

Current Trends & Future Challenges in Programming MAS

Panel at **ProMAS 2004**
New York, USA, July 20, 2004

Our Issues Today

- What are the Existing MAS Programming Languages?
- What are the Existing MAS Development Methodologies?
- What are the Existing MAS Infrastructures and Tools?
- Requirements?
- Why Everything *Ad Hoc*?
- Vision?

What are the Existing MAS Programming Languages?

- How *easy* is to *implement* concepts such as beliefs, goals, roles, etc. in these programming languages?
- Are these languages *practical* for the industry?
- Are there agent programming languages we know badly designed so that they are not helpful although the concepts on which they are based are?
- What are the *problems* with these languages?
- Which *extensions* of the current mainstream languages are suitable for implementing MAS systems?
- Should MAS programming languages be declarative, imperative, or a combination of them?
- How can the gap between MAS design and MAS implementations be bridged?

What are the Existing MAS Development Methodologies?

- MAS development methodologies uses concepts such as beliefs, goals, roles, etc. to make dealing with complex systems easier
- Do these methodologies provide or suggest programming languages to implement these concepts and the architectures that are designed by the methodologies?
- Has these methodologies been tried in industry, and do they really help?

What are the Existing MAS Infrastructures and Tools?

- Are they adopted by the industry?
- What are the problems with these infrastructures and tools preventing them to be used by the industry?

Requirements?

- Can we formulate some *requirements*
 - theoretical
 - practical
- that MAS programming languages, tools, and infrastructures should satisfy in order to make them interesting for both academia and industry?

Why Everything *Ad Hoc*?

- Why is it that each company seems to have its own ad hoc implementation of the usual agent architectures, and use this for their systems (e.g., Jack, Agentis, etc.)?
- Is this a good and natural thing or just a consequence of the fact that there are no academic tools (which have the right theoretical basis) that are practical yet?
- Should the industry come up with its own solution and perspective to MAS technology?
- In which way can academia help?

Vision?

- What is our *vision* on the development of MAS programming languages, tools, and platforms for the (near) future?

Our Panelists Today

- Monique Calisti
- David Kinny
- Michael Luck
- Onn Shehory
- Franco Zambonelli

Andrea Omicini (Moderator)