network

Takahiro Takayama, Takao Kawamura, Toshihiko Sasama, and Kazunori Sugahara

Abstract—Recently, web services to access from many type devices are often used. We have developed the shortest path planning system called "Bus-Net" in Tottori prefecture as a web application to sustain the public transport. And it used the same user interface for both devices. To support both devices, the interface cannot use JavaScript and so on.

Thus, we developed the method that use individual user interface for each device type to improve its convenience. To be concrete, we defined formats of condition input to the path planning system and result output from it and separate the system into the request processing part and user interface parts that depend on device types. By this method, we have also developed special device for Bus-Net named "Intelligent-Bus-Stop".

Keywords—Bus, Path planning, Public transport, User interface

I. INTRODUCTION

RECENTLY, web services to access from many type devices are often used. For example, "Google Calendar" [1] can be use from a PC at home or office, a mobile phone at anywhere and so on. These services can use advantages of each device due to the user interface is arranged according to the device type.

We have developed the shortest path planning system called "Bus-Net" [2] in Tottori prefecture as a web application to sustain the public transport. "Bus-Net" is assumed to use from PC and mobile phone. And it used the same user interface for both devices. To support both devices, the interface cannot use JavaScript, and it has limited space to display for mobile phone. For this reason, characteristics of the device are not utilized when users access "Bus-Net" from PC. Moreover, it was difficult to support new type devices, because to need remodeling of this user interface for limitations from new devices. Thus, we developed the method that use individual user interface for each device type to improve its convenience. To be concrete, we defined formats of condition input to the path planning system and result output from it and separate the system into the request processing part and user interface parts that depend on device types. The individual user interface for a new type device can be added easily by this method that we developed.

By this method, we have also developed special device for Bus-Net named "Intelligent-Bus-Stop". It put on public spaces like a station and it use touch panel for interface. On account of using touch panel, the person who is weak in operation of computer such as the mobile phone can use "Bus-Net".

T. Takayama is with ME student in Graduate School of Engineering, Tottori University.

T Kawamura, T Sasama and K Sugahara are with Graduate School of Engineering, Tottori University.

II. PATH PLANNING SYSTEM

Path planning function [3], [4] is a main function of "Bus-Net". Our path planning method can find paths including not only bus transfers but walking transfers.

In this system, a user selects departure-landmark and destination-landmark for searching paths. Landmarks are registered into the "Bus-Net" database, and they are not only bus stops and railroad stations, but also sightseeing spots, shops, and so on. Now, about 10,000 landmarks are registered. Moreover, the place of departure and destination can be specified by using the "Google Maps" [5]. Thus, even if people can use this system who doesn't know bus stop names or bus stop positions, they can use a bus on a path. The system has many unique aspects such as an original path searching algorithm, however, they are not referred in this paper. Fig.1 shows the result of path planning.

This is previous user interface used from both PC and mobile phone. We thought that in many cases, this system was may be used from PC or mobile phone. The site of Bus-Net has not utilize full ability of PC's web browsers. Because it also has to support mobile phone. For example, Browser on mobile phone lacks Javascript and output information is reduced for small resolution display. Thus, we develop optimized user interfaces for each device.
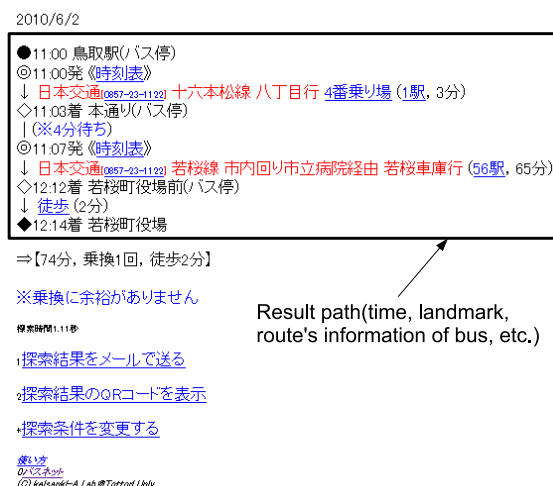

Fig. 1.   Result of path planning(previous user interface)

III. User Interface corresponding to the device

In our previous study, the path planning system of "Bus-Net" outputs results directly by HTML format. This HTML results are dependent on device type. For control of multiple devices, we have separated the system into the request processing part and user interface parts, and the request processing part outputs results by XML format that are not dependent on device type. For each device type, the corresponding user interface part are prepared, and each user interface part outputs optimized HTML or other formats from inputted XML results.

A. Define I/O format of path planning program

We defined formats of condition input to path planning program and result output from it for control of multiple devices.

Set of CGI parameter is used for condition which is an input to a path planning program. And, XML is used for result which is an output from a path planning program. As an advantage, the device using "Bus-Net" only follows this format, each device can design the user interface freely. Moreover only follows this format, it can support a new device easily. Fig.2. shows this concept.
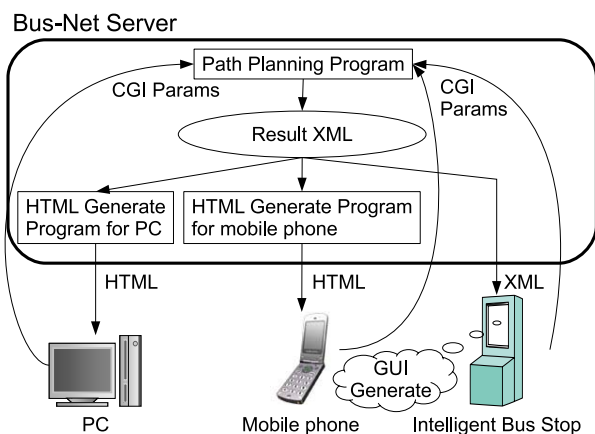

Fig. 2.   Concept figure of support multiple device

1) Input(path planning condition): Set of CGI parameter consists of key and value is used for condition which is an input to a path planning program, because in order to make request from the mobile phone. Since mobile phone can't use JavaScript, it is difficult to generate the path planning condition dynamically as XML. Moreover it has many disadvantages to send path planning condition by XML, for example, increase of communication charge. Thus, path planning condition is expressed using set of CGI parameter which is simply expression. Fig.3 shows set of CGI parameter which is path planning condition.

2) Output(path planning result): XML is used for result which is an output from a path planning program. Because XML is suitable to express path planning result. Moreover the library parsing XML is substantial generally, XML can be used at multiple devices. Fig.4 shows XML which is path planning result.

```
search.cgi?s_id=1791&d_id=3541&perpage=7&map=&wmap=&
timetype=just&after=10&date=6%2C9&hour=11&min=0&dir=
forward&train=with&bus=hinomaru&bus=nikkou&target=
%8Co%98H%92T%8D%F5
```

Fig. 3.   Set of CGI parameter of path planning condition

```
<?xml version='1.0' charset="UTF-8"?>
<Result>
 <Date>
  <Year>2010</Year>
  <Month>6</Month>
  <Day>2</Day>
 </Date>
 <Path>
  <Start>
   <Time type='minute'>11:00</Time>
   <Anchor>鳥取駅(バス停)</Anchor>
  </Start>
  <Departure>
   <Time type='departure'>11:00</Time>
   <Schedule>../timetable/timetable.cgi?target=%BB%FE%B9%EF
%C9%BD%C9%BD%BC%A8&amp;date=6,2&amp;s_id=1791&
amp;d_id=2077&amp;now=660</Schedule>
   <Company>
    <CName>日本交通</CName>
    <Tel>0857-23-1122</Tel>
   </Company>
   <Line>十六本松線 八丁目行</Line>
   <PlatForm>
    <Picture>platform.cgi?file=platform%2Fpc%2F1791-4.png
</Picture>
    <Number>4</Number>
   </PlatForm>
   <StationList>rosen.cgi?table=174&amp;connect=3594&amp;
now=660&amp;num=1</StationList>
   <StationNum>1</StationNum>
   <Time type='minute'>3</Time>
  </Departure>
…
<Result>
```

Fig. 4.   XML of path planning result

B. Kind of devices

Devices which is available "Bus-Net" is given below.

1) Mobile phone: Mobile phone version assumes the use at anywhere. User can confirm a path at near bus stop. It is difficult for mobile phone to parse XML of path planning result and generate user interface. Thus, the server of "Bus-Net" generates HTML of mobile phone version and sends to a mobile phone. Fig.1 shows user interface of mobile phone version.

2) PC: PC version assumes the use at home or office. Since mobile phone has limited space to display, user interface of previous version displayed only the minimum required information. Moreover since it didn't use JavaScript, when the setting of departure place and the destination got new information from a server, there was an inconvenient point that it was necessary to do reloading of a Web page each time.

Thus, user interface of PC new version attempted the improvement of the visuality by using the table, the icon, and so on. Moreover the setting of the place of departure and destination made it open the sub window by using JavaScript

and do. As a result, the input of the condition came to be possible without reloading of a Web page. Since PC version assumes the use from a Web browser as well as a case of mobile phone version, the server of "Bus-Net" generates HTML of PC version and sends to a PC. Fig.5 shows user interface of PC new version.

Fig. 5.   Result of path planning(PC user interface)

3) Intelligent-Bus-Stop: "Intelligent-Bus-Stop" assumes a person not good at the operation of the equipment such as mobile phones and the person who doesn't know the existence of "Bus-Net" because he visited Tottori Prefecture for the first time. "Intelligent-Bus-Stop" is implemented by using the "Microsoft Visual Studio C#" [6]. Because it can do various processing to improve operationally and visuality. Thus, a server sends XML of path planning result directly, in order to perform display processing by the "Intelligent-Bus-Stop" side. We will take up detail of "Intelligent-Bus-Stop" in the next chapter.

4) iPhone: "Intelligent-Bus-Stop" is a device that attached a high value to operationally and visuality. Thus, we thought that "Bus-Net" could be used more conveniently, when making carrying possible after maintaining such character. We think that we may become portable at operationally and visuality characteristics if we use iPhone.

5) iPad: iPad is portable device and it has more spread screen than iPhone's screen. Thus, we thought that iPad version can be used when not only user of the bus using the "Bus-Net" but also the staffs of the municipality and the bus company explain the path of the bus to the senior citizen and the tourist. We develop the iPad version in parallel with iPhone version.

IV. INTELLIGENT-BUS-STOP

We made new types of device to prove a new kind of device can be added by the method that we were able to the device made individual user interface available. The device name is "Intelligent-Bus-Stop". "Bus-Net" used from PC or mobile phone. But a person didn't use "Bus-Net" who doesn't good at the operation of the equipment such as mobile phones and doesn't know the existence of "Bus-Net" because it visited Tottori Prefecture for the first time use it. Thus, as a device of a new kind, we developed a device of the indoor setting type for the purpose of the setting to bus terminals by the name of "Intelligent-Bus-Stop". A characteristic of "Intelligent-Bus-Stop" includes the point that "Bus-Net" is available to using touch panel.

Thus, the person who is weak in operation of the machinery such as the mobile phone can use "Bus-Net". Moreover, because we assume that "Intelligent-Bus-Stop" is installed in the place where a lot of people gather including the bus terminal, the tourist etc. who do not know the existence of "Bus-Net" either can use it.

A. Construction

We constructed "Intelligent-Bus-Stop" by normal PC set in an exclusive chassis. TABLE I shows the spec of PC.

TABLE I
SPEC OF PC

CPU	Core 2 Duo E8500 3.16GHz
Mother board	ASUS P5KPL-CM
Memory	2GB
HDD	SATA 320GB

"Intelligent-Bus-Stop" uses touch panel of 19 inches as an I/O device. Fig.6 shows the face of "Intelligent-Bus-Stop". System's OS uses "Windows". "Intelligent-Bus-Stop" is implemented by using the "Microsoft Visual Studio C#". We made the user interface that considered operationally and visuality.

Fig. 6.   Intelligent-Bus-Stop

B. User interface

User interface of "Intelligent-Bus-Stop" is given below. Program of Intelligent-Bus-Stop" is if only processed according to the I/O form shown in section III. Thus, "Intelligent-Bus-Stop" can make a user interface freely taking advantage of the characteristic of a device.

1) Input landmark name by "Incremental Search": When the departure-landmark(destination-landmark) is input from the name, input the landmark name set to the departure-landmark(destination-landmark). And, the user chooses the landmark from a displayed candidate and decides it. When PC is used, a comparatively long landmark name can be easily input with a keyboard. But, since the operation number of times increases, it is hard for user to input a long landmark name with the device like the "Intelligent-Bus-Stop".

Thus, we use a search method called "Incremental Search". This is the search method how a candidate of landmarks is updated when user input a hiragana of landmark. Thus, even if all hiraganas are not input, the user can decide the landmark. It is not necessary to input a hiragana from a head. Even if the hiragana on the way is input, it is possible to search landmark. Fig.7 shows "Incremental Search".

The input landmark is sent to the server as sets of CGI parameters shown in section III.

Fig. 7.   Input landmark name by "Incremental Search"

2) Input position by "Google Maps": When the user input from a map, he input from a map screen using "Google Maps". The screen shown in Fig.8 uses Web a browser. And the user decides the position by using this. Since "Intelligent-Bus-Stop" uses touch panel, the user can decide the position to be set by touching directly. Since it can be directly operated with a finger, it is easy to use for a person who is weak in operation of device like a mouse.

The input position is sent to the server as set of CGI parameters as well as an "Incremental Search".

3) Display of path planning result: When it finishes inputting all conditions necessary for the path planning, "Intelligent-Bus-Stop" send the condition of inputting it to the server of the "Bus-Net" as set of CGI parameters. The path planning is done with the server, and the result is returned as XML. The "Intelligent-Bus-Stop" picks up necessary information from XML. And it displays path planning result easy to see. Fig.9 shows it example. Moreover it displays a QR cord expressing URL of "Bus-Net" which displays the result that

Fig. 8.   Input position by "Google Maps"

the user searched for in the same condition on the left of the screen. The user read this with the camera of the mobile phone and can confirm the same search result from a mobile phone by accessing it. Since the "Intelligent-Bus-Stop" cooperates with the mobile phone, the user gets on the bus actually, a path can be confirmed. Thus, this leads to improvement of the convenience.

Fig. 9.   Path planning result of the "Intelligent-Bus-Stop"

V. CONCLUSION

In this paper, we have developed user interfaces for multiple devices using "Bus-Net", and we have also developed new special device named "Intelligent-Bus-Stop" that use a touch panel to input interface and it puts on public spaces like a station.

In our previous study, the path planning system of "Bus-Net" outputs results directly by HTML format. This HTML results are dependent on device type. For control of multiple

devices, we have separated the system into the request process-ing part and user interface parts, and the request processing part outputs results by XML format that are not dependent on device type. For each device type, the corresponding user interface part are prepared, and each user interface part outputs optimized HTML or other formats from inputted XML results.For optimization example, the user interface of "Intelligent-Bus-Stop" has a function of "Incremental Search" that support users not good at the input of computers. Now, we are developing new user interfaces for iPhone and iPad. Like this, web services need to response new devices that put on the market one after another. In this situation, our approaches to separate the part of user interface from the system will be widely used.

REFERENCES

[1] "Google calender http://www.google.com/calendar/render."
[2] "Bus-net http://www.ikisaki.jp/."
[3] T. Kawamura, G. Kusugami, and K. Sugahara, "Path planning system for bus network including walking transfer," *IPSJ Journal*, vol. 46, no. 5, pp. 1207–1210, 5 2005.
[4] T. Kawamura and K. Sugahara, "Practical path planning system for bus network," *IPSJ Journal*, vol. 48, no. 2, pp. 780–790, 2 2007.
[5] "Google maps http://maps.google.co.jp/."
[6] "Microsoft Visual Studio http://www.microsoft.com/japan/msdn/vstudio/."