Social Knowledge in Multi-Agent System

Michal Pechoucek, Vladimir Marik, Olga Stepankova

Gerstner Laboratory for Intelligent Decision Making,
Czech Technical University in Prague

http://labe.felk.cvut.cz/~pechouc/acai/lecture.zip

What to look forward to?

- the concept of social knowledge
- agents' acquaintance models, 3ba model, knowledge maintenance
- communication traffic experimental results
- social knowledge improvement, meta-reasoning
- meta-agents and reflection in multi-agent systems
- application of social knowledge in industry
- ProPlanT demo

Motivation

- agents' are autonomous, reactive, proactive but also socially capable
- through social-oriented decision making (aimed at social behaviuor) the agent deliberates about existence, properties and behaviour of the other agents
- why so?
  - (i) smarter and more efficient coordination in the case of cooperative agents and
  - (ii) getting an advantage for the self-interested agents in negotiations, auctions, etc...
- an important component of the social-oriented decision making is social knowledge – we shall investigate social knowledge representation, maintenance and exploitation

we will be discussing the concept of social reasoning with respect to agents decision making autonomy

Social Knowledge (SK)

- agent's knowledge is either:
  - problem solving knowledge – "asocial" type of skill – guide agent's autonomous local decision making processes (aimed e.g. at providing an expertise or search in the agent's database)
  - self knowledge – knowledge about agent's behavior, status and commitments (a special instance of social knowledge – below)
  - social knowledge – knowledge about other agents, their behavioral patterns, their capabilities, load, experiences, commitments, but also knowledge and belief

knowledge is a true belief:

$$(\text{know } A\,\varphi) = (\text{bel } A\,\varphi) \wedge \varphi$$

therefore we assume that all the agent is aware of is true

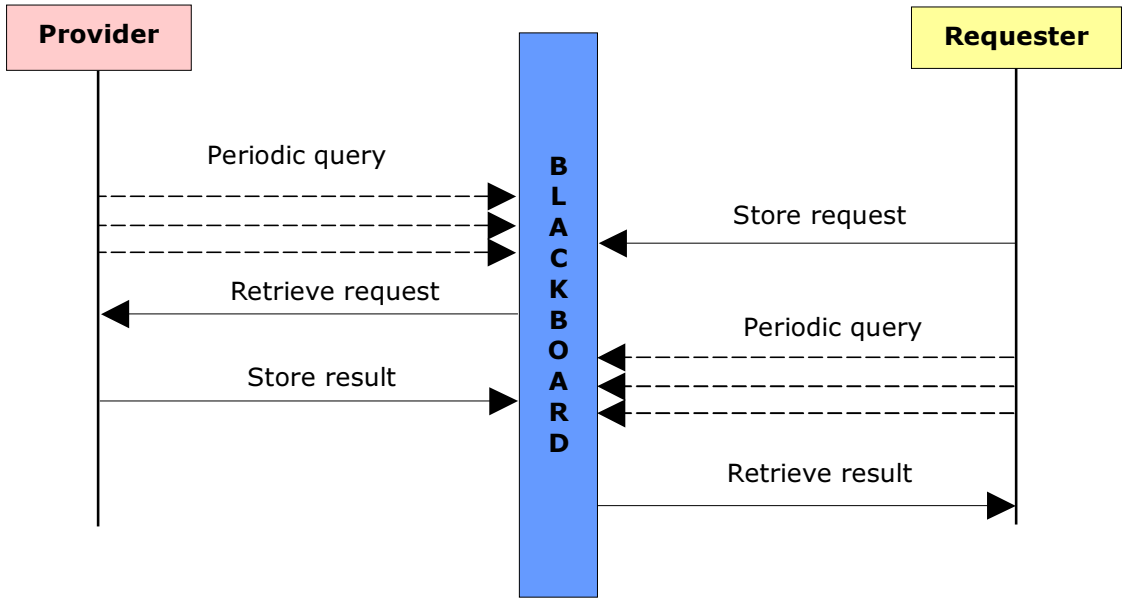
Social Knowledge (cont.)

- Different Kinds of Social Knowledge:
  - white-page list – inevitable for any MAS, stored in AMS
  - yellow-page list – usually provided by a DF, or broker
  - knowledge about other agents' status, load, trust, …
  - knowledge about other agents' beliefs, intentions, commitments
  - meta-knowledge – knowledge how to infer social knowledge

- Examples of agent's social ability to:
  - delegate responsibility,
  - decompose a task into subtask,
  - contract optimal collaborators,
  - form team and coalitions,
  - findout a missing information,
  - detect intruders …

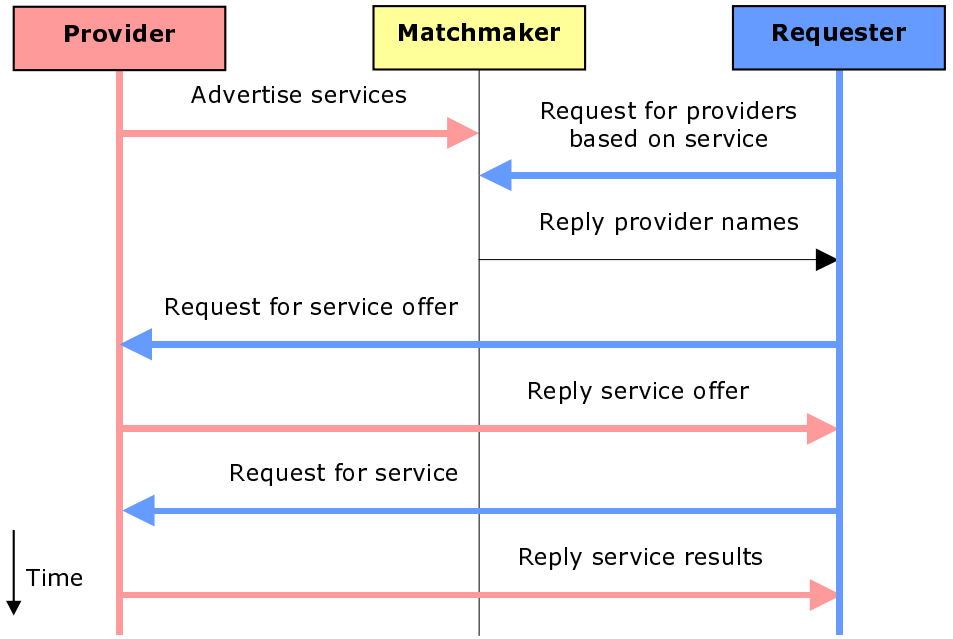
Social Knowledge Providers

- MA platform components:
  - facilitators (central communication component) [McGuire],
  - yellow-pages agents (FIPA-directory facilitator),
  - ANS (agent name server)
  - blackboard architecture (agents avoid needing SK)



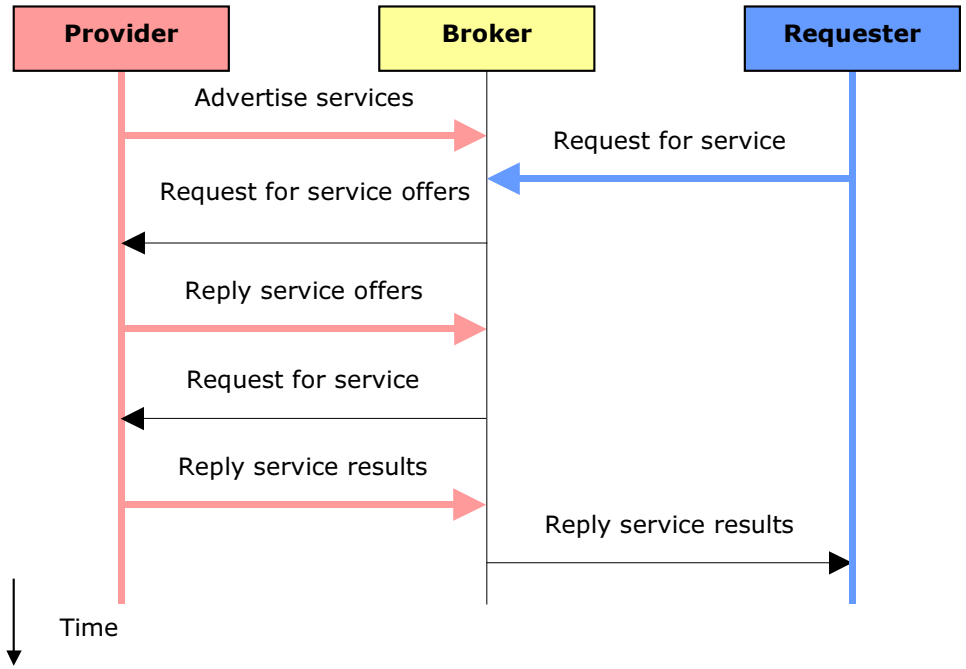
Social Knowledge Providers

- middle agents:
  - matchmakers (suggest a service provider, may be reused later)
  - brokers (act on behalf of a service requestor)
  - mediators (based on matchmaker and broker, decomposes and learns)



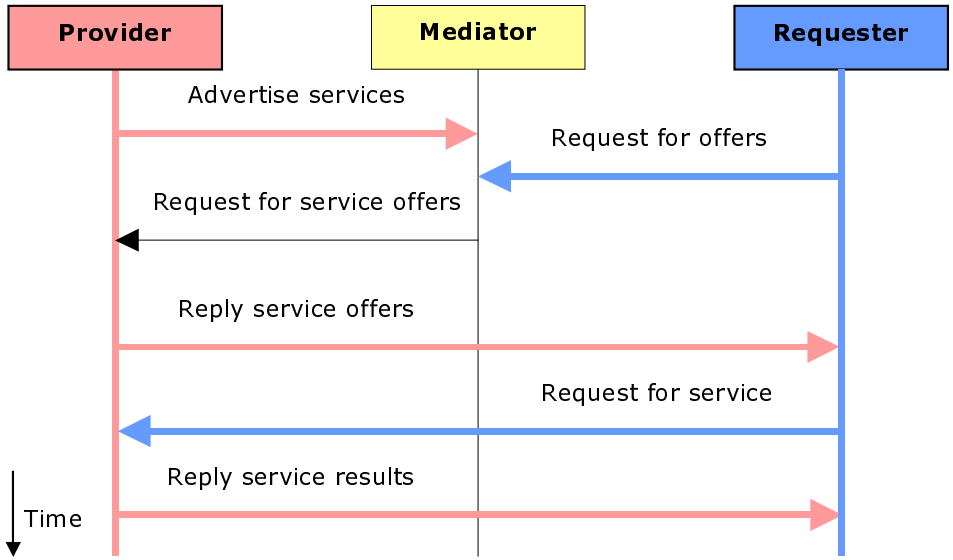
Social Knowledge Providers

- middle agents:
  - matchmakers (suggest a service provider, may reused later)
  - brokers (act on behalf of a service requestor)
  - mediators (based on matchmaker and broker, decomposes and learns)



Social Knowledge Providers

- middle agents:
  - matchmakers (suggest a service provider, may reused later)
  - brokers (act on behalf of a service requestor)
  - mediators (based on matchmaker and broker, decomposes and learns)



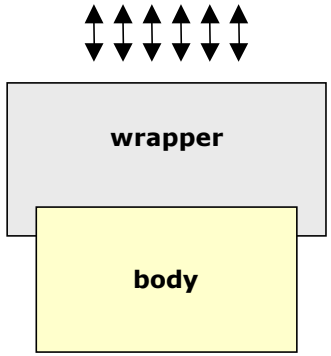
Social Knowledge Providers

- platform agents:
  - facilitators (central communication agents),
  - yellow-pages agents (FIPA-directory facilitator),
  - ANS (agent name server)
  - blackboard architecture (agents avoid needing SK)
- middle agents:
  - matchmakers (suggest a service provider, may reused later)
  - brokers (acts on behalf of a service requestor)
  - mediators (based on matchmaker and broker, decomposes and learns)

ordinary agents loose certain amount of independence as they depend on a SK provider – distributing SK within the ordinary agents will increase agents' autonomy and the MA system's robustness

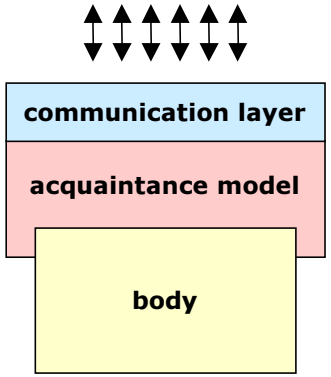
Architecture of an Agent

- usualy an agent consists of
  - wrapper and
  - body



- as we talk about integration: the body will be regarded to have no awarness about the community and the wrapper will be responsible for planning and carrying out social interaction in the broader sence (which is not the case of an ordinary agents)

- the wrapper thus consists of
  - communication layer
  - acquaintance model



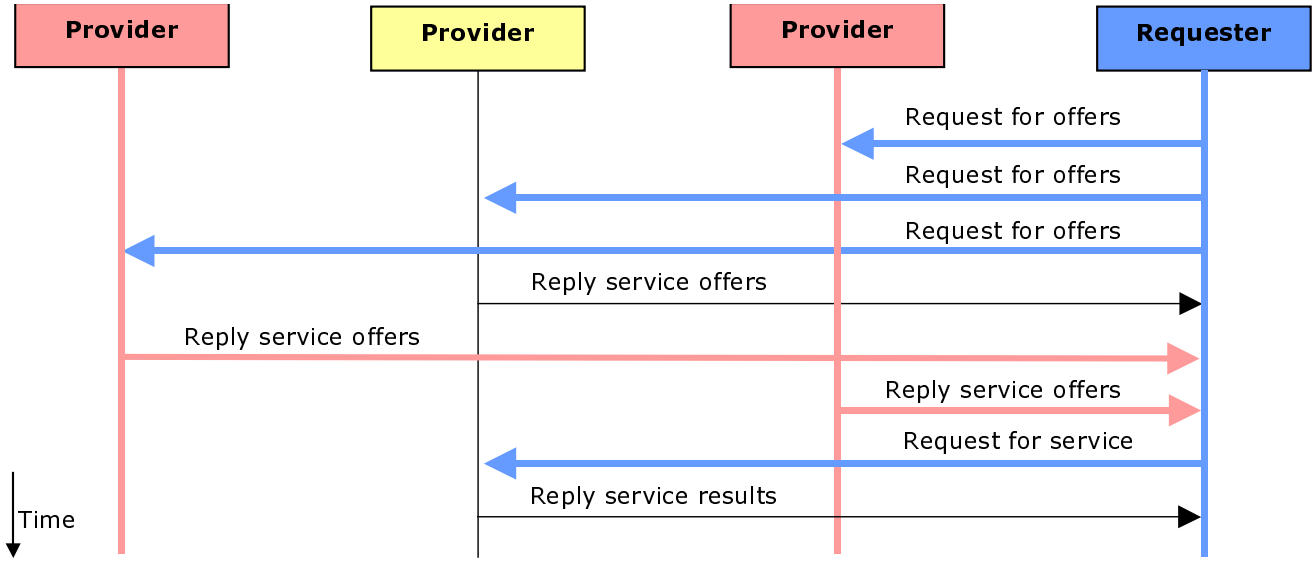
Brief Notation

- Θ – all agents within the community,
- Ψ – all services (tasks) Θ provides,
- $\alpha(A) \subseteq \Theta$ – agent's total neighborhood, agents an agent A is aware of,
- $\beta(A) \subseteq \Psi$ be the set of services the agent A provides,
- $\gamma(T)$ – all possible plans for providing a service $T \in \Psi$ *)
- $\omega(A,T) \subseteq \gamma(T)$ – plans for the service T an agent A is aware of
- $\varepsilon^t(A) \subseteq \alpha(A)$ – agent's current cooperation neighborhood
- $\tau^t(A) \subseteq \beta(A)$ – set of the services being currently solved by the agent A

*) plans are conditional, with conditions expressed in LORA (Logic of Rational Agency) [Wooldridge]

SK in Contractor Allocation

- three models of communication when seeking a contractor –
  - central communication unit – SK stored and managed centrally
  - autonomous broadcasting (contract-net-protocol) – no SK
  - acquaintance-model-based contraction – SK distributed among the agents



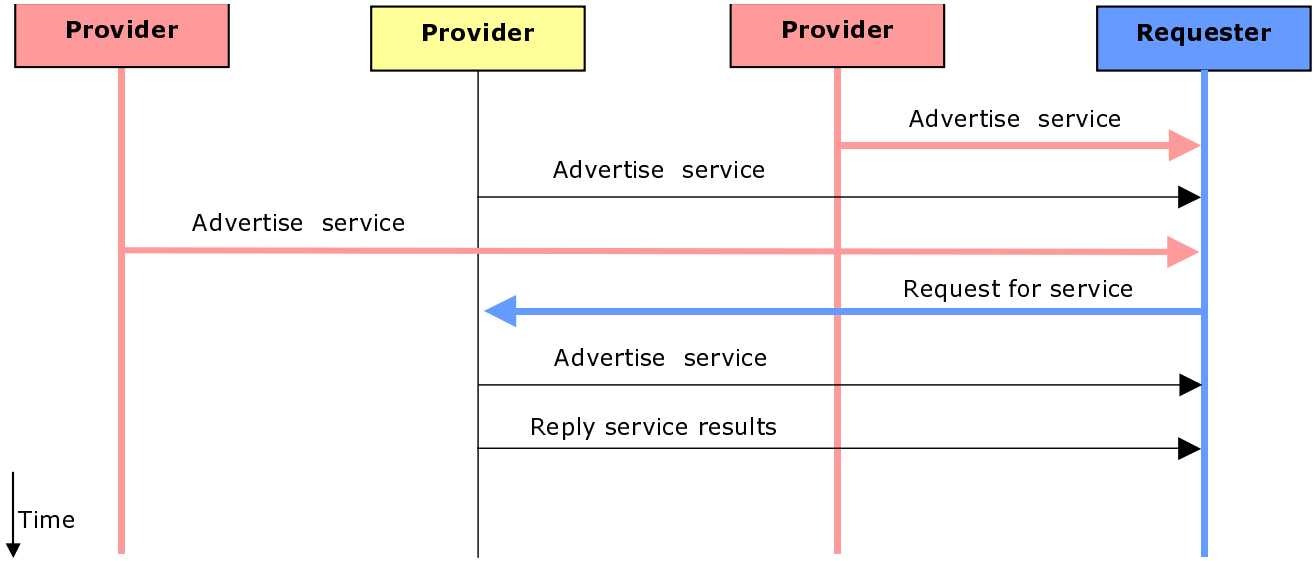
SK in Contractor Allocation

- three models of communication when seeking a contractor –
  - central communication unit – SK stored and managed centrally
  - autonomous broadcasting (contract-net-protocol) – no SK
  - acquaintance-model-based contraction – SK distributed among the agents

- an acquaintance model stores agents' computational models of their mutual awareness by which social reasoning (reasoning about others) is facilitated.

SK in Contractor Allocation

- three models of communication when seeking a contractor –
  - central communication unit – SK stored and managed centrally
  - autonomous broadcasting (contract-net-protocol) – no SK
  - acquaintance-model-based contraction – SK distributed among the agents



Acquaintance Models

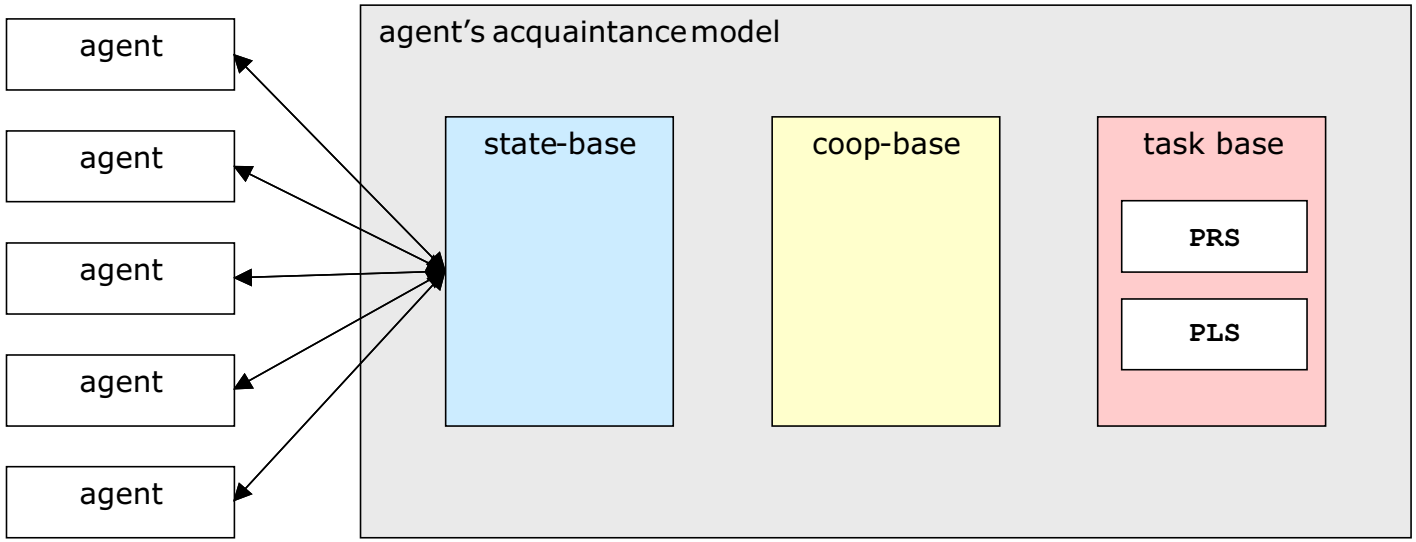
- ARCHON (*Architecture for Cooperating Heterogeneous On-line Systems*) [Wittig, Wooldridge] knowledge layers:
  - planning and co-ordination knowledge,
  - knowledge about agent internal state, and
  - knowledge about collaborating agents

  } – Archon Layer
  - problem solving knowledge – Intelligent System Layer

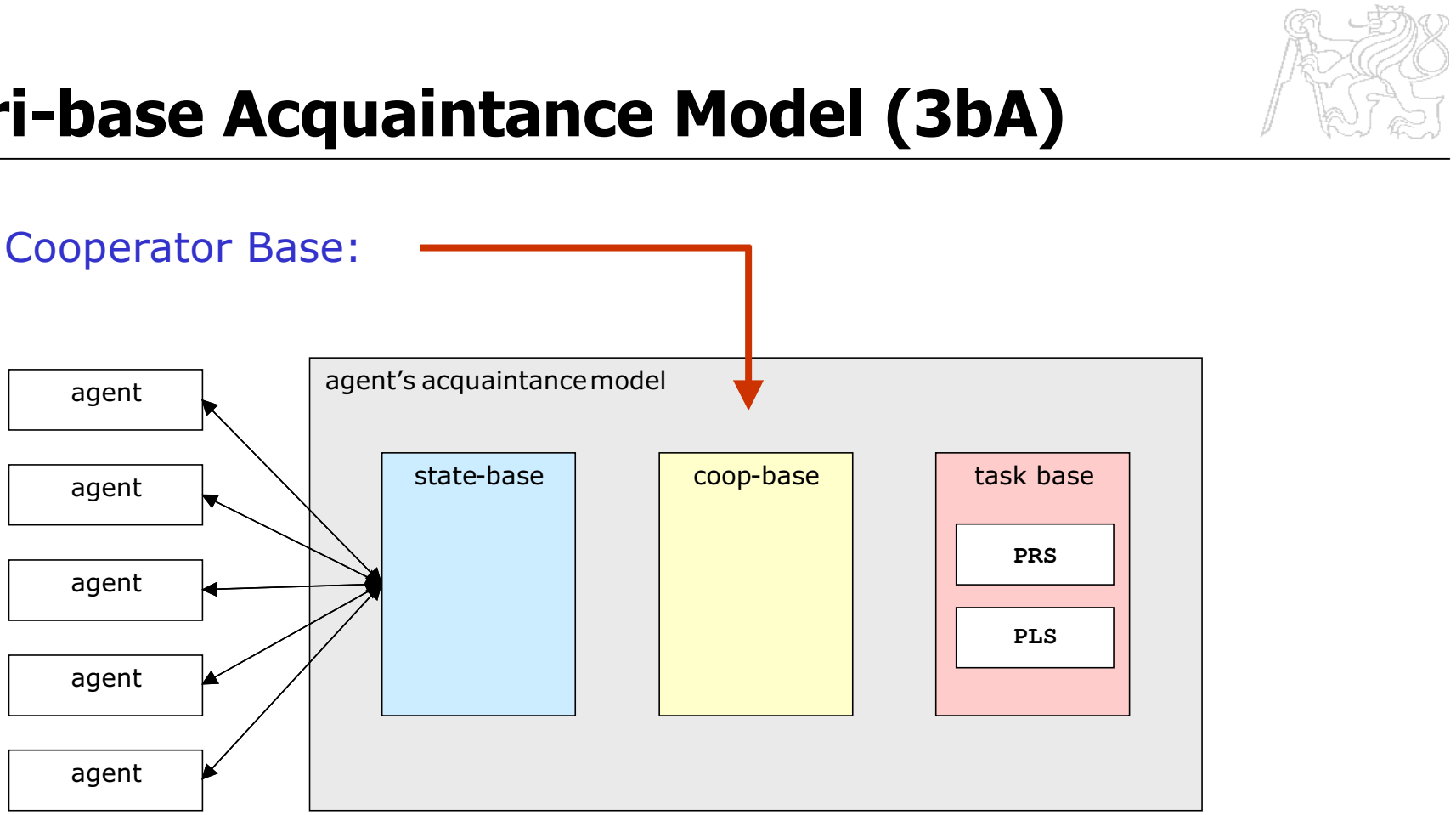
- ViSe (Virtual Secretary) MAS [Cao] - twin-base model
  - cooperator base (permanent data about the collaboration)
  - task base (ordered up-to-date task decomposition rules)
  - cooperation trader maintains (in idle times) agents task bases

- 3bA model general, features, and intro

Tri-base Acquaintance Model (3bA)



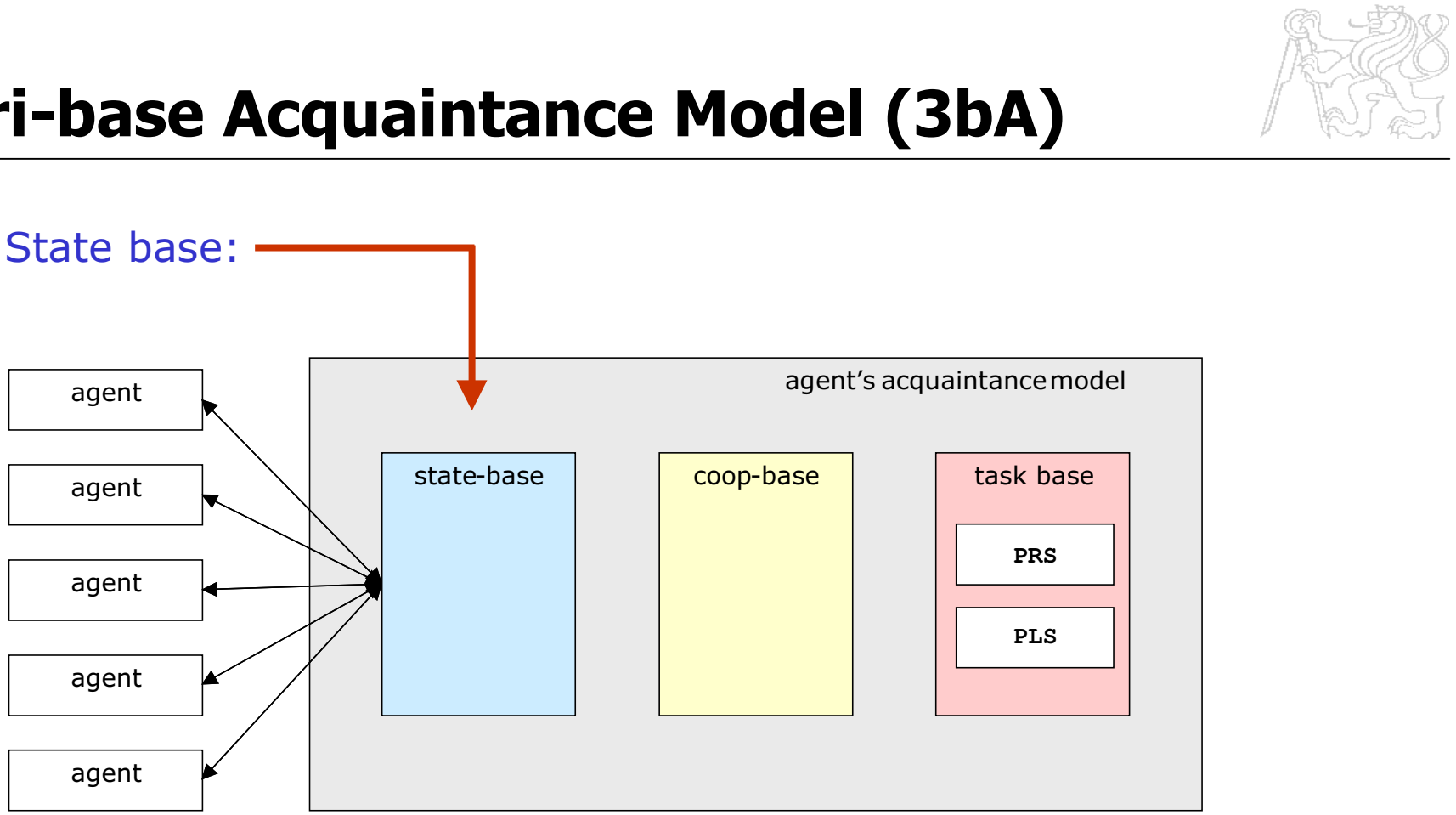
Tri-base Acquaintance Model (3bA)



- Cooperator Base:

$$CB(A) \stackrel{\text{def}}{=} \{\langle B, \text{Addr}(B), \text{Lang}(B), \beta(B)\rangle\}_{B \in \alpha(A)}$$

Tri-base Acquaintance Model (3bA)

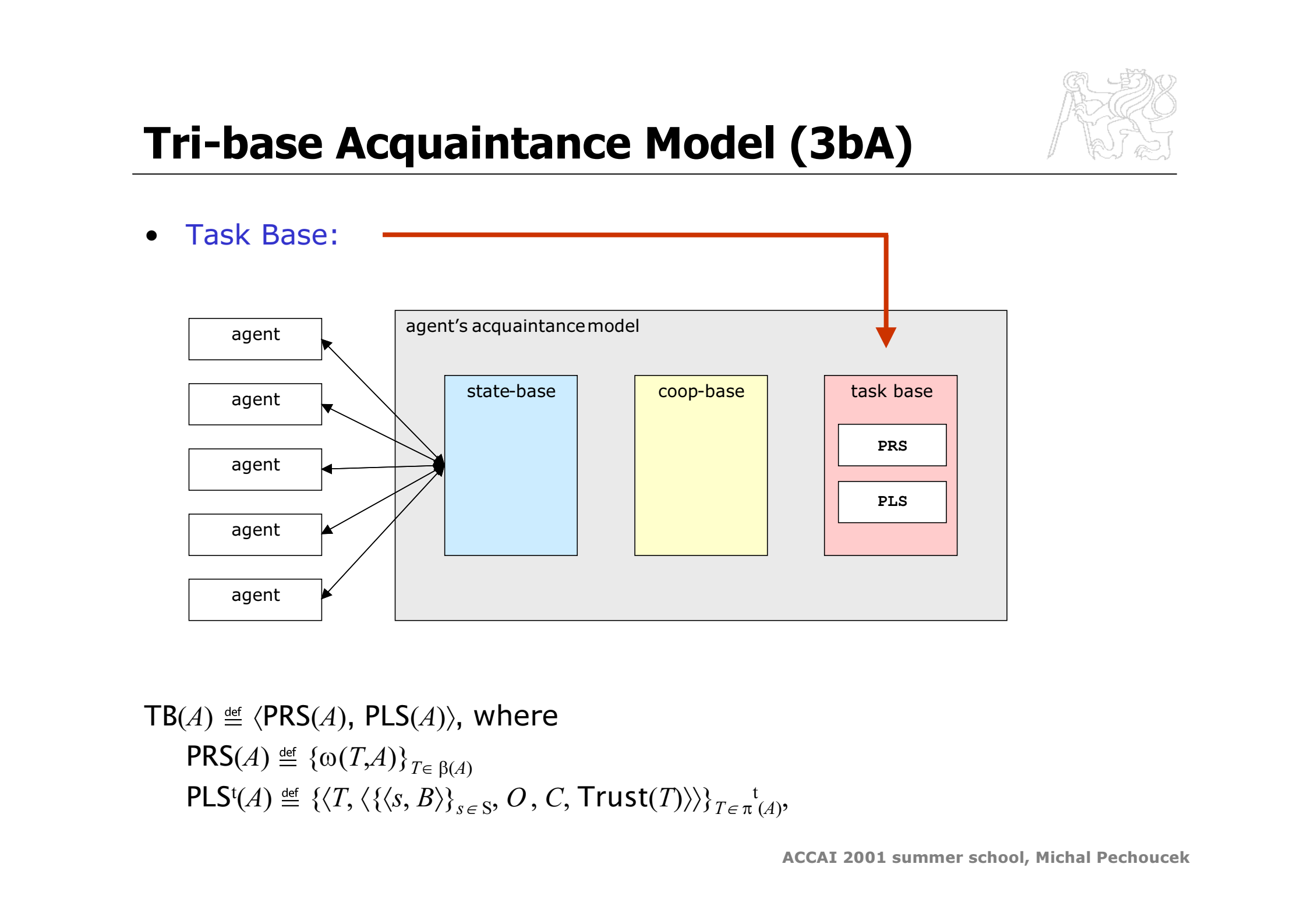


SB(A) $\stackrel{\text{def}}{=}$ ⟨AS(A), TS(A)⟩, where

AS(A) $\stackrel{\text{def}}{=}$ $\{\langle B, \text{Cap}(B), \text{Load}(B), \text{Trust}(B), \text{Knowledge}(B)\rangle\}_{B \in \varepsilon^{t}(A)}$

TS(A) $\stackrel{\text{def}}{=}$ $\{\langle T, \text{Dec}(T), \text{State}(T), \text{Trust}(T)\rangle\}_{T \in \tau^{t}(A)}$,

Tri-base Acquaintance Model (3bA)



- Task Base:

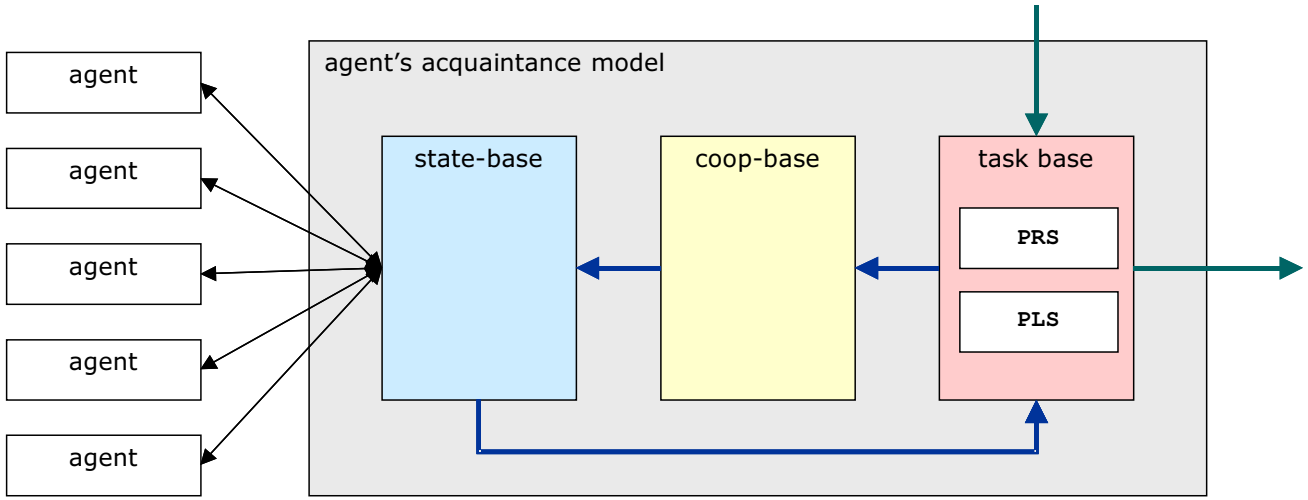
TB(A) $\stackrel{\text{def}}{=}$ ⟨PRS(A), PLS(A)⟩, where

PRS(A) $\stackrel{\text{def}}{=}$ $\{\omega(T,A)\}_{T \in \beta(A)}$

PLS^t(A) $\stackrel{\text{def}}{=}$ $\{\langle T, \langle\{\langle s, B\rangle\}_{s \in S}, O, C, \text{Trust}(T)\rangle\rangle\}_{T \in \pi^t(A)}$,

Generation of Plans

- decomposition on request:
  - exploitation of the pre-prepared plan
  - new plan generation (based on SB knowledge)
  - new plan generation (broadcasting)



- replanning driven by state-base update

Knowledge Maintenance

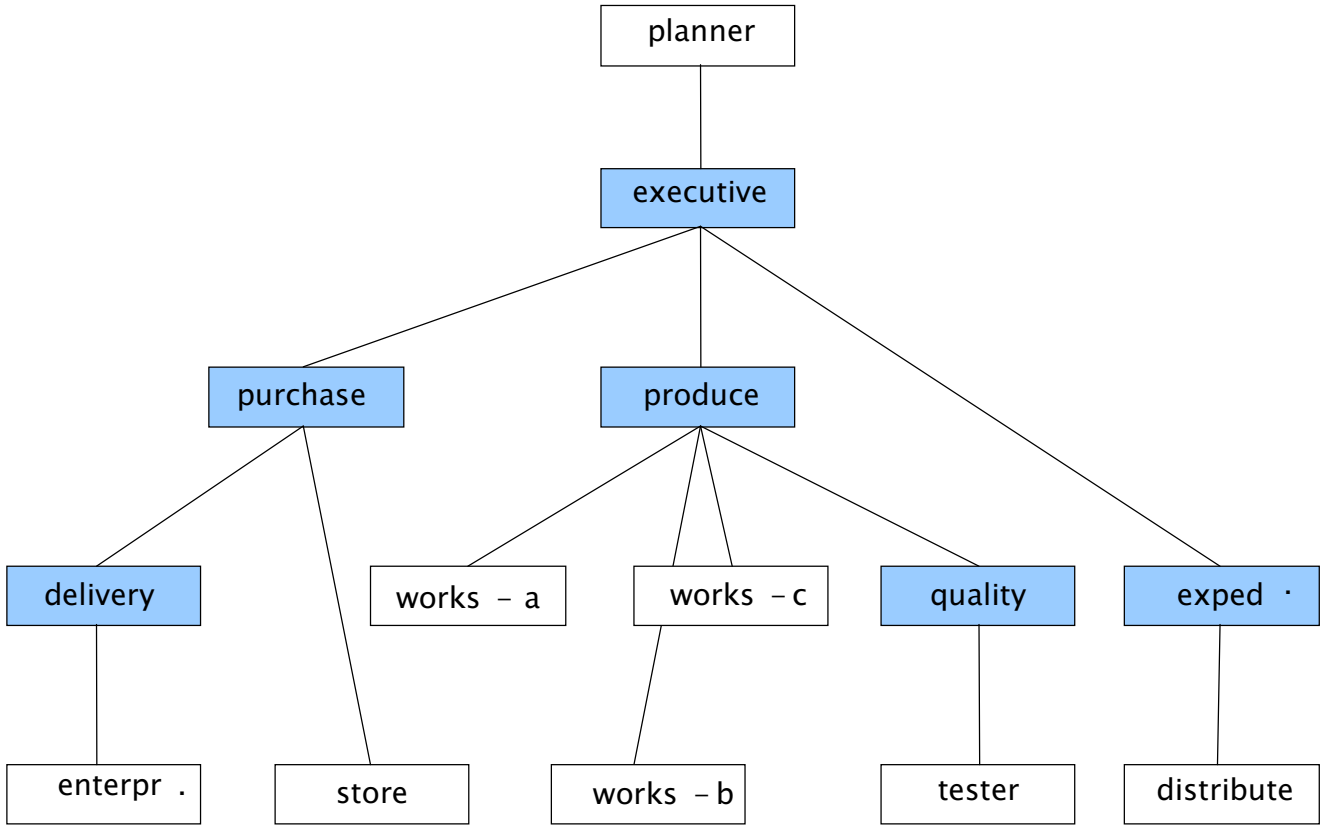
- cooperator-base is regarded as permanent
- task-base PrS is permanent and PlS is maintained by planning and replanning
  - agent's current problem solving neighborhood $\pi^t(A) \subseteq \beta(A)$
- state-base is maintained either by:
  - periodical revisions – tried out, good for frequent changes and infrequent meta-reasoning (driven either by the agent or by a cooperation trader)
  - subscription based maintenance – suitable in communication intensive applications (am information push)
  - blackboard based maintenance – centralistic approach (fragile)
  - non-cooperative knowledge maintenance – intrusion detection, visualization, etc. (in ProPlanT)
- agent's monitoring neighborhood $\varepsilon^t(A) \subseteq \mu(A) \subseteq \alpha(A) \subseteq \Theta$

Impact of SK on MAS Behavior

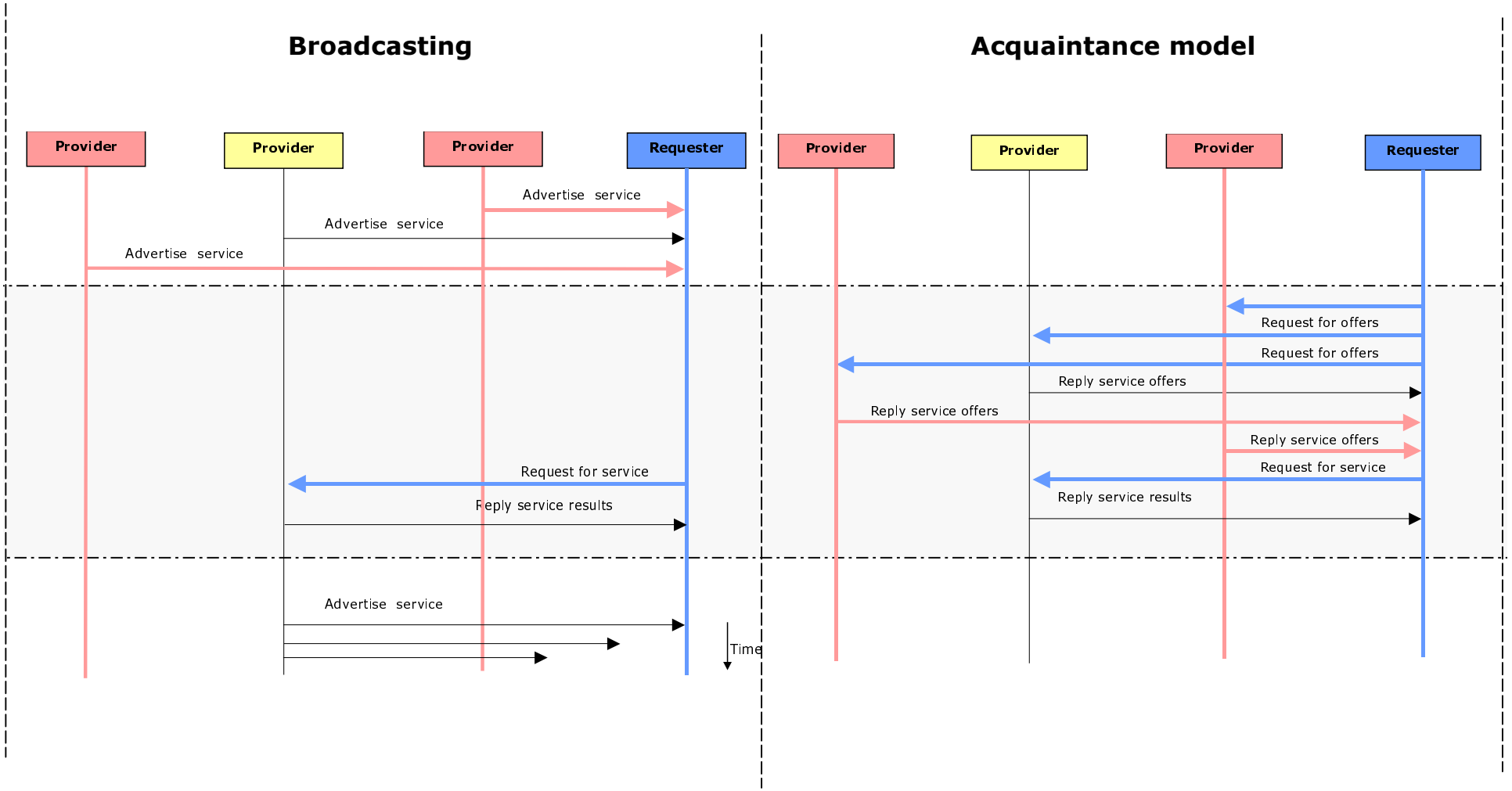
- communication traffic efficiency improvement (ProPlanT)
- quality of the distributed decision making
  - deadline/strategy specification (ExPlanTech)
- distributed SK assures robustness of MAS operation
  - failure impact: best-case-failure-impact(m,Θ)=max(0,1+m-$|\Theta|$)

    worst-case-failure-impact(m,Θ)= $\frac{m}{|\Theta|}$
  - inaccessibility of the agent

Experiments and Testing of the 3bA Model



Tested Communication Protcols



Agents with different Knowledge

- degree of freedom – total number of agents free to decide about their collaborators ($\varphi(\Theta)$)

$\varphi(\Theta) = |\{ A: A \in \Theta \wedge \texttt{free}(A)\}|$, where $\texttt{free}(A)$ iff $\exists t \in ß(A)\ (|?\ (A,t)| > 1)$, or

$\varphi(\Theta) = \sum_{t \in \Psi} \varphi_t(\Theta)$, where $\varphi_t(\Theta) = |\{A: A \in \Theta \wedge |\omega(A, t)| > 1\}|$

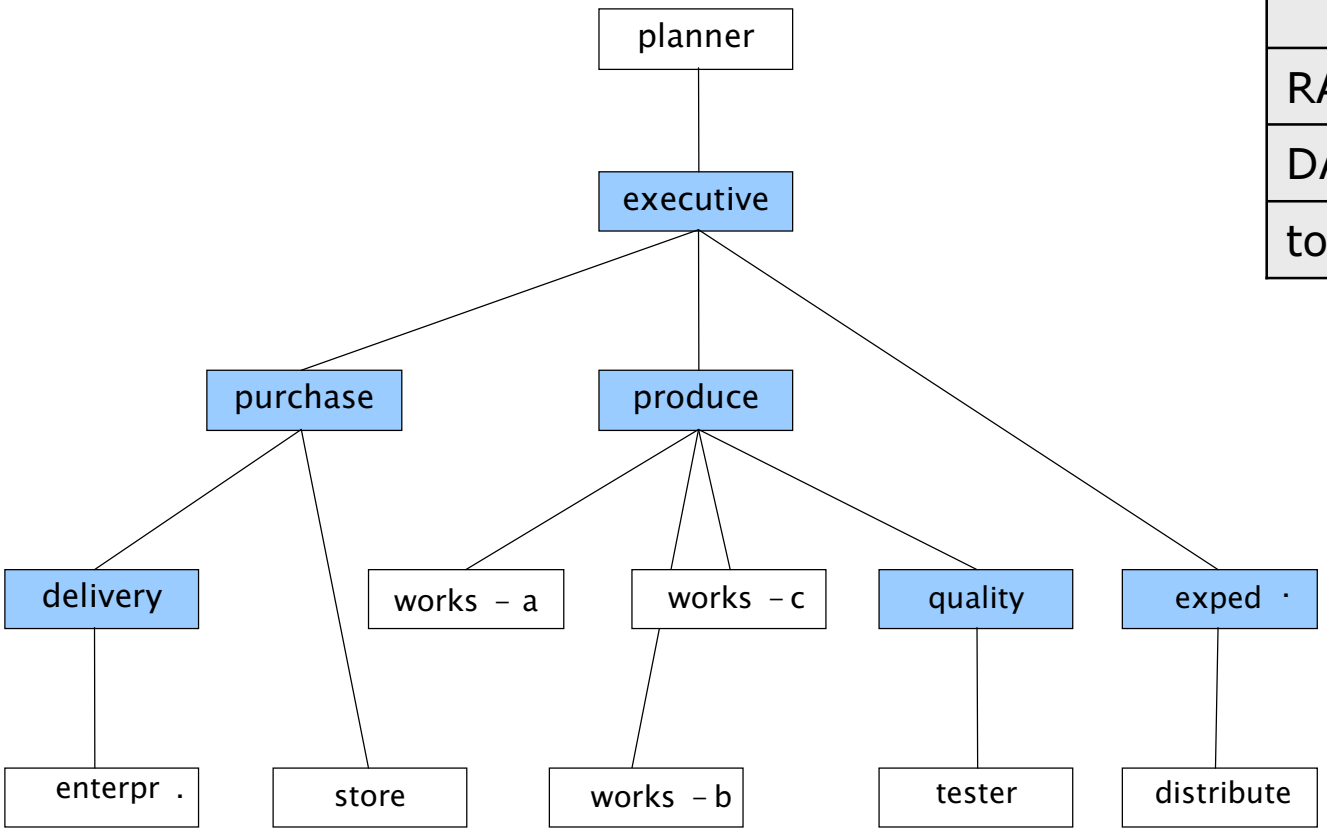
$\varphi(\Theta)$	average number of messages		
	m(tri-base)	m(broadcast)	m(maint)
0	30	30	123
2	30	46	123
7	30	180	123

Communities with Different Architecture

- different architectures (in number of agents) but identical functionality – collection of requests the system is able to process

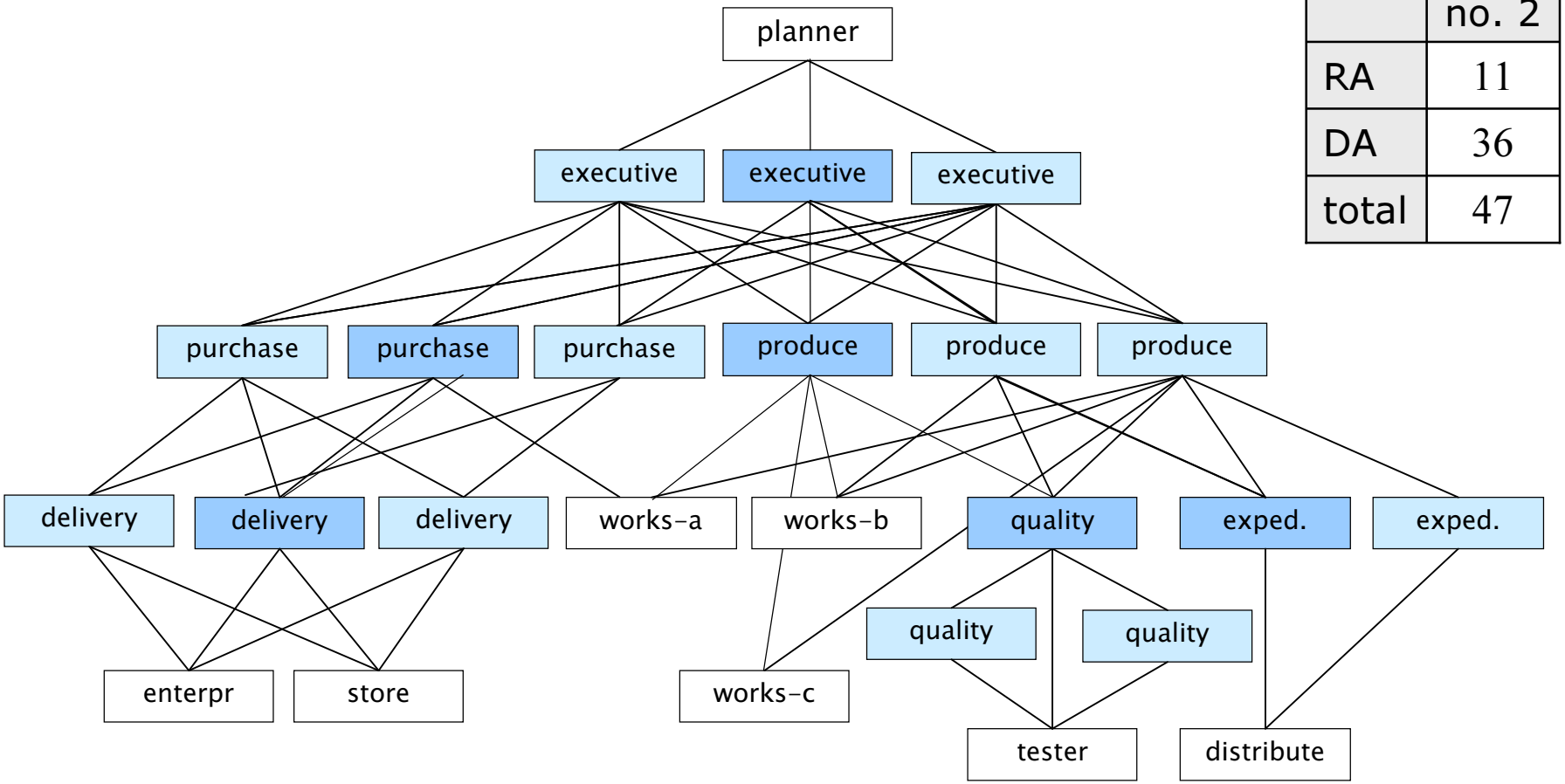
	community no. 1	community no. 2	community no. 3
number of RA	11	11	78
number of DA	6	36	36
total no. of agents	17	47	115

Community no. 1



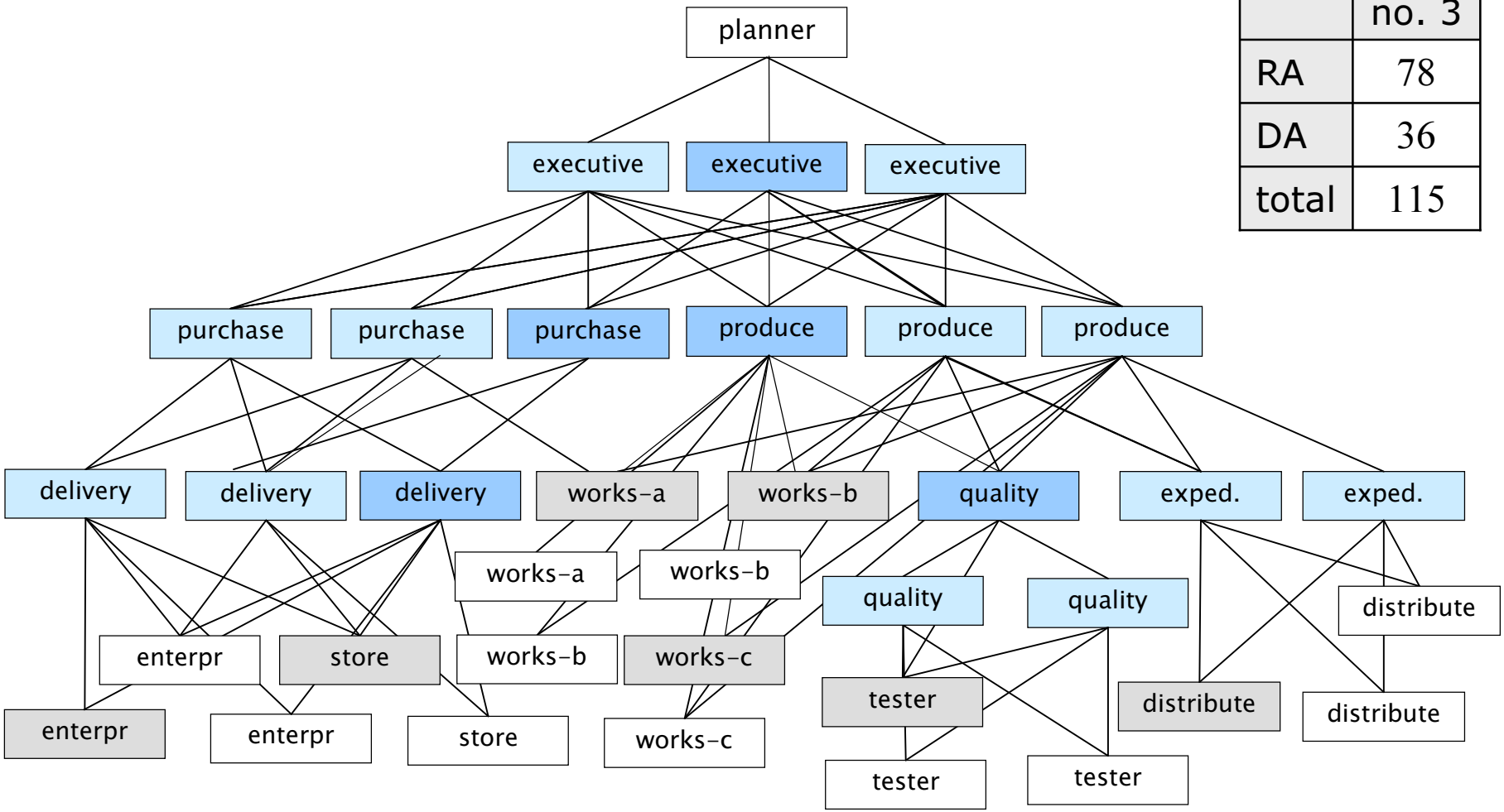
	no. 1
RA	11
DA	6
total	17

Community no. 2.



	no. 2
RA	11
DA	36
total	47

Community no. 3



	no. 3
RA	78
DA	36
total	115

Communities with Different Architecture

	community no. 1	community no. 2	community no. 3
	acquaintance based decomposition		
register	173	1871	4517
contract	30	30	30
maintenance	123	206	31
total per request	153	236	61
	broadcasting based decomposition		
register	27	58	196
evaluate	150	285	788
achieve	30	30	30
total per request	183	315	818

Remarks on Experiments

- communication traffic savings do not depend only on the degree of freedom and number of agents but also on the topology of the entire topology of mutual acquaintances
- request frequency requirements (in order to guarantee savings):

  $1/f \geq t_c + t_i$,
- parallel requests?
- conflicts resolution
  - failure
  - sub-optimality
  - re-planning
- register phase communication, advertise message content

Knowledge Improvement

- it is always difficult to set the permanent knowledge and parameters of the model – they have been given to the agents explicitly – agents only maintain
- neighborhood:
  - increasing $\mu(A)$, $\pi^t(A) \rightarrow$ fast response and complex maintenance
  - decreasing $\mu(A)$, $\pi^t(A) \rightarrow$ slow response and easy maintenance
- permanent knowledge:
  - task-base – inappropriate knowledge decomposition
  - cooperator-base – revision of the agent's capabilities

when SK is maintained we, based on communicated information or observations, maintain truthfulness of the hypothesis about agents

when SK is improved we infer, using meta-reasoning mechanism, new hypothesis about agents (that may affect the permanent knowledge and nieghbourhood parameters of the model)

Knowledge Improvement Cont'

- monitoring neighborhood optimization
  - subscribe-link may be unsubscribed if not exploited by (i) subscribee or (ii) subscriber for a threshold period of time t_{old}
  - monitoring neighborhood can grow or restrict in time
- relevance of update – specification of a quantative threshold t_{rel}

$$\frac{\left| \text{Load}_1(A) - \text{Load}_2(A) \right|}{\text{Load}_1(A)} \geq t_{rel}$$

- problem-solving neighborhood – depends on $\mu(A)$, retraction of unexploited plans

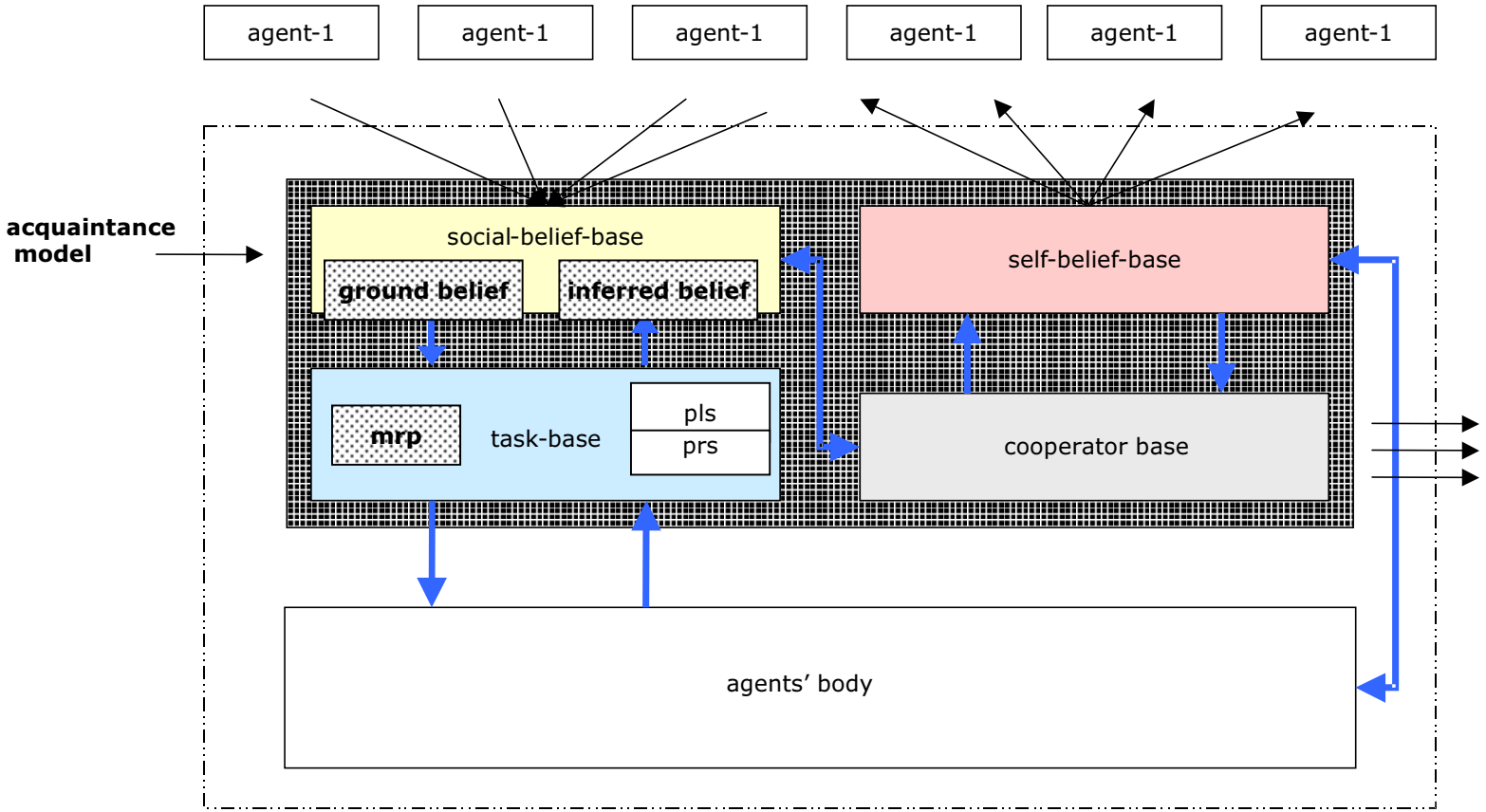
in practical applications we recommend to start with empty $\mu(A)$ and $\pi(A)$ and to learn with experience (we avoid communication overload)

Meta-reasoning

- the nature of the acquaintance model changes - we have to introduce the concept of meta-reasoning:
  - we reason about social knowledge not only in order to exhibit socially-aware behavior, but also for inferring other pieces of social knowledge
  - meta-knowledge is a rule of inference how to deduce new piece of social knowledge from the available social knowledge

- we have to update our acquaintance model by:
  - distinguishing among ground and inferred social beliefs
  - formalizing MRP – meta-reasoning program knowledge structure
  - distinguishing among self-knowledge and social knowledge

Meta-Reasoning Acquaintance Model



Meta-Reasoning Program Example


```
(defrule
        (int ?agent (bid comodity ?y)),
        (bel the-agent (has-got ?agent ?z)),
        (> ?y ?z)
        =>
        (bel the-agent ?agent fakes)
)

(defrule
        (bel the-agent ?agent fakes)
        =>
        (retract monitor ?agent)
)
```

Meta-Reasoning Program Implementation

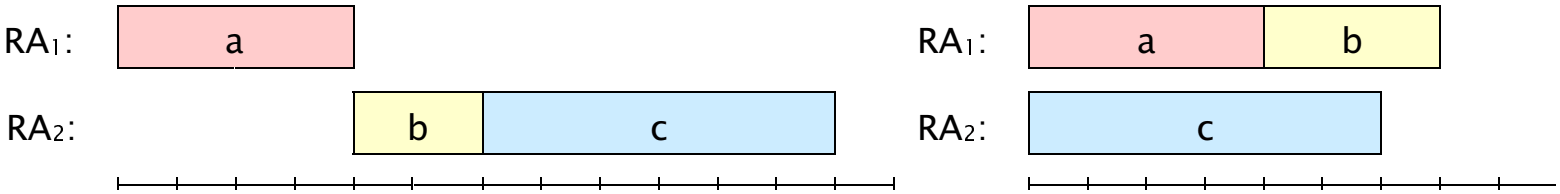
- logic programming – Prolog (problem with maintenance of all deducible facts)
- rule-based systems – Clips, Jess, OPS/5
- integrated in agent's body – case-based reasoning, machine learning, simulation, …

```
(defrule
        (bel the-agent (nasdaq today ?t)),
        (bel the-agent (nasdaq history ?h)),
        [call-body prediction-algorithm(cons ?t ?h) → ?y],
        (bel the-agent (nasdaq tomorrow ?y)),
        (> ?y ?t))
        =>
        (int the-agent (buy nasdaq))
)
```

Meta-agents

- the ordinary agents – object agents – cannot often comprehend the entire whole of the community (SCM, e-commerce, lumber industry, spying, sniffing) - object level × meta level knowledge improvement

$(x = \langle a < b \rangle, y = \langle c \rangle)$, $RA_1(a)=4$, $RA_1(b)=3$, $RA_2(b)=2$, $RA_2(b)=6$



Meta-agents

- the ordinary agents – object agents – cannot often comprehend the entire whole of the community (SCM, e-commerce, lumber industry, spying, sniffing) - object level × meta level knowledge improvement
- central (meta-) agent acts as a communication/ collaboration center and may directly control the actions and knowledge: facilitators, brokers, matchmakers, mediators, or middle agents.
- meta-agent a loosely coupled agent whose prime design objective is to reason independently about other agents:
  - active meta-agent directly affects some or all of the object agents within the community. By directly delivered messages, the meta-agent may revise the acquaintance models of the object agents.
  - passive meta-agent does not influence the community lifecycle. It just simply observes and provides the user with suitable information about how the community is evolving over time. It is up to the user to perform such a change as a feedback.

Reflection in Multi-Agent Systems

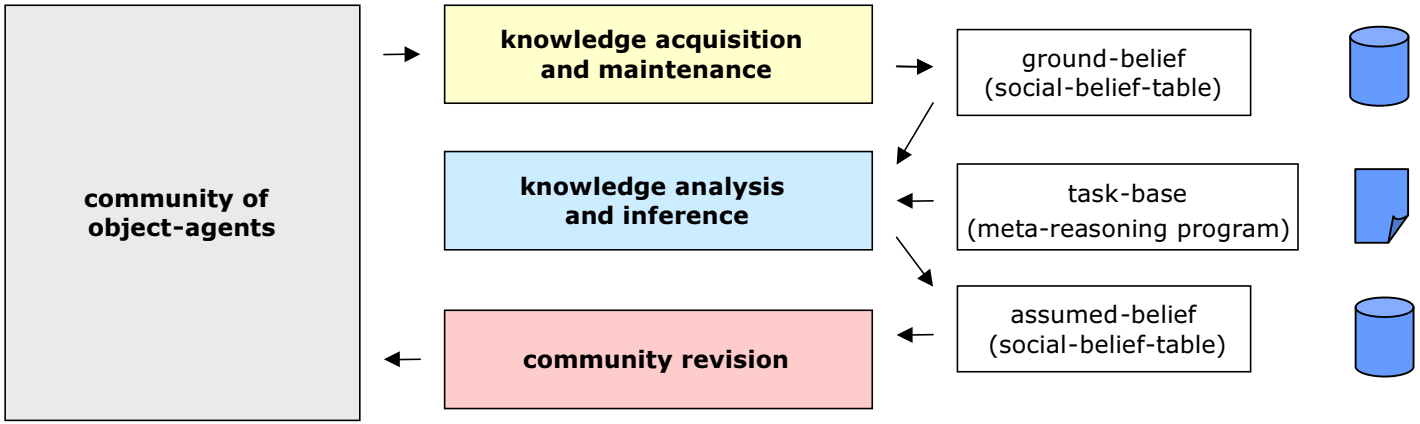
- primary reasoning (object-level reasoning) of a computational system is a performed computation (manipulation with symbols) that implements behavior meeting systems designed objective (planning, configuration, expertise, …)
- reflective reasoning allows the system to analyze its own primary reasoning (by analyzing past, reality checking, adaptation)

- a reflective computational system contains an object component and a reflective component

- Systems' reflective reasoning (reflective component itself) requires [Maes]:
  - self-representation,
  - meta-reasoning mechanism and
  - causal connection (introspective integrity and introspective force)

Meta-Agent as a Reflective Component

- meta-agent is an agent the one who implements reflection in multi-agent system:
  - contains/maintains sk about other agents (introspective integrity)
  - is capable of meta-reasoning capability
  - may/may not reflect back sk to community (introspective force)
  - objects agents prime functionality is independent on meta-agent



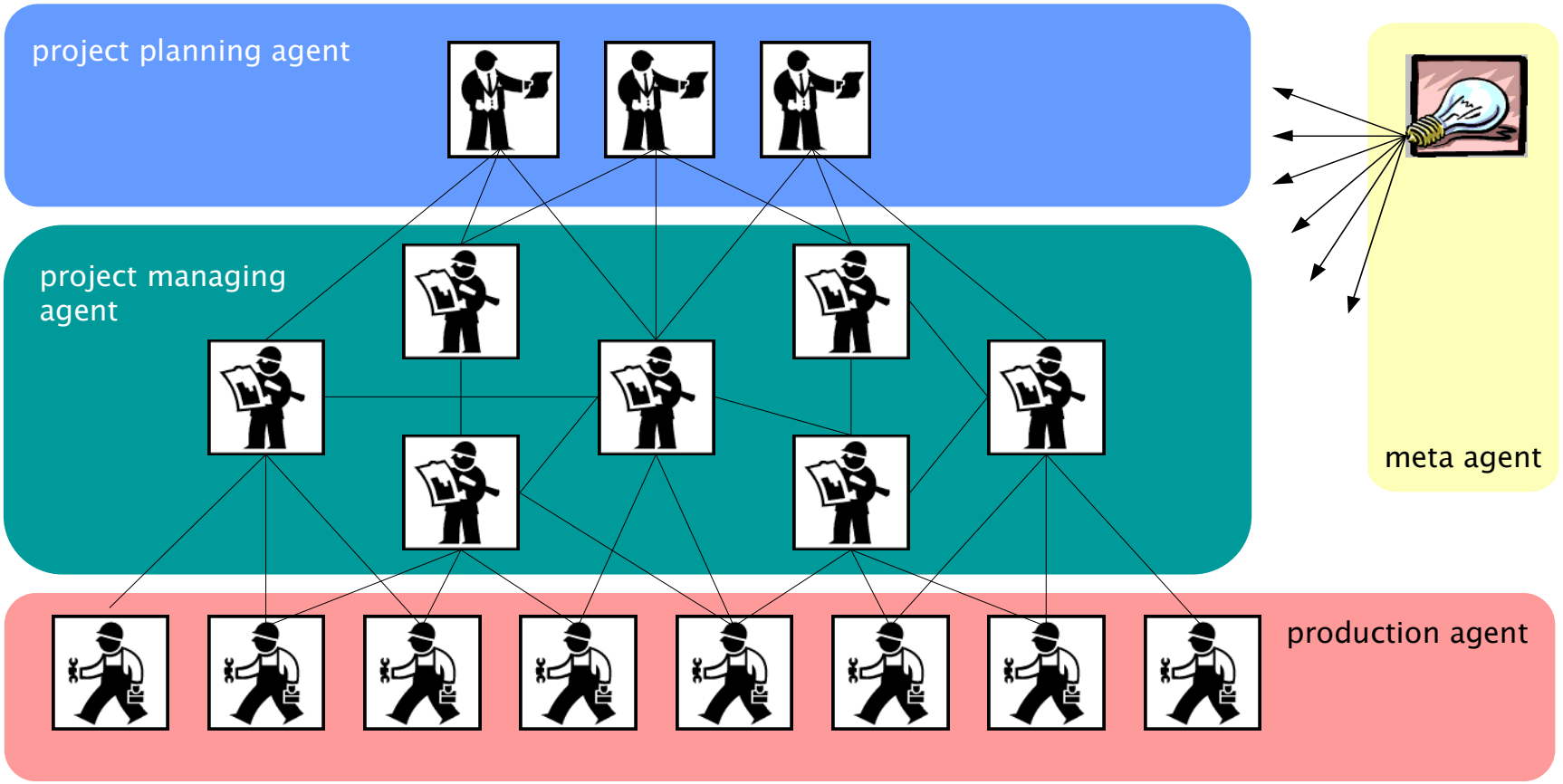
Meta-Reasoning × Reflective Reasoning

- meta-reasoning (MR) of an agent - agent's reasoning about and forming new social knowledge about other members of the multi-agent community.
- reflective reasoning (RR) of the MAS – reasoning about itself – by meta-reasoning distributed within the meta-agents,
- reflective reasoning (RR) of an agent – agent's reasoning about his own self-representation and reasoning processes
- sometimes meta-agents meta-reasons less than object-agents
- danger of meta-meta- … whatever
  - always depends on definition of the primary and reflective reasoning

What is this all good for?

- meta-agent meta-reasoning:
  - visualization, explanation (passive MA)
  - intrusion detection and fault diagnosis (active/passive MA)
  - bottleneck detection, multi-level optimization (active MA)
- object-agent meta-reasoning
  - communication (thus problem solving complexity) reduction
  - improved quality of distributed decision making and collaboration
  - improved robustness of the MA system
  - increased autonomy of the agents
- these concepts have been used in
  - production planning and supply chain management MAS
  - safe multi-agent systems intrusion detection, diagnostics
  - collaborative coalition formation
- future plans – coalition re-planning and reconfiguration, robocup rescue, teams structure reconstruction, intelligent task decomposition, …

ProPlanT Architecture



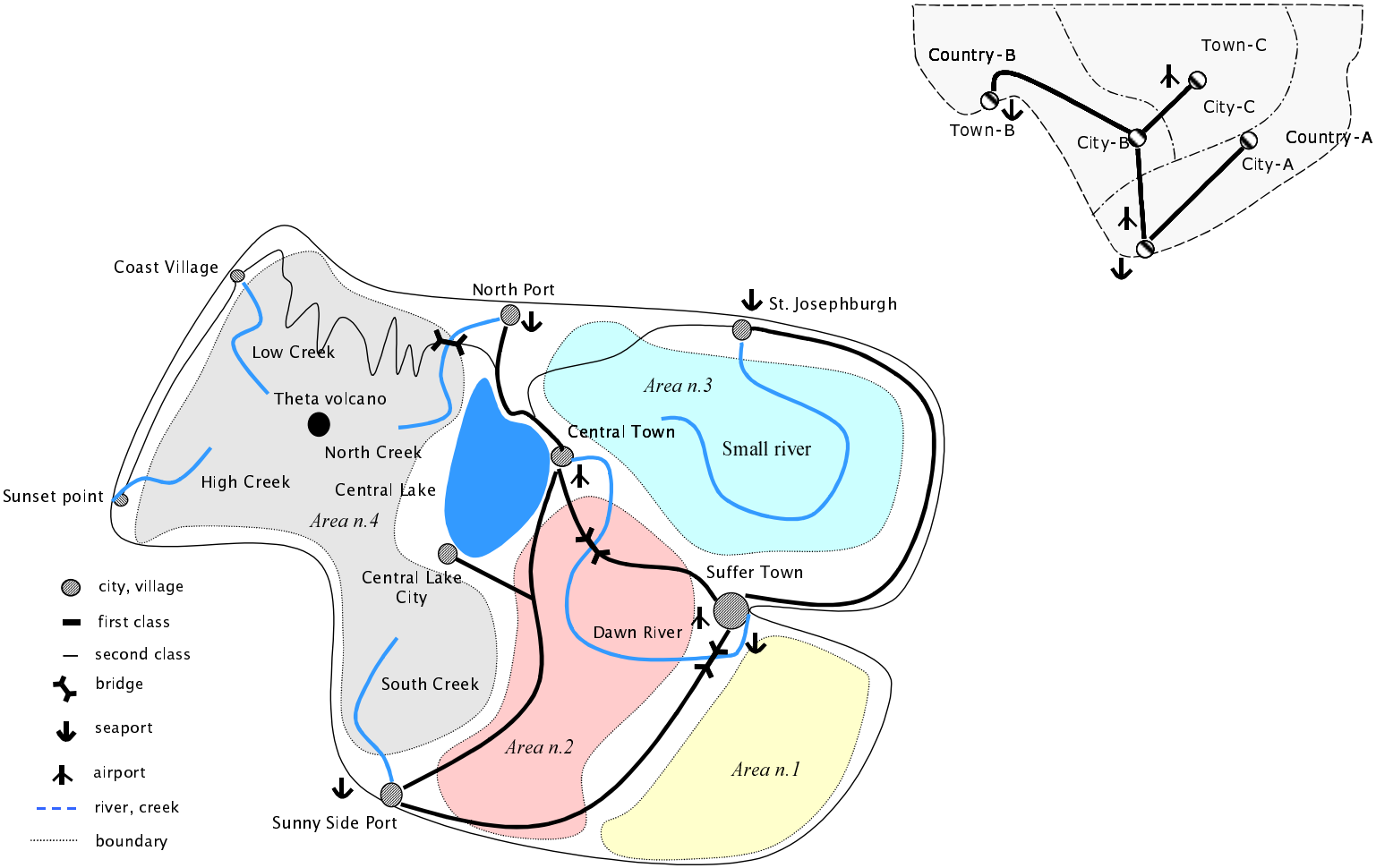
ExPlanTech

- an IST project – Exploitation of the ProPlanT technology in operating manufacturing environment:
  - Liaz Pattern Shop, Czech Republic
  - Hatzapulous Packaging, Greece
- implemented in JADE, fully FiPA compliant, build in as a planning component in company ERP systems

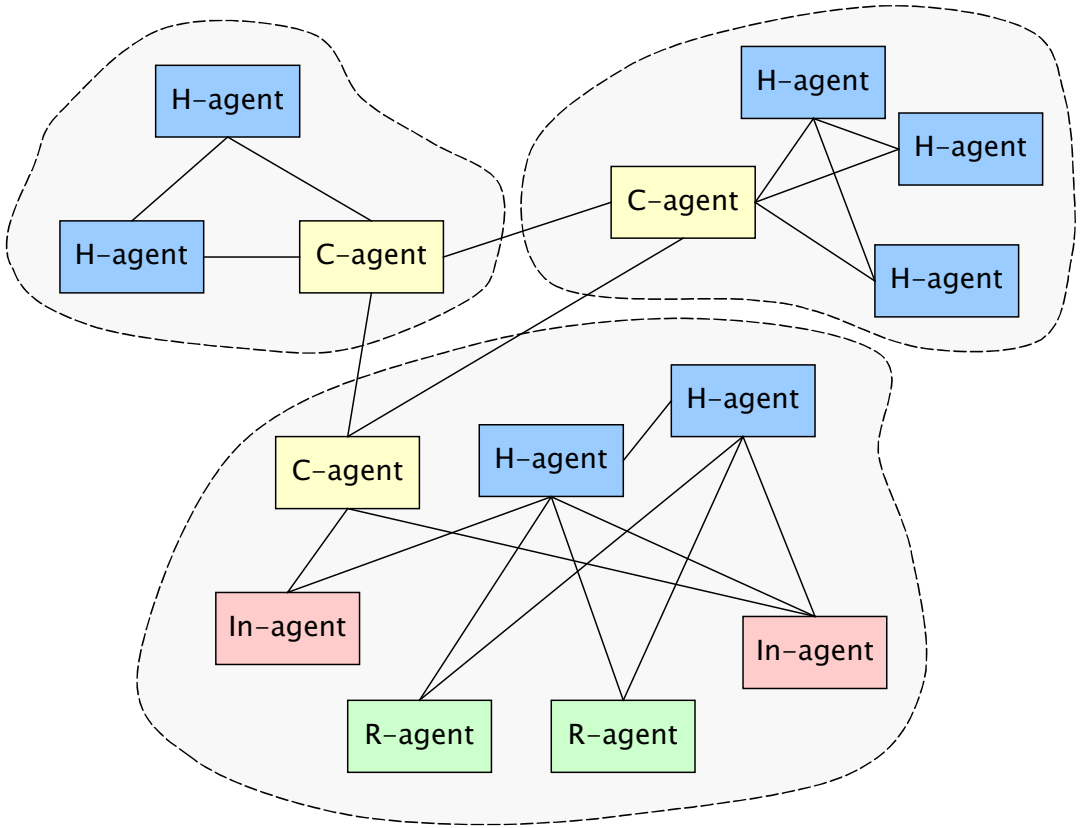
CPlanT – Coalition Planning

- classical war operations – technology control is strictly hierarchical, the command flow is managed by a hierarchy of central control elements
- operations other than war (OOTW) – cooperation of a number of different, quasi-volunteered, vaguely organized groups of people, non-governmental organizations (NGO's), institutions providing humanitarian aid but also army troops and official governmental initiatives
- two key issues related to OOTW coalition planning:
  - reluctance to provide information about their intentions, goals and resources
  - information inaccessibility caused by unreliable communication channels or unavailable agent

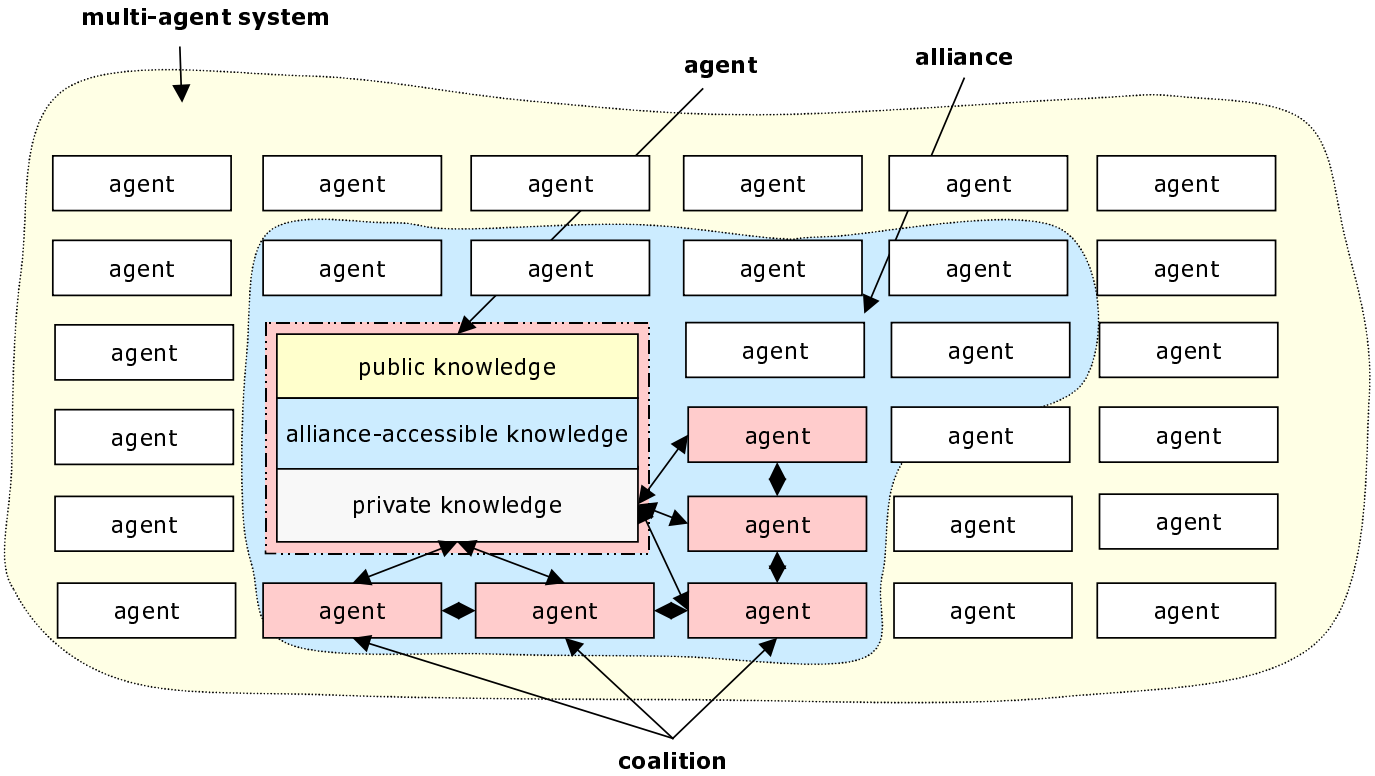
Sufferterra Scenario



CPlanT Multi-Agent System



Concept of Alliance, Agents' Knowledge



ProPlanT Multi-Agent System

Demonstration

provided there is time left