A Help Desk Application for Sharing Resources Across High-Speed Networks Using a Multi-Agent Network Architecture

Abu-Hakima, Suhayya; Ferguson, I.; Stonelake, N.; Bijman, E.; Deadman, R.

A Help Desk Application for Sharing Resources Across High Speed Networks Using a Multi-Agent Network Architecture

Suhayya Abu-Hakima[1], Innes Ferguson
Seamless Personal Information Networking Group, Institute for Information Technology
National Research Council of Canada, Building M-50, Montreal Road, Ottawa, Canada, K1A 0R6

Nancy Stonelake, Ed Bijman, Richard Deadman
MITEL Corporation, 350 Legget Drive, P.O. Box 13089, Kanata, Ontario, Canada, K2K 1X3

Abstract

The advent of complex heterogeneous network applications requires the use of distributed and intelligent mechanisms for sharing resources. In this paper, a help desk service that demonstrates the use of multi-agent systems for sharing multimedia resources across a high speed network is described. The help desk is modelled as a practical service within an organization using a service provisioning Tool for Enterprise Definition and then compiled into a set of agents as defined by the Multi-Agent Network Architecture. The help desk is a real-world computer-telephony integration (CTI) application implemented using a multi-agent system. It demonstrates the practical use of multi-agent systems for sharing distributed resources such as speech recognizers, databases and World Wide Web Mosaic servers and clients. It is also the first successful CTI prototype of the service provisioning tool and the agent architecture.

Introduction[1]

The advent of complex networks that integrate telecommunications, cable, wireless and satellite applications requires the use of highly distributed and intelligent network management mechanisms. It is no longer acceptable to rely on monolithic systems such as large telephone switches and mainframes to support complex network applications. There is also a need to provide tools to dynamically re-configure networks as failures and bottlenecks in resource use arise. A key problem in the new heterogeneous networks is the modelling and equitable sharing of remote resources by new network management tools. A Tool for Enterprise Definition (TED) has been developed to address the problem of modelling the shared organizational resources that provide services across networks. TED generates a description of a service within an enterprise network that is then compiled into a set of runtime Multi-Agent Network Architecture (MANA) agents [Gray et al. 94].

This paper describes a help desk application implemented using multiple agents that share resources distributed across a high speed network. The help desk is first modelled as a service within an organization. The modelled service is then compiled into a set of agents. The help desk agents use shared network resources designed to support MITEL Technical Service Support (TSS) personnel in responding to field service queries about Mitel products. A technician servicing Mitel equipment is expected to call the Mitel help desk. The technician is then lead through a dialogue with the automated help desk to focus in on the specific product they need help with. Once all supporting customer information is collected using the multi-agent help desk service, the call and information is then automatically routed to an appropriate TSS.

Several contributions are made in this work. The automated help desk is a real-world computer-telephony integration (CTI) application implemented using a multi-agent system. It demonstrates the practical use of multi-agent systems for sharing distributed resources such as speech recognizers, databases and World Wide Web (WWW) Mosaic clients. It is also the first successful application of the Tool for Enterprise Definition and the Multi-Agent Network Architecture designed as general purpose CTI tools for an enterprise.

In the remainder of the paper relevant work is first described. This is followed by a description of how services within an organization can be modelled using TED and then compiled to MANA agents. A description of how the help desk is modelled by a service designer with a tool such as TED is also included.Details on how the modelled help desk service is implemented by MANA agents is also given. Finally, the paper is concluded with some ideas for future work.

1. email: suhayya@ai.iit.nrc.ca, tel: (613) 993-8551.

Relevant Work

Wide area communications 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 such as that used by MANA agents), the growing need for privacy, the sustained demands for high performance, and the increasing desire for "intelligence" in the network [Weihmayer and Velthuijsen 94].

A primary objective of the MANA architecture is to facilitate the modelling of organizational services through decoupling of service definition and service provisioning [Weiss et al. 95]. Through modelling organizational services as collections of coordinated agents, a number of benefits are made possible. For example, a degree of virtual homogeneity is brought to an otherwise heterogeneous network of computer-telephony services and devices; relatedly, a more open network architecture facilitating more rapid and effective deployment of "plug and play" 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 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, MANA agents [Gray et al. 94] are analogous in scope and purpose to Laufmann's coarse-grained agents or CGAs [Laufmann 94].

MANA agents, like those supported by the Carnot project's DCA tool [Huhns et al. 93], are essentially high-performance problem solvers which can be located anywhere within and among enterprises (at present, Mitel Corporation and the National Research Council of Canada in Ottawa). These agents can communicate and cooperate with each other, and with human agents. Through the use of models of other agents and resources within the enterprise, MANA agents, like DCA agents, can cooperate to provide integrated and coherent management of information in heterogeneous computing-communications environments.

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]. MANA agents have been developed to enable users to share valuable resources such as databases and speech recognizers across networks. In Rosenschein and Zlotkin's telephone call competition example, an embedded telephone agent must negotiate the best deal for a customer call between three bids from telephone carriers (e.g. MCI, AT&T or Sprint). A MANA agent achieves this and more in allowing its broker to collect and choose a winning bid from potential suppliers of a resource (e.g. a telephone carrier, a speech recognition provider, etc.) based on an organizational policy. What is interesting in a MANA agent, is its ability to cooperate with other agents based on changing organizational policies input through the TED service provisioning tool.

Various tools have been developed to model organizations and streamline their processes [Cypress 94]. According to Cypress we are now moving toward the development of a second generation of re-engineering tools. The first generation of tools included implementations for management of customer orders and services, production, manufacturing, information access and inventory. He believes that a second generation of tools that streamline technical, social, innovation and enabling processes in an organization will follow. These tools are intended to provide seamless communications in organizations and enhance cooperative work environments. TED and MANA will play a key role in second generation re-engineering processes within an organization.

Several world initiatives are underway to provide seamless communications networks. RACE is the European initiative for advancing technology in communications. ROSA is the RACE Open Services Architecture [Oshisanwo et al. 92]. ROSA is the definition of an open architecture for integrated broadband communications services. This architecture is designed to be open to meet the rapid changes in software, hardware and networking. Its goal is to provide network services in the form of enterprise capabilities to all users. ROSA is not yet implemented and is based on modelling network interaction using an object model. MANA is an alternative to the ROSA architecture. The two architectures differ in the specification and interactions between network entities. In MANA, an agent rather than an object mode, is used. The agent paradigm makes MANA a more powerful architecture for modelling cooperative work in an enterprise.

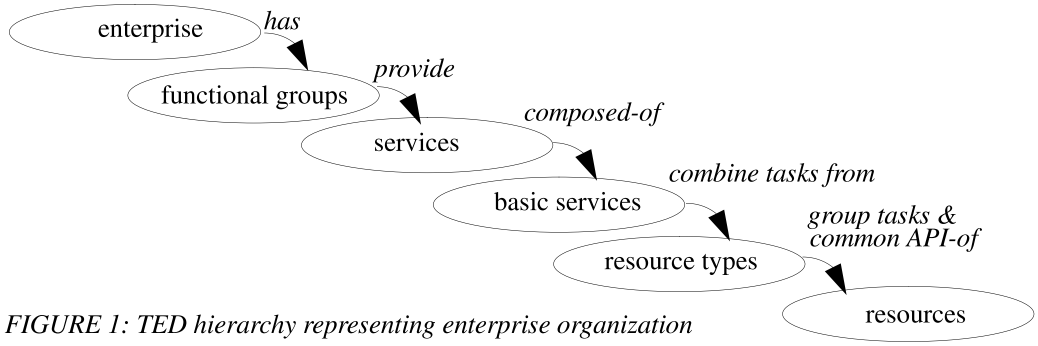

FIGURE 1: TED hierarchy representing enterprise organization

Modelling Resources in an Enterprise

The objective of the Multi-Agent Network Architecture is to decouple service definition from service provisioning [Weiss et al. 95]. The Tool for Enterprise Definition permits a service designer to iconically model a functional group within an enterprise and the services it performs in terms of its underlying resources. TED organizes an enterprise into a loose hierarchy (shown in Figure 1) of six layers: enterprise, functional groups, services, basic services, resource types and resources.

At the lowest level, resources (e.g. a speech recognizer, a WWW Mosaic server, a phone line or a database) are grouped into resource types. The resource types group resources based on the tasks they can perform and a common resource application program interface (API). Basic services combine tasks from resource types. All resource types in a basic service must be capable of performing all the tasks of the basic service but may have different APIs. A resource type can be shared by any number of basic services. Basic services which are organizationally related are combined to provide services. At the highest level, within an enterprise, functional groups combine services.

Figure 2 illustrates the three step process by which TED services are transformed to MANA agents. The services are first modelled with TED (a detailed description of the help desk service model is given in a later section). TED models are then compiled to MANA agents (again, this is described in detail for the help desk application agents). Thirdly, MANA agents cooperate to provide the runtime services.

At setup, TED maps its functional groups, basic services and resources to MANA agents as described for the MANA agents in the help desk). A MANA agent has three main areas, a setup, an act and an information area (see Figure 3). The setup area contains the capabilities definition mechanisms the agent uses to set up its act area and the admission control mechanisms that provide the agent with policies for the use of other agents [Gray et al. 94]. The setup area is scripted when an agent is created and is not active during runtime; for this reason the mechanisms of setup are shown in a dashed rectangle in Figure 3.

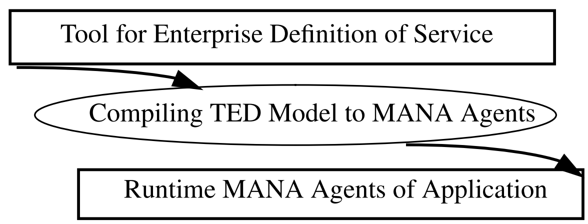

FIGURE 2: Creating MANA agents in an enterprise

The information area shows the status of an agent and is read by other agents which have the right to examine it dynamically when needed. The act area consists of goal resolution, session control and resource allocation mechanisms. Note that the goal resolution is not illustrated pictorially since its functionality is subsumed within session control. The goal resolution and session control provide the runtime agent with data, a control, an initialization and an exception script so that it may receive and resolve goals. The resource allocation mechanism combines brokers for resources and respective bidders with servants that have appropriate APIs to resources. Servants are used to interface to the physical devices essential in providing the basic services of the application (e.g. a speech generation servant would have an API that interfaces to a respective physical device that provides speech generation).

The process of compiling TED models to MANA agents transforms the defined enterprise service into agents and agent mechanisms. Specifically, TED service provision of resource types map to MANA servants which play a key role since they hold the APIs to interact with the resources provided by the physical devices.

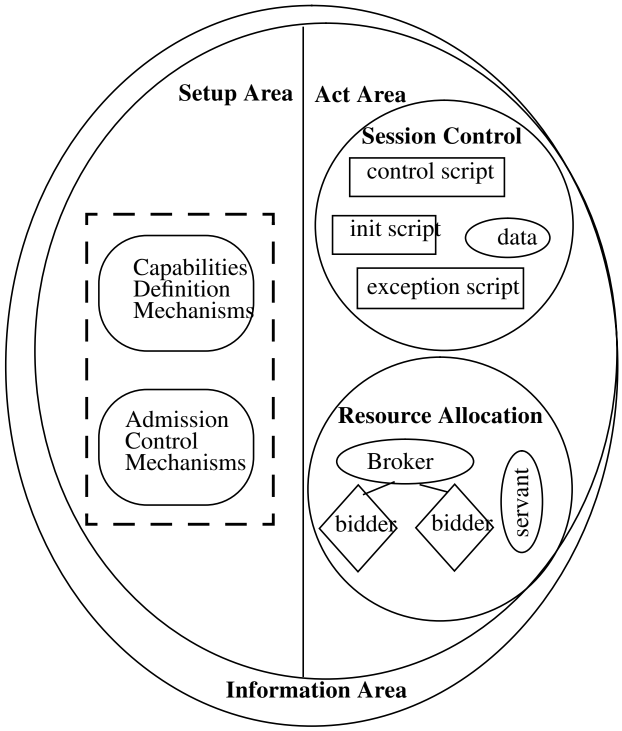

FIGURE 3: Basic MANA agent template

A functional group in TED maps to a MANA agent which provides the functional group service at runtime. The path defining the service is downloaded as a script of goals to achieve the service in the service agent. TED basic services map to MANA agents which can provide servants to a functional group agent at setup. At runtime the basic service is encapsulated into a broker within the functional group agent. Resources map to MANA resource agents at setup. At runtime, a broker requests bids from the representatives of the resource agents (bidders) and then selects the respective servant of the winning bidder.

Modelling the Help Desk

The help desk application (shown in Figure 4) is triggered by a service technician calling the customer service support line at Mitel. The call control mechanism provided by the Mitel switch or PABX receives the call and connects the line to an available speech generator and a speech recognizer resource either at Mitel or at the National Research Council (NRC) of Canada. The resources are connected across the Ottawa-Carleton Research Institute high speed broadband network (OCRINet). The first episode in the dialogue which welcomes the caller to Mitel's automated help desk is played. Within this first episode, the caller is asked to speak their identification (id) number. If the identification number matches a pre-existing trouble ticket in the customer database registered from an earlier call (shown in Figure 4 at the Mitel site), the caller is routed to the human Technical Support Specialist (TSS) who serviced them earlier. In this case, the speech resources are deallocated and the customer information is displayed on the TSS screen as the caller is routed.

If the identification number matches a field service representative id in the database, the second speech episode asking for the customer id as authentication is generated. If the id matches that expected in the database, the next speech episode is played and it asks for the product that the representative is trying to service.

Once the speech recognizer identifies the product, the caller is routed to the respective TSS and a trouble ticket is automatically generated and stored in the database. In the case of no matches with any of the database information, the caller is politely told that there is an error in their identification, they are given the option to call another customer service number to investigate the problem with their id and the automated help desk disconnects their call.

The simplified TED service view of the field service representative calling the automated help desk is shown in Figure 5. Each box indicates a basic service, the grey and white decision boxes indicate an operation with a resultant and the grey boxes indicate race condition operators where multiple paths join.

The service is triggered on an incoming call as indicated by the basic service "await notification of call". On receipt of notification, a speech generator (SG) and a speech recognizer (SR) are allocated and connected (since there is a choice of resources local to the help desk at Mitel and others remotely connected through the network at NRC). The first speech episode greeting the caller is generated and the caller's response recognized. If the caller speaks a trouble ticket number (implying this is a case already assigned to a TSS), the right branch of the service flow is followed and the number is compared to that in the database.

If the trouble ticket is successfully matched to one in the database, the caller is simply routed to the respective TSS agent for the product to be serviced, the speech resources released and the TSS service path for answering the call is entered. The "TSS answer call service" would be another functional group service modelled with TED requiring resources such as Mosaic (to interface to product documentation) and TSS personnel to help the service representative troubleshoot the product of interest.

If the caller did not speak a trouble ticket number, any database records that match the id spoken by the

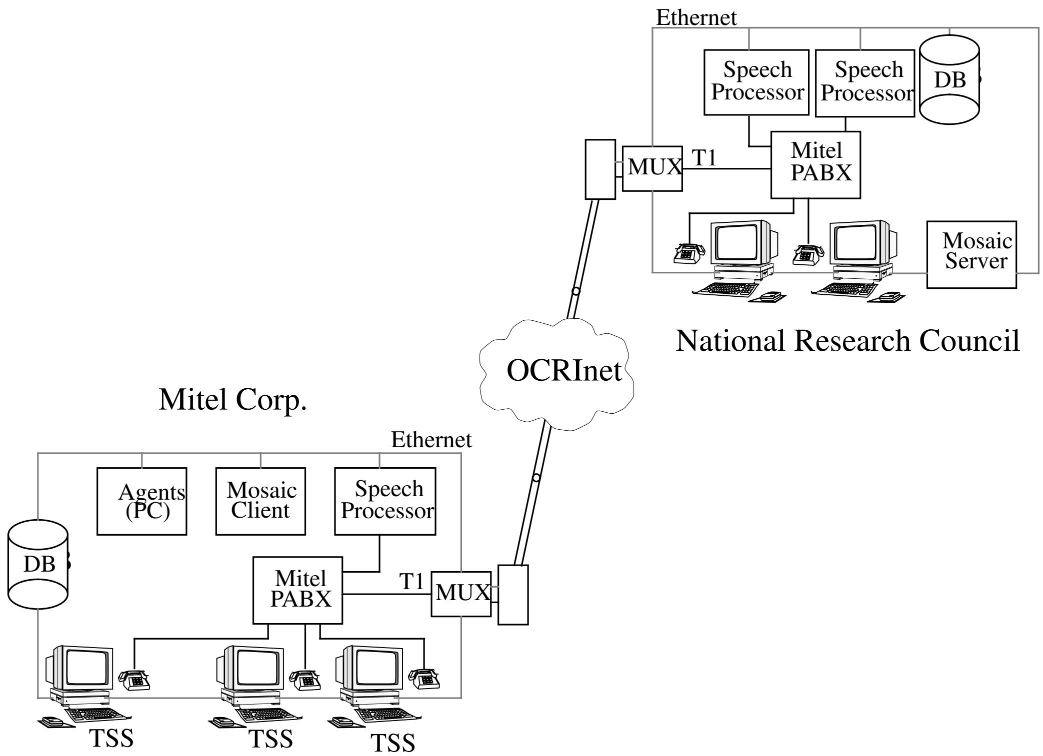

FIGURE 4: High-speed broadband multimedia network used for help desk application

caller are extracted. If a record that matches is found, then the speech episodes that identify the customer id and the product in question are generated and the caller responses recognized. Note that the rectangular box entitled "voice activated dialogue to extract customer information" is a simplification and combines three episodes of "generate and recognize speech and match to customer database records".

If the caller interacts successfully with the help desk and the customer information is collected, a new trouble ticket is created by a basic service and the race condition of the right path of the service is continued.

If the caller fails at any point during the service to provide the required information, the left hand exception path is followed. The race operator shows the joining of the four paths. An "error" basic service outputs a polite message to the caller using the speech generator. The speech resources are subsequently released in preparation for further incoming calls.

Runtime MANA Agents for the Help Desk

As described under modelling resources in an enterprise, TED service models are compiled into MANA agents. Specifically, for the help desk Support Services Functional Group, the six runtime MANA agents of Figure 6 are generated. Three additional resource agents are generated: a WWW Mosaic resource agent and two TSS personnel agents. They are not shown since they relate to TED's Technical Service Support functional group model rather than the Support Services group.

The paths of the TED service in Figure 5 form the episode control script, the initialization script and the exception script in the act area of the agent shown in Figure 6. Each script is composed of the goal resolution scripts for any of the goals that can be expected from other agents interacting with the Support Services agent. The data introduced into the session control is any data required by the agent to act in response to other agent goals in various contexts at runtime.

As described in the section describing the modelling of resources in an enterprise, resource types map to MANA servants. Basic service tasks such as allocate, connect and release speech resources are tasks associated with the resource device types at Mitel and NRC. The speech recognizer and generator physical devices at the two sites have different APIs which require different servants within the agent. The servants hold the APIs necessary to interact with the physical device resource agents (e.g. the "await notification of call" basic service is provided by a servant: "call control"). A Sybase database servant holds the API to interact with a database resource's customer information.

The string and set operation servants have APIs for software resources that fulfil the tasks of comparing the information received from the caller with that extracted from the database.

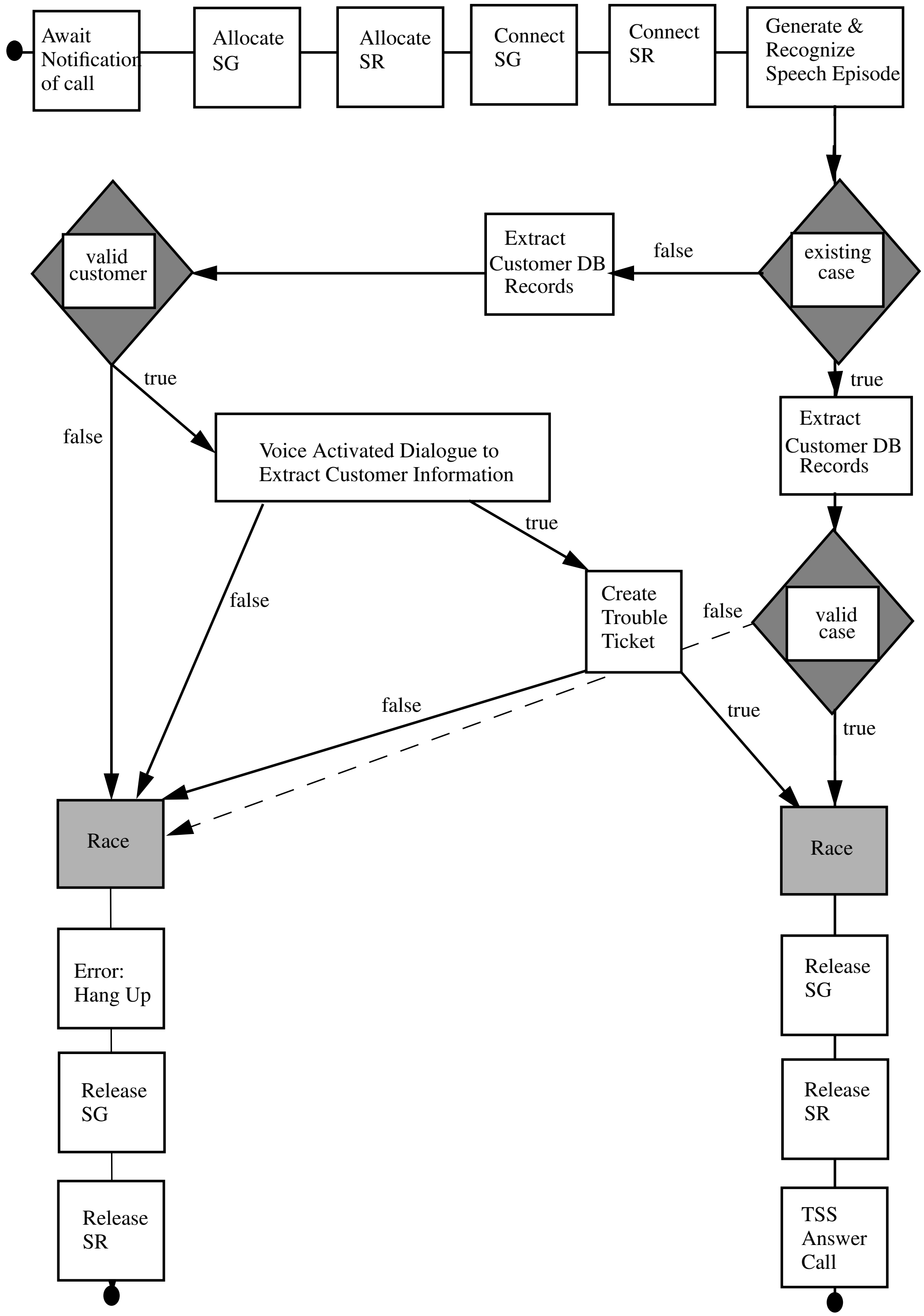

FIGURE 5: Support Services functional group view of help desk in TED

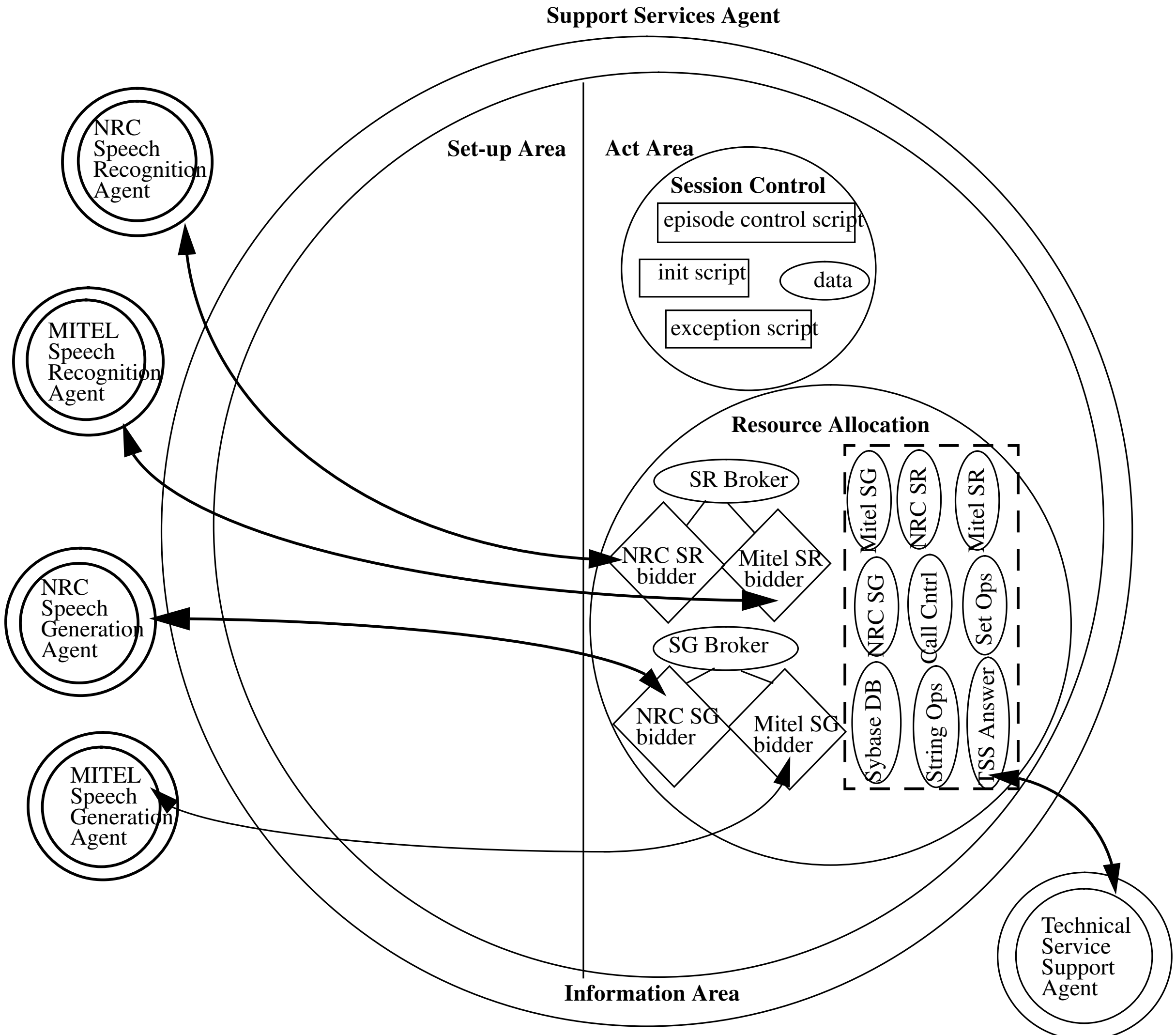

FIGURE 6: Runtime Support Services agent with subset of resource agents and TSS agent

Finally, the "TSS answer servant" interacts with the Technical Service Support agent when the information is collected from the caller and the call is to be routed to a human TSS with the supporting database information. All intra-agent interaction is achieved through the posting of goals from one agent to the next. The goals are resolved through the use of the session control scripts as previously described.

The Technical Service Support agent is similar in complexity to the Support Services agent. Together with the resource agents, the two agents cooperate to provide the help desk service. In TED the TSS service model includes the identification of the caller's product type through the speech recognizer and generator resources. Once the product is identified, the respective human TSS is identified to receive the call. If the TSS is immediately available, they are notified and their screen updated with the relevant customer database records from the caller and the supporting troubleshooting information displayed in Mosaic.

Thus, the TSS agent has the speech recognition and generation servants with the respective APIs for the Mitel and NRC resources. In addition, it has a Sybase database servant so that it may interface to the customer database records, as well as a Mosaic servant and a personal computer (pc) servant to display the prompt to the TSS (since the human TSS is supported by a pc). It also has the call control servant for receiving the routed caller and a string operations servant to identify the TSS to handle the product type uttered by the caller. Each resource type has its respective broker. Thus, the agent has four brokers: a speech recognition, speech generation, Mosaic, and TSS personnel broker. Each broker has the respective bidders for the resource types (for example, NRC and Mitel speech recognition bidders).

After the runtime MANA agents are created they are launched in the network to provide the automated help desk service. The Technical Service Support and Support Services agents reside in a MANA server (shown as "agents (pc)" in Figure 4) and their respective resource agents are distributed across the network and generally reside local to the resource they represent.

An incoming call sends a goal to the Support Services agent which readies its speech resources so that the first speech episode is handled and the incoming call is connected to the speech resources through the call control servant. Thus, the agent asks its speech recognition and generation brokers to collect bids to service a call. The servant associated with the winning bid (Mitel's or NRC's) then sends a task to the speech resource agent initialising it. The speech generator agent is then sent a goal through the agent's SG servant with the data (the utterance welcoming the user to the help desk). Once the caller's utterance is spoken, it is sent through the SR servant to the agent with the winning bid at Mitel or at NRC. The SR agent then returns the recognized id to the Support Services agent which then triggers the next goal script (match the id to the trouble ticket in the customer database). If the trouble ticket is matched, the TSS Answer servant then sends a goal to the Technical Service Support agent with the collected data.

Conclusions

This paper has described a help desk application implemented using multiple agents that share resources distributed across a high speed network. The help desk is first provisioned as an organizational service using TED. The modelled service is then compiled into a set of agents as defined by MANA.

The automated help desk is a real-world CTI application implemented using a multi-agent system. It demonstrates the practical use of multi-agent systems for sharing distributed resources such as speech recognizers, databases and World Wide Web Mosaic clients across heterogeneous networks. It is also the first successful application of the Tool for Enterprise Definition and the Multi-Agent Network Architecture designed as general purpose CTI tools for an enterprise.

Future work includes the evolution of the MANA architecture to further explore telecommunication feature interaction problems across services. It will also include the identification and development of specialists to monitor, diagnose and police MANA environments. In parallel, the implementation of the Help Desk and similar real-world applications that reflect more of the TED and MANA utility will continue.

Acknowledgments

The project team is a large one and we would like to acknowledge many contributors from Mitel and NRC. We thank Debbie Pinard of Mitel who managed the full effort. We also acknowledge Eliana Peres and Ajit Pardasani for their efforts on EDT. Also, we thank Tom Gray, Aurora Diaz and Michael Weiss for work on the MANA architecture. Further, we acknowledge Mark Vangulik of Phoenix for his work. Finally we would like to acknowledge CANARIE of Industry Canada who partially funded this collaboration.

References

[Cypress 94] R. Cypress. Re-Engineering: Make the Second Generation of Business Process Improvement Mode Work. *OR/MS Today magazine*. Published by Operations Research Society of America & Institute of Management Sciences. Vol. 21, #1. Feb. 1994. pp.18-29.

[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 AI in Business Process Re-engineering*, Seattle, July 1994, pp. 65-72. NRC 37140.

[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.

[Oshisanwo et al. 92] A. Oshisanwo, O., M.D. Chapman, M. Ke, A.P. Mullery, and J. Saint-Blancat, The RACE Open Services Architecture Project. *IBM Systems Journal*. Vol. 31, #4. 1992. pp. 691-710.

[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 Intelligence*, Lake Quinault WA, July 1994.

[Weiss et al. 95] M. Weiss, T. Gray, and A. Diaz. A Multi-agent Architecture for Networking Applications. Submitted to *First International Conference on Multiagent Systems*, San Francisco CA, June, 1995.

Acronyms

TED - Tool for Enterprise Definition
MANA - Multi-Agent Network Architecture
TSS - Technical Support Service Representative
CTI - Computer Telephony Integration
API - Application Program Interface
PABX - Private Branch Exchange
pc - IBM or compatible personal computer
SG - Speech Generator
SR - Speech Recognizer
WWW - World Wide Web