

CASSIUS: Designing Dynamic Subscription and Awareness Services

Michael Kantor

Institute for Software Research
University of California, Irvine
Irvine, CA 92612 USA
mkantor@ics.uci.edu

David Redmiles

Institute for Software Research
University of California, Irvine
Irvine, CA 92612 USA
redmiles@ics.uci.edu

ABSTRACT

CASSIUS is an awareness server which assists users in designing subscriptions for maintaining awareness of events within work, physical and social environments. This environment is designed to work with a wide range of awareness tools using desktop computers, mobile devices and ambient fixtures[4]. This work investigates the requirements for creating ad-hoc subscriptions – a subscription that is created either by the user or a software agent, and which only exists for a brief period of time. Design guidelines are proposed that help address the problem inherent in having users invest effort in creating a subscription which may last for only the few minutes in which they are in a specific location or context.

Keywords

Awareness tools, Notification servers, agents, CASSIUS, ad-hoc networks

INTRODUCTION

The "Creating Awareness with Subscription Services" (CASS) strategy is an approach for creating a ubiquitous awareness environment [5]. The goal is to enhance people's ability to coordinate with various actors within work, physical and social environments by providing a usable and useful environment for awareness and coordination. The CASS strategy consists of a set of guidelines for the design of software based awareness environments.

This paper begins by presenting an overview of these guidelines and presents our implementation of this ubiquitous awareness. We then discuss potential extensions to the guidelines and implementation which address the issues of designing an awareness environment that is usable for the ad-hoc creation of subscriptions for monitoring contextual information.

CASS Guidelines

The CASS strategy consists of a set of guidelines for creating a usable and useful awareness environment. These guidelines can be divided into three categories: provide access to diverse information, remove guesswork from

specifying the information of interest and support flexibility in the choice of awareness styles for representing awareness information.

Provide Access to Diverse Information

Research in awareness technologies has focused upon tools designed to monitor a single source (or a narrowly defined set of sources) of awareness information. This has been the case because the projects were either experimental, investigating a style of presenting awareness information with some demonstration source of awareness information or because they were implemented within a context where there was only one information source that the designer was interested in.

In a ubiquitous awareness environment, an awareness tool has access to diverse sources of awareness information allowing each user to monitor the kinds of information that matter to them. This was done by the Elvin Tickertape [3] which could monitor discussion groups, news, email, and other notifications sent to the notification server. To support awareness and coordination in diverse environments, we can not limit ourselves to monitoring a single source of information. An awareness tool needs to be able to obtain information from multiple sources of awareness information, and integrate them together to give users a broader understanding of what is happening within their work, social or physical environments. Nor can we limit users by telling them that the only information that they can become aware of is news, photos of offices [1], or any other single source.

Remove Guesswork from Specifying Information of Interest

Having access to diverse sources of awareness information would be insufficient if the user does not know what sources of information are available. To provide a *usable* awareness environment, the user needs to be informed (preferably by the awareness tools rather than by coworkers) of what sources of awareness information are available, what each source monitors, and what kinds of changes and events can be detected. The awareness environment must provide users with meta information describing the awareness information accessible to the environment.

For example, if a source of information is a research paper, the sections and subsections could be monitored for changes, as could word or page counts. As a second example, if users monitor for traffic problems, they need to

*LEAVE BLANK THE LAST 2.5 cm (1") OF THE LEFT
COLUMN ON THE FIRST PAGE FOR THE
COPYRIGHT NOTICE.*

know what freeways and roads are monitored and what kinds of traffic events are reported so that they can choose which ones to monitor.

Without this meta information, any attempt by the user to describe their interests involves a great deal of guesswork, leading at best to partial success, and more likely to frustration. Access to this type of information is a prerequisite for a usable ubiquitous awareness environment.

Support Choice of Awareness Styles

A common problem with awareness technologies is that they tend to provide a fixed awareness style with very little room for selection of alternatives. In this work, the term awareness style refers to the manner in which information is presented to users and varies along a variety of dimensions including:

1. Intrusive vs. peripheral dimension: how intrusive/disruptive is the presentation of new awareness information? If the goal is to be immediately notified of information as it occurs, an intrusive style is needed. If the goal is to maintain general awareness of ones environment, utilization of peripheral senses may be more appropriate.
2. Mobility dimension: Can the awareness tool be used as a person's work moves through different physical and social contexts? Can it use mobile devices, or does it require greater display, networking or computational resources? Does its presentation style require the kind of user attention only available within an office or control room?
3. Information Representation dimension: What kinds of information does the representation of the information focus upon?
4. Cognitive Effort dimension: How much effort is needed to interpret the representation?

To provide an awareness environment that is *useful*, people need to not only be able to choose what information to monitor, they need to be able to choose how to be made aware of the information. Ideally, they would have hundreds of different awareness tools to choose from, and could choose the one which best fits their work environment, work practices and their needs with respect to some subset of the information they intend to monitor.

Further, the user should not be limited to one awareness tool at a time, nor one awareness tool for any source of awareness information. As a user leaves an office setting for a meeting, lunch or other situations, the style of awareness that suits this new environment may change and the user needs to have the option of changing awareness tools to match the new environment. When the user is in the office, there may be many sources of information, one subset of which is monitored with an intrusive tool, and a second subset of which is monitored with a peripheral awareness tool.

CASSIUS

Our implementation of the CASS strategy is called CASSIUS (CASS Information Update Server). It is a notification server [7] which has been optimized for usability as an awareness server. Figure 1 shows a service-based architecture that CASSIUS implements, and Figure 2 shows an awareness source browser and subscription editor provided with our CASSandra toolkit.

As shown in the top two services of Figure 1, sources of awareness information must register with the server, listing the objects that they monitor and describing the types of events that can affect those objects. The awareness tool can then support users browsing through lists of sources of awareness information (shown in the top left column of Figure 2). For each information source, the user can browse through hierarchies of objects and properties monitored by that information source (top center column of Figure 2). When the user selects an object to monitor, lists of events that can affect the object are listed, allowing the user to optionally refine their subscription to just those types of events (top right column of Figure 2). A single awareness tool can monitor as many subscriptions and information sources as suits the user's needs and the tool's awareness style.

Representation of Arbitrary Information

A key issue in our design involves the representation of awareness information from any information source. If the designer of the awareness tool does not know in advance what the source of awareness information is, how can the tool represent that information? The answer is that all notifications, regardless of what software sent them, must be formatted using data fields that have a fixed interpretation shared by all CASSIUS awareness tools and information sources. The awareness tool need not understand the meaning of the data sent in a notification, but does need to understand its nature, that one field contains verbal/textual descriptions of the event, another field quantifies the extent of change, etc... Our design attempts to account for the information needs of a broad range of awareness styles by using the notification fields of Table 1.

Sample Applications

We currently have a WebDAV server (CassDAV), and an AWACS simulator which send notifications to CASSIUS,

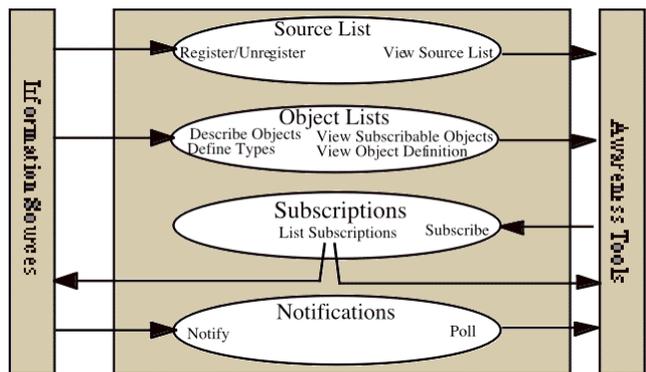


Figure 1: CASSIUS service architecture

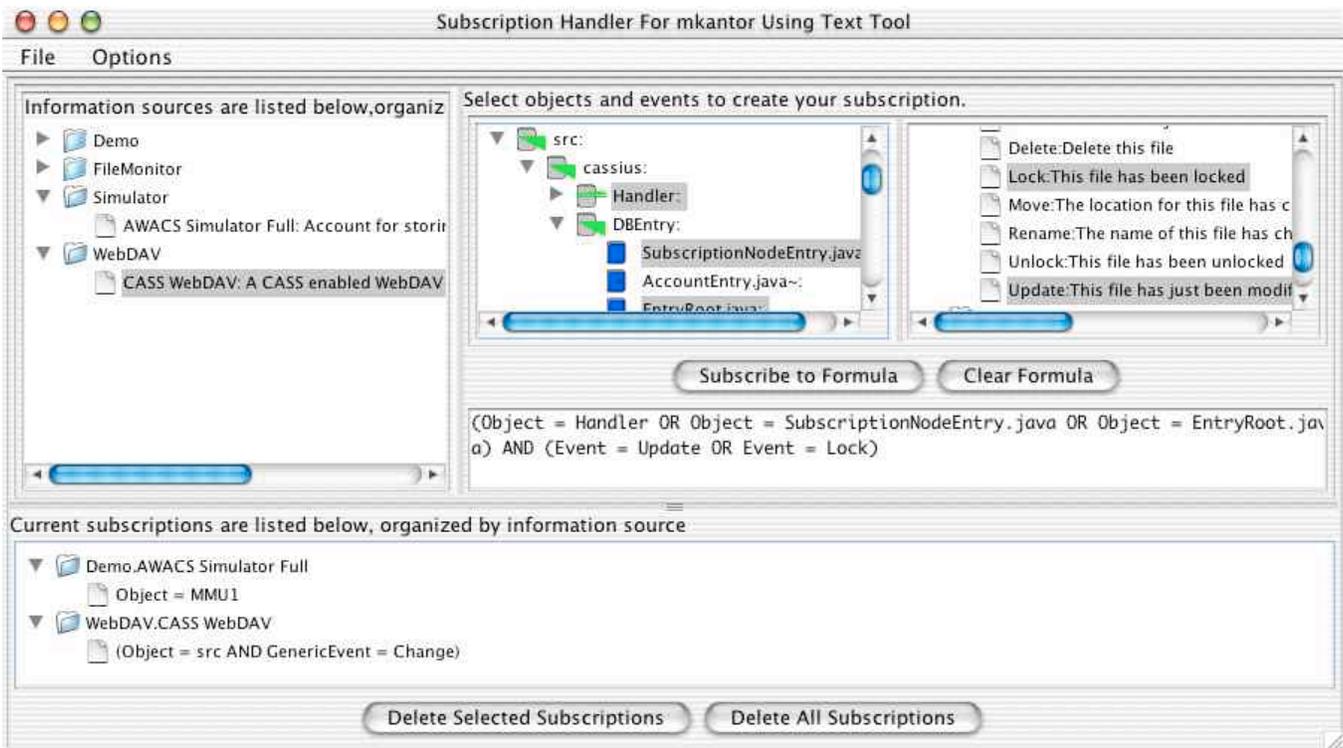


Figure 2: CASSandra information source browser and subscription editor

and we are working on a CVS repository and a Portholes implementation [1]. In the case of WebDAV and CVS repositories, the monitored objects are files and folders, which are described to the server so that the user, using an interface such as that presented in Figure 2, can browse through a representation of the file system to find and select files and folders to monitor. Notifications report on the nature and extent of the changes or operations performed upon the files and folders.

In the case of Portholes, which creates awareness by distributing photos of people at work in their offices, the monitored objects are groups and individuals, the notifications indicate the extent of changes between successive photos, and contain a URL to the photo.

To monitor these and future sources of awareness information we have a growing body of awareness tools including simpleScroller (a tickertape such as was illustrated by Elvin [3]), EventLister (a debugging tool to help developers see the notifications that their code sends) and BiffArray (Figure 3). We are also working on an email-based tool for sending digests of events, and are planning to adapt our mobile awareness technology called MiniPortholes.

BiffArray

BiffArray (Figure 3) is modeled on Xbiff, a common mail awareness indicator in unix windowing environments. It provides a row of Biffs, where the graphics within the icon show the most recent event to come from the objects being monitored. Rather than a mailbox with flag up or down graphic (as was done in XBiff), it shows the GenericEvent

field (see Table 1) of the most recent notification to be received. As there are five values of Generic Event, there are 5 images used to represent the different states. Each biff in the display can be configured both in what it monitors and in what sounds it uses to notify the user [2].

Each biff can monitor a different source of information. For example, if you have six biffs, two could monitor files and folders that you work with, two could monitor coworkers, one could monitor activity on a chat group, and the last could monitor the state of your group's printer.

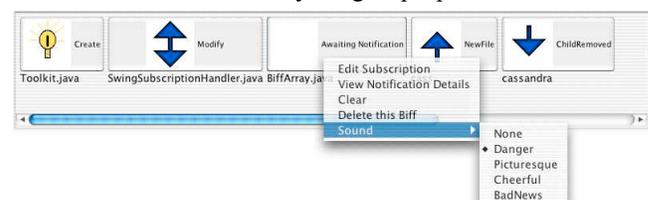


Figure 3: BiffArray: Visual and Audio Icons

Mobile Awareness

MiniPortholes (Figure 4) is a mobile awareness technology implemented in J2ME. It allows users to subscribe to maintain awareness of individuals such as coworkers and family. When this tool uses the CASSIUS server, it enables users to not only subscribe to monitor other MiniPortholes users but also monitor all types of CASSIUS information sources. This means that system administrators can monitor their servers, salesmen can monitor their inventory, parents can monitor their children, and in fact, a parent who is a system administrator and

Summary	One line textual summary of the change
GenericEvent	An event name chosen from a list of generic event names. Generic event names are shared by all information sources and enable awareness tools to understand the general nature of the event even if they can't interpret the specific nature of the event represented by the Event field. Currently supports "Activate", "Deactivate", "Increase", "Decrease" and "Change" (the last being a catch-all for events not fitting other categories).
Event	An event name specific to the information source and to a type of object within the information source. Events reporting on a section of a document might include "Text Added", "Text Removed", "Subsection Added", and "Subsection Removed".
URL	Optional link to more information about a notification. Leads users to text, images or information source specific data files.
Person	Optional person associated with event.
Place	Optional place associated with event.
Object	Identifies the object or property that has changed.
AccountPath	Identifies the information source.
NumericalValue	Optional numerical value to quantify the change.

Table 1: CASSIUS Notifications

salesman can monitor all three simultaneously – hopefully not while driving.

While currently using a simplified version of CASSIUS, we hope to integrate MiniPortholes with CASSIUS soon.

AD HOC AWARENESS INFORMATION

The high level goals of this work (the creation of a ubiquitous awareness environment) are important whether one is talking about work (awareness and coordination among coworkers, often distributed both spatially and organizationally), family (awareness and coordination with family members scattered around a city) or a physical environment (awareness of problems such as upcoming traffic, weather, riots, parades, statistics related to a sporting event you attend and special deals at your favorite coffee shop just down the street). Effective support of these diverse environments requires:

- 1) Creation of ad-hoc subscriptions, whose life span may be as little as 10 minutes (and where the time to specify the subscription must be comparably short).
- 2) Location based awareness servers that awareness tools connect to on-the-fly to discover new sources of awareness information.

The principal of what must be done remains unchanged: 1) the users must be provided with meta information telling them what information sources are available and what types of information can be subscribed to within each information source, and 2) users choose awareness styles for each type of information. However, in this new



Figure 4: MiniPortholes, mobile awareness

environment, extensions are needed in how these services are provided.

Extension 1: Detecting and Logging Information Sources

In our current implementation, users can view lists of information sources on the servers that they have permission to access. If we introduce location-based awareness servers (perhaps for sending traffic awareness to people on freeways) and time-based awareness servers (a server which only exists for a short period of time, such as for the duration of a county fair, or a festival), the nature of these lists must change. To effectively provide users with lists of information sources that they can monitor, mobile awareness tools need to be able to

1. Detect the presence of awareness servers as they come into range,
2. Obtain lists of information sources from these servers that users can browse through,
3. Store the lists of information sources and information about the awareness servers (such as that it was running on a traffic monitoring server, or on some stranger's PDA),
4. Categorize the stored information according to the nature of the awareness server (group all traffic awareness servers together, group all PDAs running their own servers together), and by the information source (all information sources that monitor a calendar get grouped together, regardless of what awareness server it came from).

Extension 2: Usability in Ad Hoc Subscriptions

A key issue in supporting ad hoc subscriptions is the efficiency with which the subscription can be created. How much examination of the display and selection of options must be done to allow the user to monitor traffic for the next 15 minutes? To address these problems, additional

guidelines have been created for the design of information sources and awareness tools.

Support a Spectrum of Complexity

Location and time based awareness servers should provide simple options for subscribing. While it should be possible to carefully refine long term subscriptions so that the awareness tool doesn't waste time presenting unwanted information, support is also needed for the fast and less precise task of creating short-term subscriptions.

For example, a person at a county fair can subscribe to the fair's scheduled events and be notified each time a new event is about to begin. Or the person can be more careful and look at the objects under "Scheduled Events" in the object hierarchy and subscribe to only be notified when musical events are about to start. Both subscriptions are useful. One requires more time and thought – time which people spending all day at the fair are more likely to invest than people attending for only part of the day.

This scaling is supported in CASSIUS in the form of notifications that can be propagated up the object hierarchy: a notification of changes to a file in a CassDAV server will result in notifications being sent to users monitoring the file (users who have carefully refined their subscription), and will also propagate the notification up to users monitoring any of the containing folders.

One new design principal for information sources is therefore to design the hierarchy of monitored objects to explicitly support both users who have time to carefully refine subscriptions by browsing deeply through object hierarchies, and to have high level, rapidly accessible objects for use in creating ad hoc subscriptions.

Consistency Across Related Information Sources

Subscriptions need to be generalizable across related information sources. For example, if a user subscribes to be notified of traffic problems while on one segment of a highway, there is a strong likelihood that when moving to a different segment of the freeway, the user will want to subscribe to the same or similar categories of information.

Support for this would require consistency across information sources that monitor the same types of information. For example, each traffic information source would have the same high level objects in its hierarchy, and only when you work your way down to monitoring certain on/off ramps do the object hierarchies of the different sources begin to look different.

Implementations of this (under the current CASSIUS architecture) would leave it to the awareness tool to

- 1) Note that two information sources are similar,
- 2) Determine that the user has subscribed to a certain set of information in the first information source,
- 3) Either automatically subscribe the user to similar information in the new information source, recommend it to the user, or make the information very easy to find and subscribe to [6].

An alternate approach would utilize the categorization and logging of awareness servers and information sources

discussed in the prior section. It would allow users to look at a variety of related information sources and design a subscription that specifies what to do if information sources of that type are encountered in the future.

CONCLUSION

We have designed a set of guidelines for creating ubiquitous awareness environments, and provided an implementation of this environment. However, without strict guidelines in the design of information sources and awareness tool that work within this environment, the environment will only be usable for static subscriptions; subscriptions to information sources that will be a part of the user's life for an extended period of time. To make this environment usable for the creation of ad hoc subscriptions, information sources need to have both high level objects for rapid subscription and low level objects for refined subscription, sources of a common type need to utilize common object hierarchies, and the awareness tools need to be able to log, organize and recommend subscriptions based on information retrieved from the awareness servers that it encounters.

ACKNOWLEDGMENTS

This effort was sponsored by the National Aeronautics and Space Administration (NASA) Research Grant NAG2-1555. The U.S. government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of NASA.

REFERENCES

1. Dourish, P., Bly, S. (1992) *Portholes: Supporting Awareness in a Distributed Work Group* In *CHI'92* ACM, pp. 541-547.
2. Gaver, W. W., R. B. Smith, and T. O'Shea (1991) *Effective Sounds in Complex Systems: The ARKola Simulation* In *CHI'91*, New Orleans.
3. Fitzpatrick, G., Mansfield, T., Arnold, D., Phelps, T., Segall B., Kaplan, S. (1999) *Instrumenting and Augmenting the Workaday World with a Generic Notification Service called Elvin* In *ECSCW'99* Copenhagen.
4. Ishii, H., Wisneski, C., Brave, S., Dahley, A., Gorbet, M., Ullmer, B., Yarin, P. (1998) *ambientROOM: integrating ambient media with architectural space* In *CHI'98* ACM, Los Angeles, pp. 173 - 174.
5. Kantor, M., Redmiles, D. *Creating an Infrastructure for Ubiquitous Awareness*, Eight IFIP TC 13 Conference on Human-Computer Interaction (INTERACT 2001 Tokyo, Japan), July 2001, pp. 431-438.
6. Maes, P. (1994) *Agents that Reduce Work and Information Overload* in *CACM*, **37**, 31-41.
7. Ramduny, D., Dix, A. and Rodden, T. (1998) *Exploring the design space for notification servers* In *CSCW'98* ACM, Seattle, pp. 227-235.

