Prospects for Agent Technology: Interviews with **Industry Experts**

Stefan Brantschen Whitestein Technologies
Patricia Charlton Motorola
Andrew Lucas Agent Oriented Software
Jörg Müller Siemens
Chris Preist Hewlett Packard Laboratories
Simon Thompson BTexact
Ken Woghiren Lost Wax

Interviewed at the AgentLink Agent Technology Conference
by Steve Munroe, Ronald Ashri and Elizabeth Coulter-Smith, University of Southampton

Stefan Brantschen

Whitestein Technologies (WT) was founded in January 1999 with the mission to become a leading provider of advanced software agent technologies, products, solutions, and services for various applications and industries. WT strongly believes that agent-based technologies and agent-oriented software engineering methodologies will be among the key concepts of a next generation of distributed information systems and network infrastructures, in particular in combination with other leading-edge technologies such as web services and mobile wireless computing.

Patricia Charlton

Motorola has a mission statement of "Intelligence Everywhere", and sees agent technology as helping to move towards intelligent and electronic lifestyles. Motorola is currently investing in various different spaces such as trust technologies and security services.

Andrew Lucas

Agent Oriented Software is a commercial product company that delivers professional agent support products for businesses, and also helps businesses to develop their own agent-based solutions. Their flagship product is the JACK Intelligent Agent Platform.

Jörg Müller

The core areas for applying agents within Siemens are agent infrastructure that facilitates communication and coordination between agents, and anything that makes it easier to build agent systems scalable over different networks and devices. Secondly, there is what Seimens call ubiquitous assistants, where the focus is on the agent as an automated assistant that helps individuals and organisations perform, for example, routine tasks in a personalized and efficient way. A third context is the interaction between humans and agents: how does the agent get to know your preferences, with you not having to explicitly instruct the agent? The fourth area is that of resource management; for example, using the multi-agent approach as an efficient market-based mechanism for allocating scarce resources in a dynamic market.

Chris Preist

Hewlett Packard (HP) has been interested in what they call e-services for about three years now. E-services are automated autonomous entities that cross organisational boundaries. HP sees agent technologies as an important way of making e-services real. Work at HP is primarily based on using agents to operate across organisational boundaries in open systems.

Simon Thompson

BTexact define their work in agents as exploring how to develop software to operate in an open environment where players interact with other players who don't know each other *a priori*. They also feel that there is fundamental research in terms of technology development and acquisition that they can do in this area as well.

Ken Woghiren

Lost Wax is a company that has been involved with agent-based systems for around three to four years. Working predominately in the financial services arena and in e-commerce, they have their own agent platform that enables buyers and sellers to get together to negotiate contracts in a generic e-commerce environment.

What are the major domains in the next couple of years that will be influenced by agent technology?

Stefan Brantschen

I wouldn't mention just one domain, but the kinds of domain that will profit from an agent-based approach are domains where you need fast decisions in dynamic (even chaotic) and possibly unstructured environments. So, it's these kinds of complex domains that are likely to profit from agent-based systems.

Andrew Lucas

We are already working to a substantial extent in defence and the military. We also have JACK in telecommunication products, and we sold JACK to Toshiba for the manufacturing group. The next area is e-commerce, but in a limited way. For example, we teamed with Logica for an electronic trading hub for the New South Wales government in Australia. The idea here is that if you are in the Department of Public Works and want to purchase laser printers, and I have signed a contract with the Government, then you can use the standard conditions we've agreed and put out a specification that I and other printer vendors can bid to. So it provides a medium for us to complete the contract with you and fulfill the order. What JACK will do is to provide advanced customization because you might be in a department that has different business practices from other departments.

Chris Preist

First of all, the area that I work on, which I believe is very important, is electronic commerce. By using agents in electronic commerce, it can become more automated and also more flexible and dynamic. Another area that Hewlett Packard is very interested in is network quality service management, which is basically how to manage the different quality of service. We are also interested in automating the purchase and sale of internet bandwidth to adjust the quality of service that internet bandwidth providers are using, depending on how important it is, and how much money you are prepared to pay, for example, for streaming video or just for standard e-mail.

Simon Thompson

Definitely network management. Anything where there is variable and unpredictable demand that you have to take action to deal with. We also use agent systems for resource management to look strategically at where we are going to deploy resources across the UK, because again we can't predict too far in advance what we are going to need. E-commerce is absolutely massive, of course.

Ken Woghiren

Any domain that is really looking at addressing the problem of what I call complex, dynamic and open environments is really going to do well to look at agent based approaches. E-commerce is one that makes sense, and so are financial services. But I think that another area is the telecommunications industry. Telecommunication networks by definition are very complex. They are large, there are lots of individual nodes, and setting up services, be they multimedia or voice-only services, is a process that is very complex. We've done work looking at agent-based techniques suggesting that this type of solution is good. Logistics and Supply Chain Management is another important area. The logistics and supply-chain management issues can also be very complex, for example with areas or situations where it is impossible to pre-plan a given path and for that plan to always be optimal. Reality shows that things go wrong and that pre-planned solutions will be next to worthless. In this situation an agent-based system that is able to dynamically reconfigure itself, or dynamically self-heal, is of much more value than one that is perfect when things are perfect. The world is not like that

Agents are being over-hyped. Do you think this is true, and if it is, how do we deal with it?

Stefan Brantschen

It is important to stress the point that you have a solution for a client, and not to talk up the agent aspects in which they are probably not particularly interested. In case the customer is interested, then of course we will explain how the agent technology works, but usually the client is only concerned with solutions, not the form of solution. Moreover, I tell customers that agents are not magical. I explain why agent technology is still only software and programs, and the part that makes an agent an agent isn't magical in any way. I explain how agent technology is just an extension of earlier software techniques with an added level of sophistication.

Patricia Charlton

I don't think that agent technology is so over-hyped. There is a community of us who do talk a lot about agent technology, its possibilities and all the research activities, but the man on the street hasn't really seen it yet and doesn't know what it is. One of the key things at Motorola is that we talk about exploitation of agent technology and we appreciate that agent technology is a broad spectrum, which goes from very simple things to very complex things.

Andrew Lucas

Lets turn to the historical example of hyping, which were expert systems. I think that was foolishly proposed in the first place, and I think that the computer scientist who originally proposed expert systems replaced experts with idiots and did a lot of disservice to the rest of us. We have been recovering from that situation ever since, but I think particularly in the case of agents we have a good technology, soundly based. The presumption behind using the word hype is that there is actually nothing there and that is not true. There is sound technology in agents; there *is* something there. Inevitably, commercial organisations may overstate the ability of agents to deliver certain things, or even for their own organizations to deliver, but I think there is much less chance of the hype being followed by complete disillusionment. So, whilst there may be a bit of excitement at times, I think that over the next few years you will see that we will actually deliver, and we will meet a large number of the claims that have been made.

Chris Preist

I believe that the problem, as several people at this conference today have spoken about, is not that agents are being over-hyped, but that agents are being sold as a technology, and the important thing is that you don't want to sell technologies, you want to sell solutions. So, yes it's true that people do say, "Agents are wonderful, they will do everything," but that's not the way to convince people. The way to convince people is to say, "I've got a way of solving your problem X." Don't bother mentioning agents.

Simon Thompson

I think that AI is a very interesting comparison, because initially AI was technology in search of a problem. It has been very successful, but it's been a quiet success. The reason for that success is because the hype went away but the deeper business needs that AI could address didn't. People then had to use rule-based systems to achieve things that could not be done with an imperative system; it was easier and better to use rule-based systems. I think the agent world will develop in the same way, apart from the fact that we have been hit with a tidal wave of software development problems, where the 3rd generation of software systems are just not adequate. Universal free bandwidth and connectivity, where the potential to exchange information is massive is something else we don't know how to deal with. Then there's the ubiquity of devices, and convergence and connection within devices, where we have devices that are extremely complicated with many different functionalities that can do many different things. I think handling this complexity is beyond the current generation of software abstractions. Furthermore, the only way we are going to successfully handle this complexity is to develop new abstractions to deal with it. I believe that the agent abstraction is a very good step towards that. It is not the end of the story but I think it is part of the story. What we will see is that agents will form part of the community's response to this wave of connectivity in much the same way that objects were part of the community's response to the wave of innovation in personal storage and access to computation that happened in the seventies and eighties. That is why we should be optimistic that the research and investment that has been done is worthwhile.

Part of the attraction of agents is that they have the potential to be autonomous and thus will be able cope with dynamic environments. However, in the event things go wrong, how do you see the legal implications?

Patricia Charlton

I think there are a lot of problems based on security, trust and privacy that we have to deal with, and not just for agent technology. Who is responsible for the software system? Is it the operator of the system? Is it the manufacturer of the system? You go into medical systems and you see this problem quite often. There are also cultural and social issues. There's no one way we are going to solve this, but we can't keep ignoring it.

Andrew Lucas

The answer that you generally find with these things is that there is a law that covers it. For example, there is a businessman in Melbourne, Australia, who is suing an American publication for libel. He is suing on the basis that he downloaded a web page in Australia and therefore the case should be heard in Melbourne, which will save him a lot of money, whereas clearly the American company would like the case heard in the USA. There is an interesting discussion here and a precedent will be set, which is probably very similar to what happens in the publishing industry where there are very well established principles for such situations. I think the legal framework is there, and it is up to us to start recognising where we are legally liable. To give you a very good example regarding unmanned air vehicles and the Ministry of Defence in the UK, they are looking to have autonomous vehicles do a lot of the work of manned aircraft partly because they don't want to lose the people. Now, there are very well established precedents for testing and certifying aeronautical software, but not for agent software in an aeronautical context, so that is the kind of thing that we are going to have to face, and that is why testing is one of the research areas that we need to address. How can I satisfy the certification authority that the JACK-based UAV will not turn around and bite the pilot?

Jörg Müller

I think that there is a short term and a long-term answer to this. The long-term answer is that it is a very serious problem. It is connected to the nature and the definition of delegation. What does it mean to delegate a task to an agent? Who is responsible in the end? The short-term answer is to what the legal implications are in practice. I don't think it is very easy for a company to sue a software vendor for something going wrong because you have to prove that it was because of the software. Longer term, it's more of a serious issue which, at some point, needs to be addressed by laws, saying how we define a machine, autonomy, etc, and how we handle delegation? So, probably, there will be a whole new type of law emerging to deal with these issues.

Simon Thompson

I think the key problem to making legal agreements with agent systems is to tie them reliably and accurately to the implementation of the agreement. So, if you make an agreement with an agent, then you must be able to bet that what it does will reflect that agreement exactly, and all sides have the same opinion as to what that is. Humans spend a lot of time learning how to do that, but with agent systems it is not quite so clear. I think that a lot of the ontology management systems and content languages that have explicit semantics will help us to do that, but it is quite a tricky problem. I don't think the actual legal framework is a problem, but actually tying the meaning of a legal agreement into computer code, and being able to show to an auditor or to the police that what you did is reasonable, is a bit harder. There is some work to be done there.

Suppose I am a developer using object-oriented methods and stuck in my ways. In simple terms, what is a quick way to convince me to switch to a different approach? In other words what are the main advantages of the agent approach?

Stefan Brantschen

I see advantages in the agent-based analysis and design methodology in that it's much more natural to design a software system by thinking in terms of agents and roles, what the responsibilities of these roles are, what the rights of these roles are, what the goals are, and what the capabilities are.

Jörg Müller

The main advantage lies in flexibility - the ability to change the topology of a system based on loads and requirements. This gives the system the capability to influence the way the interactions go - channels of interaction - which makes these kinds of systems very robust in terms of failure. Another main advantage, looking at agents in a less resource management way, is that an agent should be able to act on behalf of a human or a team. And I think this is a very important issue, although there are many important questions that need to be answered before this can be exploited, like security, like how to delegate these tasks, etc.

Chris Preist

I see agent-based software engineering as something that is working at a more abstract level than more traditional object-oriented software engineering. I see it as a methodology that sits on top of traditional object-oriented software engineering. The advantage with the agent-oriented approach is that it allows you to reason about systems that cross organisational boundaries and that are open. In other words, the advantage lies in systems where you don't know in advance who all the participants will be, whereas traditional approaches assume that you have some kind of closed system, even if it's distributed.

Ken Woghiren

The real essence of agent-based systems is the de-coupling of the constituent elements of the agents. An OO approach means that Object A invokes methods on Object B, and once Object A gets hold of Object B then it jolly well better do what its API or its interface says it is going to do. If Object A gets a response back from Object B that is not accepted, then, to use a metaphor, Object A throws its hands up and sulks. That provides possibly for an efficient system, but also a very tightly coupled and very inflexible one. You can't for example introduce a better version of Object B, because Object A only knows about Object B and not Object B++. If you take away Object B, then Object A is left high and dry.

When you develop agent-based systems, you have at the back of your mind that you don't instruct an agent to carry out a method. You send an agent a message requesting it to perform some sort of action, and it is up to the individual agent to determine whether it is going to support the message in the way it is requested. The real essence to that is that failure is not an exception, and failure is something that Object A has not only to accept and expect, but is something it is actually built to cope with it. If I ask you to do something, by the very fact that I am asking, there is the possibility you are going to say no. If you say no, then my next course of action is not to be thrown by that but to find somebody else to carry out that action, so we must have the ability to discover agents. You loosely couple agents by saying that Agent A will find an agent with the services it is looking for. So the element of message sending and discovery are what you don't find in object-oriented systems. For me that is the real essence of building flexible self-healing systems.

Agentcities.NET Deployment Grants

Simon Thompson, BTexact

The Agentcities.NET project sponsored by the EU's Framework V recently called for proposals for 30 grants of EUR10,000 to be allocated to European institutions, research centres and commercial organisations. The call was issued at the Agentcities information day in Lausanne on February 7th, and closed on 22nd March 2002.

The response to the call was extremely good, demonstrating the vibrancy of the Agent R&D community in Europe currently. 61 expressions of interest were made, and 57 completed applications were submitted. All of these applications were then subjected to an anonymous refereeing process to assess their quality. A panel of 34 referees were recruited for the task, and included many of the academics currently active in the field of agent research, industry based researchers and people involved in standards initiatives related to the agent field such as FIPA, the GRID, Web Services, the Semantic Web and Cluster Computing. Referees were recruited from the UK, France, Germany, Spain, Italy, Holland, Switzerland and the United States.

After the refereeing process the results were collated and it was decided to fund 22 projects, all of which had been rated extremely highly by their respective referees. The projects selected demonstrate the quality and breadth of the vision that European agent researchers have developed in recent years, and promise results & service deployments in personnel management & recruiting; web service deployment; ontology management & information integration; travel and tourism services; legal advice; network analysis; health care; logistics and supply chain management.

A further group of projects were selected for development by their authors before resubmission for further assessment and possible funding.

The projects, which are listed below, are slated to begin in July 2002 and will complete before 1st March 2003. Further information will become available on the Agentcities.NET web site in due course: http://www.agentcities.org/EUNET/

Organisation	Project
iSOCO, S. A., Spain	iBundler, Intelligent Bundler
University Rovira i Virgili, Spain	Deployment of agent-based health-care services
emorphia Ltd, UK	Cognac
Computer and Automation Research Institute, Hungary	Budapest Library Agent Mini-Project
University of Aberdeen, UK	Deploying AKT Services on an AgentCities Network
UPM Technical University of Madrid, Spain	Traffic city agent for recommendation
France Telecom R&D, France	AgentCities Travel Assistant
Acklin B.V., The Netherlands	European Competition Law Assitance
University of Bath, UK	An ontology server
University Of Bath, UK	A Bed and Breakfast Reservation Service
Czech Technical University, Czech Republic	AgentExchange
European Media Laboratory GmbH, Germany	Integrated Hotel Reservation Service
CERE - CNR (Consiglio Nazionale delle Ricerche), Italy	Agent Factory
V-GUIDE, France	MAS-TER : M.A.S for local development of tourism
Computer Science Institute of the University of Ancona, Italy	A MAS based on the P2P model to Information Integration
University of Manchester Institute of Science and Technology, UK	Flexible Contracting for Dynamic Management of Supply Chains
Vector TRI Ltd, UK	Application of Small-World Theory to Knowledge Agents
University of Girona, Spain	On the integration of restaurant services
Siemens AG, Germany	Agent based Human Resource Management
Manchester Metropolitan University, UK	Intelligent Simulation Models
University of Liverpool, UK	A shared ontology for negotiation protocols
Profactor Research, Austria	Integrated Logistics and Production Planning