Adaptive Navigation Support with Public Displays

Christian Kray
Computing Department
Lancaster University
Lancaster, United Kingdom
kray@comp.lancs.ac.uk

Gerd Kortuem
Computing Department
Lancaster University
Lancaster, United Kingdom
kortuem@comp.lancs.ac.uk

Antonio Krüger
Institute for Geoinformatics
University of Münster
Münster, Germany
krueger@un-muenster.de

ABSTRACT

In this paper, we describe a public navigation system which uses adaptive displays as directional signs. The displays are mounted to walls where they provide passersby with directional information. Each sign is an autonomous, wirelessly networked digital displays connected to a central server. The signs are position-aware and able to adapt their display content in accordance with their current position. Advantages of such a navigation system include improved flexibility and dynamic adaptation as well as ease of setup and maintenance. ...

1. INTRODUCTION

Effective support for wayfinding is an important design aspect of large building complexes such as airports, hospitals, and office buildings. People who find themselves in unfamiliar environments often face the challenge of having to find their way to a particular destination. This implies the need to know where they actually are in the complex, the layout of the complex, and the location of their destination in order to formulate their action plans.

Most traditional wayfinding systems are either static - once designed and setup they are difficult to change - or require users to carry some device with them all the time. However, both approaches have some major drawbacks since modern building environments are not static and information needs are dynamic. Oftentimes wayfinding needs are related to events or people rather than locations. At a typical University campus, for example, visitors, staff and students regularly need to find their way to another person's office, a lecture or a conference being hosted in a particular building. The destination is often dynamic as a particular person's office or event may have been moved to a different building or various sessions relating to the same event might be taking place in multiple buildings or the event may be held in different buildings on different days of the week. Similarly, though less dynamic than navigation information relating to events, a department may move to a different building or expand to take up additional space in another building. Similarly, alternative wayfinding information may be required in case a particular path is blocked due to building or renovation works or when the navigating person has special requirements such as wheelchair accessibility.

Existing wayfinding systems are not able to adequately address these changing needs. Handheld maps as well as stationary signs such as posted maps and directional signs have the problem that they are static and difficult to update. Recent mobile computing solutions (PDA + GPS + mapping software) are much easier to update, but have a number of shortcomings: availability of navigation information is limited especially for semi-public buildings such as airports and hospitals, accuracy and availability of navigation information indoors is restricted due to shielding of GPS signals, usability is poor which limits the users to computer literate people.

TODO: shorten, add some related work and do Conclusion: There is a need for dynamic adaptive wayfindig systems that is easily accessible for a wide range of user groups. ...)

2. GAUDI: PERVASIVE NAVIGATION SUPPORT USING ADAPTIVE DISPLAYS

To address the problem outlined above we have developed GAUDI, a prototype of a pervasive wayfinding system which is dynamic, adaptive and embedded in the build environment. The GAUDI system (Grid of Autonomous Displays) consists of set of autonomous wireless displays and a navigation server. The displays, shown in Figure 1, function as adaptive wayfinding signage and are intended to be deployed at strategic public locations across the Lancaster University campus. Their purpose is to assist visitors (and also staff and students) in navigating their way around campus. Each GAUDI display presents the following information: the name of the event or *destination,* the *direction* in which to go to reach the destination, and the approximate *distance* from the display to the destination.

2.1 Supporting Event-based Navigation

The first release of GAUDI, described in this paper, is designed to support *temporary signage for event-based navigation.* A large number of temporary events are held at the University throughout the year, which attract large numbers of outside visitors. These include concerts, performances, public lecturers, and award ceremonies. Most of these events

Figure 1: GAUDI Display TODO need better photo

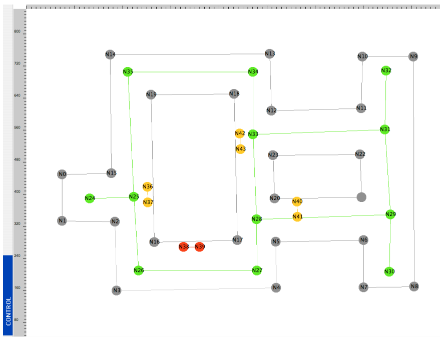

Figure 2: Setting the Position of a GAUDI Display

are not held at obvious and clearly identifiable locations, but in small venues which are difficult to find. Traditional wayfinding systems are especially poor in their support for temporary events. Hence, a pervasive wayfinding system for temporary events must satisfy the following requirements **TODO needs reformulation**:

- Easily movable signs: if a display is moved to a different location it should be adapted to display the content in a correct fashion based on its new location
- Simple way of adding or removing signs
- Simple way of authoring information
- Simple (possibly centralized) modification of wayfinding information

GAUDI supports temporary signage and event-based navigation as follows. A basic GAUDI display unit can be relocated easily since it only consists of a screen with limited computing power and a wireless network connection. It can either detect its new location via an attached sensor (e. g. ultrasound transducers) or an administrator can set the location manually using the GUI provided by the display (cf. Fig. 2). Once the display detects that is has been relocated, it automatically adapts its interface to accommodate for its new location and orientation.

Navigation Server	Display
spatial model definition route definition interface generation interface storage display management	position determination interface adaptation interface presentation

Table 1: Roles of Navigation Server and Displays in GAUDI

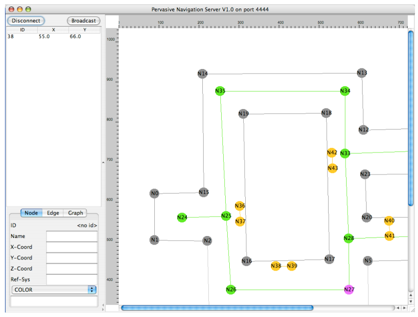

Figure 3: Administration Interface of the GAUDI Server

Adding and removing a display corresponds is done simply by turning a GAUDI display unit on and off as it will automatically connect to and disconnect from the navigation server.

3. POSITION-AWARE ADAPTATION OF WAYFINDING INFORMATION

The unique feature of GAUDI display is their ability to automatically adapt their presentation to their position. In the following we will outline the overall GAUDI system architecture and the adaptation process.

3.1 GAUDI System Architecture

A complete GAUDI system consists of two main components: a navigation server and an arbitrary number of autonomous displays. Each display is a self-contained wireless computer running GAUDI client software. The navigation server manages navigation information and publishes it to the displays. The distribution of functionality between displays and server is as follows:

The server runs an interactive application which allows administrators of the wayfinding system to define destinations in a 2-dimensional campus model. Figure 3 shows the user interface of the navigation server. The model indicates buildings and passage ways, and the current location and orientation of all displays. Clicking on a location followed by the 'broadcast' button will update all connected displays to point indicate the direction and distance to the new destination. Similarly, the interface can also be used to modify the existing spatial model or to author a new one.

3.2 Adaptation Process

The adaptation of navigation information is shown in Fig. 4. An administrator specifies a destination in the geometric

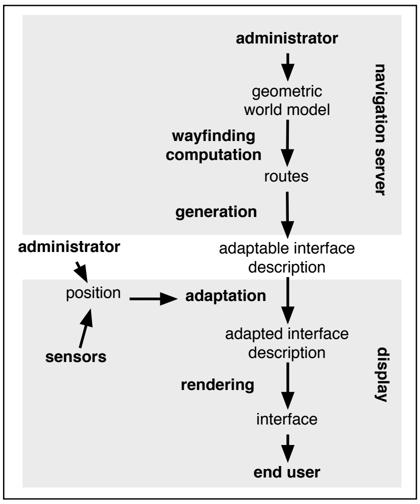

Figure 4: The adaptation process in GAUDI

world model, which in turn triggers the wayfinding computation. The resulting routes inform the generation process that produces an adaptable interface description. This description is then passed from the navigation server to the displays. On each individual display, the adaptable interface description is first stored locally, and then adapted according to the location of the display. Its location is either set by an administrator (who physically sets the display up) or by sensors that automatically determine the position of the display. The outcome of the adaptation process is the adapted interface description (encoded in plain SMIL), which is then rendered to produce the actual interface that an end user will see.

3.3 Spatial Model and Route Calculation

The spatial model used in the adaptation process is encoded as an annotated graph. It consists of nodes and edges connecting nodes, which are annotated with information such as coordinates, names, or whether or not they are part of the route. The calculation of routes operates on this graph, and is based on an A* algorithm.

3.4 SMIL++ Adaptable Interface Description

The adaptable interface description consists of an annotated graph – which models the route network as well as geometric properties of the area covered by the navigation application – and an abstract representation of the interface. This representation is encoded in a standard format (SMIL, cf. [6]), that we extended to incorporate adaptation rules. These adaptation rules are embedded in the SMIL code and are evaluated by the display clients using the annotated graph as well as information about the current location of a display. The evaluation results again in a standard SMIL document, which is then displayed by the client. We encode the rules using a small set of geometrical and logical primitives such as

- $< AngularDeviation > e_1, e_2 < /AngularDeviation >$ (computes the angular deviation between two edges e_1 and e_2 in the annotated graph)
- $< Select > key, item1...itemN < /Select >$ (evaluates the *key* expression and selects the corresponding item from a list of given alternatives)

Figure 5: Interface after adaptation to the location of the display *TODO: better image with distance and destinaton information*

3.5 Position-aware Interface Adaptation

Whenever a display client receives a new adaptable interface description, it first caches all related files (graph, interface representation, images) locally before evaluating the rules embedded in the interface representation. It first extracts all adaptation rules from the interface representation and then evaluates using a lazy evaluation scheme: whenever it encounters a function within an adaptation rules, it uses the annotated graph (e. g. by searching for the nearest edge to its own location) to replace the function with its corresponding result. Once the rule has been fully evaluated, the display client replaces the code representing the rule with its evaluation.[1] After all rules have been replaced, the resulting SMIL file is stored locally and displayed by the client (see Figure 5).

3.6 Display Management

Since the original adaptable interface description is kept as well, the client is able to re-adapt its interface without further interaction with the navigation server. For example, if the display client is re-located, it only has to re-evaluate the adaptable interface description using its new location in order to generate the proper interface. Hence, the server only needs to contact the clients when a new target location is set, i. e. a new adaptable interface description has been generated. Adding a new display to the network corresponds to starting the client application (respectively to turning the device on if the client application is started during the boot-up procedure). In order to remove a sign from the network, it is sufficient to shut down the display (or to client application).

Both the server and the client application are written in Java and communicate using standard TCP/IP sockets. The interface is encoded using the SMIL 2 standard, and is ren-

[1]Note that only those parts of the rule are evaluated that are required to determine its result. For example, in a $<$ *Select* $>$ function, only the item corresponding to the *key* will be evaluated.

dered using the XSmiles XML rendering engine [7]. The client provides a simple control GUI to define the display location in the absence of automatic sensing of its current position.

4. DISCUSSION

4.1 Related Work

TODO shorten and move to introduction

Interactive display systems have been one of the most prolific research areas in computer science and human-computer interaction. In recent years, a lot of work has centred on situated displays and on the use of large displays as public artefacts [3, 5, 4]. While most public display systems make use of single large-scale interactive surfaces, a few projects have explored the coordinated use of multiple, distributed displays. Examples include RoomWizard [**?**], Hermes [1] and the Plasma Poster Network [2], both systems comprised of collections of small displays attached next to office doors, and the Plasma Display Network [**?**] which uses poster-sized display. The Plasma Display Network is a content storage and distribution infrastructure that allows for the posting of content to all registered Plasma Posters. RoomWizard and Hermes demonstrate the potential of combining clusters of networked displays with context-awareness. Each of these systems is comprised of collections of small interactive display units placed outside people's offices or meeting rooms. The information displayed on each individual display is tailored in a context-dependent manner to its location: a RoomWizard display informs users about when or for what purpose a meeting room has been reserved; in Outcast and Hermes, when a visitor comes to an office and finds the person not there, a display provides information where to locate them, or when to come back. In both systems, the context-dependent information presentation is achieved by logically binding each display to a concrete context, which can either be a location, a room, or a person.

4.2 Experiences

...

5. CONCLUSION AND OUTLOOK

In this paper, we presented GAUDI, an initial version of a pervasive navigation system. It enables untrained users to easily set up a set of dynamic signs that will automatically adapt to their current location. The whole network of displays can be controlled from a single server, where clicking on a target location is sufficient to trigger the automatic adaptation of all connected displays. The proposed approach not only simplifies the setup of dynamic signs but also minimizes bandwidth use, since only the definition of a new target location requires the re-transmission of the interface descriptions. Otherwise, displays can locally adapt the interface without interaction with the server.

The work presented in this paper is a first step towards a transparent pervasive navigation system. In the future, we will extend the system to support multiple concurrent targets as well as individual routes. Furthermore, we will provide access to the route planning mechanism on the displays in order to enable users of the public screens to specify their destination locally. Finally, as more displays are deployed throughout the campus, we intend to perform empirical studies on the acceptance and effectiveness of our approach.

Acknowledgements

6. REFERENCES

[1] K. Cheverst, D. Fitton, A. Dix, and M. Rouncefield. Exploring situated interaction with ubiquitous office door displays. In *Public, Community and Situated Displays (Workshop at CSCW 2002)*, New Orleans, LA, USA, 2003.

[2] E. Churchill and L. Nelson. The plasma poster network: Postign multimedia content in public places. In *Proceedings of INTERACT'03*, pages 599–606, MISSING, 2003. IOS Press.

[3] S. Greenberg. Designing computers as public artifacts. *International Journal of Design Computing: Special Issue on Design Computing on the Net (DCNet'99)*, MISSING(MISSING):MISSING, 1999.

[4] K. O'Hara, M. Perry, and S. Lewis. Social coordination around situated display appliances. In *Proceedings of CHI'03*, page MISSING, MISSING, 2003. MISSING.

[5] D. M. Russell and R. Gossweiler. On the design of personal and communal large information appliances. In *Proceedings of UbiComp'01*, page MISSING, Berlin, Heidelberg, New York, 2001. Springer.

[6] The WWW Consortium. Synchronized multimedia integration language. http://www.w3c.org/SMIL, 2004.

[7] XSmiles. Xsmiles. http://www.xsmiles.org, 2004.