

DX-BASED NAVIGATION LINKED WITH KYOTO CITY BUS LOCATION SYSTEM FOR NATIONAL TREASURE VISITORS

Hiroshi Sunaga[†]

[†]Osaka Institute of Technology

Mako Kawasaki^{††}

^{††}ex-student

ABSTRACT

Many tourists take the city bus to visit national treasure sites in Kyoto. In this study, we propose an application that can manage tourist spot and bus service information. With our application, it is possible to access the same information offered by bus companies by simply touching the bus stop on the street map or on the bus route map displayed on the phone. In addition, GPS displays the user's present location on the map, which makes it easier to tell where the nearest bus stop is. It lets them know if there is a bus route that goes directly to their destinations or if the desired bus is coming shortly. It also offers easily-browsable basic information about their destinations before reaching them. To make this technically possible, we use Android to create street-map- or bus-route-map pages and implement functions for necessary information to pop up and for page transition, which can be activated by touch operation. The data itself is linked to existing websites, and their key words have been extracted in advance. We define such data as general-purpose data structure. The prototype application has received a lot of favorable feedback.

Keywords: Services Computing, Android applications, Kyoto City Bus, National treasures in Kyoto

1. BACKGROUND OF THE STUDY

Kyoto, one of the most popular tourist destinations in the world, is home to numerous cultural assets including a number of national treasures, and tons of visitors take the Kyoto City Bus to travel around places that have these treasures. This is because the bus often takes them near these places and affordable one-day tickets are available. However, the bus service is unstable and buses are not always punctual.

In view of these situations, the Kyoto Municipal Transportation Bureau has installed warning display boards showing the approach of buses in the operating area (Fig. 1). With this equipment, visitors can tell which numbered bus is approaching the bus stop even when there are bus service disruptions. Also, visitors can access their Web site that provides the same information [1] from the user's smartphone. Visitors can judge

whether they should wait until the bus arrives, use another bus route, or spend time elsewhere.



Fig. 1: Warning display board and Web site

If a visitor wants to go from the current place to a tourist spot, e.g. a national treasure place, it helps if they can also find out whether there is a direct bus route to the place or if the desired bus is coming soon. This paper proposes a digital transformative (DX) based new application with a much more user friendly interface providing bus approaching information and useful tips for visiting national treasure places by Kyoto City Bus. Note that data necessary for implementing this application are currently protected by copyright. Therefore, we use data retrieved from Web sites and process them only for the purpose of making this research prototype.

2. DATA NECESSARY FOR APPLICATION

2.1. Bus approaching information

The Kyoto Municipal Transportation Bureau offers timetable data as an open data set [2], with which some websites provide timetable or transit information search engine services. It is presumed that "Bus and Train Veteran -Arukumachi KYOTO Planner-" [3] and "KYOTO CITY BUS SEARCH" [4] use the open data set.

The official Kyoto City bus location site requires rather complicated procedures to obtain necessary information the user needs. First, the user needs to designate the bus stop by entering the name or bus stop number, or by using GPS information. Then, the site shows all the combinations of route numbers and destinations as candidates, from which the user is asked to select up to five candidates. Only after that can the user get bus approaching information (Fig. 2). Our application can omit these steps and obtain information

intuitively. Our application provides bus approaching information by designating a position on the street map or bus route map as show Fig. 3. The street map type may be easier to use for visitors, while the route map type may be preferred by users familiar with the district.



Fig. 2: Warning display board and Web site

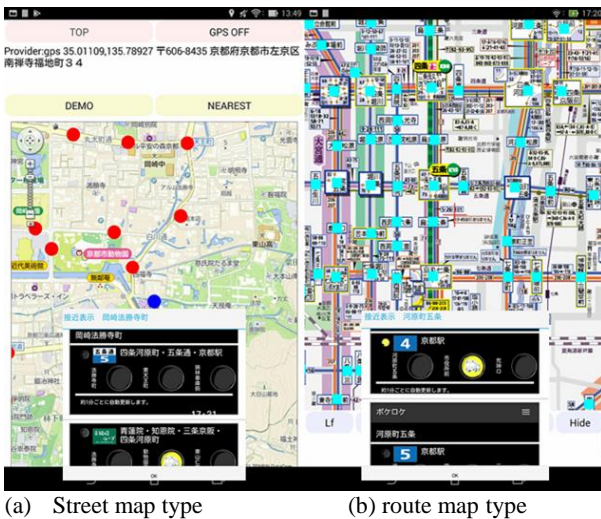


Fig. 3: Proposed application images

It is impossible to construct this application with open time table data available, so it is necessary to obtain and process data retrieved from the location system web site. In this paper, we aim at defining a data structure that can be used by many bus companies in the future when the copyright is lifted [5][6]. To this end, we propose using the JSON format for its versatility.

2.2. National Treasure Navigation Information

There are many sightseeing spots featuring national treasures and they are approachable by the City Bus. Since these spots do not offer copyright-free open data, general information about national treasures in Kyoto City and photo data are obtained from another website. By using this information combined with bus approaching information, we construct a prototype for

research purposes. As a website comprehensively covering national treasure information, displaying photos and providing bus access information, we choose "Kyoto ni Kanpai (Cheers to Kyoto)" [7]. To construct our application, the following data are necessary, i.e., historic site names, treasure names, treasure types (building, statue, painting), general information about treasure places, URLs for relevant sites, URLs for treasure photos, names of the bus stops near the treasure places, and the bus numbers. These data are structured in the JSON format.

3. IMPLEMENTATION TECHNIQUE

3.1. Bus approaching display

As shown in Fig.3, if a user touches a city map or route map, an AlertDialog window will pop up. In this window, several bus approaching information sub windows are displayed by using WebView class of the Android System. By touching a position, a specific bus stop is identified and the bus stop data related to this stop is retrieved from the JSON data. The JSON data are constructed as described below.

There are about 700 bus stops in the Kyoto City Bus service area. First, all the names of relevant bus stops need to be collected. From the bus stop name search page of the Kyoto Municipal Transportation Bureau site, these names and bus stop codes can be obtained. For each bus stop, there are URLs to provide bus approaching information which is displayed by route number and destination. The total number of the data is more than 5000. The data obtained in the HTML form by accessing these sites are analyzed. To narrow down the HTML source text to essential information, the regular expression handling mechanism is used.

Also by using the latitude and longitude for each bus stop, mapping between a position in the map and bus stop information can be done. In the route map case, relative position data for all the bus stops are necessary. To obtain the data, we make a special program which shows the bus route map and gets x, y coordinates for all the bus stops touched by the programmer. These coordinate data are mapped with bus stop information.

The JSON data structure is shown in Fig. 4. The obtained data from Web sites are managed in String arrays. Our JSON making program converts array data to the JSON format. The bus stop number and name, latitude/longitude, x/y-coordinates, and approaching information block are grouped as JSONObject. The information block is a JSONArray, which contains elements constructed with the bus number, destination and approaching information URL as JSONObject.

We use the OpenLayer API [8] to show the map. In an HTML file, JavaScript programs are coded. Functions are added for displaying the map in the designated center position, getting the latitude and longitude of the point touched and drawing circles at the

designated latitude and longitude. This HTML file is loaded in the WebView element of the Android System, on which Web browsing is possible.

When the user starts navigating the street map, the system gives GPS data to the WebView element to set a map center at this position and then the JSON data are passed to bus stops in the map. The JavaScript program judges what bus stop is touched on the map and then notifies the bus stop name to the WebView handling program. The program shows all the buses approaching the bus stop in the WebView.

In contrast, the route map is displayed by converting the route map file to the Bitmap instance. Then, by retrieving all the coordinates from the JSON data, a square is drawn at each bus stop position by the drawRect() method as shown in Fig. 3 right. When the user touches a square, the program judges what bus stop is touched and a WebView is created as in the city map case.

```

{
  "busstop": [
    {
      "stpname": "Aobashi-nishizume",
      "stpnnum": "481",
      "lat": "35.0307370000",
      "lng": "135.7697330000",
      "coord": {
        "x": "353",
        "y": "630"
      },
      "info": [
        {
          "dest": "Kita-oji Bus Terminal & Kinkakuji",
          "num": "205",
          "url": "http://blsetup.city.kyoto.jp/blsp/show.php?..."
        },
        ...repetition of approaching info element...
      ]
    },
    ...repetition of "busstop" element...
  ]
}

```

Fig. 4: JSON data structure

3.2. Navigation to Sightseeing Spots

The navigation capability supports visitors to national treasure places with its bus approaching information. The user can access this information by pop up view or page transition. The following data elements are necessary to make it possible:

- list of national treasure buildings, statues and paintings
- URLs to Web pages provided by each spot
- Bus stop and route number information near each spot
- URLs to national treasure photo sites

When the user selects a national treasure page, the list of national treasure places is displayed as shown in Fig. 5 left. If one of the buttons is touched, a window will pop up as shown in Fig.5 right. It contains its photo, keywords, the link to the detailed explanation page of this spot, and the link to the bus approaching information page. This window is constructed as an AlertDialog containing TextView, Button, WebView and link elements.

To implement the selection page, necessary data, i.e., the names of sightseeing spots, are extracted from the HTML text of the relevant site [7]. The data required for the detail pages and bus information are as follows:

- names and keywords for national treasure buildings, statues and paintings, defined as "kyototreasures.json"
- links to each detail page, defined as "spotlinks.json"
- links to photo URLs of the spots, defined as



Fig. 5: Option page and selected detail page

"treasureinfo.json"

As for "kyototreasures.json", the data structure is shown in Fig. 6.

In the pop-up there is a link to a Web site. A TextView element including plain text of <a> tag for the URL does not work for the purpose of page transition. To effectuate it, first the LinkMovementMethod.getInstance() method instantiates the MovementMethod class. Then, the setMovementMethod() method sets this instance to a variable (url_link). The link URL text (html) is transformed by the Html.fromHtml() method. Moreover, it is set to url_link by the setText() method with casting of CharSequence.

```

{
  "historicalbuildings": [
    {
      "name": "東寺(教王護国寺)",
      "items": "金堂 五重塔 大師堂 蓮花門",
      "img": "<img src='photo link' alt='東寺(教王護国寺)'>",
      "spot": "<a href='spot info link'>東寺(教王護国寺)</a>",
      "area": "<a href='area info link'>京都駅</a>"
    },
    ...repetition...
  ]
}

```

Fig. 6: JSON structure for treasure information

Fig.7 shows a list of links to attractions at the designated spot. These links are processed as described above. The right half of Fig.7 shows a photo of the designated attraction from the linked photo site.

Fig.8 shows the linkage between bus stops and bus operation information. The left half lists major stops along the user's bus route. The right half shows bus approaching information of a selected bus stop. Note that only the designated bus stop and two preceding ones are displayed to show whether a bus is approaching one of the three stops. Together with the JSON data of Fig.4, the bus stop numbers and names per bus route are necessary. The Fig.9 data are defined for this purpose.

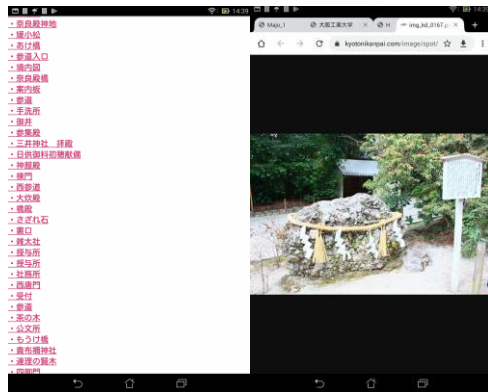


Fig. 7: List of attractions and photo site



Fig. 8: Linkage between spot and bus information

```

{
  "bus routes": [
    "keitoS": [
      {
        "num": "1",
        "stops": [
          {
            "stpnum": "225",
            "stpname": "大徳寺前"
          },
            Daitokuji
          {
            "stpnum": "016",
            "stpname": "植物園前"
          },
            Botanic Garden
          ... repetition ...
        ]
      }
    ]
  }
}

```

Fig. 9: Bus route data JSON

4. CONCLUSION AND FUTURE WORK

We proposed an application that can simultaneously manage tourist spot and bus service information in order to effectively travel around sightseeing spots with national treasures in Kyoto. To construct a versatile application, we defined a general-purpose data structure that can be used not only in Kyoto but also in other cities.

With regard to our approaching display application prototype, a survey was conducted on 20 people who have done some sightseeing in Kyoto. We asked them to

evaluate its different aspects on a scale of 1 to 5 (5 being the best) and it received an average score of 4.3 for its usefulness, convenience, and user-friendliness.

Our future research challenge involves situations where bus stops are abolished, moved, or where their names are changed. We would also like to provide transfer information between multiple buses.

5. REFERENCES

[1] "Kyoto City Bus Location System for Mobile", <http://www2.city.kyoto.lg.jp/kotsu/blssmart/index.html>

[2] "KYOTO OPEN DATA", <https://data.city.kyoto.lg.jp/node/14553>

[3] "Bus and Train Veteran -Arukumachi kYOTO Planner-", <http://www.arukumachikyoto.jp//index.php?lang=en>

[4] "Kyoto City Bus Search", <http://kyoto-bt.sakura.ne.jp/index.html>

[5] "Kyoto City Bus Operation Information Notification System by Voice Interaction", <http://sap.ist.i.kyoto-u.ac.jp/members/kawahara/paper/bus-navi.pdf>

[6] Hiroshi Sunaga, Takashi Aihara, Rina Furukawa, "Services Computing Based Mashing-up of Practical Applications", IEICE Technical Report, SC2012-9 (2012-08).

[7] "Kyoto ni Kanpai (A Toast to Kyoto)", https://www.kyotonikanpai.com/purpose/culture/cultural_property/kokuhou.shtml

[8] OpenLayers, <https://openlayers.org/>