Distributed Agents Researchers around the World, Unite!

Olivier Marin, Laboratoire d'Informatique de Paris 6
Niek Wijngaards, DECIS Lab/Thales Research & Technology Netherlands

Attending a discussion on distributed agents among computer scientists is a bit like watching an episode of Fox Television's 24. It generally starts with all the great things that are about to happen: in our case, for instance, all the issues that should find a childishly simple solution thanks to agents. At this point, everyone can express only genuine enthusiasm. Then, the plot thickens: an argument breaks out over the use of agents for a complex issue, say distributed scheduling. This is when the viewer should begin to sit back and really enjoy the show. Unfortunately, and invariably, it's when the show starts heading downhill, because regardless of the contortions, the plot grows surreal. Any argument falls short of fitting in a consistent, agent-related debate. The viewer is left wondering at what point reality became twisted without any hope for redemption and ultimately why none of the speakers appear to be thinking along the same line. The final impression is usually disappointment. Whenever people discuss agents, they're bound to end up thinking afterward, "Oh well, back to reality, then eh?" Agents seem so nice from a distance, but up close it looks like there's hardly a way to integrate them properly in any real project.

Maybe that's exactly the problem: agents do look really nice from a distance. In its own blurry kind of way, the agent paradigm comes across as one of those whatever-it-is-I-can-fix-it tools we've all dreamed of. And, as it should be with such dreams, reality—that most sober of party crashers—never fails to intrude with a vengeance.

One might think the problem lies in the lack of a proper definition for an agent. The strangest part of it all is that the current situation doesn't result from a lack of effort and compromise. At least two standards define what an agent platform is supposed to be: the OMG Mobile Agent System Interoperability Facility (MASIF) and Foundations for Intelligent Physical Agents (FIPA). If you proceed carefully through founding papers, you can discern key elements for defining agents as virtual entities that

- possess individual goals, resources, and competences;
- perceive and act, to some degree, on the near environment including communications with other agents (and humans);
- change their own states and perform actions with some level of autonomy and reactivity; and
- provide services.

So, agents do have a solid theoretical background. Researchers from the AI community have done their job. Could it be that researchers from the distributed systems community didn't follow? Not quite so. Both aforementioned standards have spawned at least one implementation each: MASIF’s Grasshopper and FIPA-OS's Jade. And countless more general and specialized agent platforms are out there waiting to be exploited.

Yet, another fact emerges from a quick scan of agent-related projects: very few of them integrate all four of the above theoretical definition's key elements. For instance, several projects obviously used the term "agent" because it once sounded sexier than "mobile code." Truth is, nowadays "agent" isn't as strong a buzzword as it once was. Some "agent platforms" have been converted to "grid computing frameworks." It gets worse: a watered-down version of agent technology is peer-to-peer technology. And, as we all know, P2P has become a stellar success both in terms of proliferation and commercial and noncommercial use.

Agent researchers sometimes blame industry for not taking up agent technology. They argue that big industries don't show up at industrial sessions or tracks at agent conferences. Invariably, more academics turn up than industrialists, and the companies that give acte de présence are (very) small, with modest and small-scale use of agent technology. It actually comes as a surprise that some software companies use agent technology. Despite almost a decade of existence, the agent paradigm has not given way to any significant software applications. Why, then, should industry trust agent technology to offer technological or economic advantages?

Is it therefore legitimate to wonder if the complete theoretical definition of an agent is too cumbersome for distributed systems-related projects? If so, we should pronounce the agent paradigm a dead-end research area. Or should we? It's us, the agent researchers, who have the burden of proof. We must demonstrate agent technology's power and benefits. This means getting our hands dirty and building applications that involve the full agent potential. These applications must show how agents can solve existing problems in new, better ways or how they can provide new solutions and services in novel ways. Possible application areas include large-scale, ad hoc monitoring systems, ad hoc network administration, and extensive human-agent interactions (for example, games).
Conclusion

The trick is to do it. Make an application work, show how it's done, get performance statistics, build the necessary tools for design, deployment, and maintenance, and make comparisons against other technologies. Distributed agents is a young research area that should now become more mature and gain the trust of key players in the world. Let's work and live for the moment that agents move away from Fox TV to the Discovery Channel!

Olivier Marin is an associate professor at Laboratoire d'Informatique de Paris 6. Contact him at olivier.marin@lip6.fr.

Niek Wijngaards is a senior researcher at DECIS Lab/Thales Research & Technology Netherlands. Contact him at niek.wijngaards@icis.decis.nl.

Related Links

DS Online's Distributed Agents Community, http://dsonline.computer.org/agents

"Future Directions for Mobile Agent Research", cms:/dsonline/past_issues/0208/f/kot_print.xml


Cite this article: Olivier Marin and Niek Wijngaards, "Distributed Agents Researchers around the World, Unite!" IEEE Distributed Systems Online, vol. 6, no. 11, 2005.