データベースシステム 121－4
情 報 学 基 礎 58－4
（2000． 5． 25）

A Position-Aware Information Delivery System for Mobile Environments

Germano Leichsenring†, Kazutoshi Sumiya‡, Kuniaki Uehara‡
†: Graduate School of Sciences and Technology, Kobe University
‡: Research Center for Urban Safety and Security, Kobe University
{germano,sumiya,uehara}@ai.cs.kobe-u.ac.jp

Abstract

We propose an information delivery system which can handle the user's current position. In the system, each piece of information is related to a point in the real world and can be delivered separately to its clients. Each piece of information, or establishment information belongs to many categories which change according to the time. These categories are used to handle dynamic links among establishments. As each component can be delivered separately, the client can connect to multiple servers which brings a flexible system with a high degree of maintainability.

1 Introduction

With the popularization of the Global Positioning System(GPS), developed by the American government, GIS is playing an important role on mobile devices. Nowadays in Japan most cars use a Car Navigation System which can resolve routes and display information about shops and gas stations. Also, Personal Digital Assistants (PDAs) and mobile cell phones are extending their usability with a GPS receiver which can provide the user position for further processing.

Their systems access information from a static data source such as a CD-ROM or receive all information from sources on the Internet. Although these devices show a high level of functionality, nowadays systems lack mainly:

- possibility to update local data dynamically.
- ability to link together information from multiple sources.

Links can improve the usability on many aspects. When linking two pieces of information together on a database where the information has a position, setting the position of the linked information is the most straightforward method. However, as Sumiya et. al.[1]

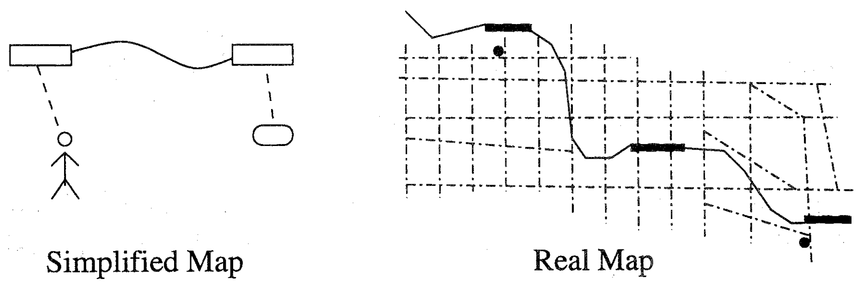

Figure 1: Simplified Map vs. Real Map

demonstrated for HTML documents, the missing feature of dynamic links on the HTML protocol has brought confusion with outdated data and broken links.

When setting links for positions, setting the type of information instead of the position itself can ease the process of setting the link and also provide links based on the user's position. This kind of link is resolved dynamically, and thus, called dynamic link throughout this paper.

To resolve dynamic links, we take into account not only the type of information requested but also the accessibility from the user's place at the present time. As shown on Figure 1, the distance meanings on a map are not directly related to our perception of distance. Accordingly, accessibility is not only physical distance but a function of travel time, expenses and the like. To use the accessibility to a full extent, we use stations which are special objects that can transport the user from one place to another. Furthermore, we extend the stations' characteristics to approach those of real life, where stations are evaluated by their surroundings. A dynamic link to a station takes into account the suitability of a station according to the constraints set by the link.

2 Related Work

Sumiya et. al.[1] developed an interesting mechanism for time dependent links embedded on hypertext. These links have time constraints and can be used on hypermedia broadcasting. Noda et. al.[2] went further and developed a version control method and proposes a component delivery mechanism for dynamically linked contents.

Several research efforts have focused on providing position-aware 2-D and 3-D media for a mobile client. HyperCampus[3] is a system for providing position-aware data according to the user's preferences through agents. At Columbia University[4] a system for providing multiple types of position-aware media to the user have been developed. Both systems focus on the position-aware problem and provide hard links between pieces of information, which are inappropriate for a distributed system where the source of information isn't centralized and managed by a single organization.

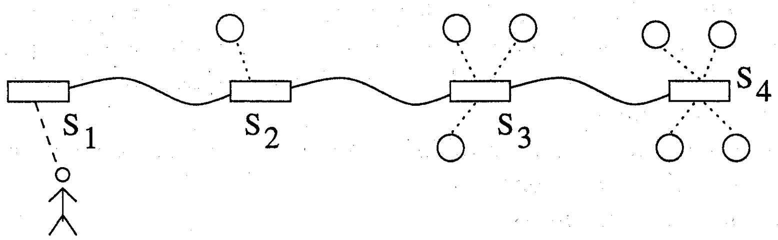

Figure 2: Accessibility and Suitability of Stations

Arikawa[5] developed methods of using position data embedded on video frames to browse and retrieve video data spatially; the method can't, at the present time, be used on mobile environments. Banjou et. al.[6] developed a system for retrieving and browsing pictures depending not on the position of the photographer but the desired object's position.

Tarumi et. al.[7] developed a system for deploying position-aware tags (SpaceTags) to users. The idea of having position-aware tags is quite interesting and opens the doors to a wide range of possible applications. Still, a common layer for developing the possible applications of SpaceTags is missing, along with the dynamic links here proposed.

3 Our Approach

We begin by proposing a component delivery method for position-related data. To classify each component, we use a category hierarchy for the components. Each component can belong to many categories, these categories changing according to time. Each component knows in advance each station which is near itself. When a component data is updated or newly registered on the client, the stations selected by the component become aware of the component's existence.

With a database of categorized components on the client, the client is ready to resolve position-aware dynamic links. The station's surroundings, then, are used for evaluating the station according to constraints of a dynamic link. For example, consider the situation shown on Figure 5.3, where each circle represents an establishment which satisfies the requirements of a link and station H_1 is the closest to the user. Station H_2, having an appropriate establishment, could be chosen as the link target; but station H_3 has much more variety of establishments. Although station H_4 has more establishments than station H_3, it is too far from the user, so H_3 should be selected.

4 Component Management

To be able to deliver position-aware components, we use common methods of component delivery where the component identifier is composed of a spatial identifier and a spatial object identifier. The spatial identifier determines uniquely a position on the globe; the spatial object identifier determines uniquely objects with the same position on the globe, such as rooms on a building. We will call an uniquely identified component an *establishment* on this paper.

To ease access to information, establishments have to be categorized. Here we use a category hierarchy of is-a relationships.

We define an establishment as:

$$E_j = (SD, SID, H, C(t), \{S_p | p = 1, 2, \cdots, n\}) \tag{1}$$

where SD and SID are, respectively, the spatial identifier and the spatial object identifier. H is the hypertext which describes the establishment. The hypertext can contain dynamic links to other establishments; the types of dynamic links and their resolution method are described on the following chapter.

Each establishment can belong to zero or more categories at a given time; these are resolved by the function $C(t)$. Belonging to zero categories means the establishment is not available at the time.

$S_1 \cdots S_n$ are stations close to the establishment.

Stations are predefined establishments which are managed separately from other establishments. Stations are defined as follows:

$$S_p = (SD, SID, \{S_i, D_i | i = 1, 2, \cdots, n\}) \tag{2}$$

$$D_i = (\{x_j | j = 1, 2, \cdots, m\}) \tag{3}$$

where S_i are stations which can be directly reached (i.e. without going through another station) from station S_p and D_i are set of elements x_j of the distance between the two stations S_i and S_p. As we don't threat distance as a constant value, the distance between two stations is a function which can rely on physical distance, average time needed to travel, money needed and so on, or m non-negative elements. Thus, the distance function can be shaped to the user's needs.

When information about an establishment is updated on the client, each station S_i included in the establishment information is notified. Each station has a dynamically constructed table of establishments near itself which is updated upon notification. This table is used when searching for specific establishments near a station.

5 Position-Aware Dynamic Links

5.1 Definitions

$d(A, B) \geq 0$ is the distance from place A to place B. The distance primarily relies on physical distance; but when the route from A to B involves passing through stations, the distance D_i between stations is taken into consideration. The distance can be calculated only when a route from place A to place B is known.

We will threat S_p as the set of establishments registered on station S_p. $x \in S_p$ means establishment x is registered on station S_p. $d(x,\ S_p)$ can be calculated for every $x \in S_p$.

A dynamic link is expressed as follows:

$$l = (\text{Origin}, \text{Destination}) \tag{4}$$

where Origin is the set of categories of the establishment which is setting the link and Destination is the set of desired categories.

5.2 Nearest Link

The nearest link is a link to the nearest establishment belonging to all of the desired categories at the time of searching. It is subdivided into nearest to user and nearest to origin, but both use the same algorithm. On a nearest to user link, the user position is the starting point; on a nearest to origin link, the origin establishment position is the starting point.

5.3 Station Link

The station link is a link to a station which is easily accessible by the user and at the same time fulfills the Destination requirements at their most. Its main purpose is to show the user a convenient or little-known cool station.

The relations between origin categories and destination categories are questioned to resolve the station link. We first explain the relations between two categories, then we generalize it with an algorithm to determine the suitableness of a station with a set of establishments. Furthermore, the suitableness and the accessibility are joined together.

5.3.1 Simplified Category Relations

To clarify the relations, we use examples based on the class hierarchy shown on Figure 3.

Origin and Destination don't have is-a relations:

On this case, the more establishments belonging to the Destination category, the most suitable the set is.

For example, on a link from a book shop to a school $l = (C_4, C_3)$ a station with many schools is suitable.

Origin is a subclass of Destination:

On this case, the set of establishments should avoid having establishments of the Origin category.

For example, on a link from a book shop to a shop $l = (C_4, C_2)$, a station with many markets is more suitable than a station with many book shops.

Destination has 2 categories with no is-a relationship:

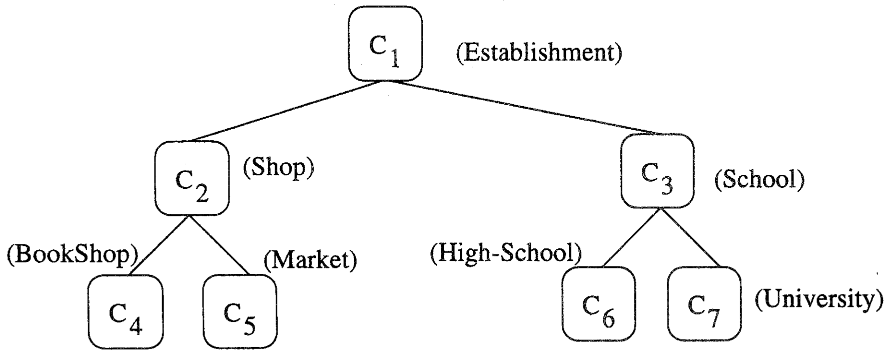

Figure 3: Category Hierarchy Example

On this case, the more establishments belonging to each category or both categories, the most suitable the set is.

For example, on a link from a book shop to a market or school $l = (C_4, \{C_3, C_5\})$, a station with many schools and markets is suitable.

Destination has 2 categories with an is-a relationship:

This case has to be separated from the previous case because it would end on a station with many establishments belonging to the child category would be the most suitable, which is not the expected behavior. On this case, a suitable station has many establishments belonging to the parent category and a good balance between establishments belonging to the child category and establishments belonging only to the parent category.

For example, on a link from a book shop to a university or school $l = (C_4, \{C_3, C_7\})$, a station with many university and high-schools is suitable.

5.3.2 Suitableness

The suitableness of a station is calculated according to the following algorithm. Here, $c(S_p, C_i)$ is the number of establishments registered on station S_p that belong to category C_i.

1. Given a link $l = (C_O, C_D)$ and a station S_p, let the suitableness s_p be zero.

2. For each category C_i in C_D which is not a child of another category in C_D, add $c(S_p, C_i)$ to s_p.

3. For each category C_i in C_D which is a child of another category C_j in C_D, add $-a\left(\frac{c(S_p,C_i)}{c(S_p,C_j)}\right)^2 + b\left(\frac{c(S_p,C_i)}{c(S_p,C_j)}\right)$ to s_p. Here, a and b are application-dependent constants which satisfy $0 < a \leq b$.

4. For each category C_i in C_O which is a child of another category C_j in C_D, subtract $c\left(\frac{c(S_p,C_i)}{c(S_p,C_j)}\right)$ from s_p. Here, c is an application-dependent positive constant.

5.3.3 Suitableness and Accessibility

First, the suitableness s_p is truncated to predefined acceptable levels, so that $s_{min} \leq s_p \leq s_{max}$. Then, if the distance from the user to the station $d(U, S_p)$ exceeds a threshold t, the suitableness is multiplicated by $k^{-1} \times (k + t - d(U, S_p))$, resulting on the following formula:

$$v_p = \begin{cases} d(U, S_p) \leq t : & s_p \\ d(U, S_p) > t : & s_p \times k^{-1} \times (k + t - d(U, S_p)) \end{cases} \tag{5}$$

where k is a positive constant.

The actual algorithm which searches for a station, then, is as follows:

1. Find the stations which are close enough to the user and put them on a queue q.
2. Let r be 0.
3. Pop the station with the highest value $d(U, S_p)$ from q as S_i.
4. If $r \neq 0$ and $max(v_i) < v_r$, stop.
5. If $r = 0$ or $v_i > v_r$, then $r \leftarrow i$.
6. Put all stations not yet on queue q and not yet processed which can be accessed from S_i into queue q.
7. Go to 3.

The resulting station S_r will have the maximum value possible for v_p.

5.4 Prototype System

We plan to develop a prototype system which can be used for a limited area. The system will include information of the surroundings of the Kobe University. There are a few bus lines which can be used at Kobe University; the bus stops will be used as stations.

The main objectives of developing a prototype system are:

- Fine tuning the equations and algorithms shown on chapter 5.3
- Finding any other dynamic link types needed for the success of the system

6 Concluding Remarks and Future Work

On this paper we have described an information delivery system for establishment information. First, a method for managing components, or information about establishments, is proposed. A category hierarchy with is-a relationships is used to categorize establishments, and the categories an establishment belongs to change according to the present time. Then, all contents have a position definition, and can be linked together based on constraints for dynamic links embedded on the hypertext describing the establishment.

For the future, experiments with the category hierarchy and the links are needed for further evaluation of the method. Then, the possibility of joining the idea with nowadays systems or systems still under development has to be considered.

References

[1] Kazutoshi Sumiya, Reiko Noda, Katsumi Tanaka, *A Temporal Link Mechanism for Hypermedia Broadcasting*, The Transactions of the Institute of Electronics, Information and Communication Engineers D-I, Vol. J82-D-I, No. 1, pp. 291-302(1999)(in Japanese)

[2] Reiko Noda, Kazutoshi Sumiya, Katsumi Tanaka, *A Valid-time Based Delivery Model and Temporal Consistency Management for Hypermedia Broadcasting*, IPSJ SIG Notes 99-TOD-4 pp 126-140 (in Japanese)

[3] Katashi Nagao, *Mobile Computing and Agent Augmented Reality*,Journal of Japanese Society for Artificial Intelligence Vol. 14 No. 4 pp 579-589 (in Japanese)

[4] Tobias Hollerer, Steven Feiner, John Pavlik, *Situated Documentaries: Embedding Multimedia Presentations in the Real World*, In Prof. of 3rd International Symposium on Wearable Computers

[5] Masatoshi Arikawa, *Spatial Browsing for Video Data Using Spatio-Temporal Descriptions of Cameras*, IPSJ SIG Notes 98-DBS-116 pp 225-232

[6] Yutaka Banjou, Hiroki Takakura, Yahiko Kambayashi, *Visual User Interface for Spatial Databases Combined with Landscape Images and Sensors*, DEWS'99, IEICE(1999) (in Japanese)

[7] Hiroyuki Tarumi, Ken Morishita, Megumi Nakao, Yahiko Kambayashi, *Space Tag: An Overlaid Virtual System and its Applications*, Proc. International Conference on Multimedia Computing and Systems (ICMS99) pp 207-212