Multi-Agent System and their Industrial Applications

Michal Pěchouček

Gerstner Laboratory, AT Group

http://agents.felk.cvut.cz/

What is an agent?

∴ agent is an encapsulated computational system, that is **situated** in some environment, and that is capable of **flexible**, **autonomous** behaviour in order to meet its design objective (Wooldridge).

∴ an agent is not only an **object**, **process**, **program**, **situated** , ..

∴ **critical difference:** agents internal decision making processes are not transparent – one cannot prove what the other agent will do.

example:

$$\{\text{Inform } A\ B\ \alpha\ \varphi\} \cong \{\text{Attempt } A\ \alpha \begin{cases} (\text{Bel } B\ \varphi) \\ \\ (\text{Bel } B\ (\text{Int } A\ (\text{Bel } B\ \varphi))) \end{cases}\}$$

∴ this property (and fact that agents are ususally developed by different developers) causes **emergent behaviour** that has not been thought of at the design time

∴ agents can be **standalone** or members of a **multi-agent system** (we need more to exploit full power of agent paradigm)

Agents – what are they like?

∴ **autonomous** – agents are proactive, goal-directed and act on their own performing tasks on your behalf without necessarily requiring user initiation, confirmation, and notification, do not have to be benevolent, have free will, can cheat, can leave/join the community

∴ **reactive** – agents are triggered by events and sensitive to real-time domain considerations; able to sense and act

∴ **intentional** – ability to maintain agents long term intention, organize its behavior in order to meet targeted goals, agent that uses speech-act-based communication (see ACL), formulates plans in pursuit of its own agenda, and uses reflective reasoning.

∴ **social** – agents collaborate together in communities to achieve a shared goals, they are aware one of the other, they perform reasoning about each other. can group into coalitions, teams, they can benefit from this

Agents – what are they like?

∴ **adaptive** - agents dynamically adapt to and learn about their environment. They are adaptive to uncertainty and change. They can adapt and improve their social role.

∴ **cooperative** - agents coordinate and negotiate to achieve common goals. They are self-organizing and can delegate.

∴ **mobile** - agents move to where they are needed, possibly following an itinerary

∴ **interactive** - agents interoperate with humans, other agents, legacy systems, and information sources

∴ **personal** - agents manifest believable traits such as emotion

∴ **rational agent is** – autonomous, proactive, reactive and social

Agent Technologies, Research Themes

∴ **agent communciation languages** – in the US: KQML ARPA Knowledge Sharing Effort (KSE), in Europe: FIPA-ACL, part of the FIPA standards, currently only the agent-to-agent interaction, agent-to-human is still a challenge

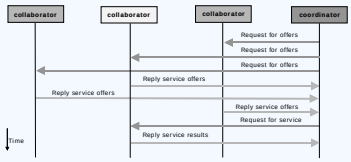
∴ **coordination mechanisms** – classical contract-net-protocol, subscribe-inform coordination, market mechanisms, investigation of truthfulness, fairness, trust

∴ **social knowledge management** – how to acquire, manipulate, store and exploit social knowledge, centrally, in agents – acquaintance models, by means of middle agents (mediators, matchmakers, brokers, facilitators)

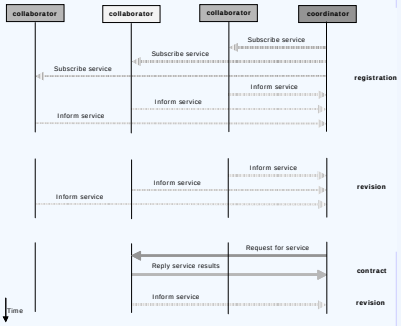
∴ **multi-agent planning** – distributed decision making, decomposition and task distribution, individual agents' conflict resolution, re-planning – partial global planning, acquaintance model team action planning

∴ **knowledge management** – knowledge sharing, ontologies (problem of building general ontologies, ontologies translation) – Ontolingua, OIL (Ontology Inference Layer), DAML (Darpa Agent Markup Language)

Agent Technologies, Research Themes



Agent Technologies, Research Themes



Agent Technologies, Research Themes

∴ **agent communciation languages** – in the US: KQML ARPA Knowledge Sharing Effort (KSE), in Europe: FIPA-ACL, part of the FIPA standards, currently only the agent-to-agent interaction, agent-to-human is still a challenge

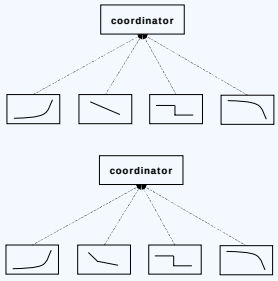
∴ **coordination mechanisms** – classical contract-net-protocol, subscribe-inform coordination, market mechanisms, investigation of truthfulness, fairness, trust

∴ **social knowledge management** – how to acquire, manipulate, store and exploit social knowledge, centrally, in agents – acquaintance models, by means of middle agents (mediators, matchmakers, brokers, facilitators)

∴ **multi-agent planning** – distributed decision making, decomposition and task distribution, individual agents' conflict resolution, re-planning – partial global planning, acquaintance model team action planning

∴ **knowledge management** – knowledge sharing, ontologies (problem of building general ontologies, ontologies translation) – Ontolingua, OIL (Ontology Inference Layer), DAML (Darpa Agent Markup Language)

Team Action Planning



Agent Technologies, Research Themes

∴ **agent communciation languages** – in the US: KQML ARPA Knowledge Sharing Effort (KSE), in Europe: FIPA-ACL, part of the FIPA standards, currently only the agent-to-agent interaction, agent-to-human is still a challenge

∴ **coordination mechanisms** – classical contract-net-protocol, subscribe-inform coordination, market mechanisms, investigation of truthfulness, fairness, trust

∴ **social knowledge management** – how to acquire, manipulate, store and exploit social knowledge, centrally, in agents – acquaintance models, by means of middle agents (mediators, matchmakers, brokers, facilitators)

∴ **multi-agent planning** – distributed decision making, decomposition and task distribution, individual agents' conflict resolution, re-planning – partial global planning, acquaintance model team action planning

∴ **knowledge management** – knowledge sharing, ontologies (problem of building general ontologies, ontologies translation) – Ontolingua, OIL (Ontology Inference Layer), DAML (Darpa Agent Markup Language)

Agent Technologies, Research Themes

∴ **negotiation strategies** – advanced auction mechanism design (in one-to-many environment), negotiation techniques on (one-to-one environment), application is stockexchange trading and supply chain management

∴ **learning** in multi-agent systems – agents adapting ot the environment (agent-human interface), improving their knowledge from the past interaction, community structure reconfiguartion – particular utilisation in the ubiqutous environment

∴ **monitoring, meta-reasoning** – a novel progressive concept for controlling and monitoring emergent behaviour in multi-agent system, for intrusion detection, modelling and simulation

∴ **coalition formation** and **teamwork** – techniques for efficient coalition formation and team-action planning – problems with centrality and reasoning complexity

∴ **architectures** – individual agents' architectures: reasoning processes, knowledge representation, integration & agents' communities archietcture – organization, roles, hierarchy, infrastructures

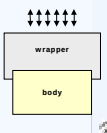
Agents Reasoning Architectures

∴ **deductive reasoning agents** – agents based on the theoretical reasoning paradigm, they operate as theorem provers, logic centred view on agency (situation calculus, AOP, ConcurentMetateM, ...)

∴ **practical reasoning agents** – deliberation (deciding which goals to achieve) and means-end-reasoning (planning how to achieve it), relies on specific plan library, concept of intentions, Belief-Desire-Intention model

∴ **reactive agents** – model of the outer environment is the environment itself, intelligence is encoded in layered if-then rules, high-level intelligence emerges from interactions between the agents and the layers – subsumption architecture – **hybrid agents** – interaction between the layers is managed by the control subsystem

∴ **integration architectures** – there is potential of integration of pre-existing computational systems. The agent is divided into body and wrapper.

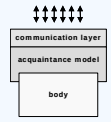


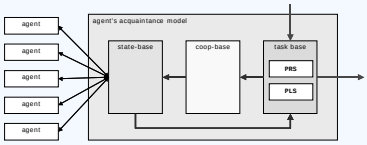
agentification is the process of converting an old, 'non-agent', 'stand-alone' system into an agent/ holon being able to be integrated with a certain multi-agent community

??? Agents Reasoning Architectures

∴ **acquaintance models** – computational models of agents mutual awarness. Safer, robust and more efficeint commucniation. Implements agents

∴ **tri-base acquaintance model** – 3bA





Agency vs. Interoperability

∴ **agent infrastructure** supports development and operation for the whole agent-based system, a technical middleware implement for implementing interaction

∴ **infrastructure technologies**: TCP/IP, CORBA, HTML, XML/RDF, Jini, JAS

∴ **available infrastructure**: agentcities (fipa-compliant), COABS (jini-based)

∴ **interoperability** is the mean for achieving agency. Interoperability can be implemented on the technological level and on the level of standards

∴ there were several standard initiatives in the agent community – KSE (Knowledge Sharing Effort), KQML (Knowledge Query Manipulation Language), KIF (Knowledge Interchange Format)

the leading and industry-wide recognised standard body for agents' interoperability is **FIPA** – Foundation for Intelligent Physical Agents

FIPA – Foundation for Intelligent Physical Agents

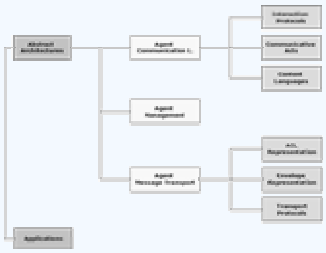
∴ a non-profit association registered in Geneva, Switzerland, founded in December 1995. The main goal of FIPA is to maximize interoperability across agent-based applications, services and equipment. It is done through FIPA specifications.

∴ specifies the set of interfaces which the agent uses for interactions with various components in the agent's environment, i.e. with humans, other agents, non-agent software and the physical world. It focuses on specifying **external communication among agents** rather than the internal processing of the communication at the receiver.

∴ a high-level, **neutral abstractions** with respect to (a) the application area and (b) hardware and software platforms

- **normative specifications** that mandate the external behaviour of an agent and ensure interoperability with other FIPA-specified subsystems and
- **informative specifications** of applications for guidance to industry on the use of FIPA technologies.

FIPA – Specifications



FIPA Agent Development Platforms

∴ **FIPA-compliant**: April Agent Platform (Fujitsu Labs of America), FIPA-OS (Emorphia), Grasshopper (IKV++), JADE (CSELT), Zeus (British Telecom).

∴ **FIPA-OS** complies with the majority of the FIPA specifications – fully implemented in Java and features two FIPA-OS distributions (http://www.emorphia.com, formerly by Nortel Networks):

- Standard FIPA-OS – one distributions running under JDK1.2 and another running under JDK1.1 and
- MicroFIPA-OS which is an extension to the JDK 1.1 version of FIPA-OS, designed primarily for execution agents on PDAs.

∴ **JADE** – Java Development Toolkit – (http://sharon.cselt.it/projects/jade/) – requires JDK LGPL (Lesser General Public License Version 2).

- LEAP (Lightweight Extensible Agent Platform) that allows FIPA-compliant agents to run on PDAs and cell phones (http://leap.crm-paris.com/).

Agency in Industry

∴ **design architecture** (e.g. Prosa architecture, Holonic Manufacturing Systems, ProPlanT architecture, etc.)

∴ **distributed decision making algorithms** (e.g. stigmergy, negotiation and auctioning, social intelligence based interaction, etc.)

∴ **integration/agentification technology** (e.g. FIPA standards, agent development environments)

Design Architectures

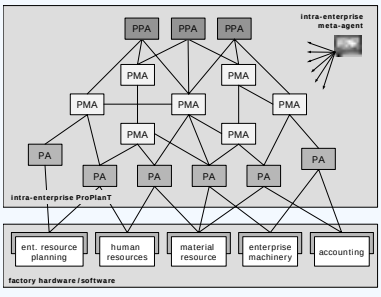
∴ **hierarchical**: In manufacturing we very often require predefined architecture with rather rigid command and control structure. These systems are very often closed where no unknown agent can easily get integrated within the community.

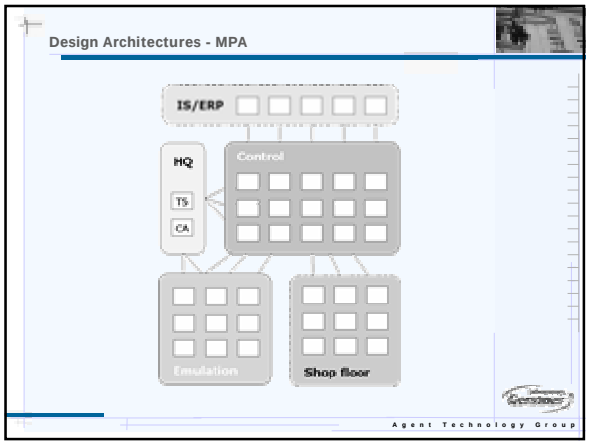
∴ **peer-to-peer**: In e-business and automated trading (including the domain of supply chain management and logistics) we usually require a community autonomous agents – such a community needs to be open to new agents

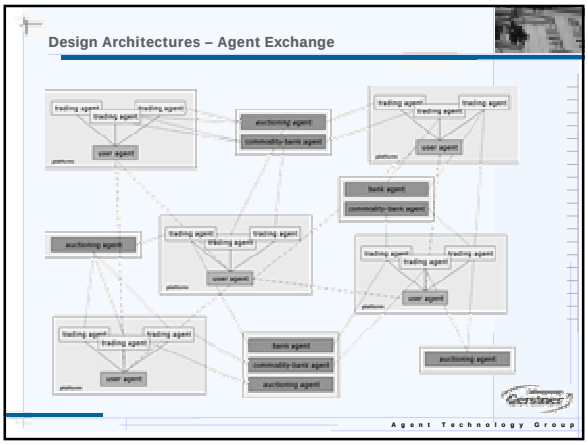
∴ PROSA – a holonic architcture with several classes of agents:

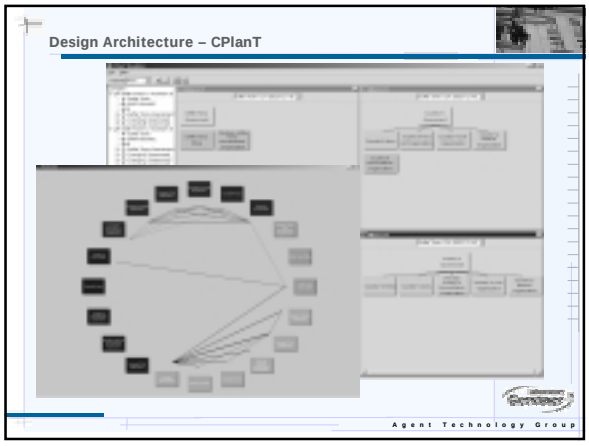
- product agent
- resource agent
- order agent
- staff agent

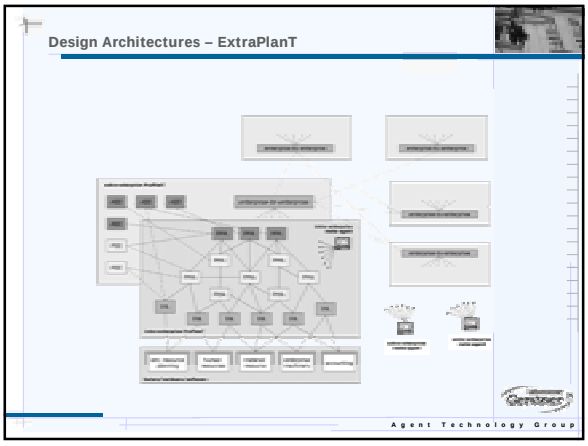
Design Architectures - ProPlanT




Design Architectures - MPA
IS/ERP
HQ
TS
CA
Control
Emulation
Shop floor
Agent Technology Group


Design Architectures – Agent Exchange
Agent Technology Group


Design Architecture – CPlanT
Agent Technology Group


Design Architectures – ExtraPlanT
Agent Technology Group