Hermes: A Platform for Context-Aware Enterprise Communication

Ajita John
Avaya Labs
New Jersey, USA
ajita@avaya.com

Reinhard Klemm
Avaya Labs
New Jersey, USA
klemm@avaya.com

Ankur Mani
Arizona State Univ.
Arizona, USA
ankur.mani@asu.edu

Doree Seligmann
Avaya Labs
New Jersey, USA
doree@avaya.com

Abstract

This paper describes our vision for next-generation context-aware enterprise communications that achieves the integration of backend business processes with user communications. We envision a new class of enterprise applications in which communications between users in response to a variety of enterprise events will be driven by an automated process in which an appropriate group of users will be selected for communicating at an appropriate time on an appropriate media. To achieve this goal, the applications must exploit a variety of context information such as enterprise knowledge, user knowledge and application knowledge. This paper describes our platform Hermes that enables the creation and execution of such context-aware applications. It briefly presents experiences with Hermes in several demonstrations.

1. Introduction

In this paper, we present our vision of context-aware enterprise communications and our platform *Hermes* for enabling the creation and execution of context-aware communication applications. Communication among people in an enterprise has been a means for knowledge transfer, knowledge generation, and decision making. Significant shifts have been occurring in the recent past that impact communications in an enterprise. The virtualization of enterprises through globalization and expansion has necessitated the increasing dependence on technology for communication. We have seen a proliferation of communication devices and media, the use of multiple devices of varying capabilities, an increasing mobility of people and devices, the permeation of work into different environments such as the home, hotel, and the road, which are conducive to varying degrees of the use of specific types of media, and an increasing specialization of enterprise workers. As a result of these changes, it is becoming increasingly difficult to manually identify who, when and how to contact people in response to an enterprise event. Additionally, there is a strong need to bridge the gap between backend business processes that compute, perform database operations, process documents, and interact with users through web pages and other clients, and the decision making processes that users constantly engage in through communications in an enterprise.

1.1. Our Vision

We envision a modern enterprise that integrates context-aware computing and communications with its knowledge management and business processes. To address this vision, we have built a platform, *Hermes*, for creating and executing rich context-aware applications that tightly integrate backend business processes with communications. The communication applications on Hermes are modeled using workflows as defined by the Workflow Management Coalition (WFMC) [1]. The nodes of the directed graph in the workflows represent communication tasks that use the context of the enterprise, of the specific application, and of users in a reasoning process to make an effective selection of the media, time, and people for a specific communication purpose. At the chosen time, the tasks set up communication sessions such as conferences, interactions, notifications involving selected groups of people on selected media. Hermes integrates the communications applications with backend business processes, thus enabling closing the loop between computing-centric enterprise processes and enterprise communications. The closing of the loop is essential for context modeling because context is a dynamic construct [3]. This is in sharp contrast to open loop systems where communications such as

Proceedings of the Fourth Annual IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOMW'06)
0-7695-2520-2/06 $20.00 © 2006 IEEE

emails or voice calls may be initiated by a backend process but complex communication processes are largely manual and their intermediate or final outcomes typically do not feed back into the business processes. The integration of context-aware communications with business processes provides:

- faster, more precise, repeatable communication processes, resulting in latency reduction for decision-making in the enterprise
- tractability of communication threads in an enterprise and the ability to automatically follow up on communication events or the lack thereof
- enhanced employee satisfaction through reduction in manual communication tasks, decreased interruption rates, and balancing of communication load
- more precise audit trails that cover communications in an enterprise, allowing improvements in reporting and optimizations in enterprise processes

1.2. Related Work

Platforms for creating context-aware applications have been discussed in [2,5,9]. These articles focus on the collection, dissemination, and processing of context information independent of the applications. However, something is context because of its relationship to a larger set of interrelated facts and thus affects decision making [7]. Thus, while [2,5,9] provide support for collecting, disseminating and processing contextual information, the application developers themselves have to provide for processing and using context.

Efforts in [4] use pieces of context information to make decisions about connecting people. User context information is used to route and migrate calls in [6]. We, on the other hand, focus on a set of communication applications that are embedded in the working environment of the modern enterprise. We use semi-static knowledge in the enterprise about work practices, organizational hierarchies, user preferences, and more and associate this knowledge with changes in user context information resulting from dynamic business processes. This association provides the required context to make communications decisions. There has been some effort in using such knowledge, for example in the workflow for multimedia conferencing [10], but it is restricted to a specific application, whereas we created a platform for designing and execution a wide range of complex and context-aware communications applications.

The organization of the rest of the paper is as follows. In Section 2, we present an enterprise communication scenario that motivates the need for a context-aware platform as we define it. Section 3 discusses our context-aware platform Hermes and the acquisition, representation, and use of context in Hermes. Section 4 discusses our experiences with Hermes and conclusions are given in Section 5.

2. An Enterprise Communications Scenario

We present an enterprise communication scenario that illustrates the key motivations for Hermes. This scenario has been drawn from an extensive case study that was conducted with insurance companies. It revolves around a scenario in which person-to-person communication barriers are reduced and users with access to multiple communication media across a global enterprise are brought together to solve a specialized and time-sensitive issue. An example of such an issue is the resolution of a claim in an insurance company by people such as a claim appraiser and an expert. We first discuss how such a scenario can be driven by an intelligent context-aware middleware and then provide the contrast to a manually handled scenario. Similar scenarios have been observed in other domains such as health care where appropriate resident doctors, nurses, and specialists are connected to respond to a patient emergency.

2.1. Intelligent Communications Using Context

A Claims Handling business process in an insurance company detects that the deadline for an unresolved car accident claim from a customer is near and triggers a communication application named *Flow 1* (Figure 1). This flow contacts the claim appraiser via phone according to the appraiser's user rules. If the appraiser does not respond, Flow 1 escalates the problem by sending invitations for a conference, to be started in say, 2 hours, to the appraiser, appraiser's supervisor, and the customer. It sends the invitations using a combination of presence and availability information, user rules, and enterprise policies. User rules include preferences that specify how users would like to be contacted at different times of the day. This may mean that, for instance, the supervisor receives the invitation via email, the appraiser via voice, and the customer via IM. The invitation can be derived from a template by including information specific to the claim such as customer data. The recipients may respond through the phone or through a user portal. If a quorum of people accepts the invitation, Flow 1 establishes a conference with the three invitees at the scheduled time. The medium of the conference (e.g. voice or IM) may be determined based on the needs of the collaboration, the presence and availability information of the participants, and the media capabilities of the devices of the participants.

During the conference, the supervisor recognizes the need for an auto collision expert. There are many

potential experts in the global locations of the company and the supervisor does not know which experts have the necessary skills and are available. Experts with prior interaction with the customer, appraiser, or supervisor may be preferred. Experts may be mobile using devices with voice and text chat capabilities. However, the noise level in the expert's environment may preclude any productive voice interaction. The interaction may require high attention levels that necessitates that the expert does not participate in any other communication session. To automate the expert finder process, the supervisor initiates *Flow 2* which selects a possible group of experts based on skill, *cohesiveness* (i.e. a history of interactions) with participants of the conference, presence, availability, location and environment, and sends out invitations to them. Depending on responses from experts, Flow 2 alerts the supervisor to the availability of an expert. Since the supervisor is present on an ongoing conference, this context information is used to expedite the delivery of the alert: If the conference is a voice conference, the alert is rendered as a *whisper* and audible only to the recipient. If the conference is an IM conference, the whisper is an IM message. If an expert is available, the expert is bridged into the conference.

After the completion of the conference, which constitutes a communications event, the supervisor receives a request to update the status of the claim. The update can be rendered either by voice interaction software or through a user portal. This step will also affect backend databases in the enterprise resulting in the resolution of the claim and closing the loop from human communications and decision making to the enterprise backend.

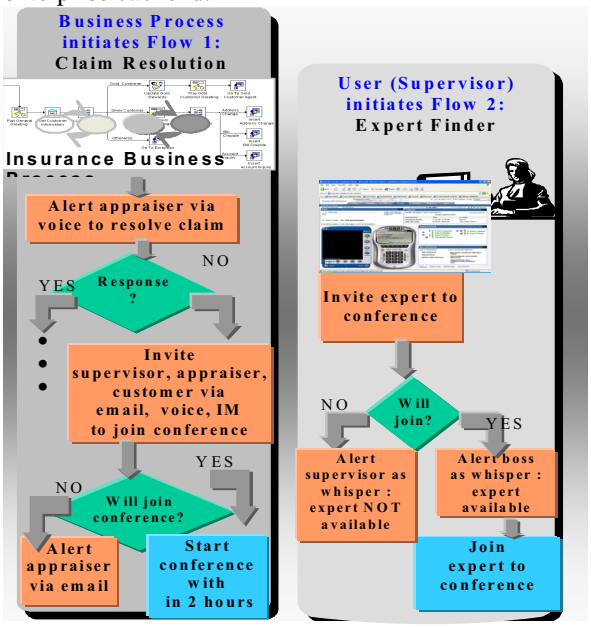

Figure 1: Example of communication flows

The above scenario shows how a system that can intelligently connect the right group of people at the right time on the right media by reasoning about contextual information such as presence, availability, cohesiveness, past and current communication sessions, user rules, skills, location, environment, enterprise organizational information and policies can result in effective communication to address the needs of the enterprise. The degree of automation of communication tasks in our scenario stands in stark contrast to a traditional communication process where people establish communications manually. Here, the human delays and the potential for errors and suboptimal communications decisions can be costly to the enterprise, especially if the pool of people for selection is large, capabilities of the media used by the people is varied, and the people are geographically distributed. From a user perspective, it protects personal information such as presence, availability, and location from exposure to other users by allowing only a trusted system to access the information.

3. Hermes Communications Middleware

We developed a communications middleware *Hermes* that allows the design and execution of communication flows such as those presented in Section 2.1. The architecture of Hermes is shown in Figure 2. Hermes consists of various layers that define, gather and use context for communication applications. In this section, we describe the layers and how they process context.

3.1. Design Environment: Specifying Workflows that Rely on Context

The design of a flow follows a visual workflow model in a graphical user interface. A flow designer selects nodes from a palette of nodes, drops these nodes onto a canvas, connects them according to the desired control flow, and configures each node for the use of context information by right-clicking on it and filling out a node type-specific property sheet. For example, the property sheet for an alert node will allow the designer to specify, among other things, whether presence, availability, and/or user preferences should be used to alert a user. It allows the specification of *application* context such as the customer claim information in Section 2.1. Workflows created in the design environment are stored in XML definitions. The Hermes *task library* contains the logic for the nodes in the workflow. Hermes provides complex communication tasks such as *conferencing* and *alert* nodes. The Hermes design environment abstracts away the inherent complexities of programming to low-level,

often proprietary, and technology-centric communication server and services APIs. Hermes communication nodes provide simple, easy-to-maintain, vendor-agnostic, and converged control over different communication media (voice, instant messaging, Web, email, etc.) and communication activities (conferencing, alerting, messaging etc.).

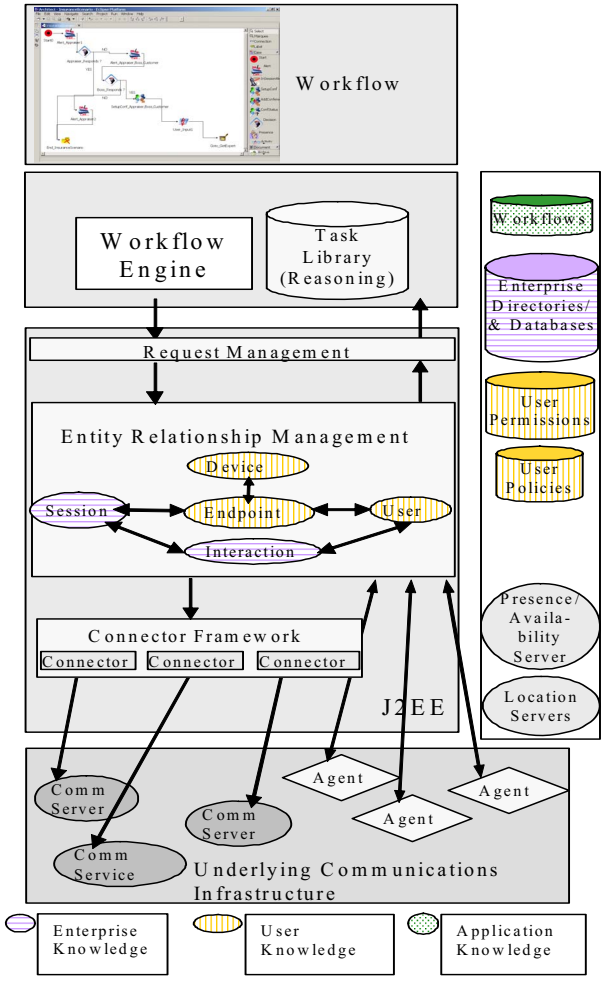

Figure 2: The Hermes architecture highlighting the various aspects of context in the system

3.2. Hermes Execution Environment

This section describes the layers in the Hermes execution environment.

3.2.1. Workflows. Hermes flows can be easily integrated into business processes either at design time through a shared workflow design environment or at runtime by giving business processes access to Hermes flows through a Web Services interface. Once deployed, Hermes flows execute on a workflow engine. The workflow engine sequences the execution of nodes in Hermes flows. The communication tasks access contextual information from different sources and use this information in reasoning processes to determine who, how, and when to contact and connect people. The reasoning processes use algorithms and rules that operate on the enterprise, application, and user context to make the appropriate selections. A detailed discussion of these approaches is beyond the scope of this paper. The communication tasks use the services of the Hermes J2EE components as well as LDAP enterprise directories, databases, and other servers that collect and store relatively static contextual information such as enterprise rules, policies, user profiles, preferences, and roles. When tasks launch communication activities such as phone calls, conferences, sending of messages, they issue abstract communication requests to the Hermes request manager which queues up requests, logs them, and ties them together with the services of the Hermes entity relationship manager.

3.2.2. Entity Relationship Manager: Aggregating Contextual Information. The entity relationship manager administers much of the dynamic contextual information that Hermes tasks rely on to reason about user presence, availability, past and present participation in communication sessions, and communication capabilities.. Hermes communication sessions capture information pertinent to establishing or modifying communications and interactions through functionality in the underlying communication infrastructure, record relevant communication activities of users, enable the delivery of whispers, gather user feedback in response to whispers and notifications, and more. The entity relationship manager reflects all the user and session communication activity that Hermes initiates or that occurs outside the Hermes boundaries and that Hermes can obtain knowledge of. The representation of contextual information occurs in an entity relationship model that includes session, user, interaction, endpoint, device and other entities and through relationships between such entities. The entity relationship model is implemented by persistent Enterprise JavaBeans backed by a relational database. Hermes' entity relationship management is converged across different media and thus allows uniform access to all the contextual information that Hermes administers as well as uniform control over communication activities (setup, start, modification, termination of communication). As part of the reasoning process, Hermes tasks can execute

sophisticated query and navigation operations against the entity relationship model.

3.2.3. Agents: Gathering Contextual information. Hermes communication agents are attached to communication servers and devices and feed raw data about users and their communication activities through a Web Services interface into the entity relationship manager. For example, a communication agent attached to an instant messaging server can relay user presence information about instant messaging clients and conferences to the entity relationship manager. In another example, a communication agent can signal the arrival of a voice mail for a specific user to Hermes.

3.3. Context in Hermes

In this section, we discuss the definition, acquisition, and use of context in Hermes. Context forms a subset of the static and dynamic knowledge about the enterprise, its structure and its processes, users and the user environment, and the application, that is important for a decision making process that requires human communication. Knowledge about an entity (here: the enterprise, the user, or the application) is a set of interrelated facts about the entity. For example, in the scenario discussed in Section 2, solving a critical customer issue is a decision to be made and it requires the scheduling of a conference. The ensuing communication process involves answering three questions as shown in Figure 3: *who* should be invited to participate, *when* the conference should be, and *what media* and devices should be used for the conference. The pieces of information needed for scheduling the conference are the presence and availability information of the users, the environment of the users, the skills, roles and positions of the users in an organization and the requirements of the current application (here the particular customer issue). These pieces of information and their relationships with each other and with the knowledge they are embedded in, form the context.

To facilitate the representation and storage in a relational database, acquisition and use, all this knowledge can be divided into three classes namely the enterprise context, the user context and the application context. Note that these contexts are not independent of each other and rather overlap. We now discuss these different contexts and the associated knowledge spaces in detail.

3.3.1. Enterprise Context The enterprise context is a subset of the enterprise knowledge. The enterprise knowledge consists of the enterprise directory and databases which store information about enterprise processes, deadlines, states of the processes, policies, concurrent and independent communication sessions, needs of the sessions (e.g. document sharing), and social network of the enterprise. The dynamic data about the current sessions, the current processes and the enterprise social network is obtained from the entity relationship manager in Hermes which persists its data in a database. An analysis of communication logs creates knowledge about the social network in the enterprise as discussed in [8]. The enterprise context consists of the subset of the enterprise knowledge relevant to a communication task (e.g. subset of the enterprise social network consisting of the people suitable for the given application) and the values of certain variables such as details of the current communication sessions in progress. The information about social networks in the enterprise as part of the enterprise context helps in choosing a cohesive subset of users from the set of suitable and available users for a given conference as described in [8]. The enterprise context may define constraints on timing and communication media selection.

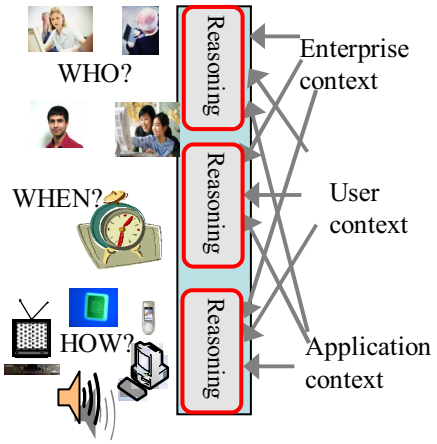

Figure 3: Conference scheduling uses the enterprise, the user, and the application contexts

3.3.2. User Context The user context allows assessing the suitability of a user for a communication event such as a conference, at a given time, and through a given communication medium. The user knowledge consists of interrelated facts about a user. This includes static and dynamic information such as user preferences regarding reachability, social networks, organizational level, roles, skills, properties of the physical environment of the user and presence and availability on different media and devices. The static information such as the user preferences, roles, skills and designations are obtained from the enterprise directory. The dynamic information about the user presence and availability are obtained from the entity

relationship manager. When such information is missing, Hermes enables the estimation of the user presence and availability using a statistical model as in [4].

3.3.3. Application Context We now discuss the application context in relation to the application knowledge. The application knowledge consists of the facts about different application subjects such as the roles of users who participate in a communication (e.g. appraiser, supervisor in section 2.1), customer issues, workflows for the customer issues, hyperlinks (e.g. claim documents in section 2.1), content of messages in the application, requirements such as maximum time for service and other fields depending on the application. This knowledge in stored in the workflows provided by the application developer. The application context is dynamically determined by the workflow engine. The application context provides conditions to identify the suitable set of users for the conference, the time period for the conference and the set of media elements required for communication.

4. Experiences with Hermes

Communication applications similar to the one presented in Section 2.1 have been demonstrated on Hermes extensively at Avaya and at various public forums such as tradeshows. This section briefly summarizes the experience with Hermes gathered from about 200 demonstrations on the system by users who are familiar with it. The experience is described from the perspective of (i) a developer to design context-aware communication applications and (ii) a user that is involved in Hermes-initiated communications.

(i) Developer Perspective: The time it took to configure a communication node (e.g. conference node to connect 3 people) in the design environment varied between 2-3 minutes assuming that the user understood the application requirements beforehand. The time it took to develop an entire application such as the expert finder flow in Section 2.1 varied between 15-20 minutes. (ii) User Perspective: The time it took a user to provide a response to an invitation through a user portal (e.g. the expert response in Section 2.1) varied between 2-4 minutes. The time it took a user to join an interaction triggered by Hermes such as a conference (e.g. the appraiser, supervisor and customer receiving the conference calls in Section 2.1) was on the order of seconds. For the application described in Section 2.1, the execution times were always dominated by the time it took users to respond to an invitation (2-4 minutes for each response step). The above experiences demonstrate the effectiveness of Hermes from the perspective of the application and the enterprise.

5. Conclusions

In this paper we presented the architecture of a platform that serves our vision of the modern global enterprise. The platform models and uses context to make effective decisions for selecting the users, media and time for a communication task in an enterprise. We observed that this context is a subset of knowledge about the enterprise, the users and the application. While we describe the various sources of context information that we use in our work so far, there are many issues that we have not studied yet. We list a few significant ones as candidates for future work: (i) Conduct extensive studies over large user groups and study the effectiveness of the reasoning processes that are used in Hermes. (ii) Identify improved reasoning strategies for scheduling communication tasks such as conferences. (iii) Conduct user studies to understand user satisfaction in interacting with Hermes.

6. References

[1] The Work Management Coalition www.wfmc.org.

[2] A. K. Dey (2000). *Providing Architectural Support for Building Context-Aware Applications*. College of Computing. Atlanta, Georgia Institute of Technology, PhD.

[3] S. Greenberg (2001). *Context as a Dynamic Construct*. Human-Computer Interaction 16: 257-268.

[4] E. Horvitz, P. Koch, C. M. Kadie, et al. (2002). *Coordinate: Probabilistic Forecasting of Presence and Availability*, Uncertainty in Artificial Intelligence, Edmonton, Canada, 224-233, August 1-4, 2002.

[5] H. Lei, D. Sow, J. Davis, et al. (2002). *The Design and Applications of a Context Service*. ACM SIGMOBILE Mobile Computing and Communications Review 4(6): 45-55.

[6] H. Lei, D. Sow and J. Davis (2004). *Context Aware Unified Communication*, IEEE International Conference on Mobile Data Management, January, 2004.

[7] A. Mani (2004). *The Networked Home as a User-Centric Multimedia System*, Workshop on Next Generation Broadband Challenges, ACM Multimedia 2004, New York,

[8] A. Mani and A. John (2005). *Cohesive group selection for a communication task in an organization*. Avaya labs technical report.

[9] A. Ranganathan and R. Campbell (2003). *A Middleware for Context-Aware Agents in Ubiquitous Computing Environments*. Lecture Notes in Computer Science, Springer Verlag 2672: 143-161.

[10] M. Weber, G. Patsch, A. Scheller-Huoy, et al. (1997). *Flexible Real-time Meeting Support for Workflow Management Systems*, 30th Hawaii International Conference on System Sciences (HICSS '97), Hawaii, 559-567, January, 1997.