Intelligent Agents for Seamless Personal Information Networking

Sue Abu-Hakima
AmikaNow! Corporation
700 March Road, Suite 201, Kanata, Ontario, Canada, K2K 2V9
suhayya@amikanow.com
www.amikanow.com

The paper describes a unique cooperative agents testbed that aims at addressing diverse applications for the difficult problem of seamless personal information networking. The real-world testbed is aimed at two difficult applications, Seamless Messaging and intelligent network management. Both applications are agent-driven and share agent behaviours and the messaging agents rely on the network management device diagnostic agents for input. The paper introduces both problem areas in a common testbed. User-centric seamless messaging assumes heterogeneous communication environments intended to support today's nomadic users. The aim of it is to intercept, filter, interpret, and deliver multi-modal messages be they voice, fax, video and/or e-mail messages. A user's Personal Communication Agent is charged with delivering messages to the recipient regardless of their target messaging device be it a telephone, a pager, a desktop, a wireless laptop or a wireless phone. A taxonomy of agents and behaviours for the Seamless Networking testbed is introduced.

1. Introduction

Today's highly distributed workspaces require users to communicate and compute across heterogeneous networks and a multitude of applications. The paper describes the design and implementation of a cooperative agents testbed that aims at addressing applications for the difficult problems of seamless personal information networking. The SPIN testbed is aimed at two difficult applications, Seamless Messaging[Abu-Hakima et al. 96; 97] and Intelligent Network Management. The paper describes the multi-agent framework intended for both problem areas. The remainder of the paper includes: section 2. on What SPIN was about and why Seamless Networking requires agents, 3. on relevant work, section 4. on intelligent network management concepts, 5. on seamless messaging concepts, 7. on future work and conclusions.

2. What is Seamless Personal Information Networking (SPIN)?

Seamless Personal Information Networking is based on the concept that today's user workspace is no longer bound by four walls. Rather, today's workspace is a virtual one with a multitude of applications. Furthermore, people work with networks of desktop and mobile computers, fixed and mobile telephones, fax machines, and pagers, with both hardware and software constantly evolving. SPIN is a simple concept that is quite complex to achieve. To make SPIN a reality, a real-world cooperative multi-agent system (MAS) testbed was put in place to prototype personalised networking applications.

Why use Agents in Seamless Networking?

Many reasons come to mind why agents are an ideal evolution of the traditional distributed computing paradigm for seamless networking. Agents are accepted as software entities that can act autonomously or with some guidance on the user or software system's behalf. Agents are active computational entities that are persistent, can perceive, reason and act in their environments and can communicate with other agents [Huhns and Singh 95]. Agents are ideal for applications that require some distributed intelligent cooperation.

Networks are inherently distributed hence making any centralised intelligent software function to support users unreliable. A centralised system will surely fail in a world with 2 billion telephones and over 140 million Internet users. Networks are managed through their partitioning into subnetworks. Thus, seamless networks designed to support user needs require distributed intelligent processes to manage them. In SPIN we are addressing this need by developing Diagnostic Agents that live in network nodes. These are introduced in section 2. under SPIN's Intelligent Network Management Applications in the MAS testbed.

Users require both heavyweight or complex reasoning processes in the form of personal assistants that can manage their computing and communication needs as well as lightweight or simple processes that can act as proxies on their behalf in the network. As a user roams from place to place, they require distributed support to access their information as the need arises. Active information processes embodied as agents can provide such support.

Intelligent personal assistants that can act autonomously on a user's behalf are essential in dealing with the overflow of information arriving in a user's workspace. If the information is urgent, a user's personal communication assistant has to track the roaming user and deliver the key information content. This is only possible through customisation of personal agents. A personal agent must be created and empowered to find the user as the need arises.

To support mobile users, networks require a form of distributed intelligence. This can be achieved in plug and play networks designed to support nomadic user information requirements. For example, in a seamless messaging application, it may only be possible to deliver an urgent email message to the user over a wireless telephone. Thus, a one page text message must be interpreted by the personal agent and its salient content delivered through a text-to-speech conversion to a wireless phone as a 30 second voice clip. The underlying network the email arrived on may be a typical TCP/IP-based local area network (LAN). Thus, the Personal Communication Agent must receive that message and route it over a wireless network seamlessly to the user. In today's networks where seamless interoperability is not yet a full reality, such an application would be impossible without the mediation capability of a personal agent.

IIT's Living Lab: SPIN's Real-World Multi-Agent System Testbed

For the SPIN vision to be achieved, a real-world networking environment was used as the underlying environment for the agent testbed. At the Institute for Information Technology (IIT) we set up the heteregeneous environment known as the IIT Living Lab. The idea behind the Living Lab was to allow agents to be created, to monitor themselves, to co-exist, to spin-off proxies and to be killed or expire when they have fulfilled their duties. Both the Seamless Messaging and the Intelligent Network Management Application Agents shared the testbed. The intent was for the Personal Communication Agents (PCAs) of the seamless messaging application to make use of the Diagnostic Agents (DAs) of the network management application. For example, if a PCA needed to forward information to a user device that was not responding it could ask the device DAor its parent DA 'why is the active user device not responding?'.

Figure 1 illustrates the IIT Living Lab. It is centered around a typical enterprise LAN. The Institute LAN has daily operational and experimental traffic for an organisation of over 100 users with hundreds of interconnected devices over seven subnetworks. Initially, we are launching the agents into the SPIN subnetwork which has over 30 desktop devices which are also accessible from home by dialing in. The SPIN subnetwork also has network devices such as printers, routers, etc. connected to it. The PCAs will typically reside on the user desktops.

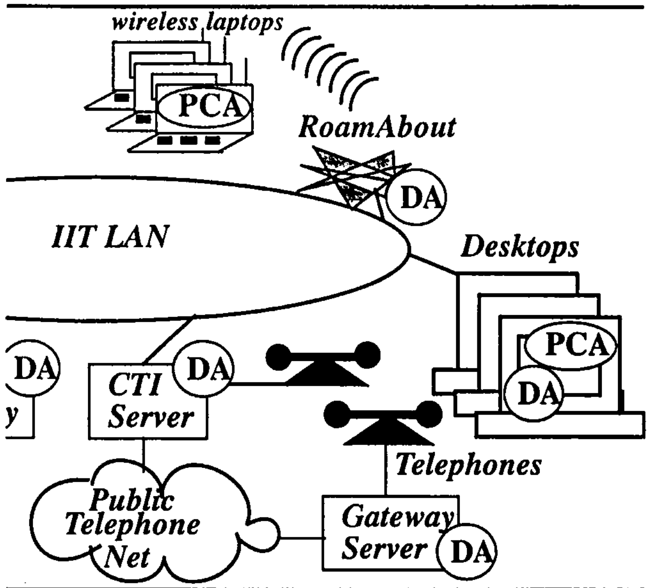

Figure 1: The IIT Living Lab

In the case of a user with a number of desktops (the average at SPIN is three: two at the office and one at home), the user PCA can reside on a single device but monitor activity on the other devices. This will be possible through the DAs which reside on every networked device and work as proxy agents of SNMP agents (SNMP is the Simple Network Management Protocol from the Internet Engineering Task Force). SNMP agents are fast becoming an Industry standard with any new network device [IETF references on SNMP are all on-line at www.ietf.org]. SNMP agents are very simple and function to place respective device information in a Management Information Base (MIB). The MIB is very much like a database that can be browsed by network management tools for simple device monitoring information.

Both the Seamless Messaging application and Intelligent Network Application have agents resident on a respective server. The idea behind the server is to house any agents that would not be device resident. It is also intended to be the place for the persistent agents to be stored if user devices are shut down (as is typically done with today's desktop PCs) or if user devices experience problems. Our strategy will be to maintain a backup of the PCA for the user with a scalable time delay (a day-old backup would be the default but if a user wants one on demand it will be provided). DAs will be backed up in a similar manner. Here we take the lessons from the real-world in computing and communications to plan ahead to avert catastrophe.

The IIT LAN has a Lucent Technologies RoamAbout wireless access point connected to it with its own Diagnostic

Agent proxy to its SNMP agent which monitors operation and access. The RoamAbout allows a user to walk around campus with a wireless laptop. Again, a user PCA may be resident on a wireless laptop or may send a PCA proxy to a laptop with a reduced set of behaviours.

Also connected to the LAN is a Computer Telephony Integration platform (CTI) which allows the user to receive telephone calls with any associated telephony information (incoming caller id, call forwarding info., etc.) to the desktop. The LAN is thus connected to the public telephone system so that it may allow the PCA access into voice mail environments that are normally telephone-driven. The CTI platform will also have a diagnostic agent proxied to its SNMP agent.

Furthermore, a wireless campus base station allows a user to roam about with a wireless phone which the PCA can also route calls to. A paging gateway is accessible through the LAN for the PCA to page the user as instructed. Both the base station and the pager gateway will have diagnostic agents attached to them to monitor their health. Proxy PCAs with simplified behaviours may be dispatched to a cellular phone or pager as long as the device can host it. For the devices to host a PCA proxy, we assume some processing capabilities on board. This will soon be facilitated to some extent as pager and wireless phone manufacturers deliver JAVA-enabled devices with JAVA virtual machines on board. In this manner, the PCA proxy implemented in JAVA can be hosted. At the moment JAVA-enabled pagers and wireless phones are not part of our testbed and we simply rely on device-compatible gateways to deliver text or voice information to the pager or cellular phone.

Finally, an Asynchronous Transfer Mode (A) switch allows the user access to high speed multimedia applications such as video on demand. Most A switches are now SNMP-enabled and thus our DA can work as a proxy to its SNMP agent.

The IIT Living Lab makes an ideal testbed into which we have launched our seamless networking applications. The Living Lab's heteregeneous networks are essentially transparent to the user and are simply used by the Personal Communication Agent to route user information as needed. The networks are also being managed by a set of diagnostic agents which share some behaviours with the PCAs. Furthermore, the PCAs rely on some of the monitoring information from the DAs to fulfil their obligations. Thus, in a *single* testbed, SPIN is ensuring the re-usability of agent behaviours and the interoperability of two agent-based applications. This is a key aspect to our work which we believe is a goal for many researchers but is fairly rare if non existent in other multi agent system testbeds.

3. Related Work

Networks can be seen as a natural domain for the application of distributed artificial intelligence, and more particularly, agent-based computing technology [Reinhardt 94]. In particular, Weihmayer and Velthuijsen suggest a number of reasons for this, including their inherent distribution (e.g. along spatial, functional, and temporal lines), the proliferation of heterogeneous devices and services associated with them (this is particularly true of multi-vendor mixed computing-communications networks), the growing need for privacy, the sustained demands for high performance, and the increasing desire for "intelligence" in the network [Weihmayer and Velthuijsen 94].

Modelling messaging in organization services as a collection of coordinated agents results in a number of benefits. For example, a degree of virtual homogeneity is brought to otherwise heterogeneous networks of computer-telephony messaging services and devices (such as voice mail, e-mail or fax mail); relatedly, a more open network architecture facilitating more rapid and effective deployment of "plug and play" messaging services is made possible. All the same, the agent metaphor does not, in and of itself, directly resolve any of the technical issues related to system interoperability such as sharing remote resources, guaranteeing a particular quality of service or resolving the network feature interaction problem. Rather, as Laufmann points out in [Laufmann 94] "the metaphor provides a model of coordination that addresses real-world issues of the computing and communications marketplaces, and in so doing leverages the deployment of new technical solutions as they become available". In this respect, the *Seamless Messaging Personal Communication Agents* are analogous in scope and purpose to Laufmann's coarse-grained agents or CGAs [Laufmann 94]. The Diagnostic Agents can also be assumed to be CGAs.

These agents, like those supported by the Carnot project's Distributed Communicating Agents (DCA) tool [Huhns et al. 93], are essentially high-performance problem solvers which can be located anywhere within and among networked enterprise resources. These agents are intended to communicate and cooperate with each other, and with human agents. Through the use of models of other agents and resources within the enterprise, SPIN Seamless Messaging and network management agents, like DCA agents, will be able to cooperate to provide integrated and coherent management of information in heterogeneous computing-communications environments [Huhns and Singh 95].

The convergence of networks and the need for personal digital assistants with embedded agents that interface to all the information accessible through a web of networks (Internet, World Wide Web, etc.) has also been cited as a key reason to develop cooperative multi-agent systems [Rosenschein and Zlotkin 94]. Agents are being developed to enable users to seamlessly share valuable messaging services such as voice and e-mail as if in a single seamless network across a variety of devices that include cellular phones, PDAs, wired telephones and wireless laptop computers.

The Multi-Agent Network Architecture (MANA) described in [Gray et al. 94; Abu-Hakima et al. 95] is designed to provide services across communication networks. In some ways, its objectives are similar to those of the concepts in Seamless Messaging. However, its agents are coarse-grained agents which are compiled at run-time and hence lack the flexibility to adapt as would be necessary in a seamless messaging environment. Personal Communication Agents launch proxy PCAs as mobile agents which are fine-grained agents that are light weight specifically designed for reactive situated behaviour. The MANA agents did not incorporate mobile or static fine-grained agents.

Bradshaw and his colleagues in describing KAoS (Knowledgeable Agent-oriented System) describe an open agent architecture that mirrors in its goals what SPIN is aiming for in its testbed [Bradshaw et al. 97]. The goal of the SPIN agent framework is not to take on monolithic reasoning capabilities but rather to tailor their behaviours to the context of their world. Thus, the personal communication agent of a user aims to provide situated personalised functionality to the user while relying on other agents such as the Secretary Agent and the Device Manager Agent to provide more generic functionality. The PCA also relies on the diagnostic agents of the network management application to provide monitoring information if the PCA fails to deliver required personal messaging information to the user. Furthermore, the PCA relies on software modules that are mediated by an agent wrapper or that are accessible through open APIs (Application Programming Interfaces) to fulfil its responsibilities. For example, a PCA does not take on any computer-telephony functionality but accesses the CTI server through a common application interface to gain knowledge about messaging information as it needs it. Thus, in one world, the agents are designed to rely on an open, mediated approach to gaining more functionality without taking it on. In some sense, this is what is known as *good old software engineering* where a system is partitioned carefully to maximise the use of generic blocks of code with well-defined interfaces rather than repeating the blocks for each instantiation.

Jim White of General Magic Inc. introduced TeleScript and its Magic Links with mobile agents in the mid 1980s [White 97]. SPIN in many ways is aiming for some of the functionality that White described in allowing users to be mobile with information that can roam on their behalf in the form of mobile agents. TeleScript and the Magic Links did not succeed as they assumed a closed world that did not interoperate with much of the user's workspace (the telephone being the most obvious example). SPIN is quite focused on user-centric computing and communications. The vision starts from the user workspace and works outwards. We do not assume that we can force the user to buy special purpose PDAs to acquire seamless messaging. We insist on making use of Industry and commonly accepted standards and good user practices. We recognise that for the vision to succeed, we must adopt the user's constraints and recognise that each user has a unique and individual workspace. This is why the PCAs are highly personalised to the user's workspace and habits. The PCAs also assume a very open world with no borders and take it on as a challenge to address user-centric requirements rationally. This is by no means an easy problem. We also believe that General Magic has continued working on their vision with more open languages and platforms.

4. SPIN's Network Management Application

What is NM?

Network Management (NM) is typically achieved in today's complex environments through a Network Control Centre NCC). The NCC has five primary functions: administration, planning, operations, security, and application development.

The *Administration-related* functions concentrate on the policy and procedures of the organization and their reflection in the implemented network. A key administrative function is ***accounting*** for both the network resources and its use.

The *Planning-related* functions include the design and the evolution of the network. Initially, the network is designed with a certain margin (sometimes 50%) for growth. The network is continuously evaluated and its capacity monitored for expansion. In addition, the planning functions may include the provision of any training required to better use the network.

The *Operations-related* functions are the primary concern of a network manager, mainly to *"keep the network running"* for the users. Operations include ***monitoring*** events, ***trending*** the performance of the network, ***fault recovery***, taking configuration actions when needed, issuing trouble tickets, the provision of a help desk and providing technical support when needed.

The *Security-related* functions include maintaining the integrity of the data on the network and putting in place procedures to prevent a breach of the network by unwanted intruders (or hackers).

NCC functions often include some specialized activities that are Application Development-related. These may include setting up electronic mail, specialized databases or workflow management software such as Lotus Notes.

The five activities that are included in the five layers of the network management standard are accounting, monitoring, performance trending, fault recovery and security. (These are shown in boldface in the text above.) Many vendors developing network management tools strive to include these five key functions in their products.

Where agents fit into SPIN's NM?

SPIN's strategy is to make use of the adopted Industry standard known as SNMP. Through the SNMP agents, we have developed proxy Diagnostic Agents (DAs) that can monitor a device and provide some self-healing functionality. The DAs have both a peer-to-peer and an adopted parent DA relationship. In a future paper we will provide greater detail on the NM agents in the SPIN MAS testbed.

5. SPIN's Seamless Messaging Application

What is SM?

Seamless Messaging provides users with the capability to work freely in distributed personal workspaces. It allows the creation, encoding, filtering and delivery of messages across heterogeneous networks. Thus, users can seamlessly deliver voice or electronic mail to wireless or wired mail environments. The SM paradigm requires that the recipient of the message be located through an electronic secretary and the message be tailored to the recipient's active device user interface. As such, SM is not an easy endeavour.

Why is SM important?

In today's distributed environments users are often faced with multiple messaging environments (e.g. voice mail, e-mail, fax mail or video mail) that do not interwork intelligently, if at all. The results are that users are overwhelmed with hundreds of messages in different messaging boxes. Many of these messages are unimportant or junk mail. As a result, users are calling for methods of seamlessly integrating these message boxes and of intelligently filtering their contents.

SPIN's Seamless Messaging Objective

The objective of the Seamless Messaging project is to achieve end-to-end personalized communication of information in distributed workspaces. This is a challenging project because of the heterogeneity of the personal distributed workspace.

Where agents fit into SM?

Each user can customise a Personal Communication Agent (PCA) through a PCA launch tool. This tool allows the user to tailor PCA behaviours in terms of classifying and acting on incoming messages from email, voice mail, fax, and video mail environments in a multi-agent system as detailed below.

6. Taxonomy of Agents for Seamless Networking

A set of agents and behaviours can be defined to satisfy the multiu-agent testbed. This section enumerates these agents and their behaviours.

6.1 SPIN's Agent Class

The top level agent class is simply the class Agent. This Agent has a set of generic functions that can be inherited by any of its subclassed agents. These generic functions can also be absent from its agent subclasses as we assume a loosely coupled rather than a strict inheritance hierarchy.

6.1.1 Generic Behaviours

Generic agent behaviours are accepted to include the agent's ability to sense its environment, its ability to reason, its ability to adapt, its ability to communicate with other agents and its abitlity to effect its environment. These behaviours are individually complex behaviours which have specilaties within them. They are described in the subsections that follow.

6.1.1.1 Sense Behaviours

Agent sensing behaviours can initially be primitive and then can evolve to more complex ones. Three classes of agent sensing behaviours are required: detecting sound, detecting images, and detecting keywords from text.

6.1.1.2 Reason Behaviours

- reason based on rules
- reason by navigating a semantic network. A semantic network represents linked entities which can be reasoned about. In AI these objects are most often represented as frames with active fields. A semantic network can be organised as an is-a or part-of hierarchy or as a network of relations between entities.

6.1.1.3 Adapt Behaviours

A set of agent learning mechanisms can include:
- a user explicitly instructing an agent
- an agent learning by monitoring or learning by rote
- an agent learning via case-based reasoning or example
- an agent learning through reinforcement learning

6.1.1.4 Communicate Behaviours

A set of agent communication mechanisms that can include:
- cooperation with another agent to solve a common goal
- coordinating with another agent
- competing with another agent
- broadcasting a goal
- identifying an agent & sending it a goal

6.1.1.5 Effect Behaviours

A set of agent mechanisms that can include:
- sensing (accept input - see, listen)
- reason (based on rules/more complex structures: semantic network, hierarchies)
- learn/adapt (modify knowledge structures)
- communicate/effect (share info. with another agent, accept a goal, respond to a query, etc.)

6.2 Agents for Seamless Messaging

Seamless Messaging agents inherit the behaviours of the Agent Class. The main agent is the Personal Communication Agent (PCA). Each user with seamless messaging is has a PCA which is personalised to their messaging needs. The PCA is supported by several other agents and behaviours as described below.

6.2.1 Personal Communication Agent (PCA) for Seamless Messaging

Ascripted agent whose role is to manage all the user's messaging needs. Its role is to accept a message and have it classified then acted on by the classifier and action behaviours

6.2.1.1 PCA Behaviours

These include partitioned behaviours such as:
- classify messages
- request Service Provider action on message
- request e-Secretary to look at the user's schedule
- request PeopleFinder to rank best guesses where user is
- act on message

6.2.1.2 PCA User Profile

The PCA User profile will include:
- user addressing information (Name, email, fax, telephone number)
- list of user devices
- list of user preferences re-device usage and messaging constraints
- directory of user's personal communication contacts

6.2.2 Message Watcher Agent

This agent waits for a message event in the form of a phone call (voice mail,, fax), desktop communication (email, multimedia message, fax, video message), fax machine communication(fax)

6.2.3 Secretary Agent

This agent is the e-Secretary which:
- receives queries about the user whereabouts
- can call on user calendar through Sense User Agent
- can call on PeopleFinder Agent for additional user sensor information

6.2.4 PeopleFinder Agent

The PeopleFinder agent cooperates with Sense User Agents and ranks best guesses where user is based on Sense User Agents.

6.2.4.1 Sense User Calendar Agent

This agent checks the user's electronic calendar to ascertain active device. The active device may either be on user or in meeting room/office/site where user is.

6.2.4.2 Sense User Desktop Agent

Sense User Desktop Agent looks for activity on user desktop device. The user desktop can either be at a home site or at a forwarded visitor site.

6.2.4.3 Sense User Wireless Laptop Agent

Sense User Wireless Laptop Agent queries the RoamAbout wireless LAN access point about any user activity from an identified user laptop that he/she may be roaming with.

5.2.4.4 Sense User Agent Off-Hook Agent

Sense User Off-Hook queries the Computer Telephony Integration Server if the user's desktop telephone is off-hook. This agent receives the telephone number the user could be at and queries the CTI box for the latest off-hook information.

6.2.4.5 Sense User Cellular Phone Agent

This agent queries the Wireless Service Provider if the user is active on the cellular phone. The query can be launched to an external wireless service provider or to an internal campus provider.

6.2.4.6 Sense User Pager Agent

This agent queries the user 2-way pager if the user is active and requests the latest paging information

6.2.5 Service Provider Agents

Service Provider Agents are used by the PCA to transform messages from one form to another (voice, text, fax, video, audio). They also provide connections that the PCA may require to arrive at the user in the form of telephony,world wide web,

6.2.5.1 Text-Speech Agent

Converts a text string to an utterance through a speech synthesis facility.

6.2.5.2 Speech-Text Agent

Recognises spoken speech through automatic speech recognition and outputs text.

6.2.5.3 Fax-Text Agent

Takes an encoded fax message as input and decodes it to an image. The text on the image is then recognised through Optical Character Recognition.

6.2.5.4 Text-Fax Agent

Converts a text message which may include graphics to an encoded fax message to be output to a fax receiving device.

6.2.5.5 CTI Telephone Connection Agent

The CTI Telephone Connection agent can provide a telephony connection on demand. It expects a telephone number to connect to and it attaches the open voice channel it is given to the called one (much as a conference connection is mad ein typical Call Control).

6.2.5.6 Web Connection Agent

The Web Connection agent provides on demand access to the world wide web. It may use either Netwscape or Microsoft Explorer as the web connection based on the environment variables it is provided to link to.

6.2.5.7 Pager Connection Agent

The Pager Connection agent provides access to a 2-way paging connection on demand. As a secondary function, the Pager Connection agent can also send a Page command with a set of digits to an external paging service.

6.2.5.8 Wireless Carrier Service Agent

The Wireless Carrier agent provides access to a wireless service on demand. It can link two ends of a wireless call together or simply dial-out a requested number to contact a particular wireless device.

6.2.5.9 Wireless RoamAbout Service Agent

The Wireless RoamAbout Service agent provides a connection on demand to a wireless LAN link.

6.2.6 Device Manager Agents

Device manager agents are responsible for delivering information to a user device on demand. The Device Manager agent is directed by the PCA to access a particular user device.

6.2.6.1 Telephone Agent

A Telephone agent receives information from the Device Manager and accesses the CTI Telephone Connection service provider agent to establish a telephone call. It also makes use of the respective service provider agent for conversion to voice, for example, a Text-Speech service provider agent to ensure that text is sent as voice.

6.2.6.2 Desktop Agent

A Desktop agent receives information from the Device Manager and forwards it to the user desktop. The desktop is identified through the PCA User Profile addressign information. This may have to eventually be converted to an IP address.

6.2.6.3 RoamAbout Laptop Agent

A RoamAbout Laptop agent delivers information to the user's wireless laptop. It again makes use of the PCA User Profile addressing information to identify the conrrect laptop.

5.3 Agents for Network Management

Network Management agents inherit the behaviours of the Agent Class. The main agent is the Diagnostic Agent (DA) which is a proxy agent to the SNMP agent in every device in SPIN's multi-agent system testbed. The SNMP agent interacts with a device Management Information Base (MIB) which can be read by other SNMP agents in the testbed. The idea behind the DAs is to allow network devices some autonomous self-healing capabilities. These allow a device to monitor itself, to perform some diagnostic capabilities and to "call for help" from a peer DA or a pearent DA. Each DA has a parent DA and a set of peer DAs. The grouping of DAs around peers and parent agents relies on the subdivision of the network into subnetworks. In the SPIN MAS five parent DAs will exist with a number of subordinates. It may be worthwhile to partition the network around device functionalities. For example, the PC-compatible devices can have a set of DAs subordinate to a parent DA, the Sun workstations can have a parent DA, the telephony devices (CTI server, the PCS Base Station, etc.) can have a parent DA, the A subnetwork (Fore, Newbridge, Passport switches) can have a parent DA, and the photonic network (connected to the A subnet) can have a parent DA.

5.3.1 SPIN's Diagnostic Agent (DA) for Network Management

The differentiation between parent DAs versus subordiante DAs versus peer DAs will be accomplished through DA behaviours. As device specifics in terms of self monitoring, diagnosis or event filtering are captured, some device-specific behaviours for PCs, Sun devices, A switches, Photonic components, may need to be added to DA behaviours.

5.3.1.1 DA Behaviours

- The set of DA behaviours include:
- subordinate DA behaviour
- peer linkage identification
- peer-to-peer communication behaviour
- self monitoring
- local filtering of alarms
- self diagnosis
- call-for-help behaviour (to peers and/or to parent)
- interact with PCA

5.3.2 DA Parent Agent Behaviours

The set of parent DA behaviours include:
- parent DA behaviour
- monitor subordinates
- local filtering of alarms
- respond to queries from parent
- inform parent of significant events
- interact with human user
- interact with PCA

7. Conclusions and Future Work

This paper has described a MAS testbed for Seamless Messaging and Network Management. We have also introduced a taxonomy of agents for this multi-purpose testbed. This work is continuing in commercial products from AmikaNow! Corporation (www.amikanow.com).

Acknowledgements

I would like to acknowledge my old SPIN team at the Institute for Information Technology at the National Research Council which included Roger Impey and Ramiro Liscano who contributed to this work between 1995 and 1998. My new company, AmikaNow! corporation is commercializing this technology.

References

[Abu-Hakima et al. 97] Abu-Hakima S., Liscano R., and Impey R., "Integrating Multi-Modal Messages across Heteregeneous Networks," Proceedings of the First IEEE Enterprise Networking Mini-Conference, ICC 1997, Montreal, Quebec, Canada, 11-12 June , 1997, pp. 45-53.

[Abu-Hakima et al. 96] Abu-Hakima S., Liscano R., and Impey R., "Cooperative agents that adapt for seamless messaging in heteregeneous communication networks," IJCAI-96 Workshop on Intelligent Adaptive Agents, August 4, 1996, Portland, Oregon, Technical Report WS-96-04, AAAI Press, Menlo Park, California, pp.94-103.

[Abu-Hakima et al. 95] S. Abu-Hakima, I. Ferguson, N. Stonelake, E. Bijman, and R. Deadman. A Help Desk Application for Sharing Resources Across High Speed Networks Using a Multi-Agent architecture. Proceedings of the workshop on AI in Distributed Information Networks, IJCAI-95, Montreal, Quebec, August 19, 1995. pp. 1-9. (available from editorial@ai.iit.nrc.ca)

[Bradshaw et al. 97] J.M. Bradshaw, S. Dutfield, P. Benoit, and J. D. Woolley, "KAoS: Toward an Industrial-Strength Open Agent Architecture", Chapter 17, Software Agents, AAAI / MIT Press, 1997.

[Ferguson and Davlouros 95] I. Ferguson and J. Davlouros, On Establishing Multi-Sensory Multi-Channel Communications Among Networked Users. Proceedings of the workshop on AI in Distributed Information Networks, IJCAI-95, Montreal, Quebec, August 19, 1995. pp. 1-9. (available from editorial@ai.iit.nrc.ca)

[Gray et al. 94] T. Gray, E. Peres, D. Pinard, S. Abu-Hakima, A. Diaz, and I. Ferguson. A Multi-Agent Architecture for Enterprise Applications. In *Working Notes AAAI-94 Workshop on Artificial Intelligence in Business Process Re-engineering*, Seattle WA, July 1994, pp. 65-72. NRC 37140.

[Huhns and Singh 95] M.N. Huhns, and M.P. Singh, Cooperative Information Systems, IJCAI-95 tutorial Notes, August 21, 1995, Montreal, Quebec, Canada, 68 pages. Published by IJCAI.

[Huhns et al. 93] M.N. Huhns, N. Jacobs, T. Ksiezyk, W. Shen, M. Singh, and P.E. Cannata. Integrating Enterprise Information Models in Carnot. In *Proceedings of the First International Conference on Intelligent and Cooperative Information Systems*, 1993.

[Laufmann 94] Laufmann, S. The Information Marketplace: The Challenge of Information Commerce. In *Proc. of the Second International Conference on Cooperative Information Systems*, May, 1994, pp. 147-157.

[Reinhardt94] A. Reinhardt. The Network with Smarts. *Byte*, October, 1994 pp. 51-64.

[Rosenschein and Zlotkin 94] J.S. Rosenschein and G. Zlotkin. Designing Conventions for Automated Negotiation. *AI Magazine* 15(3), 1994, pp. 29-46.

[Weihmayer and Velthuijsen 94] R. Weihmayer and H. Velthuijsen. Application of Distributed AI and Cooperative Problem Solving to Telecommunications. In *Proc. of the International Workshop on Distributed Artificial Intelli-*