

Multi-agent Software Systems: What we've learned, and next steps

Geoff Arnold
Kate Stout



VicAgents, Oct.27,2000

Overview

- Over the last 3 years we have engaged in the agents research community
- This report provides
 - Overview of this domain
 - Status on standards efforts
 - Summary of what we've learned
 - Recommendations for what Sun should do next

Executive summary

- Multi-agent systems ideas are important to Sun
 - The MAS model reflects a natural way of thinking about how Sun – and our customers – will build systems in the future
 - MAS represents a paradigm shift akin to object oriented programming
- MAS is *inevitable*
 - If we do not anticipate how it will affect our businesses and technologies, we may be blind-sided

Why are agents interesting?

- The growing importance of flexible, decentralized peer-to-peer interactions vs. rigid client-server architectures
 - For example, Bluetooth, Jini™
- The trend towards black-box components, both hardware ("appliances") and software
 - Self-diagnosing, self-configuring → autonomous

Does this mean that Sun needs to build MAS?

- Not *necessarily*
- We must understand what our customers intend to do
 - Make Sun platforms best for MAS applications
- We should apply MAS ideas to selected systems and application areas
- We must understand how MAS ideas may affect the evolution of Java, Jini, etc.
 - Or their successors...

Agents: some background

“Agent” - an imprecise term

- The term “agent” is widely used in software engineering, computer science, and artificial intelligence (not to mention philosophy)
- Some uses of the term are related
- Others are completely incompatible
 - Often this is simply confusing or inconvenient
 - For some, these are questions of dogma

Interesting agents

- Multi-agent systems
- Characterized as follows: [Jennings/Woolridge/Sycara, 1998]
 - *Each agent has incomplete information or capabilities for solving the problem, and thus a limited viewpoint*
 - *There is no global system control*
 - *Data is decentralized*
 - *Computation is asynchronous*
- These ideas are characteristic of the “Post-PC” worldview

Agents not in this discussion...

- We're not thinking about
 - “Shop-bots” and similar web tools
 - Desktop agents, like the “Microsoft paperclip”
 - Traditional SNMP/CMIP agents
 - Simple mobile code (e.g. “aglets”) – unconvincing, probably not scalable
- On the borderline: “agent” as a design pattern for “an object with a thread”
 - An interesting programming language idea
 - Probably an important evolutionary step

Agents and "intelligence"

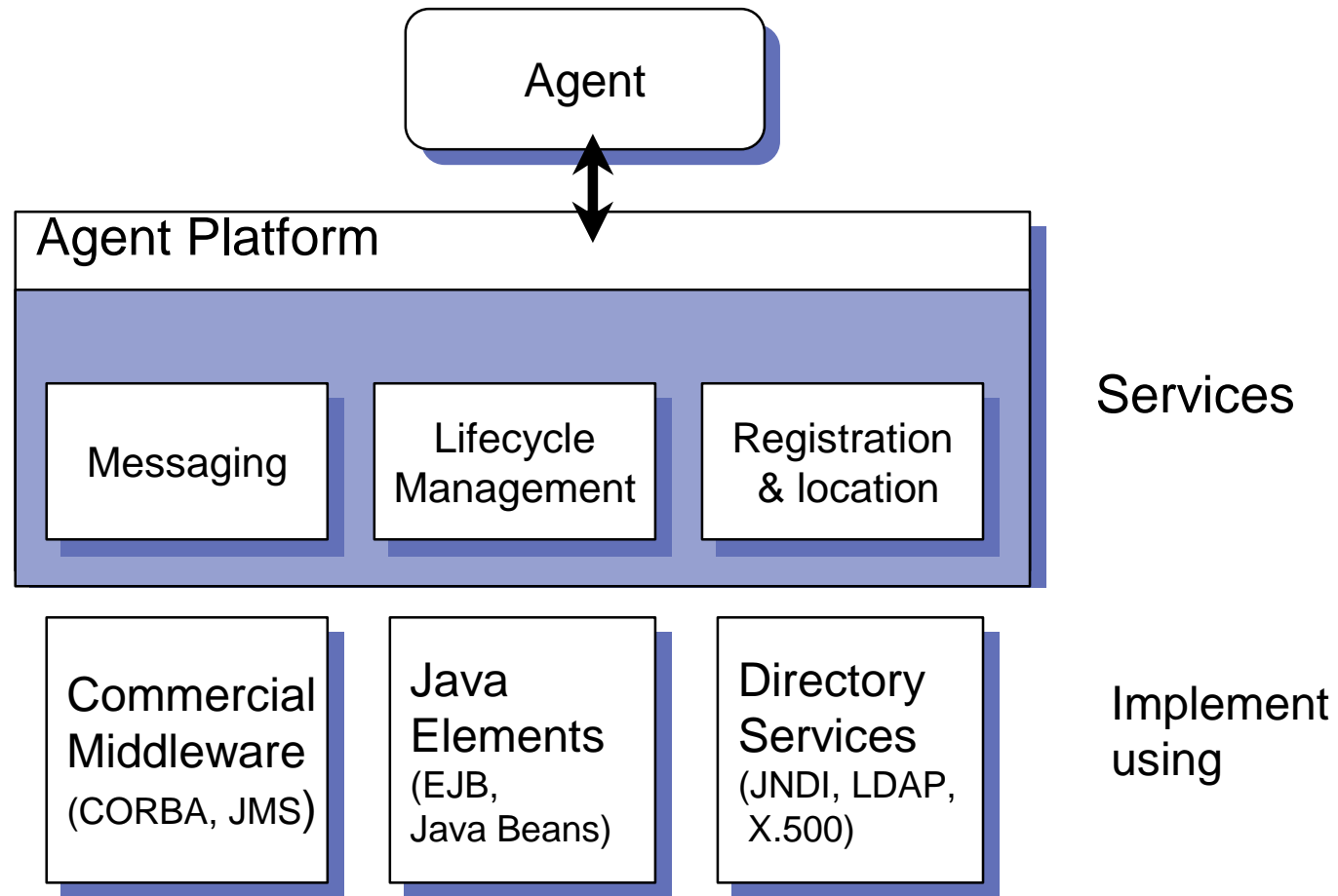
- Goal is that agents are more autonomous, or "intelligent"
- Accomplished by
 - Use of multi-level planning, procedural reasoning, etc.
 - Adaptation, learning, collaboration
 - Semantically rich communication & negotiation
- The added reasoning moves them from a distributed compute platform to a model for creating complex ad-hoc interactions

Agents: Architecture

Parts of a MAS architecture

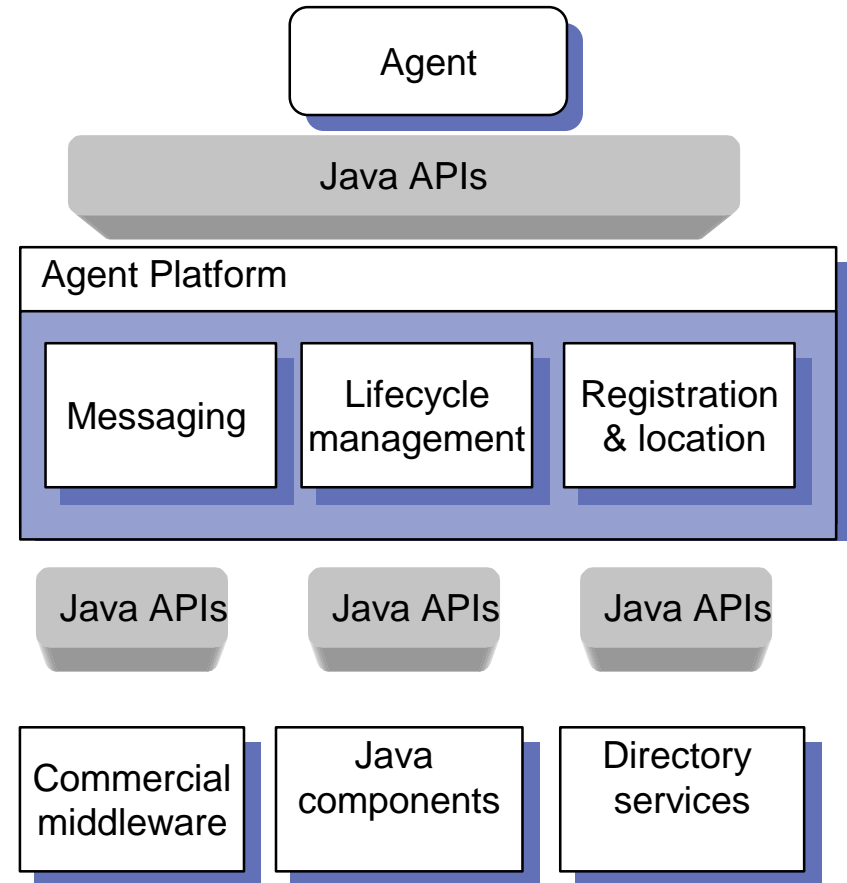
- Agents
 - Capable of communication using complex message content
- Message delivery mechanisms
- Directory services for agent location
- Agent platform to unify services, provide lifecycle management

Model for MAS architecture



Java standards for MAS

- Specify interface
 - between agent and services
 - for implementations of services
- Sweet spot for Sun - servers & services
- Common infrastructure would encourage deployment of agents
- Leverage existing and emerging technologies



Agent Standards

Java MAS standards - status

- JSR for agent standards being drafted by Fujitsu
 - Supported by Sun, IBM and HP
 - JAS – Java Agent Services – JSR 87
- JSR is direct result of our work on architecture in FIPA, an agents standards body.
- Will use existing Java standards (EJB, JMS,)
- Focus is on...
 - Agent services (infrastructure)
 - message structure
 - NOT semantics



Other MAS standards work

- Over the past two years we've worked in the two main forums for agent standards
- **FIPA** *Foundation for Intelligent Physical Agents*
 - FIPA's stated goal
 - To promote the development of specifications of generic agent technologies that maximise interoperability within and across agent based applications.*
 - Established in 1996 by BT and other agent researchers
 - We joined fall 1998
- **OMG Agents working group**
 - We joined at time of founding, fall 1998
 - Sponsored by Internet and E-commerce Task Forces

Outcomes of standards work

- FIPA
 - We partnered with IBM and led effort to do architecture for agents infrastructure
 - Architecture was well received technically, but...
 - FIPA focus changing
 - Solve difficult problems in agent communication
 - Many researchers are uninterested in commercial issues
 - Commercially-relevant outcome unlikely soon
- Our status
 - Geoff Arnold on FIPA board, continue monitoring

Outcomes of standards work

- **OMG**
 - We made substantial contributions to Agent Green paper
 - This work had seemed to be going nowhere – there was no real customer within OMG
 - However, recently upgraded from WG to SIG
- **Our status**
 - Monitor from a distance -
 - If new and interesting work happens, re-engage

Agents: What have we learned?

Observations on agents

- There's no agreement on what "agents" are
 - Research factions, not a single community
- The most successful agents-related research has focused on particular application problems, not on agents per se:
 - Distributed control systems
 - Simulation and modeling
 - Distributed problem solving

Observations on agents research

- MAS research efforts have addressed:
 - Flexible, intentional peer-to-peer communication – conversations, negotiations, commitment
 - Application of MAS ideas to middleware itself
 - e.g. directory services as agents that can negotiate
 - Knowledge representation, semantics, and ontology management
- The research often overlooks real-world considerations
 - Performance, scalability, interoperability, usability

Market Predictions

- MAS techniques will first be used in closed systems:
 - No externally-visible “agent” interfaces
 - Likely arenas: web-based B2B e-commerce; workflow; factory automation
- It is unclear when explicitly interoperable MAS applications will emerge
 - Most likely domain is personalization – location, preferences, personal workflow, affinity
 - Unclear if the term “agent” will be used

Standards Predictions

- Standards work is appropriate for MAS infrastructure and messaging
 - Existing standards are biased towards client-server, synchronous, transaction-oriented systems
 - Standards will facilitate code re-use, and foster tool-building
 - Necessary now to deal with XML
- It is premature to think of standardizing content languages, conversations, etc.
 - These are still research topics

The bottom line

- Think of multi-agent systems as a way to create robust, highly distributed, highly decentralized applications
- Many customer scenarios (vertical and horizontal)
- This also applies to many Sun domains
 - Hardware configuration and management
 - Personalization
 - Ad-hoc networking

What next?

What should Sun do?

- Short term actions – e.g. partnering or acquisition
 - Obvious application areas....
- Support Java standards for infrastructure
- Research in key topics
 - In-house, plus collaborative/sponsored
- Continue to monitor customer needs
 - Internal and external

Research needs for distributed computing

- There has been relatively little work on several problems which are critical to the acceptance of MAS applications:
 - Security and delegated authority
 - Long-term identity
 - Ontology management
 - Contractual commitment
- There has been insufficient work on making agent ideas and techniques accessible to the wider software engineering community

Summary

- Multi-agent software systems are inevitable
 - Not “if”, but “when”
 - However “how” remains unclear
- There are short term actions we can take
 - and longer term plans to discuss

Contact info

Geoff Arnold

Kate Stout

Sun Labs

geoff.arnold@sun.com

kate.stout@sun.com

Sun Microsystems Laboratories;

MS USUR02-309

One Network Drive

Burlington, MA 01801-0902



VicAgents, Oct.27,2000