A Robust Environment for Agent Deployment

Dick Cowan
Hewlett-Packard Labs
1501 Page Mill Road
Palo Alto, CA 94304 USA
01-650 857-8038
dick_cowan@hp.com

Martin Griss
Hewlett-Packard Labs
1501 Page Mill Road
Palo Alto, CA 94304 USA
01-650 857-8715
martin_griss@hp.com

Robert Kessler
University of Utah
50. S. Central Campus Drive
Salt Lake City, UT 84104
01-801 581-4653
kessler@cs.utah.edu

Brian Remick
University of Utah
50. S. Central Campus Drive
Salt Lake City, UT 84104
01-801 588-1776
remick@cs.utah.edu

Bernard Burg
Hewlett-Packard Labs
1501 Page Mill Road
Palo Alto, CA 94304 USA
01-650 857-2403
bernard_burg@hp.com

ABSTRACT

The Achilles' heel of agent technology lies in its insular position when trying to tackle all the problems at once: from the new semantic models, to new design metaphors, to new tools, reaching all the way to robust industrial implementations to become part of the mainstream of enterprise application development implementations and to support large scale deployment. Agent environments should not re-invent the wheel for non–agent specific tasks, but should instead adopt existing solutions and benefit from the robustness and experience accumulated over time by enterprise class application servers and management systems, just like any other large-scale application. This article presents our approach integrating the JADE agent platform into the HP application server HP-AS, and into the HP OpenView management system, allowing agents to be managed and monitored through these well known, distributed and supported tools. The integration into the HP application server is operational in our four Agentcities platforms. The complete package of an Agentcity can be downloaded from our website.

Categories and Subject Descriptors

I.2.11[**Distributed Artificial Intelligence**]: multiagent systems

General Terms

Reliability, Experimentation, Standardization.

Keywords

Agent platforms, industrial components, Agentcities

1. INTRODUCTION

As manufacturers migrate to adaptive supply networks, they will exploit intelligent-agent technology to detect and resolve operational glitches proactively, in particular: Analysis agents will interpret data in real-time; Exception handling agents will resolve glitches; Interaction agents will regulate access and security; and Policing agents will coordinate the network [Forrester Brief, 2002] [Forrester Wholeview, 2002].

In order for agent-based systems to become part of the mainstream of enterprise application development, we need agent-based systems to become manageable, to be monitored, reported and controlled using robust enterprise class application servers (like HP-AS) and management systems (like HP OpenView.) Just like any other industrial application, they should work 24/7, be capable of starting and shutting down and have a reliable and trusted configuration management. Rather than building these non-agent specific capabilities into a standalone agent-infrastructure, we suggest morphing existing agent capabilities into embedded capabilities within well-deployed application servers. This will allow additional agent-based capabilities for the existing operational applications in order to satisfy the needs predicted by Forrester for the future supply networks. In short, we want a seamless integration of agents and J2EE services in an application service context.

HP's solution is to work with JADE [JADE] one of the most promising Agent platforms and increase its management capabilities by embedding and integrating it with proven application server and management environments like HP-AS and OpenView. By linking existing agent systems with those tools we will get the best of both worlds. In the remainder of this paper, we describe the goals and status of our integration of HP-AS 8.0 and JADE, called BlueJADE [Cowan et al., 2001] and OpenView [Remick and Kessler, 2002]. We describe our deployment of BlueJADE which supports our Agentcities sites as well as our in-house research platforms where it supports a multitude of agents and several communication features, via email, instant messaging and VoiceXML. The conclusion closes the loop with the Supply network application, by analyzing the added value of BlueJADE.

2. BlueJADE

By leveraging J2EE technology we solve many of the industrialization issues, provide a base for solving other issues, and most importantly begin to properly partition responsibilities of effort. Clearly, from the enterprise view, agent technology should be viewed as just another manageable service. Making JADE into a service manageable by HP-AS is just the first necessary step to be capable of leveraging all the other services.

By "management" we include a number of typical system, application and service configuration, monitoring and control actions, such as: a) detecting and restarting a faulty element; b) detecting load conditions and adjusting resources or moving an element to another processor; c) restarting an element when its configuration changes; d) collecting information on the normal and abnormal client and resource state, statistics and usage of each component; and e) element lifecycle monitoring and control. By "element" here we mean a single agent, a related group of agents, an agent container[1], a complete agent platform, or even a set of agent platforms on multiple machines with associated other software.

We provide a "standard" API that management systems can access. The benefits of making an agent platform operate well as a managed J2EE service include: a) more robust intelligent web-service platform; b) additional resources to help industrialize the technology; c) distribution and advocacy, leading to a 'de facto' default standard platform for further research and deployment; and d) creating links to other standard activities.

HP-AS is a fully functional, scalable, J2EE application server, built upon the Core Services Framework, and complete with load balancing, dynamic application launching and Hot Versioning functionality. Figure 1 shows a simplified view of HP-AS and shows just 4 of the many services. The JADE service is shown as just another managed service under HP-AS.

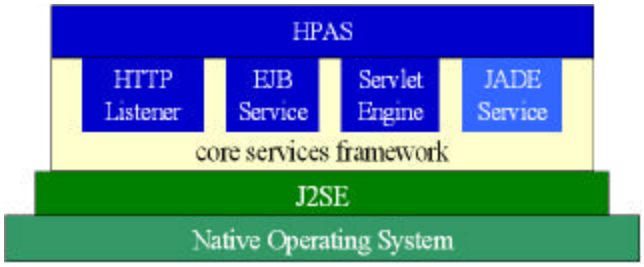

Figure 1. Simplified view of HP-AS

JADE's service description and configuration, like all other services, is done with XML files. HP-AS monitors these configuration files and notifies the owning service if they are modified. The JADE service, like all other HP-AS services, may be managed using HP-AS's JMX viewer. The JADE agent platform (version 2.5) causes the JADE service to be notified when agent or agent platform events occur. HP-AS notifies the JADE service when application server events (e.g. configuration changes, start-up, shut-down) occur. The service processes these events and takes appropriate action. Future work could use this basis to construct a new class that is part JADE Agent and part EJB to interact directly with other EJB's and services. More details on these interfaces and mechanisms can be found in [Cowan et al., 2001]. Figure 2 shows the HP-AS management console while running the BlueJADE service and the collection of JADE agents supporting our current work. Using this console one may control any service. The JMX view enable interaction with a selected service and the ability to view and/or modify any attributes a service exposes through its MBean interface. The log view presents a consolidated log containing output from all active services. The window labeled "JADE's Agent view" is produced by JADE's Remote Management Agent which is just one of many active agents. This agent provides a simple mechanism for controlling individual agents.

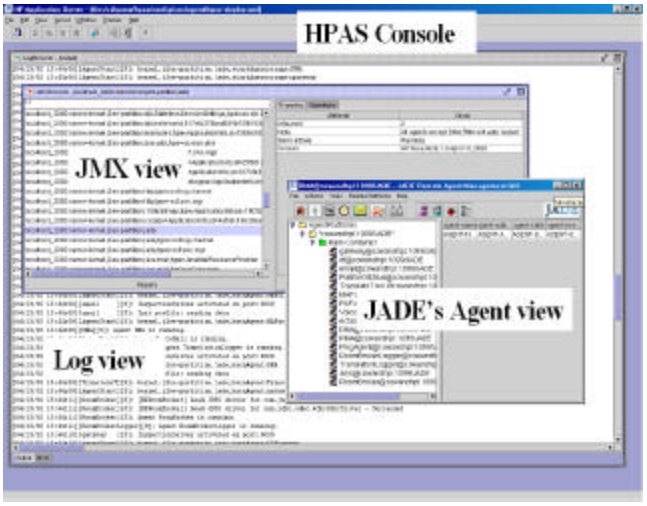

Figure 2. Screenshop of HP-AS management console.

3. AgentSNMP

Network management software like HP OpenView provides a powerful, off-the-shelf means of managing complex distributed systems. The AgentSNMP system takes advantage of this power by extending it to managing agent platforms themselves.

Utilizing the flexibility of SNMP (Simple Network Management Protocol), an SNMP interface (called a Management Information Base, or MIB) is provided to expose information about elements of the agent platform, including agent information, message traffic, etc. Through the MIB, management software like HP OpenView can track temporal changes in data as well as receive events from the agent platform. The AgentSNMP MIB is modeled after the FIPA agent platform specification, meaning it can be used to manage any FIPA-compliant agent platform [FIPA 2001]. The JADE system is used as a proof of concept.

[1] By Agent container here, we mean some sort of agent or element grouping capability, such as the JADE container or J2EE container [Cowan et al., 2001].

On the JADE platform itself, an SnmpAgent class implements the interface and translates generic SNMP requests from outside the platform into JADE-specific management tasks. For basic management tasks, such as agent creation/destruction and message tracking, the SnmpAgent communicates with the AMS using the Introspection ontology provided by JADE to perform the requested operations. More complex requests require the SnmpAgent to perform customized behavior, such as pinging an agent when load balancing is requested.

Beyond polling and data management, HP OpenView provides a powerful visual display normally used to describe the topology of the managed network. This display can be easily utilized by AgentSNMP to visualize the agents running on an agent platform. Using OpenView's customization capabilities, a plug-in is provided with AgentSNMP to translate SNMP events from the AgentSNMP MIB into a graphical display of the agent platform. Figure 3 shows an example of this functionality in HP OpenView during a client/server agent simulation. Agents are represented by symbols in OpenView's map architecture, and messages sent between agents are represented by connections between the corresponding agent symbols. OpenView has the ability to automatically arrange symbols to easily recognize common network topologies, which is helpful in AgentSNMP to be able to recognize patterns of agent communication.

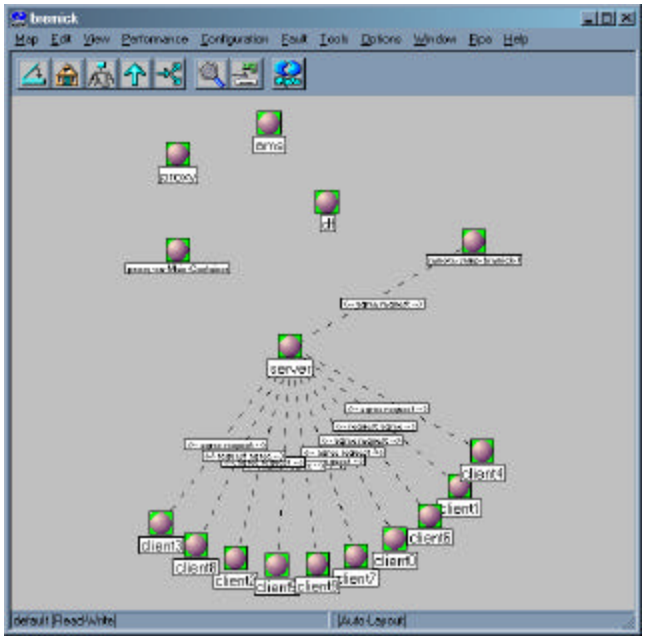

Figure 3. HP OpenView management display

4. APPLICATION TO AGENTCITIES

As one test of BlueJADE to demonstrate its robustness and features, we have implemented four Agentcities sites: Palo Alto, Honolulu, Miami, and Salt Lake City.

Agentcities is a worldwide agent interoperability testbed, where some 40 organizations have created compatible sites using their favorite agent platform (we use JADE upon BlueJADE), some infrastructure, and some web pages. To be consistent with the Agentcities community, the platform must implement a ping agent, must publish its Apdescription.txt so that other FIPA compliant platforms can exchange ACL messages with it, and register itself with a central agentcities registry, so that other systems and an automated periodic pinger can find it.

We have a BlueJADE group installation process that installs a new HP-AS group (a set of file and execution script templates) to a known location. Invocation of this group's start script causes HP-AS to start, which then starts the JADE service, which in turn loads a specific set of agents, and applies appropriate property files to customize those agents. Included with the BlueJADE package are two sample applications with a minimal agentcities implementation being one of them.

We have also constructed a richer agentcities implementation using our CoolAgent toolkit. These sites has a human readable top-page, that describes the site and its capabilities, and links to useful information about the research project, the city, and to several dynamic pages created when the agents are running.

We have also implemented several agents, using our CoolAgent toolkit to provide capabilities, such as properties for customization, embedded web-server for agent visualization and state-machine implementation of the behaviors. The agents we have are:

- PingAgent – responds to periodic ping messages from Agentcities central or other agentcities sites
- Weather Service – accesses a web site to get local weather conditions, and present these in an ACL accessible form
- Watchdog – periodically pings our other agentcities sites to see if they are still up; if a failure occurs, these will send an alert (via email) to several administrators, who can take corrective action.

In addition, using our Web Object Inspector, we provide a dynamically updated human readable web page that displays the state of the ping agent, the weather agent, and the watchdogs. Figure 4 shows a browsers view of the log of the incoming ACL messages and responses.

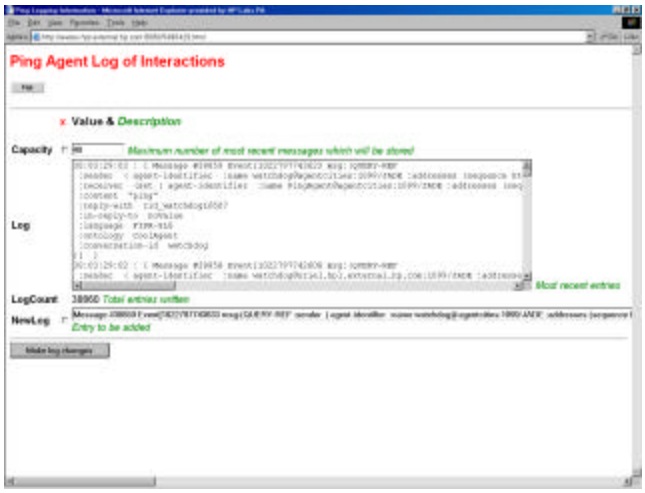

Figure 4. Web based agent inspection

As a consequence of the combination of BlueJADE and the cross-platform watchdog, we have a very reliable system. If an agent should crash for some reason, BlueJADE will restart it automatically. If the agent slows down and stops responding, the network becomes disconnected, the platform crashes, or it is taken down for maintenance then the watchdog will immediately notice it and inform the "administrators." Typically, we run the watchdogs to ping more frequently than Agentcities central, so that we can take corrective action before the central pinger recognizes a problem.

We have started to use these stable platforms for a number of other experimental purposes. Since they are outside the HP and University firewalls, they can host additional services that we can use for experiments and public demonstrations. We already mentioned that the watchdog agent can send email; we have also implemented a Jabber connection to our agents, and a thin connection to agents inside our firewall. This allows an ordinary Jabber client to talk to agents that activate the jabber interface—one can "chat" with one's agent.

We have used both Email and instant messaging (Jabber) with an experimental VoiceXML interface to a multi-agent based meeting scheduling application that currently runs inside our firewall. This allows us to use a phone to call the VoiceXML portal, which then invokes an agent on the Agentcity site, which can then (via email or Jabber) communicate with the meeting arranger.

5. LARGE SCALE APPLICATIONS

BlueJADE was released internally to HP in March 2002 for beta testing, and we are working with selected outside partners. These platforms will prove their robustness once used in demanding applications, needing to run and maintain populations of agents interacting to provide services. As these services have dependency links inside and outside the agent infrastructure, BlueJADE and AgentSNMP can monitor their availability, launch the missing services and manage a complete configuration of a population.

Seen from the user side, agents are integrated in their applications, and are managed exactly like any other component allowing agents to bring added value to existing deployed applications. This smooth transition offering a gradual integration of agents into applications, answers the very basic needs of application managers who are in charge of the day to day operations and need to re-use as much as possible existing and well performing applications. Therefore agents appear to become an evolution towards added value bringing new possibilities to existing applications, instead of being a complete redesign of applications with some unproven technology.

This evolution is a needed step towards the deployment of adaptive supply network flexibility, as the major players in the field cannot drop their current infrastructure for a new one, but are seeking for a smooth evolution path to increase the capability of their supply chains to incorporate agents at every individual node of the network to perform exception handling and resolve glitches at the earliest level, in real-time and avoiding errors to propagate and amplify with time. The agent decentralization, autonomy and proactive behavior are key to success. As application contexts evolve towards more interoperability, static regulation and access policy show their limits. In particular the current access policies, firewall etc, impose a strong burden to applications, often leading them to errors, access denial and loss of time, for problems that are actually not related to the supply chain management, but rather to their current implementation. Finally there is a need of coordination this decentralized network of semi-autonomous agents, to make sure that some of the agents will not compromise the equilibrium of the complete network and to supervise the mission of the complete supply chain. [Forrester Brief, 2002].

6. CONCLUSION

The combination of BlueJADE and AgentSNMP makes for a powerful agent management solution. This industrial strength platform provides the ability to keep agent systems up 24/7 and facilitates the management of what could be thousands of agents. This enabling technology facilitates the use of intelligent agents to solve problems in important domains such as those outlined by Forrester's articles. We are moving forward with external testing and deploying outside of our immediate groups. This is just the beginning. Now we can expand on these capabilities, add more agent services, and take more advantage of the robustness and management capabilities of BlueJADE.

7. ACKNOWLEDGMENTS

We thank Fabio Bellifemine (Telecom Italia Lab) and other members of the JADE team for working with us to help shape and implement the modifications to JADE.

8. REFERENCES

[Griss 2002 et al] Martin L Griss, Steven Fonseca, Dick Cowan and Robert Kessler, "SmartAgent: Extending the JADE Agent

Behavior Model," Proceedings of SEMAS, Orlando, Florida, July 2002, (to appear). [*HPL Technical Report HPL-2002-18, January 2002.]*

[Cowan and Griss 2001] Dick Cowan and Martin Griss, "Web Object Inspector," HP Laboratories Technical Report HPL-2001-297, Dec 2001.

[Cowan et al., 2001] Dick Cowan Martin Griss, and Bernard Burg, "BlueJADE- A service for managing software agents," HP Laboratories Technical Report HPL-2001-296 (R.1), Mar 2002.

[FIPA 2001] FIPA Agent Management Specification, Aug 2001. http://www.fipa.org/specs/fipa00023/

[Forrester Brief, 2002] N. Radjou, L.M. Orlov and T. Nakashima. Adaptive Agents Boost Supply Network flexibility. Forrester Brief, March 11 2002. http://www.forrester.com/ER/Research/Brief/Excerpt/0,1317,14684,00.html

[Forrester Wholeview, 2002] N. Radjou, L.M. Orlov and T. Nakashima. Adapting to Supply Network Change. Forrester Wholeview, March 2002. http://www.forrester.com/ER/Research/Report/Summary/0,1338,12182,FF.html

[JADE] Java Agent DEvelopment Framework http://jade.cselt.it/

[Remick and Kessler, 2002] Brian Remick and Robert Kessler, “Managing Agent Platforms with AgentSNMP,” submitted to this workshop.