



Software Agents Theory and Practice

Frank Dignum, Virginia Dignum,
Mehdi Dastani,

Utrecht University



Issues to be discussed

- Software agents?
- Definition of software agents
- Theory and architecture of agents
- Multi-agent systems
- Building Agents
- Applications (general)
- Agents and Information Management
- Applications
- Current developments



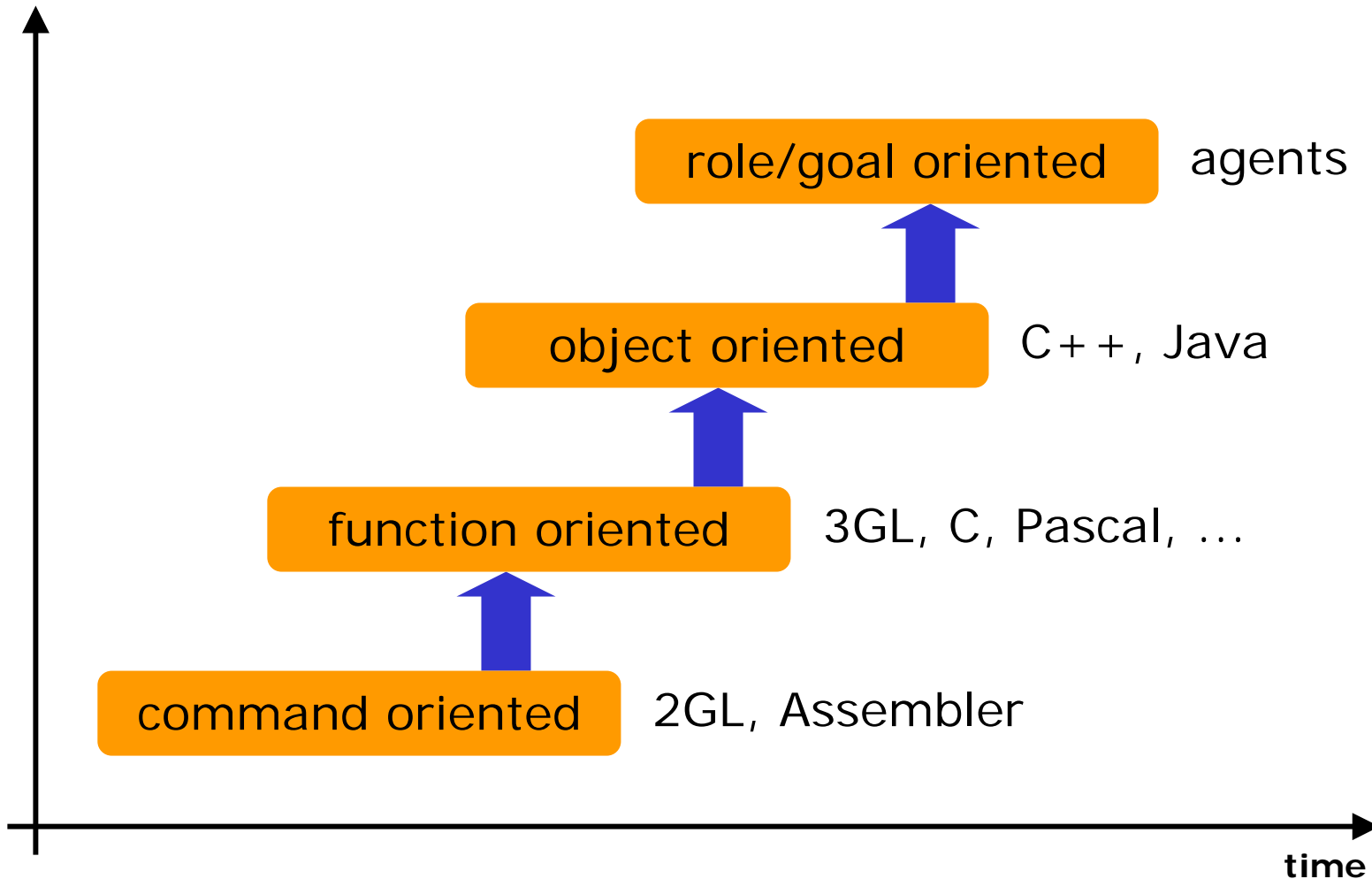
Software agents?

- **Development metaphor**
 - Software Engineering
 - Methodology
- **Technology**
 - agent theory
 - agent architectures
 - agent languages



Paradigm Shifts

real world mapping
(abstraction level)





Definition of agents (weak notion)

- Autonomous
- Pro-active
- Reactive
- Social ability



Definition of agents (strong notion)

- Belief
- Desire
- Intentions
- Goals
- Knowledge
- Obligations



Other possible attributes

- Rationality
- Veracity
- Mobility
- Learning capacity
- Cooperativeness



Theory of agents

- Represent mental attitudes
 - knowledge, belief, goals, etc.
- Reason about mental attitudes
- Plan actions
- “Observe” changes
- Update mental attitudes



From Theory to Architecture

- How to use the theory in architecture of agents:
 - Only for representation of attitudes
 - As formal specification of agent implementation
 - Also use the logical inferencing of logic in deliberative agent architecture



Agent architectures

- **Deliberative agents**
 - BDI agents (mostly theory)
 - planning agents (IRMA)
- **Reactive agents**
 - Brooks' subsumption architecture
- **Hybrid agents**
 - Interrap

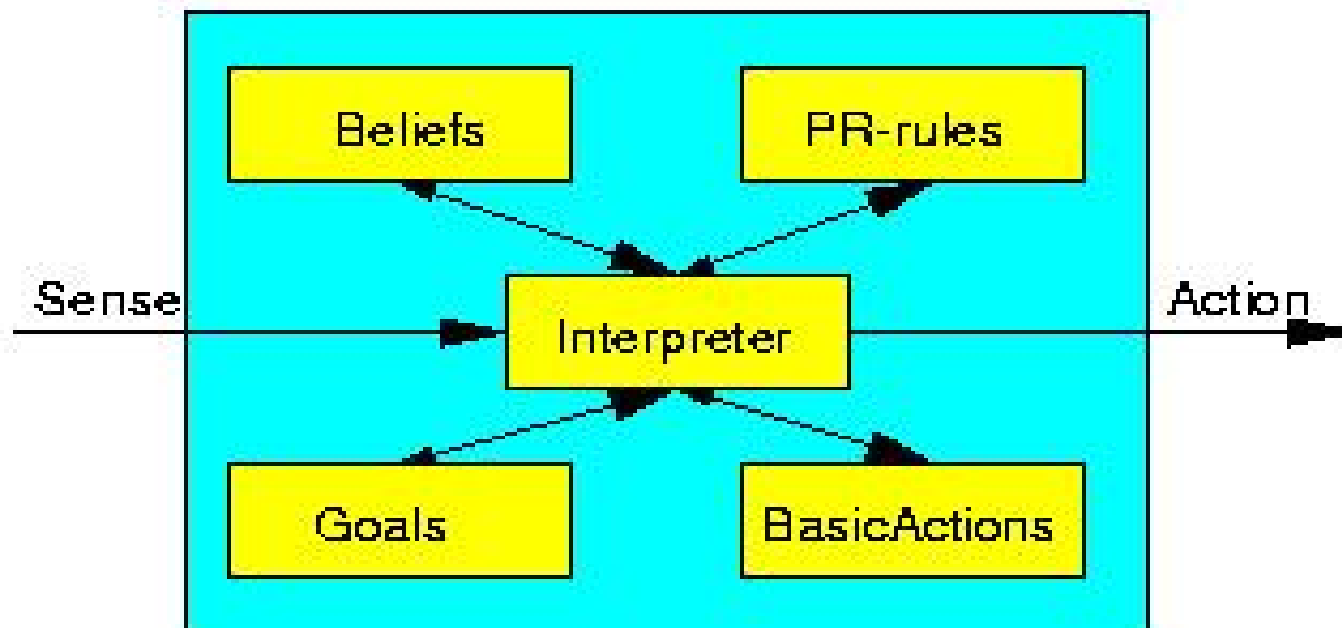


Building Agents

1. Agent Oriented Programming (e.g. **3APL**)
 - Close to agent theory, but far from industrial use
2. Based on Java components (e.g. **Jade**)
 - More robust, but build intelligence yourself in Java
3. Based on robust infrastructure (e.g. **Tryllian ADK**)
 - Industry standard systems (robust, efficient, scalable), but no intelligence



3APL agents (I)



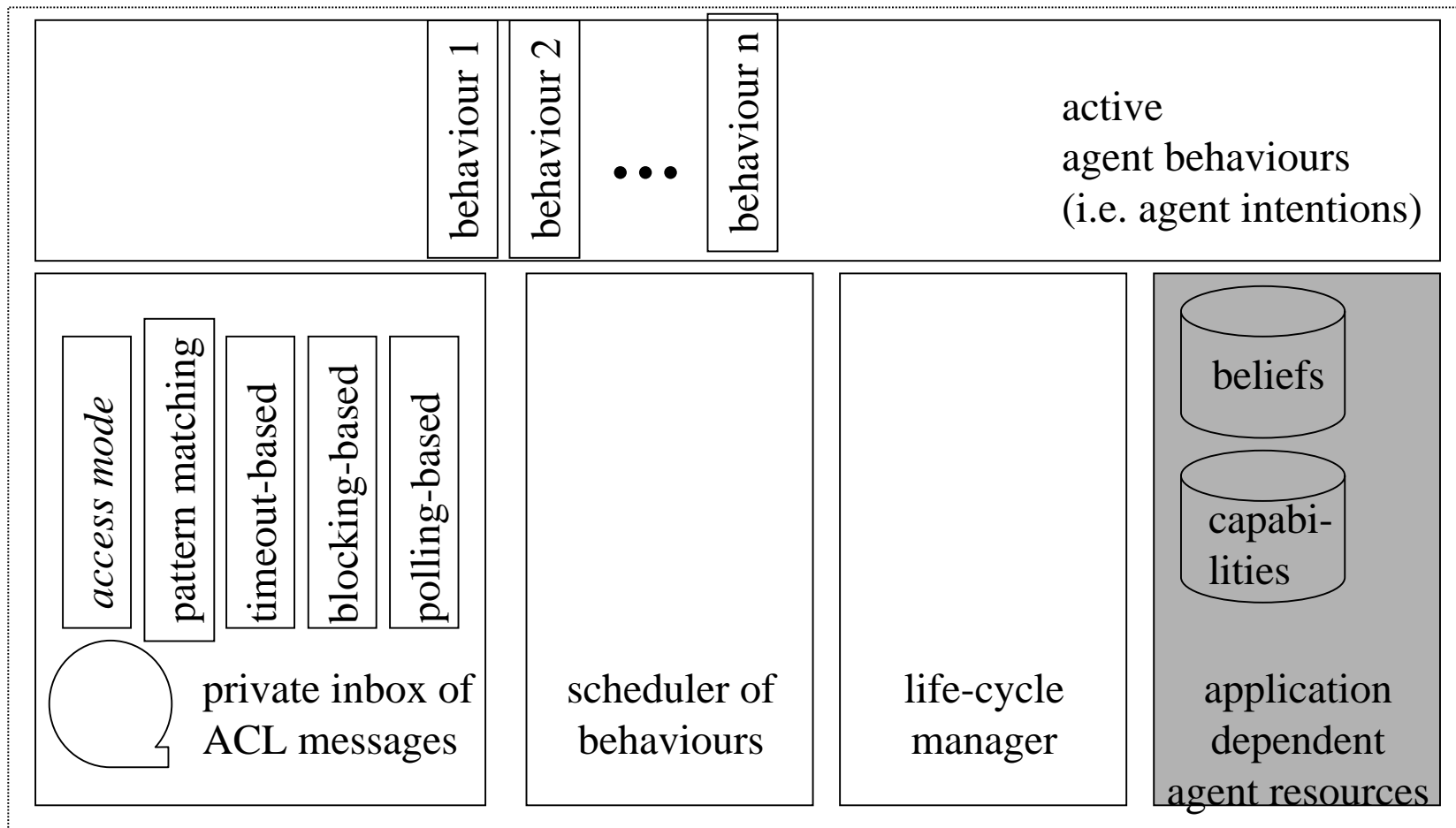


3APL agents (II)

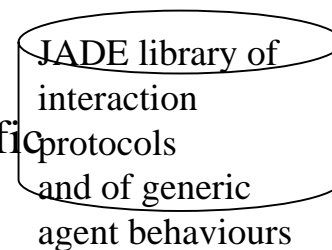
- **PROGRAM** "patrol_agent.3apl"
- **CAPABILITIES:**
{ at_east(self) } WalkWest() { **NOT** at_east(self) , at_west(self) } ,
{ at_west(self) } WalkEast() { at_east(self) , **NOT** at_west(self) }
- **BELIEFBASE:**
at_west(self)
- **GOALBASE:**
patrol()
- **RULEBASE:**
patrol() ← at_east(self) | WalkWest() ; patrol() ,
patrol() ← at_west(self) | WalkEast() ; patrol() .



JADE agents



The JADE framework includes a library of interaction protocols and generic agent behaviours, that must be customized for the specific application needs in order to create the agent capabilities





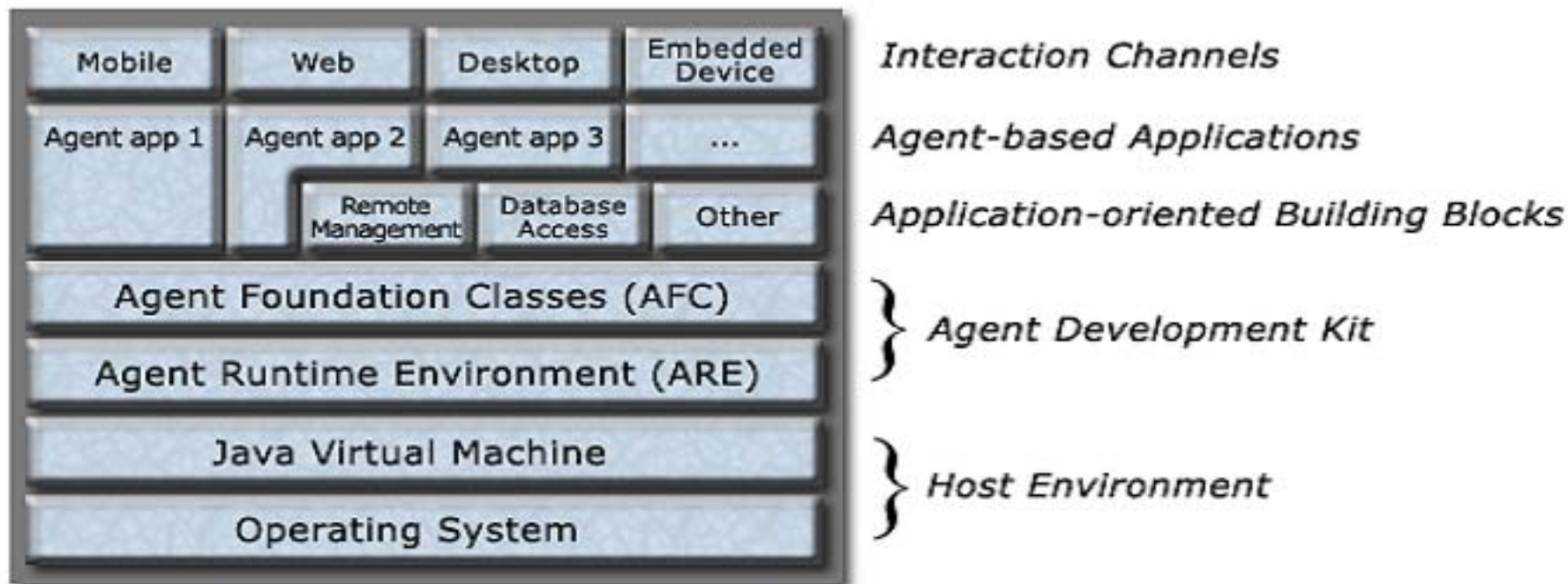
Tryllian ADK (I)

- The *Agent Foundation Classes* (AFC) providing tools, libraries, building blocks and examples for building Agent Based Applications
- The *Agent Runtime Environment* (ARE) a “habitat” for hosting agents developed using the AFC
- *Agent Management Tools* for managing agents on a server



Tryllian ADK (II)

System Architecture





From Agents to Multi-Agent Systems

- Agent **communication**
- Agent **societies**
 - realise there are other agents
 - use other agents for your actions
- **Collective** plans, goals, etc.
- MAS vs. Autonomous agents



Some General Application Areas:

1. Industrial applications

- manufacturing
- process control
- telecommunications
- transportation systems

2. Electronic Commerce

- electronic markets/auctions
- buying agents (e.g. Jango, shopbot, etc.)

3. Business Process Management

4. Information Management

- information gathering
- information filtering



Industrial applications (I)

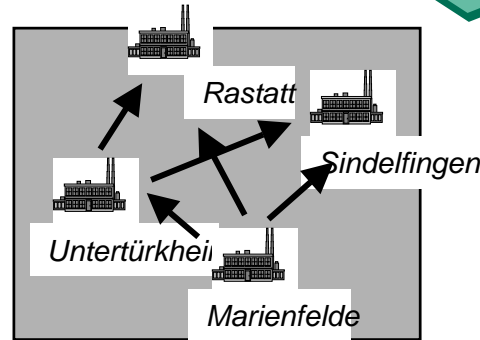
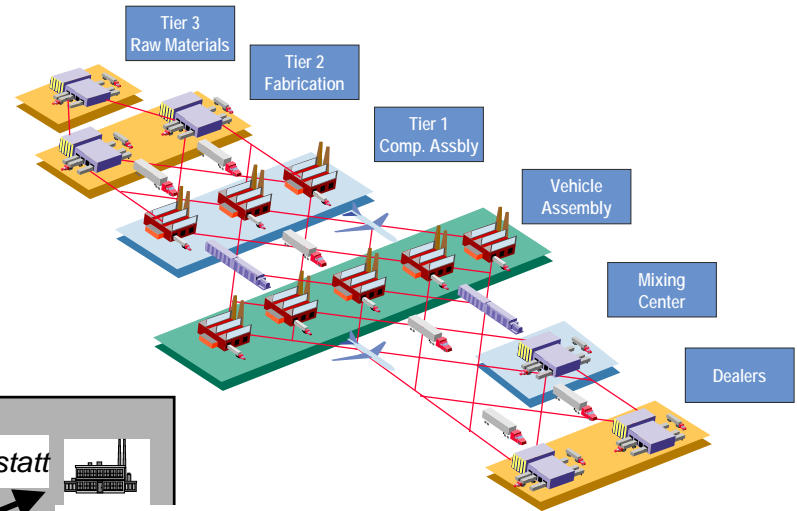
Manufacturing at Daimler Benz



Manufacturing and supply nets Daimler Benz

Objective

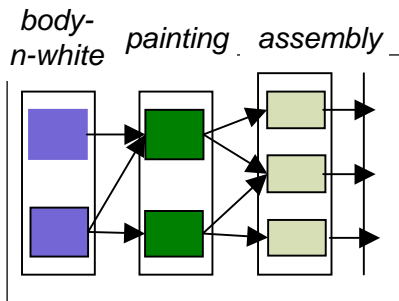
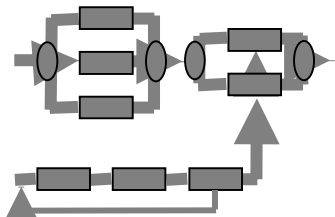
improve manufacturing
and supply processes



plant network

inter-organizational
supply net

production lines



vehicle assembly plant

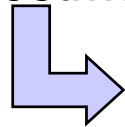




Motivation: *Business trends*

Growing surplus of industrial capacity and globalization lead to

- shorter product life-cycles
- reduced time-to-market
- increased product variety
- volatile demand
- reduced investment costs



New manufacturing requirements

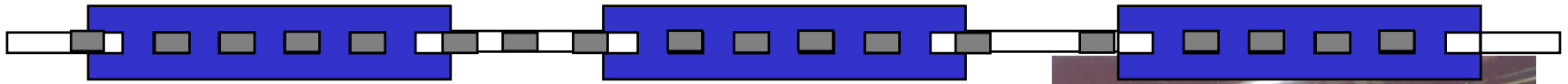
→ **product flexibility**

→ **volume scalability**

→ **robustness**

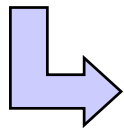


Production 2000+: Flexible and robust production system

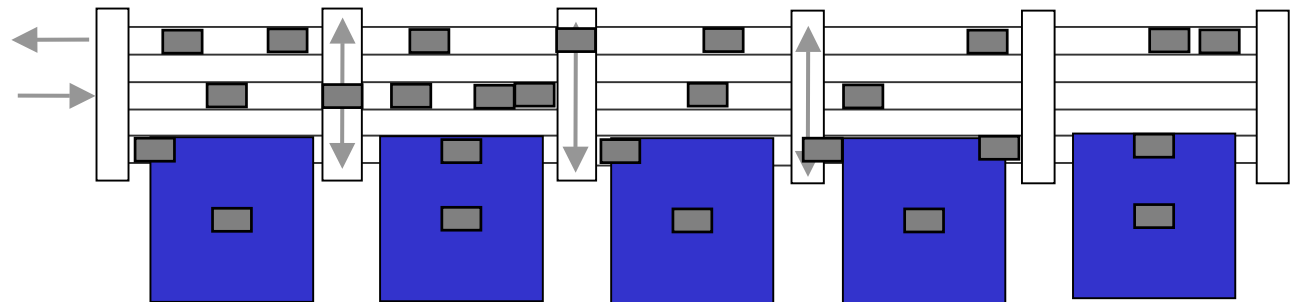


Disadvantages of transfer lines

- dedicated machines
- rigid material flow



introduce more hardware flexibility

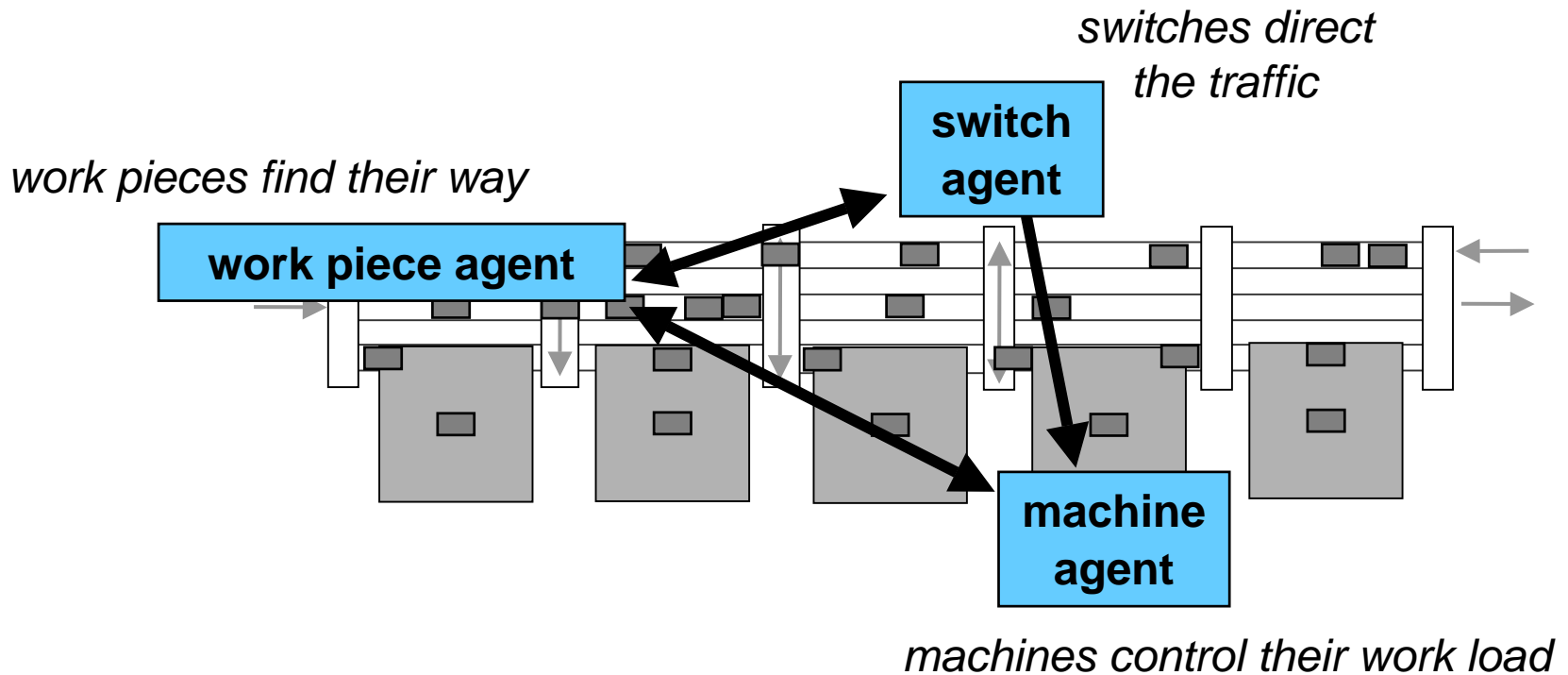


flexible machine

flexible transportation system



Production 2000+: An agent-based control system

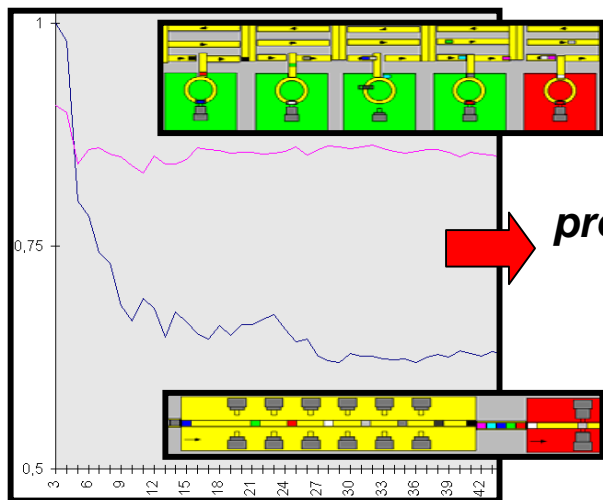


Control forces:

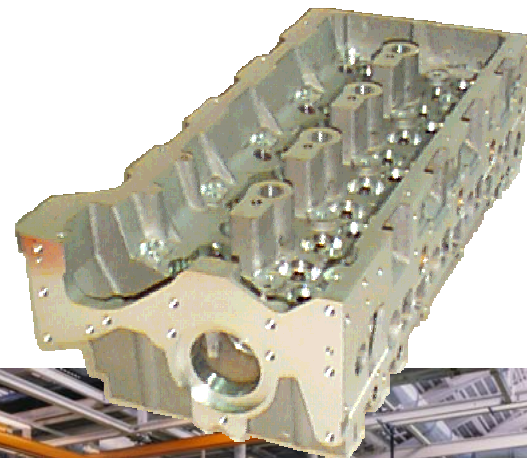
- work pieces push themselves
- machines limit work-in-process



Production 2000+: *Field test*



**productivity increase
at least 10%**



**by-pass to
existing transfer line**





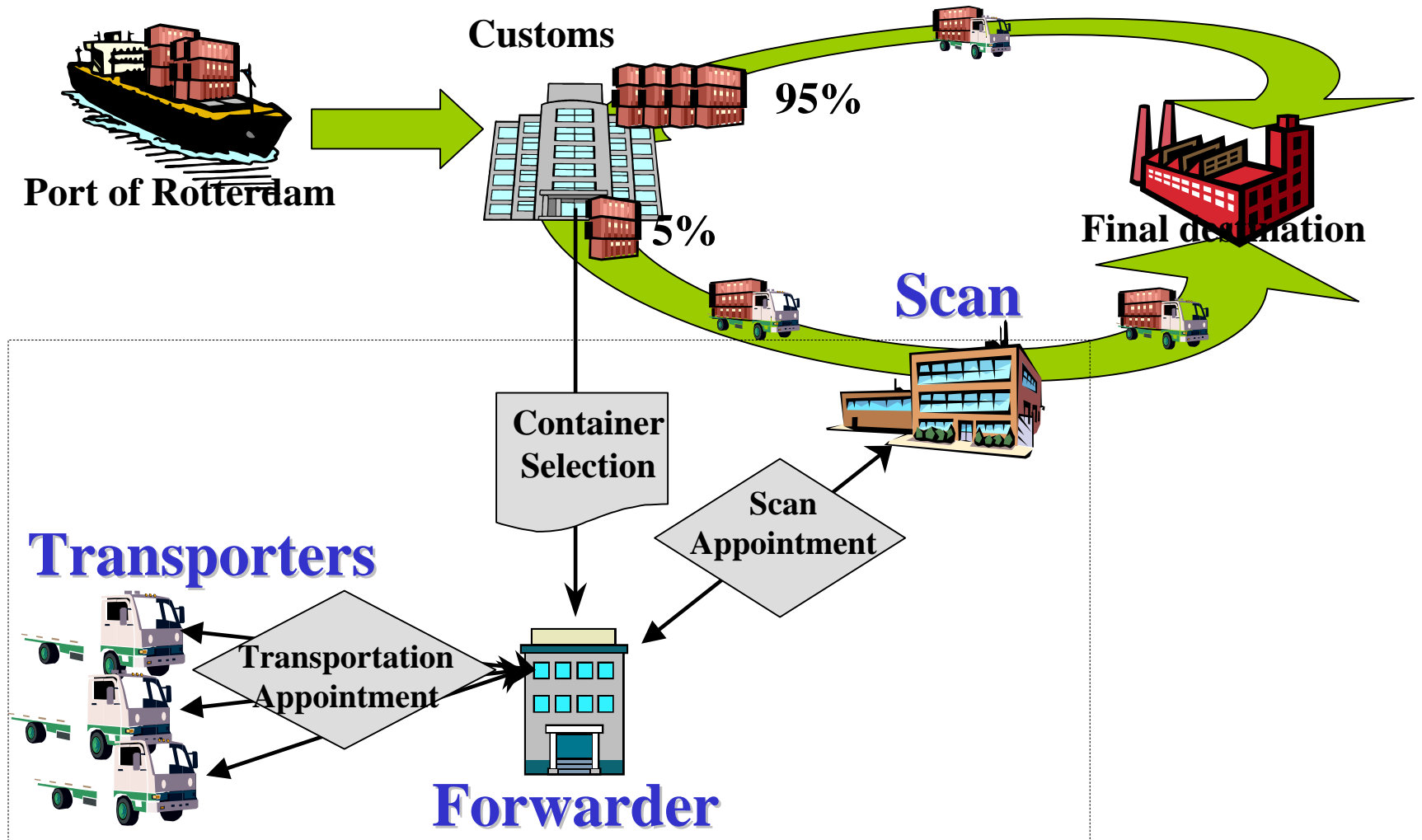
Industrial applications (II)

Container transport in Rotterdam

Tryllian

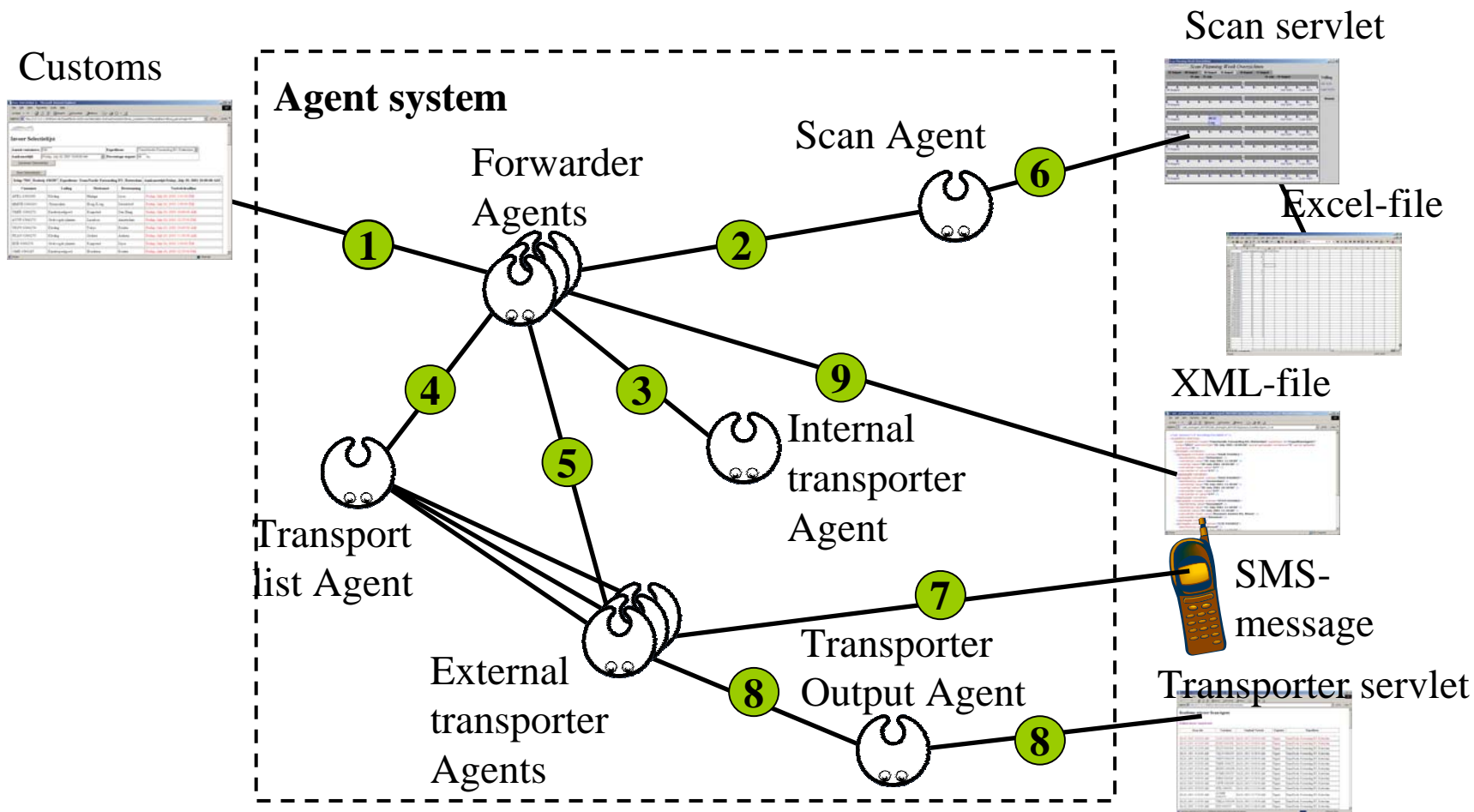


Container scanning





System overview





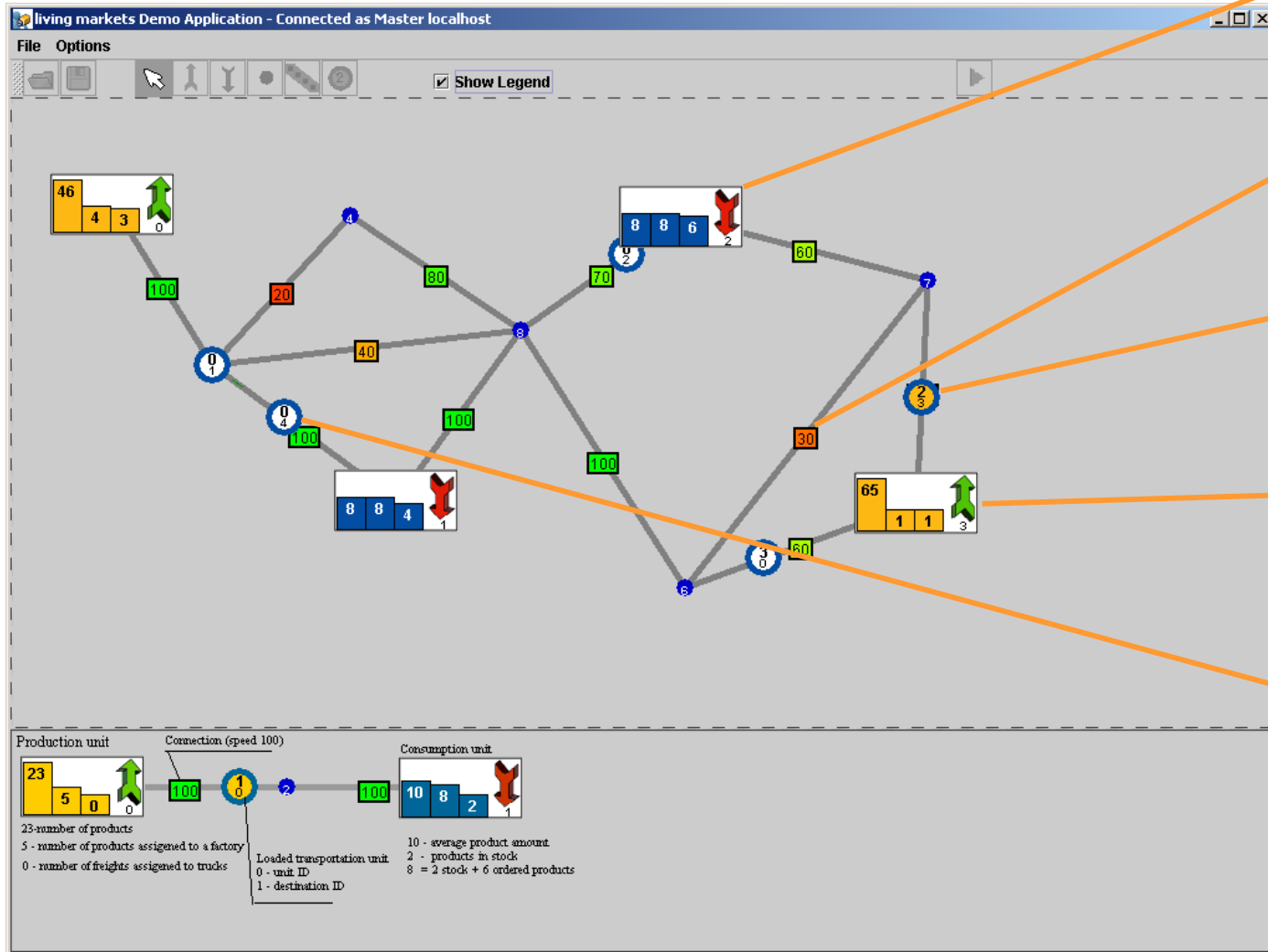
Industrial applications (III)

Managing transportation

Living Systems



MAS for transport management



consumption units (store)

connections with varying speed (road)

transportation units (truck)

production units (factory)

trucks inform each other about slow traffic



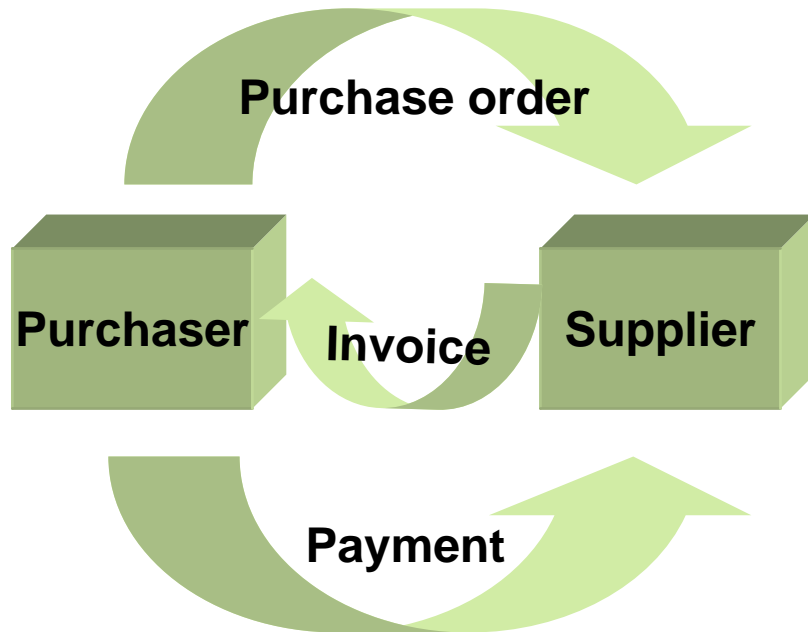
Electronic Commerce (I)

Electronic Markets and Auctions

Hewlett Packard



Electronic Data Interchange (EDI)



Business Partners electronically exchange trading documents in a pre-arranged format.

Advantages:

Reduced paperwork, faster transactions, easy to automate.

Disadvantages:

Lock-in to small number of business partners.



Electronic Markets

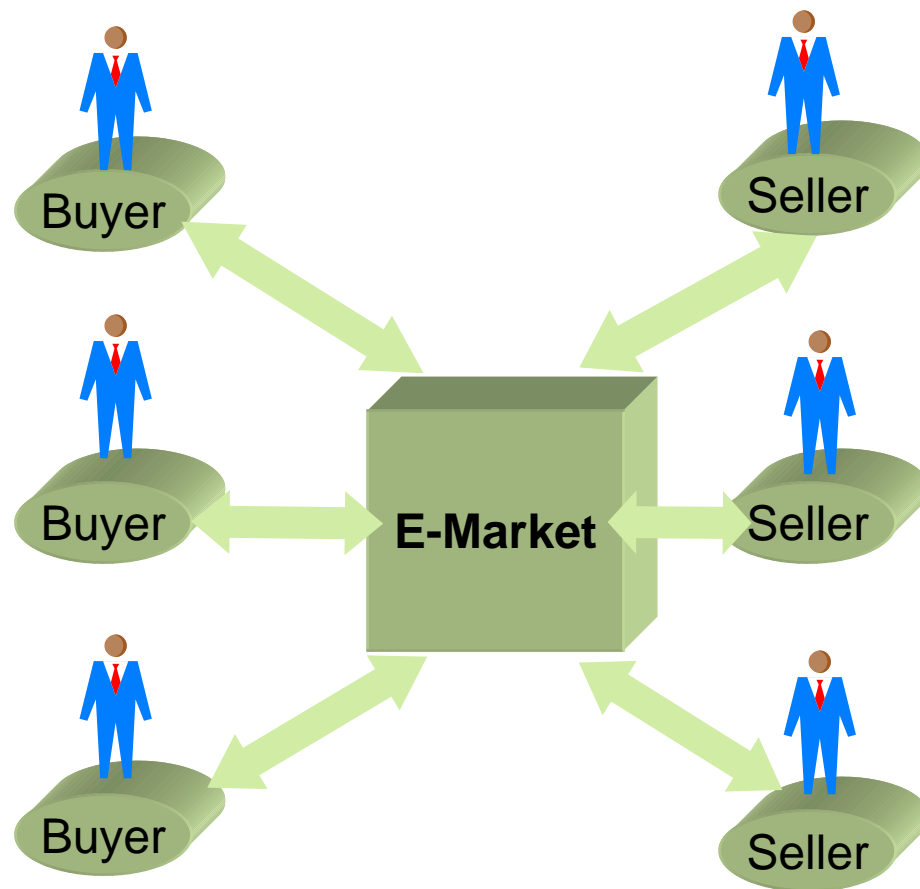
Many potential trading partners meet and negotiate at an internet site.

Advantages:

Very flexible. More competition, leading to better deals.

Disadvantages:

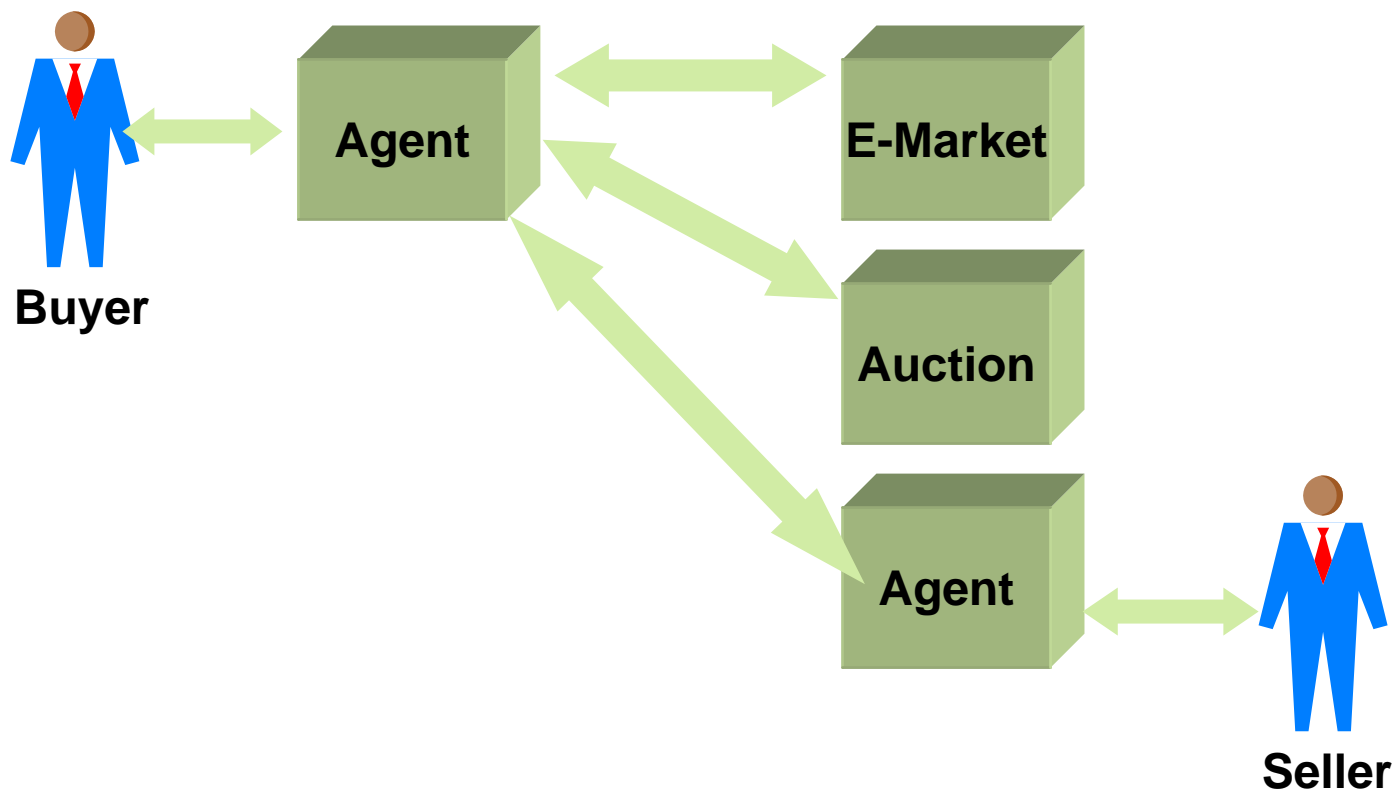
Not automated, so slower and more labour-intensive than EDI.





Agent-Mediated Electronic Commerce

Agents represent buyers and sellers, and can participate in many Electronic markets.



Advantages: Best of both worlds – Flexible but automated.

Disadvantages: Technology not yet widely accepted.

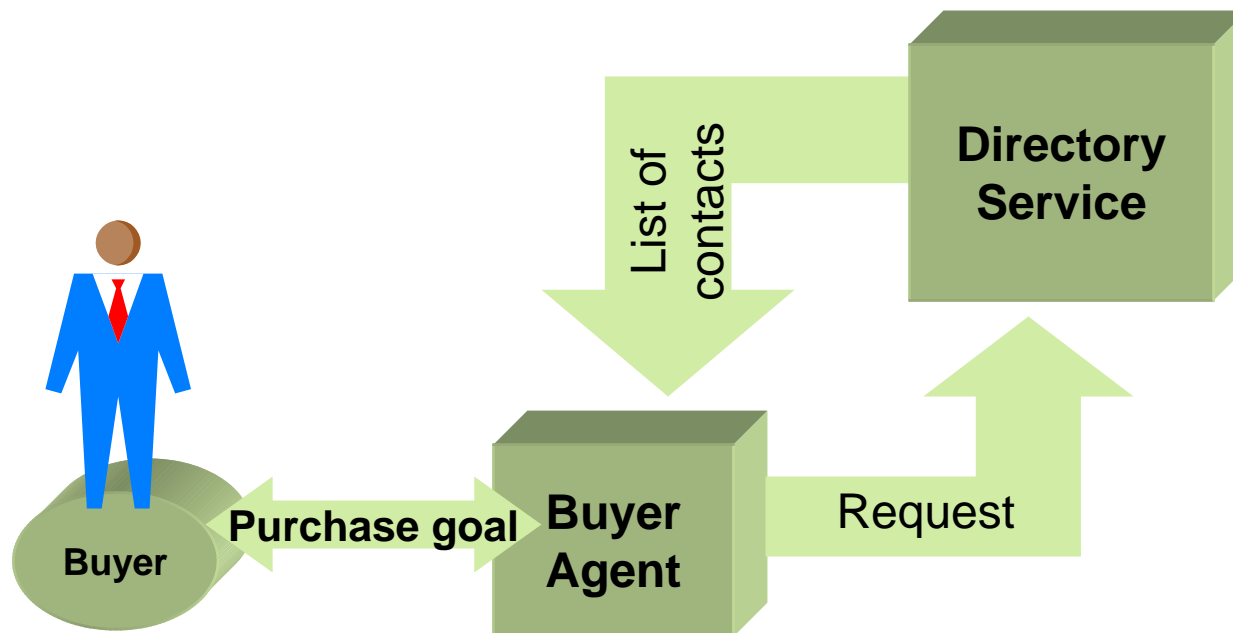


The Stages of E-Commerce

- Matchmaking
- Negotiation
- Contracting
- Contract Fulfilment

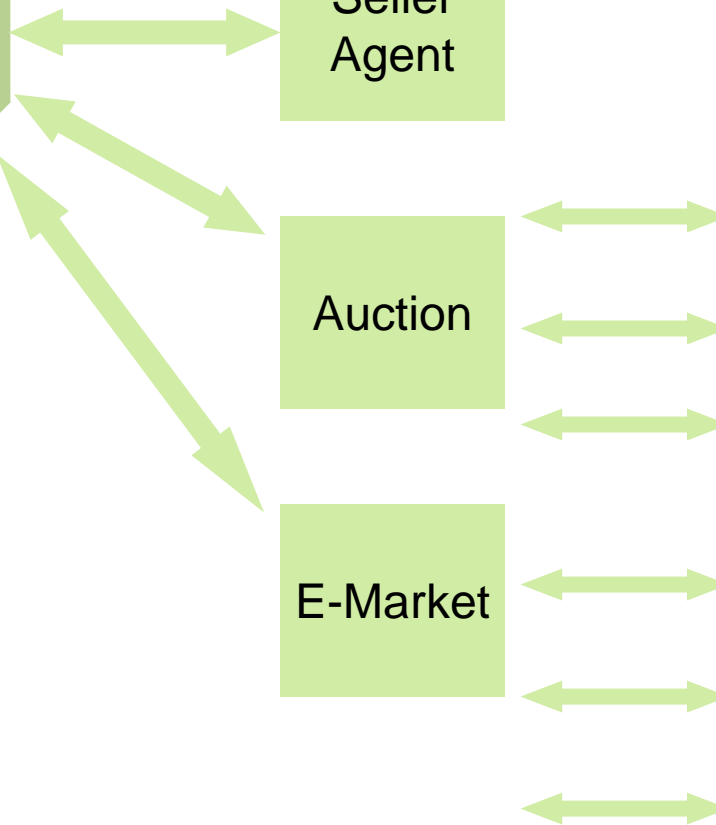
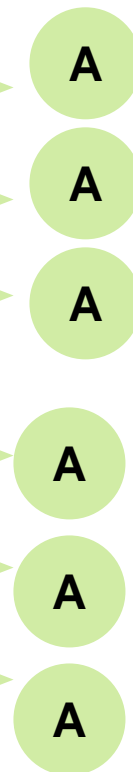
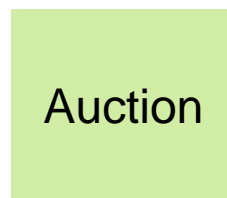


Matchmaking



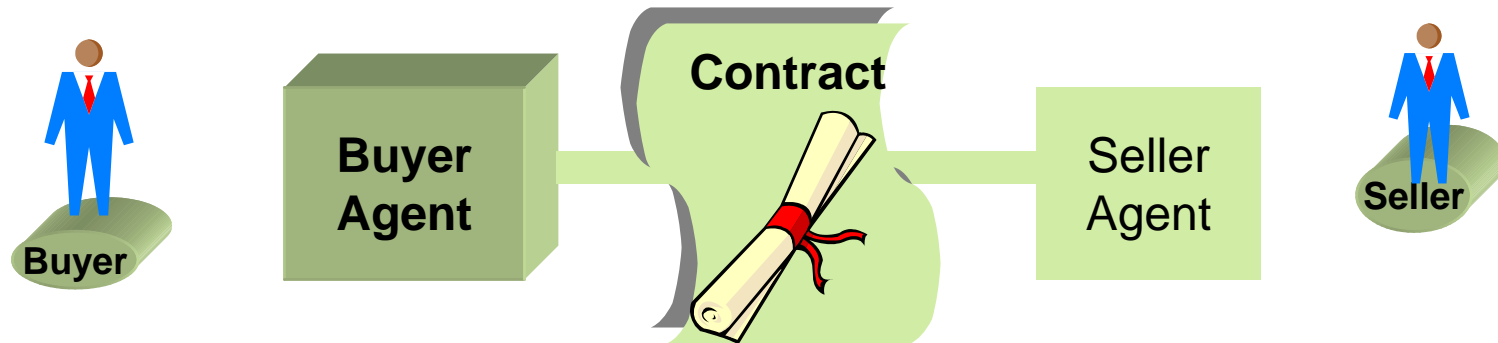


Negotiation



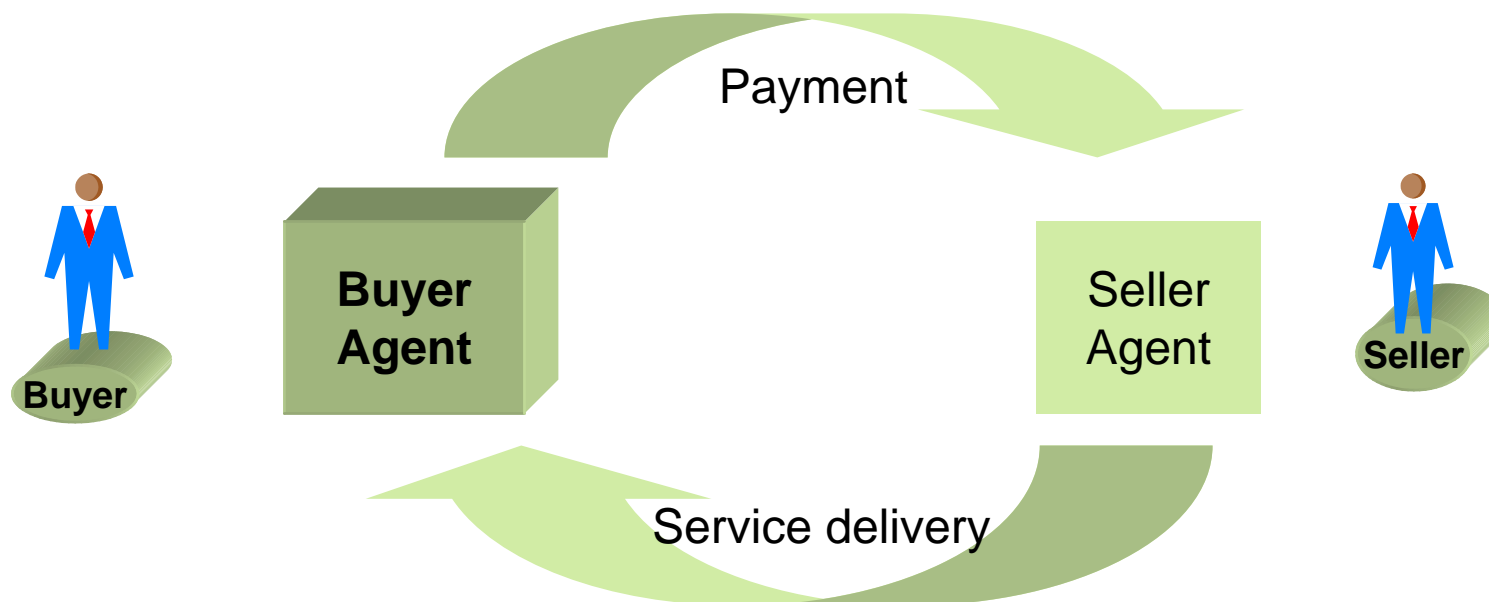


Contracting





Contract Fulfilment





Electronic Commerce (II)

Research competition:

Trading Agent Competition

www.sics.se/tac/

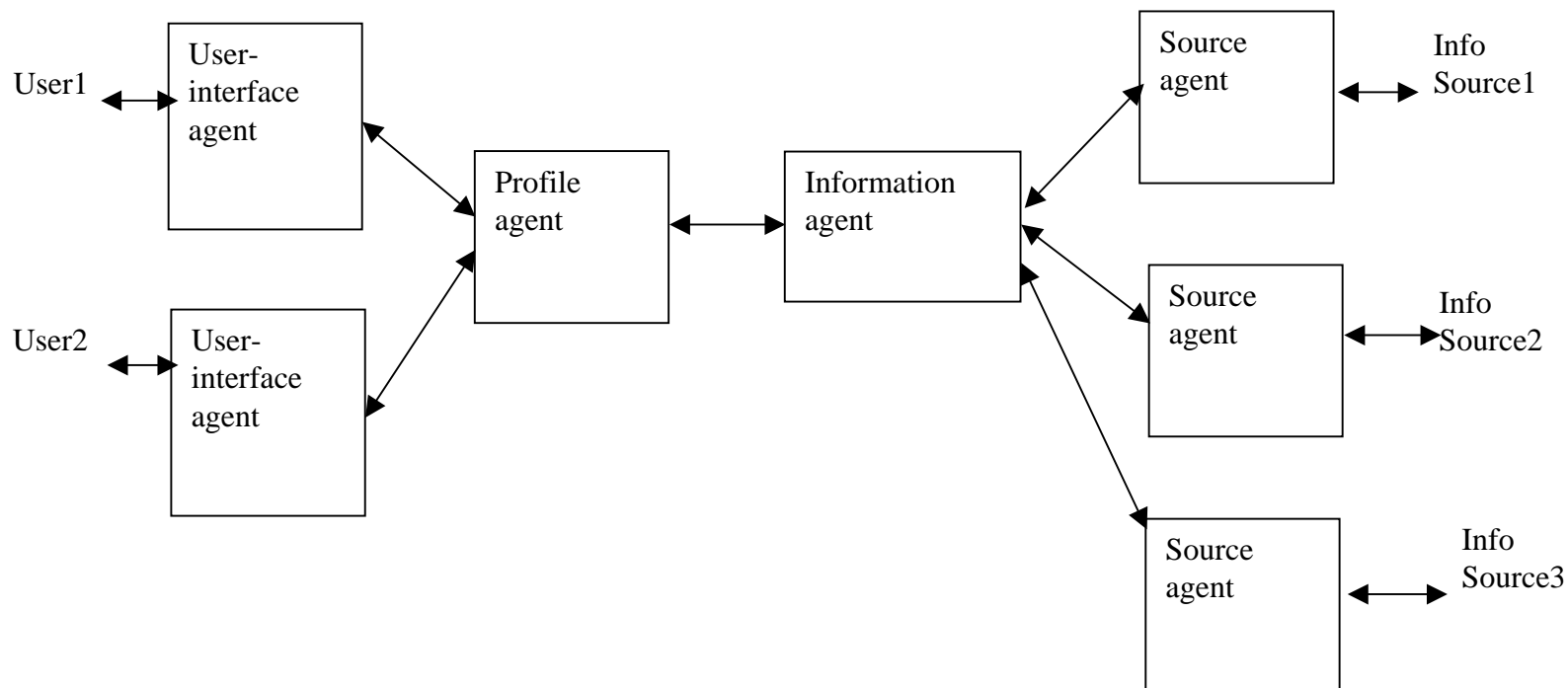


Information Management

1. Theoretical architecture
2. Examples from Practice

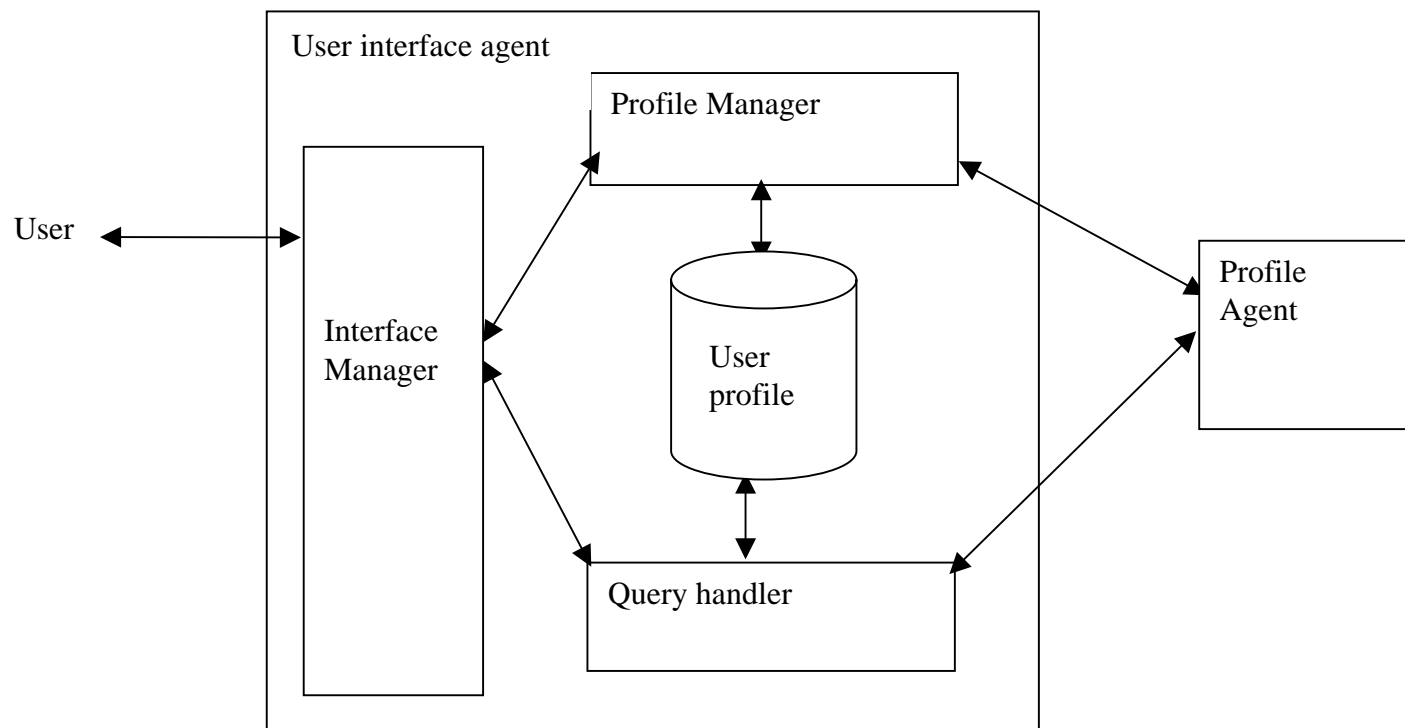


General MAS Architecture for Information Management





The User interface agent



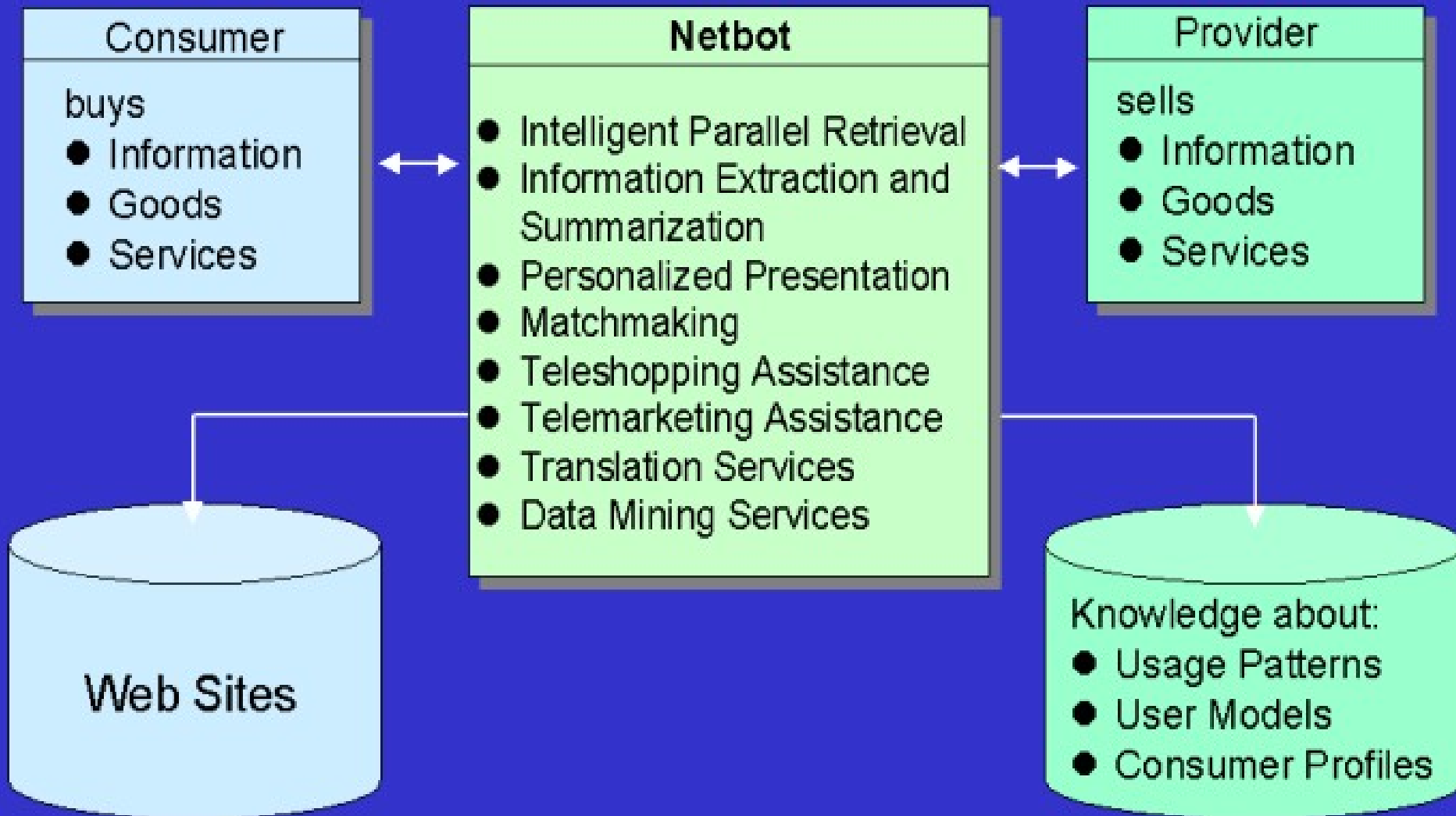


Information Management

Gathering tourist information

DFKI

Intelligent Web Services



AiA: Information Integration for Virtual Webpages

PAN Travel Agent Andi

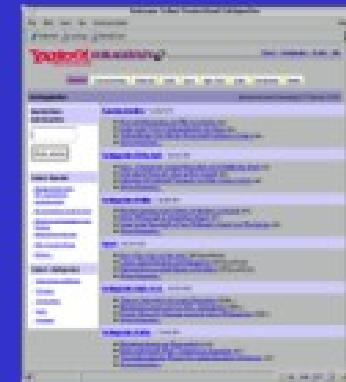


Car Route Planner



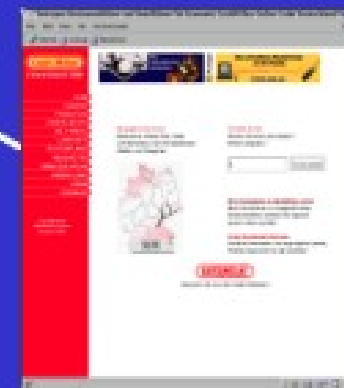
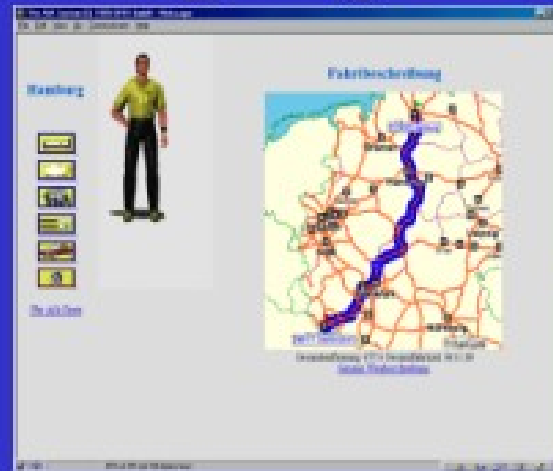
Yahoo
Weather
Server

Yahoo
News
Server



Hotel
Guide

Gault Millau
Restaurant
Guide





Information Management

Remote system and data
management

Tryllian

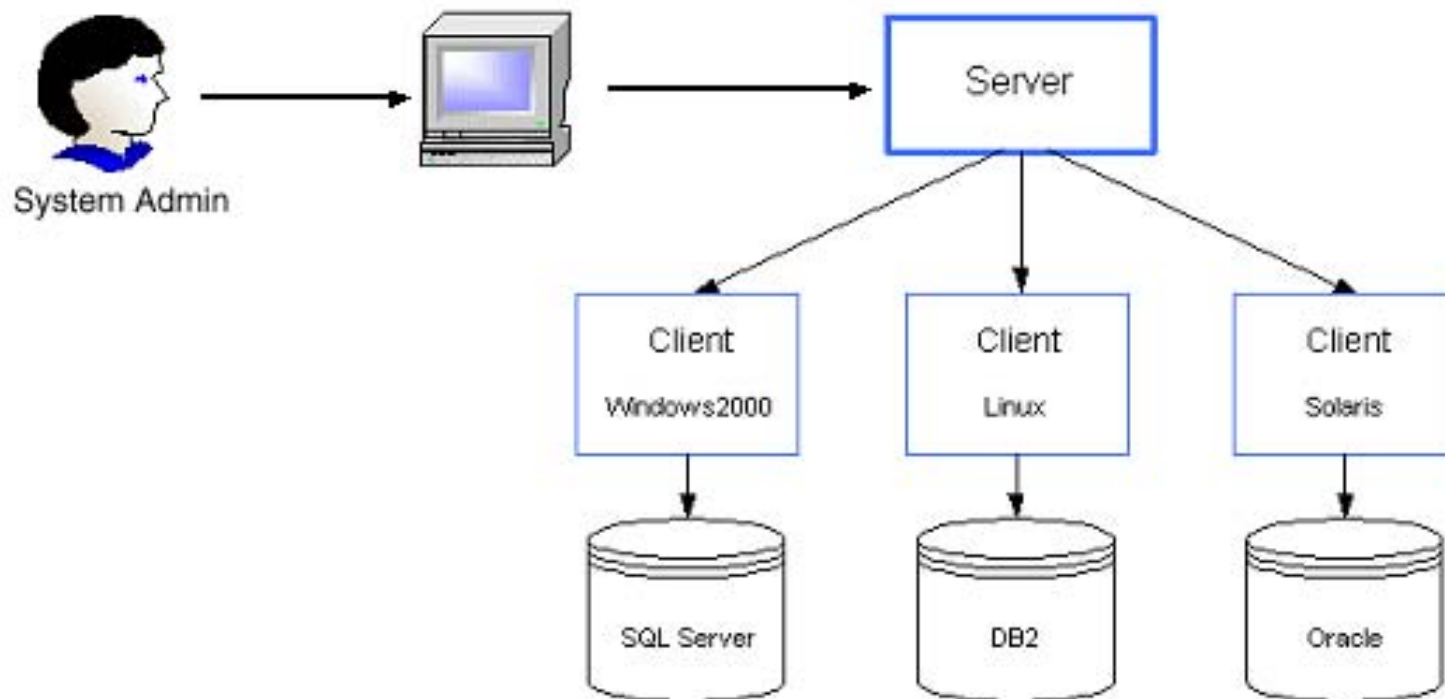


Overview

- Network discovery
- Remote software installation
- Database discovery
- Data retrieval
- Upgrade Agent functionality dynamically



Architecture





Agent research and practice

- AgentLink: Network of excellence
 - Research groups in agents
 - www.agentlink.org
- Agent Cities: Network for deploying agent applications
 - www.agentcities.org
- FIPA: foundation for agent standards
 - www.fipa.org



Conclusions

- Agents are here to stay!
- Agents are NOT simple
- Difference between theory and implementation
- Useful in complex domains:
 - Inter-organizational applications
 - Internet applications
 - Distributed applications
- Agents have a high potential
- Agents are increasingly used in practice