COOPERATION LAYERS IN AGENT-ENABLED BUSINESS PROCESS MANAGEMENT[1]

Ermolayev, V. A., Plaksin, S. L.

Dept. of Mathematical Modelling and Information Technologies,

Zaporozhye State University,

66, Zhukovskogo st., 69063, Zaporozhye, Ukraine, tel/fax:+380 61 264 17 24,

E-mail: {eva, psl}@zsu.zp.ua

Abstract. The paper proposes a four-layer cooperation framework for agent-enabled business process management and performance in distributed open structured organizations comprising economically rational executives. It is assumed that the executives possess various capabilities, roles, authorities, commit to follow some joint rules while performing tasks. The layers of the framework are: Cooperation Models Layer, Interoperability Layer, Communication Layer, Transport Layer. Cooperation Models Layer provides the models of an organization, an organizational unit, a business process, the patterns for executives' coordination, performance and behaviour monitoring. Interoperability Layer contains intraction protocols, the patterns for inter-agent conversations, the formalisms for parametric feedbacks and for the shared concepts for the data interchanged by the agents in the course of their cooperative performance. The lower two layers are the slots for the widely used and/or standard primitives and components. These are Communicative acts' specifications formalized in agent communication languages and transport services and envelopes respectively. The paper also provides the review of the related work structured along the proposed framework layers. This review shows no vital contradictions with the proposed cooperation framework layering.

Аннотация: В статье предложена четырехуровневая структурная схема организации управления и выполнения бизнес-процессов на базе агентов, составляющих открытые распределенные организации экономически рациональных исполнителей. Предполагается, что исполнители обладают различными специализациями, ролями, сферами ответственности, обязуются следовать некоторым общим правилам в процессе выполнения заданий. Уровнями структурной схемы являются: Уровень моделей кооперации, Уровень интероперабельности, Уровень коммуникации, Уровень транспорта. Уровень моделей кооперации предоставляет модели организации, подразделения, бизнес-процесса, шаблоны для координации работы исполнителей, мониторинга их деятельности и поведения. Уровень интероперабельности содержит протоколы взаимодействия, шаблоны диалогов между агентами, формализмы для параметрических ответов и общих концепций данных, которыми обмениваются агенты в процессе кооперативной работы. Два нижних уровня являются слотами для подключения широко применяемых и/или стандартных примитивов и компонент. Этими компонентами соответственно являются спецификации коммуникативных актов, формализованные на языках коммуникации агентов, и транспортные сервисы и конверты сообщений. В статье также приводится обзор работ в области кооперации, структурированный в соответствии с уровнями предложенной схемы. Обзор показывает, что в работах других авторов нет существенных противоречий с предложенным способом расслоения.

Keywords: Cooperation, Business Process Management, Coordination, Organization, Coalition, Performance Monitoring, Behaviour Monitoring, Interoperability, Communication, Transport, Agent.

Ключевые слова: Кооперация, Управление бизнес-процессом, Координация, Организация, Коалиция, Мониторинг деятельности, Мониторинг поведения, Интероперабельность, Коммуникация, Транспорт, Агент.

Paper status: Submitted to the 3-d Intl. Scientific - Practical Conference on Programming (UkrPROG'2002), Kiev, 22-24 May 2002

Last modified: 22.01.2002

[1] Presented research was run in frame of the Project financed by Ukrainian Ministry of Education, Grant № 0199Y1571.

COOPERATION LAYERS IN AGENT-ENABLED BUSINESS PROCESS MANAGEMENT[1]

Ermolayev, V. A., Plaksin, S. L.

Dept. of Mathematical Modelling and Information Technologies,

Zaporozhye State University,

66, Zhukovskogo st., 69063, Zaporozhye, Ukraine, tel/fax:+380 61 264 17 24,

E-mail: {eva, psl}@zsu.zp.ua

Abstract. The paper proposes a four-layer cooperation framework for agent-enabled business process management and performance in distributed open structured organizations comprising economically rational executives. It is assumed that the executives possess various capabilities, roles, authorities, commit to follow some joint rules while performing tasks. The layers of the framework are: Cooperation Models Layer, Interoperability Layer, Communication Layer, Transport Layer. Cooperation Models Layer provides the models of an organization, an organizational unit, a business process, the patterns for executives' coordination, performance and behaviour monitoring. Interoperability Layer contains interaction protocols, the patterns for inter-agent conversations, the formalisms for parametric feedbacks and for the shared concepts for the data interchanged by the agents in the course of their cooperative performance. The lower two layers are the slots for the widely used and/or standard primitives and components. These are Communicative acts' specifications formalized in agent communication languages and transport services and envelopes respectively. The paper also provides the review of the related work structured along the proposed framework layers. This review shows no vital contradictions with the proposed cooperation framework layering.

Аннотация: В статье предложена четырехуровневая структурная схема организации управления и выполнения бизнес-процессов на базе агентов, составляющих открытые распределенные организации экономически рациональных исполнителей. Предполагается, что исполнители обладают различными специализациями, ролями, сферами ответственности, обязуются следовать некоторым общим правилам в процессе выполнения заданий. Уровнями структурной схемы являются: Уровень моделей кооперации, Уровень интероперабельности, Уровень коммуникации, Уровень транспорта. Уровень моделей кооперации предоставляет модели организации, подразделения, бизнес-процесса, шаблоны для координации работы исполнителей, мониторинга их деятельности и поведения. Уровень интероперабельности содержит протоколы взаимодействия, шаблоны диалогов между агентами, формализмы для параметрических ответов и общих концепций данных, которыми обмениваются агенты в процессе кооперативной работы. Два нижних уровня являются слотами для подключения широко применяемых и/или стандартных примитивов и компонент. Этими компонентами соответственно являются спецификации коммуникативных актов, формализованные на языках коммуникации агентов, и транспортные сервисы и конверты сообщений. В статье также приводится обзор работ в области кооперации, структурированный в соответствии с уровнями предложенной схемы. Обзор показывает, что в работах других авторов нет существенных противоречий с предложенным способом расслоения.

Introduction

Business process management and performance automation proved to be a hot issue in Enterprise Modeling, Corporate/Enterprise Information Systems domains. A dominating approach to model a business process is to represent it in the form of a workflow. "Workflow systems have proved successful for the management of "administrative" processes – characterized by clear, well-defined structure and constant predictable form – for some time. However there is a consensus that current systems are insufficiently flexible to deal with complex, dynamic processes within a changing context."(cf. [Moore et al, 2000]). Yet one more complication is that normally the knowledge on how to perform a business process is not concentrated at one point, but is distributed among the various executives along several organizational levels and units. It even becomes impossible to define the whole workflow in case an organization is open one [Richard Scott, 1987] (e.g., electronic marketplace) and the constituents are more self-interested than benevolent or act in a limited trust environment.

The paper addresses cooperation issues in dynamic distributed business process management enabled by the coalitions (eg., [Tsvetovat et al., 2000]) of task performing economically rational [Nwana, 1996] agents. Following BPML specification [BPMI, 2001], business processes are understood as partially ordered sets of either atomic or non-atomic activities [Borue et at., 2001] performed in interaction between participants according to a defined set of rules in order to achieve a common goal. Agents represent intelligent actors having their roles [Moore et al, 2000] as organizational unit community members and possess their own knowledge on how to decompose and/or perform one or another activity. These actors form task coalitions for optimal task execution in the course of business process performance. Workflow model of the performed business process is thus generated "on-the-fly" and may be further used for the performance analysis and fine-tuning.

Inter-agent cooperation models and patterns are the key issue for agent enabled distributed business process management and performance. An important aspect is that these patterns should take into account the desired balance

[1] Presented research was run in frame of the Project financed by Ukrainian Ministry of Education, Grant № 0199Y1571.

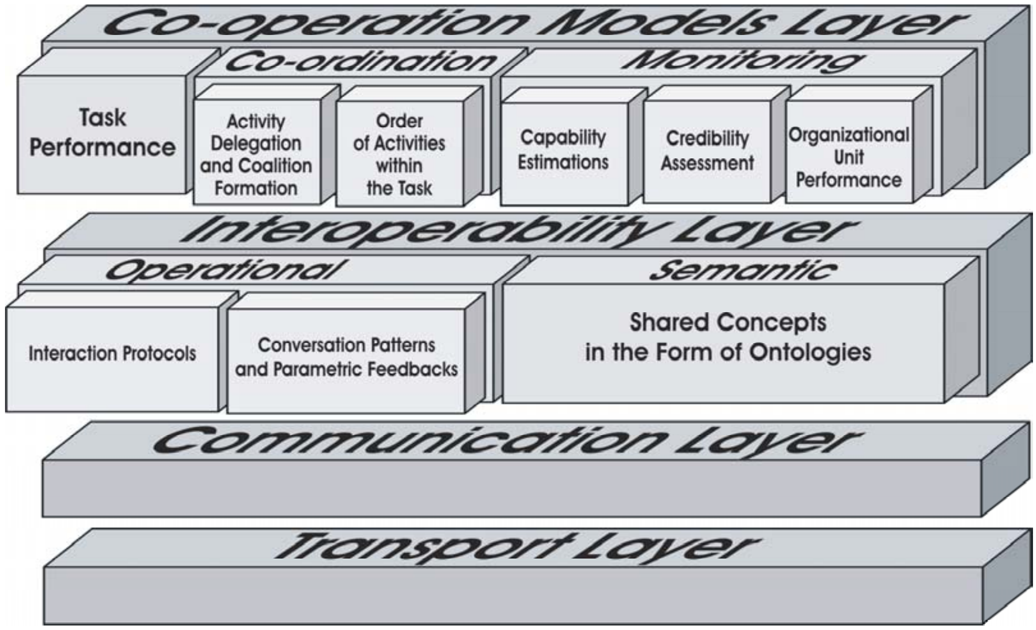


Fig. 1. The layers of cooperation patterns in business process management and performance.

[Lesser, 1998] of the actors' features of rationality, self-interest and benevolence (in the sense of being rationally ready to collaborative performance). Another important issue for open organizations is to cope with the variety of behavioral patterns, architectural, technical platforms at various levels of abstraction used to implement the actors by different independent parties in a uniform, coherent way to provide proper level of interoperability and commonly acceptable environment. The paper proposes a multi-layered agent-enabled cooperation framework in business process management domain and provides a detailed discussion of its Cooperation Models Layer and Interoperability Layer.

The reminder of the paper is structured as follows: Section 1 is devoted to the high-level description of the multi-layer cooperation framework; Section 2 present Cooperation Models Layer providing in-depth discussion of organization and business process models, the models of coordination of cooperative activities and the monitoring of organizational unit performance; Section 3 discusses Operational and Semantic slots of the Interoperability Layer; Section 4 sketches out the related work in the field and depicts the roles of the lower Communication and Transport framework Layers; Section 5 provides authors' conclusions and prospects for future work.

1 Multi-Layered Cooperation Framework in a Nutshell

Cooperation while performing business processes by autonomous, distributed actors possessing rational, uncertain and, sometimes, contradictory behaviors within an open organization is rather a complex utility. One of the widely used approaches to cope with the solution of such a multi-aspect task is to provide a proper framework layering. Layered Agent Shell is, for instance, used as the architectural solution for InfoSleuth [Nodine et al., 1998]. In the discussed context the higher the layer we are at – the more domain specific, from one hand, and the more abstract and complex components, from the other hand, are plugged into its slots. The lower we go down – the more standard, concrete, formalized elements we encounter as the layer plug-ins.

This very approach to arrange cooperation is used in reported research. Proposed is a four-layer agent-based cooperation framework for business process management and performance. The layers of the framework (refer to Fig. 1) are:

– Cooperation Models Layer
– Interoperability Layer
– Communication Layer
– Transport Layer

The upper two layers provide the models, the patterns, the formalisms for rather complex domain-specific constructs and activities. Cooperation Models Layer provides the models of an organization, an organizational unit, a business process, the patterns for executives' coordination, performance and behavior monitoring. Interoperability Layer contains interaction protocols the patterns for inter-agent conversations, the formalisms for parametric feedbacks and for the shared concepts for the data interchanged by the agents in the course of their cooperative performance.

The lower two layers are the slots for the widely used and/or standard primitives and components. These are Communicative acts' specifications formalized in agent communication languages KQML [Labrou and Finin, 1997] and FIPA ACL [FIPA, 1999] for the Communication layer and Transport services and envelopes [FIPA, 2000] for the Transport Layer.

From the conceptual point of view it is assumed the upper-layer components are assembled of the elements of the next-lower layer. A relevant alternative (e.g. KQML performative or ACL message) is used to base on the required design platform. From the implementation side, mainly on the lower layers, the lower-layer primitives are used as standardized wrappers for the more complex and domain-specific upper-layer components. A KQML performative is, for example, used as a container for activity results desirability advertisements, parametric responses, ontologies, etc.

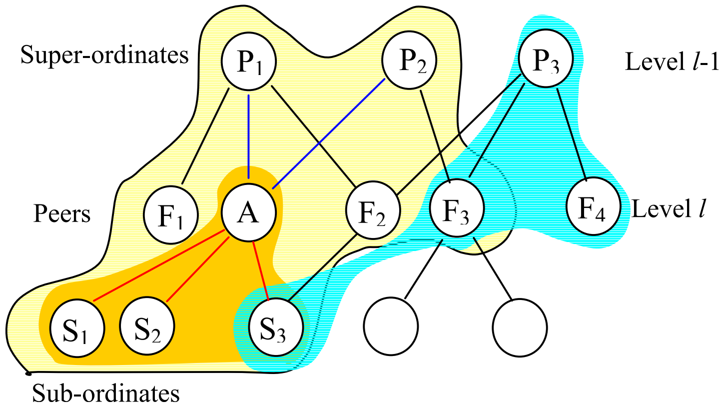


Fig. 2. A graph model of an organization.

2 Cooperation Models

Cooperation models provide a framework to formalize executives' cooperative behaviors and performance in the environment influencing them to participate in business processes management and execution. An organization and a business process models are presented in details in Section 2.1. The models of the coordination of cooperative business process performance are the subject of Sections 2.2 and 2.3. The model of Organizational unit performance monitoring is given in Section 2.4. Cooperation models are the top-level models within cooperation layers hierarchy. The in-depth development or the implementation of these models will therefore require the utilization of the more formalized and more specialized components of the lower layers of Interoperability, Communication and Transport. The layered organization of the proposed cooperation framework however provides for flexible usage of any available alternative components of the lower layers.

2.1 Functional System, Functional Component and Task Execution Models

The existence of business processes in the real world merely reflects the fact that there exist no executives in modern organizations, which are capable to perform one or another job entirely by themselves. These jobs are rather performed as cooperative processes by the groups of human and/or artificial actors. Each of these executives occupies a definite position within one or another level of an organization and is characterized by his/her/its capabilities, commitments, authority. Each of the executives possesses its own knowledge on what does this or that job mean, how it may be decomposed into the partially ordered set of simple activities, which of these activities should be delegated to its peers or subordinates. Normally, the execution of a job is initiated by the executives of the upper organizational levels, whose/which knowledge of a job is rather abstract and general. The parts of a job get more detailed context while going down the organizational structure to the executors with more specific capabilities and authorities. The activities, being atomic simple ones for a boss may be evidently considered as complex jobs by its sub-ordinates. Normally, at any level, an executor co-operates with its super-ordinate(s), fellow-peers and its sub-ordinate(s) (if any) and has no need to be aware of all the executives of the upper, of his own or of the lower levels. In case the structure of an organization is presented in a form of graph (see Fig. 2), the sphere of actor A awareness may be limited by the nodes of the upper (P_1 and P_2) and lower levels (S_1, S_2, S_3) adjacent to A as well as the nodes of the same level two branches away from A via a super-ordinate (F_1, F_2, F_3). An organizational unit (e.g., a subsidiary, a department) of level l consists of an executive of level l plus all its sub-ordinates (e.g., $\{A, S_1, S_2, S_3\}$). Some executives may participate in several organizational units (e.g., $\boldsymbol{S_3}$ in $\{A, S_1, S_2, \boldsymbol{S_3}\}$ at level l and $\{P_3, \boldsymbol{S_3}, F_3, F_4\}$ at level l-1). Such executives may belong to the spheres of awareness of outside actors and may accept external influences from the members of different organizational units. The executives capable to accept external influences from the exterior of their organizational unit and are called Proxies (e.g., A, $\boldsymbol{S_3}$). A proxy, when viewed from the outside of the organizational unit, is seen as a simple executive – *a functional component*. It represents its organizational unit (*a functional system*) in another organizational unit of a higher level. Organization is evidently the set of its organizational units at level 1. In accordance to the principles of organizational structuring (see, eg. [Gasser, 1992]) it is therefore assumed that an organizational unit is the set of active entities (actors) possessing respective capabilities and communicating according to the given shared cluster of patterns. The actors are modeled by economically rational [Nwana, 1996] software agents designed in frame of [Ermolayev et al., 2000]. The capabilities of an agent are provided by the set of macro-model programs for activity performance [Ermolayev et al., 2000].

Organizations and their functional units act within an environment. A functional system provides the environment for its functional components, which, in turn may expand into functional systems of the lower level(s). As far as the mission of an organization is to perform business processes, it is assumed that the environment is modeled by a generator function providing *tasks* T (refer to [Unruh and Nodine, 2000] for a similar terminology) as the sets of activities w^i :

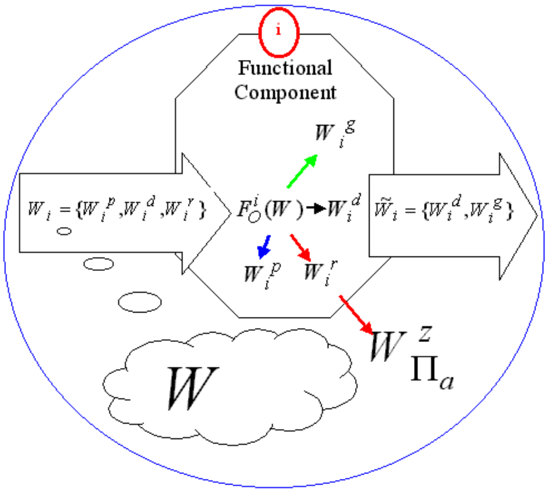


Fig. 3. Functional Component model.

$$\mathrm{E} \rightarrow \mathrm{T} = \{w^1, w^2, ..., w^k\} \quad (1)$$

These tasks are accepted by Proxies. Organization is thus tailored to perform the tasks provided by the environment as external influences.

It is assumed that a task $\mathrm{T} = \{w^1, w^2, ..., w^k\}$ is the set of atomic (for the given actor) activities. The actors within the organization are capable to perform the atomic activities belonging to the sets of their permissible atomic activities W_A. They are as well capable to generate (sub-)tasks without any external influence reacting to some events (internal to the organization) or in the course of performing of one or another atomic activity.

An actor, say *A*, involved in task execution has its own beliefs on how to perform atomic activities and how much effort should it spend to accomplish the activity w^j, provided that it possesses certain working *capacity* $N_A(w^j)$ related to this certain atomic activity.

Capacity is understood as actor's ability to accomplish the activity per unit time interval τ. It may be **unlimited** in case at any time A is able to concurrently accomplish as many w^j-s per τ as needed, and **limited** in case if the maximal quantity of concurrently running w^j-s is constrained with an upper bound. In case $N_A(w^j)$ is limited it may be evenly, or not really evenly distributed over the activities w^j to be performed.

Activity w^j may be constrained by the *deadline* d_{w^j}. The deadline is the point in time after which w^j results are not needed anymore by the customer agent. This means that w^j customer agent's results *desirability* function value:

$$des_{w^j}(t, d_{w^j}) = \begin{cases} tdf(t), t \le d_{w^j} \\ 0, t > d_{w^j} \end{cases} \quad (2)$$

falls down to zero after the deadline has passed and promises changing incentive $tdf(t)$ as a kind of a trade-off over time.

These beliefs form executives' subjective Partial Local Plans (PLP) for performing certain atomic activities. PLPs are formalized by lightweight task ontologies [Ermolayev et al., 2001] coded in standard OIL [Fensel et al., 2000]. PLP differ from, say, GPGP [Decker, 1995] by the fact they do not contain the subjective beliefs on what would be the actions of the fellow actors. Alternatively, the updates of the information on changing fellows' capabilities, fellows' credibility evaluations are performed by the actors individually in the course of their cooperative work. The actors are involved into the cooperative task execution either by the results of negotiation on delegating the task or the activity, or by accepting a targeted directive from the super-ordinate.

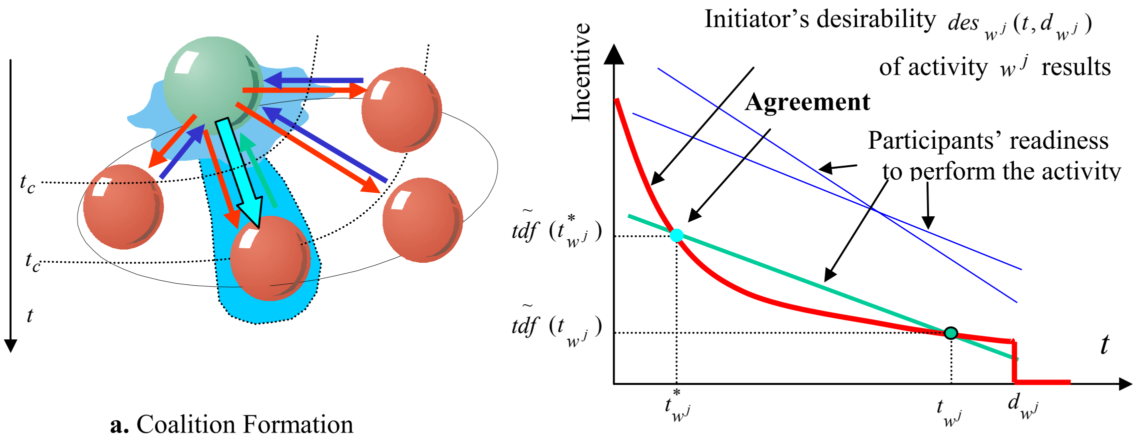
After an influence is perceived by an actor it may:

- Accept and perform some of the activities contained within the task
- Decline some of the activities
- Decide to delegate a (sub-)set of the activities to one its fellows according to it's beliefs on the fellows' capabilities (Section 2.2.3), credibility (Section 2.2.3), and readiness to perform the activity(ies) (Section 2.2.1)
- Require the performance of some new activities, the execution of which (as it knows from its knowledge formalized by the Task Ontology) is essential to successfully complete the overall accepted task execution

The model of a functional component (See Fig. 3) is built upon the idea of "absorption" and "generation" of activities from the set of the permissible activities $W = \{w^1, w^2, ...\}$ of this functional component. It is considered that the sensory input of the functional component i admits a task $W_i \subseteq W$. A certain part of its activities W_i^p may be performed ("absorbed") by the given component and the remaining part of activities may be either delegated to another system's components – W_i^d, or rejected – W_i^r. Functional component may as well generate additional set of activities W_i^g to facilitate to the execution of activities W_i^p. W_i^g as well as W_i^d are delegated to another components:

$$W_i \rightarrow F_O^i(W) \rightarrow \tilde{W}_i, \quad (3a)$$

where: $W_i = \{W_i^p, W_i^d, W_i^r\}$, $\tilde{W}_i = \{W_i^d, W_i^g\}$, $F_O^i(W)$ - macro-model program.



a. Coalition Formation

b. Proposition and Feedbacks

Fig. 4. Arrangement phase. Negotiation on delegating the activity and joining the coalition.

In a special case component i may generate a new set of activities W_i^g without been invoked by incoming influence W_i - i.e. may "summon" a new task:

$$F_O^i(W) \rightarrow \widetilde{W}_i \,, \tag{3b}$$

where: $\widetilde{W}_i = \{W_i^g\}$, $F_O^i(W)$ - macro-model program.

Process Π_a of task execution begins with the acceptance or the generation of the new task $W_a \subseteq W$. Task W_a, as well as the derived tasks $\widetilde{W}_a$, are considered to be linked to process Π_a and labeled with the unique identifier of this process. The component is considered to be *linked to process* Π_a in case it has absorbed the part of W_a, $\widetilde{W}_a$, or has generated W_a^g.

Process Π_a is considered to be completed in case all the components stopped to absorb the atomic activities of the tasks linked to process Π_a. The set of activities $W_{\Pi_a}^z$ not absorbed in the process of Π_a is denoted as the set of *inexecutable* activities.

For practice the set of permissible activities of a functional component is constrained to be finite: $W = \{w^1, w^2, ..., w^n\}$.

2.2 Coordination Models

Coordination in a functional system performing concurrent tasks is critically important as it provides the means of making the executives perform in a coherent manner. Coordination models for cooperative task performance cover two major management issues: 1-st – negotiation on activity delegation and task coalition formation arrangement and 2-nd – providing the proper order of activities' performance. One more function of the coordination slot is to provide the environment for monitoring cooperative activities of organizational unit members. The further analysis of the monitoring results may provide to proper fine-tuning the staff of an organizational unit, the capabilities and the behaviors of its functional components.

2.2.1 Negotiation on activity delegation

As it was shown in section 2.1 in a described kind of organization no executive performs the entire task. It performs the activities it is capable to do and delegates the rest to its fellow colleges. Delegation procedure takes place each time the need in such assistance arises. Coordination pattern for activity delegation is called an *Arrangement Phase*. The goal of the Arrangement Phase is to seek for the executor of the certain activity, which is decided (according to PLP or because of the overload) to be delegated to another actor. Arrangement is performed via negotiation between the *Initiator* and the group of *Participant* actors. Initiator's goal is to perform the activity in the most optimal way. It is assumed that in the course of negotiation on activity delegation an Initiator attempts to solve two-criteria optimization problem. First criterion is related to the believed optimal *time* of the activity accomplishment. Second one is the optimal *incentive* to be paid to the fellow(s). Initiator proposes the activity to some of its fellows about whom it believes that they are capable to perform the activity and that they are credible enough to accomplish the activity in the agreed time. The actors, which received the proposition (red arrows on Fig. 4a, red curve on Fig. 4b), are considered to become negotiation Participants. The Participants reply with their 2-point trade-off assessment feedbacks indicating their readiness to perform the proposed activity (Fig.4b). The participant is ready to perform the activity in case its feedback contains intersections with the desirability function declared by the Initiator. Otherwise, it rejects the proposition. Initiator than chooses the best bid from the set of received intersections. The participant, which feedback has been

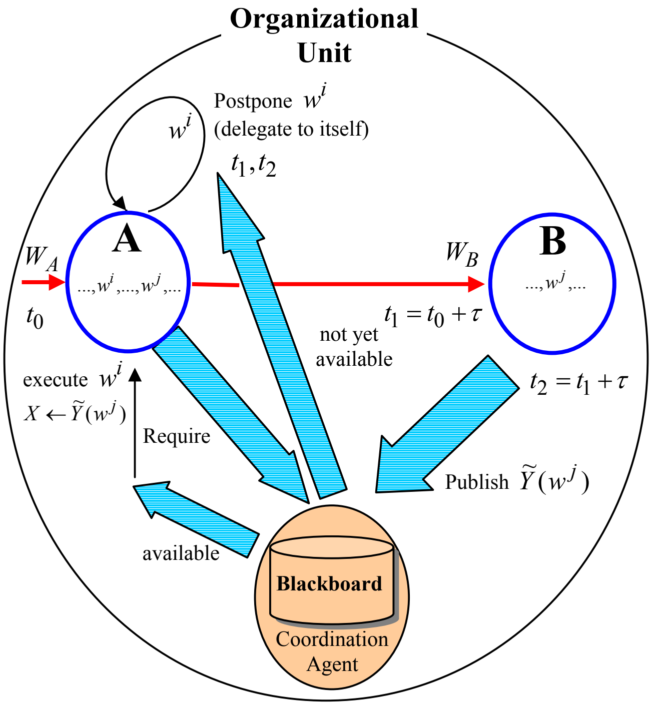


Fig. 5. Coordination of the sequence of activities execution. Assumed is that activity w^i strongly depends on activity w^j.

chosen as the optimal bid, becomes the Contractor. It thus joins the task coalition (green arrow on Fig. 4a) and proceeds with delegated activity performance.

By joining the coalition an actor pledges to follow some system rules, which regulate the proportion of its benevolence and self-interest. These rules may be classified, following Jennings Commitment-Convention hypothesis [Jennings, 1996] as actor's Individual and Joint Commitments and Coalition Conventions:

Rule 1: **Relative cooperation commitment**. Coalition members are **relatively** committed to cooperatively achieve the overall goal: to accomplish the task with maximally achievable effectiveness (maximal quality, balanced load, minimal time, ...). The ratio of this commitment depends upon the discrepancy between the actor's autonomous intentions and the overall goal of the task coalition.

Rule 2: **Activity arrangement convention**. Within the Arrangement Phase the coalition member proposing the activity (the Initiator) pledges to truthfully advertise ***desirability*** function related to the proposed activity. In response, perspective contractors (the Participants) are committed to truthfully report about their readiness to perform the activity providing the information about their ***capacity*** share by replying with the duration of activity execution in the form of parametric feedback [Ermolayev et al., 2000].

Rule 3: **Results delivery commitment.** Since an activity is accepted by an actor for the performance, the actor pledges to unconditionally accomplish this activity and to bring up the results to public immediately after the work is done.

2.2.2 Coordinating the flow of activities

Activities, which are performed in the course of a task execution, may be indirectly interrelated (as in *TÆMS* [Decker, 1995]) with strong relationships – activity w^i strongly depends on activity w^j in case the results of activity w^j are essential to start activity w^i performance, or with weak relationships – activity w^j facilitates to activity w^i performance. The flow of activities is thus constrained by partial order relationships. Sometimes this flow may contain iterative or recursive cycles. An actor may generate a subtask containing or leading to the performance of the atomic activity, which may finally be assigned to this very actor. Coordination of activity performance sequence is therefore needed to facilitate to effective and coherent activities performance as well as to the entire task execution.

The model of activity performance sequence coordination is based upon the usage of inter-activity relationships. The subjective knowledge of these relationships is contained in the actors' PLPs and is formalized by Task Ontology. Activity w^i should be performed after the activity w^j performance has been started or accomplished in case w^i weakly or strongly depends on activity w^j respectively. Otherwise w^i and w^j may be performed concurrently. It is thus sufficient to make the results of the activities performed by organizational unit executives publicly available on a shared blackboard. In frame of the reported coordination model this blackboard is under control of a dedicated utility-agent – the Coordination Agent (CoA). The executors, which have accomplished their activities, publish the results to CoA. The actors, which are going to perform the activities depending on the other activities, require CoA for the relevant results. In case these results are available on the blackboard they are further used as the parameters for the targeted activity. Otherwise the targeted activity is postponed until the next time point $t+\tau$ (refer to Fig. 5).

Let's assume that at time point t_0 actor A commences to perform the task $W_A = \{w^i, w^j\}$. After W_A decomposition and analysis has been performed A decides to perform activity w^i by itself and to delegate w^j to its fellows. After the Arrangement Phase is performed, A delegates w^j to B and proceeds with w^i execution. A's PLP for w^i points to the fact, that w^i strongly depends on w^j. A requires the results $\tilde{Y}(w^j)$ from CoA and, as far as they are not yet available, postpones w^i execution to t_1. Meanwhile, B performs w^j, accomplishes it right in agreed time ($1 \times \tau$) and publishes $\tilde{Y}(w^j)$ to CoA. By t_1 $\tilde{Y}(w^j)$ are therefore available. At the next time point A requires $\tilde{Y}(w^j)$ from CoA ones more, receives them and uses $\tilde{Y}(w^j)$ as the parameters for w^i.

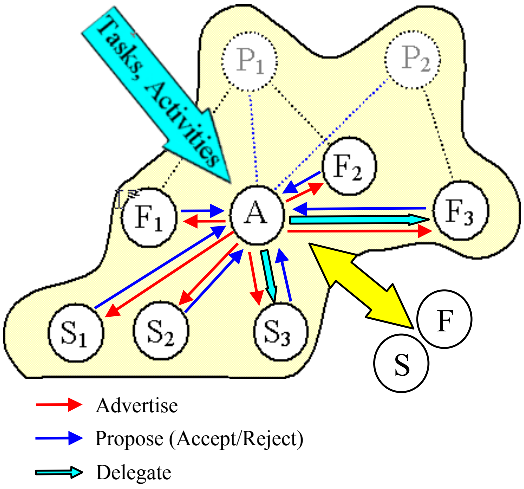


Fig. 6. Interactions providing new knowledge to update Fellows' Capability Expectations.

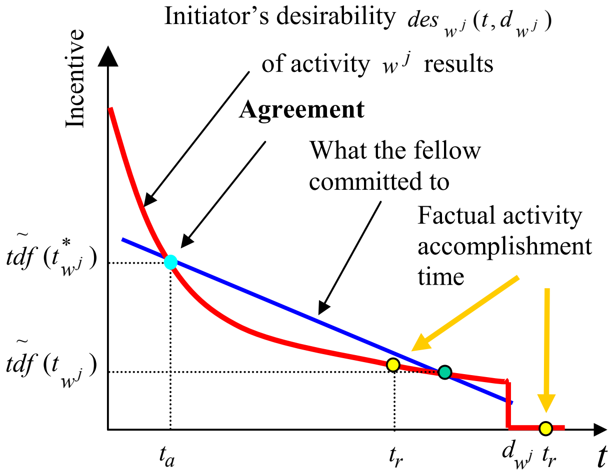


Fig. 7. Inputs to fellow credibility adjustment.

2.2.3 Capability and Credibility Estimations

Proxy actors accumulate the knowledge on the capabilities of their peers and sub-ordinates in the course of their cooperative performance of incoming tasks. New portions of this knowledge appear each time an Arrangement Phase is performed by a Proxy to delegate an activity to an executive within the sphere of its awareness (see Fig. 6) thus updating Proxy's subjective beliefs on the probability of its fellow capability to perform the given activity. These beliefs are autonomously maintained by each Proxy in the form of its Capability Expectations Matrix:

$$\mathbf{C} = \begin{array}{c} \\ F_1 \\ \dots \\ F_n \\ S_1 \\ \dots \\ S_m \end{array} \begin{array}{c} \begin{array}{ccccc} w^1 & \dots & w^j & \dots & w^k \end{array} \\ \begin{bmatrix} c_1^1 & & c_1^j & & c_1^k \\ & & \dots & & \\ & \dots & c_i^j = (q_i^j, p_i^j) & \dots & \\ & & & & \\ & & \dots & & \\ c_{n+m}^1 & & c_{n+m}^j & & c_{n+m}^k \end{bmatrix} \end{array}, \tag{4}$$

where dimensions $n+m$ and k change in the process of actor's evolution reflecting the appearance of new incoming activities and the actors' stuff within the proxy's sphere of awareness. Capability estimations c_i^j change each time the Proxy negotiates with its fellow for to delegate an activity. Element q_i^j in c_i^j tuple stands for the quantity of recorded negotiations with agent i concerning activity w^j. Element q_i^j stands for the capability expectation. The rule for c_i^j updates is as follows:

$$\begin{array}{ll} 1. & p_i^j \leftarrow p_i^j + \dfrac{r}{q_i^j}, \\ 2. & q_i^j \leftarrow q_i^j + 1 \end{array} \tag{5}$$

where r is equal to: 0 if the fellow rejected the activity, 0.5 if the fellow replied that it can accept the activity and 1 if the activity was finally delegated to the fellow.

One more aspect providing influence on a Proxy decision to delegate an activity to one or another negotiation participant is its estimation of the participant's *credibility*. A self-interested actor, due to the appearance of the new highly attractive activity offers in the competitive environment or due to the peculiarity of its behavior, may lower previously declared capacity it is spending for the bulk of the activities under execution. This will lead to the increase of the duration of these tasks execution and may seriously decrease the customer actors' desirability of these results (refer to Fig. 7) and, thus, lower the credibility value for actor selling its' fellows short.

The mechanism of accounting actors' credibility values is merely the same as that of adjusting the beliefs on changing fellow capabilities (4). Credibility estimations change over time as an actor adapts its subjective beliefs by comparing the desirability values (see Fig. 7) derived from 1-st — activity duration the executive committed to within the arrangement phase and 2-nd — actual time the executive consumed for providing the results.

Corresponding credibility matrix elements are than recomputed due to the following:

$$Cr_{i,j} := Cr_{i,j} \times \begin{cases} 1, t_r \le t_a \\ p_{w^j}(t_a / t_r), t_a < t_r \le d_{w^j} \\ 0, t_r > d \end{cases}, \tag{6}$$

where: t_a is the time the parties have agreed to accomplish the activity w^j, t_r is the actual time of w^j results delivery, d is the deadline and p_{w^j} is the weight coefficient characterizing the current priority of w^j for the customer actor.

2.2.4 Monitoring Organizational Unit Performance

The purpose of the models presented in the previous Sections 2.2.1-2.2.3 is to effectively arrange the execution of a current task. Negotiation on activity delegation, coordination of the activities flow performance, fellows capabilities and credibility assessment are tailored to optimally perform the task by exploiting the indirect knowledge on actors' capabilities, workload and behavior peculiarities. The answer to the question if the organizational unit possesses enough resources and capabilities to perform typical tasks can be however obtained only by monitoring the overall performance of the unit within some period of time.

The goal of the monitoring model is to provide the shell to analyze if the unit organization (the staff, the capabilities of the executives, their capacities) is optimal for the execution of the tasks the organization most frequently accepts for the performance. Monitoring function is vested in the Coordination Agent. CoA monitors its organizational unit by maintaining its simplified model of a task flow, provided that organizational unit members inform CoA about their states.

CoA maintains $W = \{w_1, w_2, ..., w_\sigma\}$ as the set of permissible activities of the organizational unit. CoA also updates organizational unit Conjoint State Matrix at each time point of the functional system performance. Modeling process is performed in two levels.

At the upper level the assembly of all the functional components' states into the conjoint system states model at the moment $t_n+\tau$ is performed. The conjoint model is presented in the form of matrix $\Omega(t_n + \tau)$ with dimension $m \times \sigma$, where m is the number of the functional components of the system and σ is the number of activities in W. The rows of matrix Ω are the vectors $\Theta_i = \{k_1, k_2, ..., k_j, ..., k_\sigma\}$ reflecting components' states, where k_j is the state of the component i with respect to the execution of activity w_j:

$k_j = 0$ - functional component is performing activity w_j; $k_j = l > 0$ - functional component is performing activity w_j and l similar activities are waiting in line; $k_j = l < 0$ - functional component was capable but has not been assigned to the performance of w_j (idle state).

Conjoint State Matrix $\Omega(t_n + \tau)$ is formed from the Component State Matrixes $\mathbf{K}_i$ (dimension $m \times \sigma$) representing functional component states. Matrixes $\mathbf{K}_i$ are provided by the functional components and are used as inputs to the following formula:

$$\Omega = \sum_{i=1}^{n} \mathbf{K}_i, \tag{7}$$

In addition activity delays vector D_a of the process is updated by CoA in case one or more activities w_j from W_i are postponed by any functional component:

$$D_a[j] = D_a[j] + 1, \tag{8}$$

At the lower level the production of $\mathbf{K}_i$ is performed by each functional component. $\mathbf{K}_i$-s are updated according to the results of incoming activity analysis, decomposition, and taken decision on its performance (delegate or perform). The elements of $\mathbf{K}_i$ receive the following values to adequately reflect the performance of the functional component within the time interval $]t_n, t_n + \tau]$:

k_{lj}, $l \neq i$: **1** - component i delegates activity w^j to component l

0 - otherwise

k_{ij} : **-1** - component is assigned to, or decides to perform, or will continue to perform the activity w^j within interval $]t_n, t_n + \tau]$

1 - component i postpones w^j by delegating it to itself

0 - component i is not capable to perform activity w^j within given time interval $]t_n, t_n + \tau]$.

Monitoring information collected by CoA may be further used by human administrators to fine-tune the organization by adjusting agents' capabilities, capacities, organizational units' staff thus making the organization more optimized to the performance of the typical tasks.

3 Interoperability Layer

The components of the interoperability layer provide the framework for the executives to be able to uniformly behave in the course of their cooperative activities. In frame of this layer the uniform interaction protocols, conversations and shared concepts are defined. Interoperability layer, being more specific in comparison with more abstract cooperation layer, provides more constraints on the executives. The executives are considered as agents and the organizational units are considered as Multi-Agent Systems (MAS) capable to cope with and to use the protocols, the conversations and the shared concepts described in this section.

Inter-agent operability enabling components are further on examined from and grouped by two tightly interrelated, but conceptually differing points of view – operational and semantic.

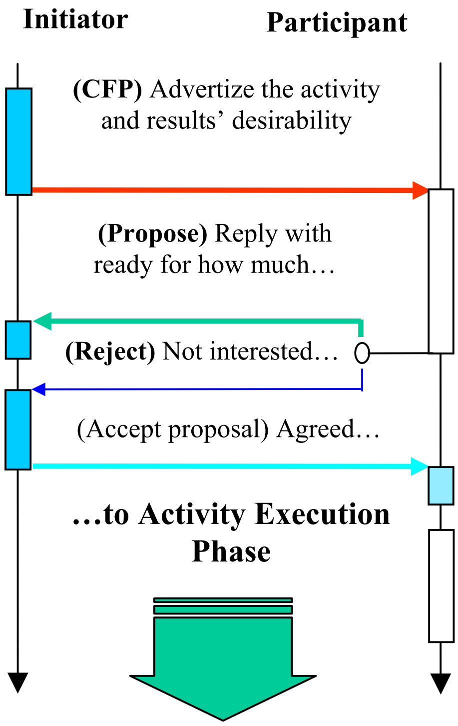


Fig. 8. Arrangement Phase protocol.

3.1 Operational Interoperability

Operational aspect comprises the interfaces and the rules for inter-agent operation formalized as the protocols and the conversations.

3.1.1 Protocols

The reported framework assumes that any relevant widely accepted interaction protocol (e.g. one of FIPA) providing the common frame for inter-agent operation may be used to facilitate to agents cooperative task performance. It is supposed that a protocol versus a conversation pattern (Section 3.1.2) is a more complex and a more purpose-specific construct and may be assembled of conversation patterns and communicative patterns of the Communication Layer.

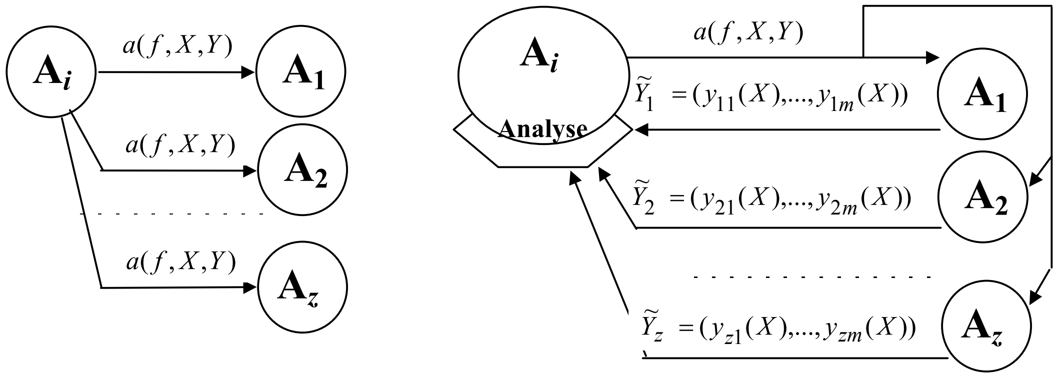
In the frame of the presented research a slightly modified FIPA CNP protocol was used to arrange negotiations on activity delegation (refer to Fig. 8). The protocol was assembled of two conversation patterns (Section 3.1.2): a parametric query with results analysis (Advertise and Propose/Reject phases) and a directive (Accept Proposal phase)

3.1.2 Conversation Patterns and Parametric Feedbacks

While the layered framework assumes that any relevant conversation formalisms may be "plugged-in" into its open slots, two different conversation patterns are distinguished in frame of the reported research: a *directive* and a *parametric query with results analysis*. A directive stands for an unconditional influence a requestor agent invokes the targeted executive(s) with. Issuing a directive thus means that the requestor is sure that the targeted executive(s) are committed to unconditionally perform the activity contained within the directive and bring the results to public. An example of the conversation performed as a directive is publishing the results of activity performance by an executive (requestor) to CoA (targeted executive). The diagram for this type of conversation is given on Fig. 9a. A parametric query with results analysis stands for a query with both the parameters and the results provided in a parametric form. An example of the conversation performed as a parametric query with results analysis is the first part of the protocol of activity delegation negotiation. The diagram for this type of conversation is given on Fig. 9b.

A parametric query conversation mechanism is base upon the usage of *parametric feedbacks*. In frame of negotiation on activity delegation, for instance, the Initiator agent advertises the activity (*f* on Fig. 9b) by providing results desirability function *des* (2) in a parametric form (as the table of values) together with activity parameters X^* ($X = \{des\} \cup X^*$ on Fig. 9b). Participants than reply with their 2-point parametric propositions ($\widetilde{Y}_1, \widetilde{Y}_2, ...\widetilde{Y}_z$ on Fig. 9b), providing the range of alternative values in response to the given parametric input. These propositions are analyzed by the Initiator (recall Section 2.2.1). More details on the parametric feedback formalism may be found in [Ermolayev et al., 2000].

Conversation patterns are to be constructed from the appropriate communicative patterns of the framework's Communication Layer.



a. a directive

b. a parametric query with results analysis

Fig. 9. Diagrams of Conversation Patterns.

3.2 Semantic Interoperability

Semantic face defines the shared concepts as ontologies, which formalize the knowledge the MAS and the agents operate with while performing tasks. Task and Negotiation ontologies were designed and formalized in Standard OIL [Horrocks et al., 2000] to serve as shared conceptualizations in frame of the reported approach to model cooperative task execution by coalitions of service providing agents – the members of organizational units. The framework and the design of these ontologies was primarily motivated by B2B E-Commerce features of uncertainty and the need of the proper balance between self-interest, rationality and benevolence. The role of the Task Ontology is to provide shared concepts of a task, an activity, a parameter, a result template an effort, a priority, a deadline, a budget as well as a Partial Local Plan for activity performance. These concepts are used by agents-executives to determine: if the incoming activity is atomic; if they are capable to perform the activity; if the parameters and the expected results match to their knowledge about the activity; if they need their fellows' assistance to perform the activity. ER-style diagram of the Task Ontology is given on Fig. 10.

Negotiation Ontology provides shared conceptualization of the terms used by agents while they participate in negotiations on activity placement. Negotiation type for the Arrangement Phase is Contracting. The difference of the negotiation approach used in the reported research from another contracting frameworks is the use of parametric trade-off assessment feedbacks. This parametrization provides more flexibility to agents' behavior and allows to avoid negotiation iterations. The shared concepts of Negotiation Ontology are: activity, results' desirability, proposed deadline, time and its granularity, incentive, trade-off point, trade-off feedback. ER-style diagram of the Negotiation Ontology is presented on Fig. 11.

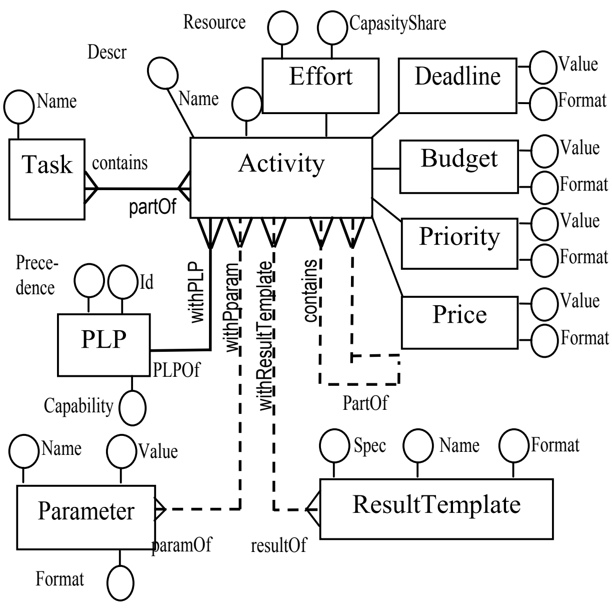


Fig. 10. ER-style diagram of Task Ontology

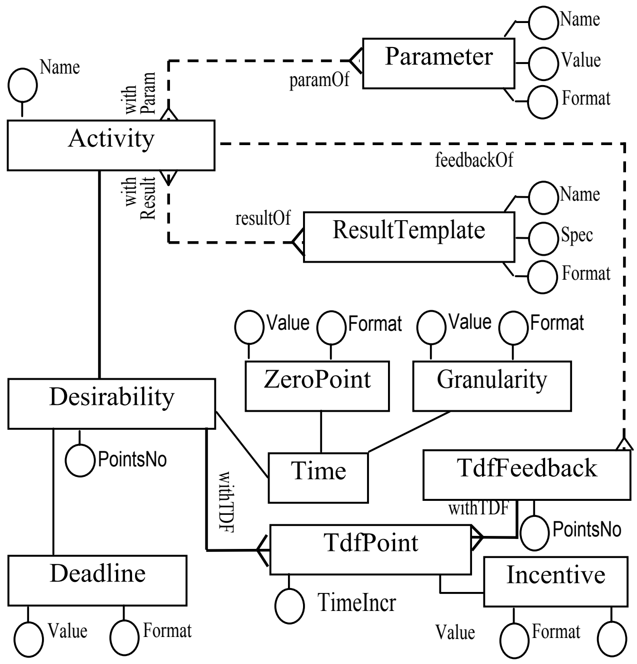


Fig. 11. ER-style diagram of Negotiation Ontology

OilEd 2.2a[1] and FACT[2] reasoner were used for ontologies design and expressiveness check respectively. OIL, RDFS, DAML and SHIQ versions of the reported Task and Negotiation Ontologies are available at http://eva.zsu.zp.ua/eva_personal/ontologies/. Further details on the design and the application of the presented ontologies may be found in [Ermolayev et al., 2001]

4 Related Work

Though the framework for arranging cooperation among the rational autonomous executives performing or managing business processes in open distributed organizations has not been proposed in the form presented in the paper, lots of substantial efforts are applied and numerous successful accomplishments may be observed in the domain. A definitely incomplete survey of the related work is further on structured along the proposed framework layering.

4.1 Models of Cooperation

It is emphasized in [Inverno et al., 1997] that "…Central to the study of cooperation is the notion of a *social structure*. A social structure is a set of relations that hold between agents in a society. These relations define the dependencies that exist between agents … and determine the rights and responsibilities of each agent in the society with respect to its peers. In order to cooperate effectively with its peers, an agent must *represent* any social structures in which it plays a part, and *reason* with these representations. This reasoning process is carried out in order to answer such questions as whether cooperation is possible, and to investigate how an agent stands in relation to other agents in the society...". Social structures are seen as organizations in the domain of business process management and performance.

Basic conceptual models of an organization, an executive, a business process (a task) may be found in [Uschold et al., 1998], [Tate, 1998], [Schlenoff et al., 1999] and many other sources. Abstract representation of *cooperation structures*, wherein agents cooperate to achieve goals on each other behalf and the study of the feasibility of such cooperation is given in [Inverno et al., 1997]. Decentralized task/activity decomposition and allocation mechanisms are presented by [Walsh and Wellman, 1998], [Stone and Veloso, 1999]. A comprehensive survey of negotiation approaches to distributed service provision may be found, for example, in [Faratin, 2000]. In E-Commerce, for instance, the models for coalition formation based on pre- and post- negotiation patterns are proposed in [Tsvetovat et al., 2000] having COALA[3] as the general-purpose testbed for studying cooperative behaviors in agent coalitions. [Shehory and Kraus, 1998] examine methods for task allocation based on agents' coalitions formation. The approach of the presented research is close to that of service oriented negotiation [Faratin et al, 2000], which involves determining a contract under certain terms and conditions. The mechanisms of service provision via negotiation discussed in this paper are more close to that of ADEPT [Jennings et al., 2000], though, more authoritarian algorithms were examined as well [Barbuceanu and Fox, 1996]. A framework for argumentation-based negotiation is proposed in [Sierra et al., 1998]. Capability based approach to service matchmaking (e.g., [Sycara et al., 1999]) is proposed to determine the proper candidate to become the service provider. Methods for adjusting agents' behaviors in cooperative environments based on learning technique are proposed in [Matsubara et al., 1996], [Claus and Boutilier, 1997], [Ermolayev, 2000b]. Coordination patterns and languages are presented in [Decker, 1995], [Barbuceanu and Fox, 1996b]

4.2 Interoperability Issues in Agents' Organizations, Communities and Coalitions

The common understanding of agent interoperability expressed by [Greaves et al., 1999] is based on three main characteristics shared by the interoperating agents:

"…They would agree on the syntax and semantics of a common agent communication language (ACL) in which to express themselves (*i.e.,* there must be *language* interoperability)".

"...They would share (possibly through translation) a common content ontology, truth theory, and method of binding objects to variables (*i.e.,* there must be *logical* interoperability)".

"..They would be able to access a set of shared infrastructure services for registration, reliable message delivery, agent naming, and so forth (*i.e.,* there must be *structural* interoperability)".

Cooperation and coordination issues are not even considered within these widely accepted inter-agent operability faces, though some emphasis to the content semantics is made. Within the framework, presented in the paper, syntactic issues of *language* interoperability belong to the Communication Layer, *logical* interoperability requirements form the part of Semantic Interoperability Slot, *structural* interoperability matters are the subject of the Transport Layer. Missed higher-level "…goaldirected interagent dialogues …" (cf. [Greaves et al., 1999]) comprise Operational Interoperability Slot as interaction protocols and conversation patterns.

4.2.1 Interaction Protocols and Conversations

Operational aspect's role in the context of agents' interoperability is to provide means to proper, commonly understandable scheduling of agents' communicative acts in order to achieve an organizational or coalitional goal – to accomplish a task with maximum achievable quality. [Wagner et al., 1999] propose to frame agents' conversations into "… the context of quantified, scheduling-specific multi-agent coordination…". In this context interaction protocols may

[1] Freely downloadable from http://img.cs.man.ac.uk/oil/. Last accessed on Nov. 3, 2001.
[2] http://www.cs.man.ac.uk/~horrocks/FaCT/. Last accessed on Nov. 3, 2001.
[3] http://www.cs.cmu.edu/~softagents/coala.html. Last accessed on Apr. 10, 2001.

be denoted as specific coordination patterns for generic inter-agent conversations in the course of performing tasks. [Pitt and Mamdani, 1999] in addition propose to concentrate on protocol-centric formulation of ACL semantics due to the fact that "…the protocol specification is considerably less complex than the intentional description; interoperability of separately developed components via a protocol-based semantics is improved; reuse of recurrent interchanges is enabled; and verification of compliance with the standard is made easier…".

Conversation policies are commonly denoted as declarative specifications that govern communications between agents that use an agent communication language and are clearly separated from the messaging mechanisms. Conversations thus may be denoted as ACL communication constructs wrapped by conversation policy constraints. Conversation policies are commonly understood as general constraints on semantically coherent ACL messaging behavior. [Unruh and Nodine, 2000] from their experience with InfoSleuth infrastructure state that the aspects of conversation policy specification that need to be addressed at the level of conversation types include the following: Role-based conversation specification, Agent location and message passing policies, Control policies for extended conversations.

Agent conversations (cf. [Unruh and Nodine, 2000]) fall into two levels, extended conversations that encompass all of the interactions among agents and agent types related to some overall task being executed on behalf of the user, and individual, localized conversations between pairs of specific agents that form the components of the extended conversations. In the given terminology from the structural point of view the protocols slot within the framework presented in the paper may stand for an extended conversation.

[Lin et al., 2000] propose to introduce a utility layer of incorporated conversation managers to facilitate to enhancing high-level communication capability.

4.2.2 Semantic Interoperability

Substantial effort is applied to resolve the problem by various research communities as well as by standardization bodies trying to unify and harmonize the means for concept sharing and semantic interoperability, coordination patterns, process description and service provision facilities. In distributed process management and service provision domains the activities in these 3 streams may be presented as follows.

The major standardization effort in process, workflow modeling and management belongs to WfMC[1]. The major accomplishments of WfMC in the field are: the Process Model – workflow and activity representation, XPDL – XML binding of PDL. Semantic interoperability solution is generally seen by WfMC as Workflow-XML binding.

As it was mentioned in [Tate, 1998], "...cooperation and coordination of the planning, monitoring and workflow of the organizations can be assisted by having a clear shared model of what comprises plans, processes and activities...". Known are the efforts aiming to define the basic shared concepts: SPAR [Tate, 1998] ontology, the Enterprise Ontology [Uschold et al., 1998], Process Specification Language [Schlenoff et al., 1999] (ordered hierarchies of activities), ToVE [Fox and Gruninger, 1998] (shared terminology for a virtual enterprise), aggregation of activities at multiply levels [Gruninger and Fox, 1994], O-Plan research [Tate, 2000] (manipulating plans of task execution) and others.

Substantial results are appearing in the development of the languages for service mark-up. DAML+OIL [Mcllraith et al., 2001] initiative promises probably the best perspectives as both languages are built on top of RDF(S) – W3C[2] metadata standard.

4.3 Communication

The definition of the place and the role of communication primitives in the interaction protocols –conversation patterns – communicative acts hierarchy still requires concretization. Some authors (e.g., KAoS [Bradshaw et al., 1995], [Bradshaw, 1997]) consider dialogue (conversation patterns) protocols as the core basics for inter-agent-communication specification. This point of view has also found its implementation in the enhanced KQML proposal [Labrou and Finin, 1997]. The InfoSleuth framework also emphasizes conversation patterns as the core elements of inter-agent communication and interoperability facilitators [Nodine et al., 1998]. Other authors propose an alternative approach to base Agent Communication Languages semantics on interaction protocols and performatives with the intentional descriptions providing exemplary guidelines [Pitt and Mamdani, 1999]. The tendency (supported by FIPA from the authors' point of view) is still to introduce a higher-level conversation patterns, interaction protocols and coordination frameworks over the more primitive, formal and standardized collections of communicative acts specified in ACL. One of the approaches to formalize communications and communication primitives is the use of Petri nets (more recently – colored Petri nets [Lin et al., 2000]).

The efforts and the accomplishments of Communication Layer are well presented in the survey article by Yannis Labrou, Tim Finin, and Yun Peng [Labrou et al., 1999]. Major standardization efforts in the domain are carried out by FIPA (e.g., [FIPA, 1999]).

4.4 Transport frameworks and services

A general consensus on the transport environment is that it should provide the shell for agent naming, location and message delivery mechanisms. According to FIPA Transport Service Reference Model [FIPA, 2000b] Agents in an

[1] Workflow Management Coalition. http://www.wfmc.org/. Last accessed on Nov. 4, 2001
[2] World Wide Web Consortium. http://www.w3c.org. Last accessed on Nov. 4, 2001.

open organization are bound to Agent Platforms (AP) and exchange messages via the Transport Services of their AP-s. The mechanism FIPA proposes as the standard to cope with various network protocols is the use of the message Envelopes. For the moment provided are the specifications for IIOP [FIPA, 2000c] and WAP [FIPA, 2000d] protocols.

5 Conclusions

Cooperation while performing business processes by autonomous, distributed actors possessing rational, uncertain and, sometimes, contradictory behaviors within an open organization is rather a complex utility. Though, Inverno, Luck and Woldridge consider cooperation to be "certainly the best studied process in multi-agent systems research" (cf. [Inverno et al., 1997]), there are still lots of open issues in the domain. One of the most difficult problems is the lack of the widely accepted consensus on how all this staff should be structured and organized. The main contribution of the paper is the proposal of a four-layer cooperation framework for agent-enabled business process management. The paper is not so ambitious as to claim the final solid word in the domain, but rather to analyze the trends, to try to put it to the reasonable places within a conceptual hierarchy. The research, the paper is based on, to some extent proves (Sections 2 and 3) that there is some sense in the proposed layering, especially in the domain of business process management and performance. From the other hand, the review of the related work (Section 4) provides no vital contradictions to the presented layering proposition. The future work planned by the authors in frame of the new project is targeted to the further verification, development and implementation of agent-enabled cooperation framework

References

[Barbuceanu and Fox, 1996] Barbuceanu, M.; Fox, M.S.: The Architecture of an Agent Building Shell. In: (Wooldridge, M.; Mueller, J.P.; Tambe, M. Eds): Intelligent Agents II: Agent Theories, Architectures and Languages. Lecture Notes in Artificial Intelligence, Vol. 1037, Springer-Verlag, Berlin Heidelberg New York (1996)

[Barbuceanu and Fox, 1996b] M. Barbuceanu and M.S. Fox. The Design of a Coordination Language for MultiAgent Systems. In J.P. Muller, M.J. Wooldridge, and N.R. Jennings, editors, Intelligent Agents III. Proceedings of Third International Workshop on Agent Theories, Architectures, and Languages (ATAL'96)), number 1193 in LNAI. Springer Verlag, August 1996.

[Borue et al., 2001] S. U. Borue, V. A. Ermolayev, N. G. Keberle, S. L. Plaksin, V. A. Tolok Mathematical Models and Methods for the Description and for the Interaction of the Elements of a Unified Information Space. Final Report on Research Project (Grant No 0197y012776 of Ministry of Education and Science of Ukraine), Zaporozhye State Univ., Zaporozhye, 2001, 174p. (in Ukrainian)

[BPMI, 2001] BPMI.org, Business Process Modeling Language (BPML), Working Draft 0.4, August, 2001, http://www.bpmi.org last accessed 12.01.2002.

[Bradshaw et al., 1995] Bradshaw J.M., Dutfield S., Carpenter B., Jeffers R., Robinson T.: KAoS: A Generic Agent Architecture for Aerospace Applications, in *Proc. of the CIKM'95 Intelligent Information Agents Workshop*, Omni Inner Harbor Hotel, Baltimore MD, Dec. 1–2, 1995.

[Bradshaw, 1997] Bradshaw J.M.: KAoS: Toward an Industrial-Strength Open Agent Architecture, in Bradshaw J.M.(ed.), *Software Agents*, AAAI Press/MIT Press, Cambridge/Menlo Park, 375–418, 1997.

[Claus and Boutilier, 1997] Claus, C., Boutilier, C.: The Dynamics of Reinforcement Learning in Cooperative Multiagent Systems. AAAI-97 Workshop on Multiagent Learning, (1997).

[Decker, 1995] Keith S. Decker, Environment Centered Analysis and Design of Coordination Mechanisms. Umass CMPSCI Tech. Rep. 95-69, 1995, 199 p.

[Ermolayev et al., 2000] V. A. Ermolayev, S. U. Borue, V. A. Tolok, N. G. Keberle: Use of Diakoptics and Finite Automata for Modelling Virtual Information Space Agent Societies // "Lecture Notes of Zaporozhye State University", ISBN 966-599-058-4, Vol. 3, No 1, 2000, pp. 34-44.

[Ermolayev, 2000b] Ermolayev, V., Dynamic Agent Communities Facilitating to Distant Learning in a Virtual University Information Space. In: Proc. of Intl. Conf. IS2000, Special Session on Virtual Universities and Distance Education, Japan, November 5-8, 2000 (2000) 488–495

[Ermolayev et al., 2001] Ermolayev, V. Keberle, N., Tolok, V.: OIL Ontologies for Collaborative Task Performance in Coalitions of Self-Interested Actors. To appear In: Proc. of the 2-nd Int. Workshop on Conceptual Modelling Approaches for E-Business (eCOMO'2001) at ER'2001, Nov. 27-30, 2001, Yokohama, Japan.

[Faratin, 2000] Faratin, P.: Automated Service Negotiation Between Autonomous Computational Agents. PhD Thesis, Univ. of London (2000)

[Faratin et al, 2000] Faratin, P., Jennings, N. R., Buckle, P., Sierra, C.: Automated Negotiation for Provisioning Virtual Private Networks Using FIPA-Compliant Agents. In: Proc. 5th Int. Conf. on the Pratcical Application of Intelligent Agents and Multi-Agent Systems (PAAM-2000), Manchester, UK (2000) 185–202

[Fensel et al., 2000] D. Fensel, M. Crubezy, F. van Harmelen, and I. Horrocks: OIL & UPML: A Unifying Framework for the Knowledge Web. In Proceedings of the Workshop on Applications of Ontologies and Problem-solving Methods, 14th European Conference on Artificial Intelligence ECAI'00, Berlin, Germany August 20-25, 2000.

[FIPA, 1999] Foundation for Intelligent Physical Agents (FIPA) Spec: DRAFT, Version 0.2, Agent Communication Language, 1999, http://www.fipa.org, last accessed Jan 19, 2002.

[FIPA, 2000] Foundation for Intelligent Physical Agents (FIPA) Spec: FIPA Agent Message Transport Service Specification Doc No XC00067B, 2000, http://www.fipa.org, last accessed Jan 19, 2002.

[FIPA, 2000b] Foundation for Intelligent Physical Agents (FIPA) Spec: FIPA Agent Message Transport Service Specification Doc No XC00067, 2000, http://www.fipa.org, last accessed Jan 19, 2002.

[FIPA, 2000c] Foundation for Intelligent Physical Agents (FIPA) Spec: FIPA Agent Message Transport Protocol for IIOP Specification C and D Doc No XC00075, 2000, http://www.fipa.org, last accessed Jan 19, 2002.

[FIPA, 2000d] Foundation for Intelligent Physical Agents (FIPA) Spec: FIPA Agent Message Transport Protocol for WAP Specification Doc No XC00076, 2000, http://www.fipa.org, last accessed Jan 19, 2002.

[Fox and Gruninger, 1998] Fox, M.C.; Gruninger, M.: Enterprise Modelling. AI Magazine 19(3) (1998) 109–121

[Gasser, 1992] Gasser, L., "DAI Approaches to Coordination" in Distributed Artificial Intelligence: Theory and Praxis (eds. N. M. Avouris and L. Gasser) Kluwer Academic Publishers pp 31-51.

[Greaves et al., 1999] Mark Greaves, M., Holback, H., Bradshaw, J.: What Is a Conversation Policy? In: Proc. Of the Workshop on Specifying and Implementing Conversation Policies, Autonomous Agents '99, Seattle, Washington May 1st, 1999

[Gruninger and Fox, 1994] Gruninger, M., Fox M. C.: An Activity Ontology for Enterprise Modelling. Workshop on Enabling Technologies – Infrastructures for Collaborative Enterprises, West Virginia Univ. (1994)

[Horrocks et al, 2000] Horrocks, I., Fensel, D., Broekstra, J., Decker, S., Erdmann, M., Goble, C., Van Harmelen, F., Klein, M., Staab, S., Studer, R.: OIL: The Ontology Inference Layer, Tech.Rep. IR-479, Vrije Universiteit Amsterdam, Sept., 2000 (2000)

[Inverno et al., 1997] M. d'Inverno, M. Luck, and M. Wooldridge, (1997). Cooperation structures. IJCAI '97.

[Jennings, 1996] Jennings, N. R.: Coordination Techniques for Distributed Artificial Intelligence. In: (O'Hare, G. M. P., Jennings, N. R. Eds.) Foundations of Distributed Artificial Intelligence. Wiley (1996)

[Jennings et al., 2000] Jennings, N. R., Faratin, P., Norman, T. J., O'Brien, P., Odgers, B. Alty, J. L.: Implementing a Business Process Management System using ADEPT: A Real-World Case Study. Int. Journal of Applied Artificial Intelligence 14 (5) (2000) 421-465

[Labrou and Finin, 1997] Labrou, Y. & Finin, T. (1997). A proposal for a new KQML specification , Technical Report TR CS-97-03, University of Maryland Baltimore County (UMBC), Computer Science and Electrical Engineering Department, Baltimore, Maryland 21250, USA. http://citeseer.nj.nec.com/article/labrou97proposal.html last accessed Jan 20, 2002

[Labrou et al., 1999] Labrou, Y., Finin, T. and Peng, Y.: Agent communication languages: the current landscape. IEEE Intelligent Systems, May 1999.

[Lesser, 1998] Lesser V.R.: Reflections on the Nature of Multi-Agent Coordination and Its Implications for an Agent Architecture, Journal of Autonomous Agents and Multi-Agent Systems, 1(1), 89–111, 1998.

[Lin et al., 2000] F. Lin, D. Norrie, R. Flores, R. Kremer, Incorporating Conversation Managers into Multi-agent Systems. In: Proc. Of Agents-2000 Workshop on Agent Communication 3 June 2000, Barcelona

[Matsubara et al., 1996] Hitoshi Matsubara, Itsuki Noda, and Kazuo Hiraki. Learning of cooperative actions in multi-agent systems: a case study of pass play in soccer. In Adaptation, Coevolution and Learning in Multiagent Systems: Papers from the 1996 AAAI Spring Symposium, pages 63--67, Menlo Park,CA, March 1996. AAAI Press. AAAI Technical Report SS-96-01.

[Mcllraith et al., 2001] Mcllraith, S. A., Son, T.C., Zeng, H.: Mobilizing the Semantic Web with DAML-Enabled Web Services. In: Proc. Of the Second Intl. Workshop on the Semantic Web – SemWeb'2001, Honkong, China, May 1, 2001 (2001) 82–87

[Moore et al, 2000] J. Moore, R. Inder, P. Chung, A. Macintosh, and J. Stader: "Who Does What? Matching Agents to Tasks in Adaptive Workflow."; In Proceedings of the 2nd International Conference on Enterprise Information Systems (ICEIS 2000), B. Sharp, J. Cordeiro, and J. Filipe (eds), Stafford, July 2000, pp 181-185, ISBN 972-98050-1-6.

[Nodine et al., 1998] M. Nodine, B. Perry, and A. Unruh. Experience with the InfoSleuth agent architecture. In: Proc. of AAAI-98 Workshop on Software Tools for Developing Agents, 1998.

[Nwana, 1996] Nwana, H. S.: Software Agents: an Overview. Knowledge Engineering Review 11(3) (1996) 205–244

[Pitt and Mamdani, 1999] Pitt, J. and Mamdani, A.: Communication Protocols in Multi-Agent Systems. In: Proc. Of the Workshop on Specifying and Implementing Conversation Policies, Autonomous Agents '99, Seattle, Washington May 1st, 1999

[Richard Scott, 1987] Richard Scott, W.: Organizations: Rational, Natural and Open Systems. Prentice-Hall, Inc. Englewood Cliffs, NJ (1987)

[Shehory and Kraus, 1998] O. Shehory and S. Kraus. Methods for task allocation via agent coalition formation. Artificial Intelligence, 101(1-2):165--200, 1998.

[Schlenoff et al., 1999] Schlenoff, C.; Ciocoiu, M.; Libes, D.; Gruninger, M.: Process Specification Language: Results of the First Pilot Implementation. In: Proc. of the Int. Mechanical Engineering Congress and Exposition, Nashville (1999)

[Sierra et al., 1998] C. Sierra, N.R. Jennings, P. Noriega, and S. Parsons. A framework for argumentation-based negotiation. In M.P. Singh, A. Rao, and M.J. Wooldridge, editors, Proc. ATAL-97, pages 177--192, Berlin, Germany, 1998. Springer-Verlag.

[Stone and Veloso, 1999] Peter Stone and Manuela Veloso. Task decomposition and dynamic role assignment for real-time strategic teamwork. In "Intelligent Agents V — Proceedings of the Fifth International Workshop on Agent Theories, Architectures, and Languages (ATAL'98)", Mueller, Singh, and Rao (Eds.), 1999. Springer Verlag, Heidelberg.

[Sycara et al., 1999] Sycara, K., Klusch, M., Widoff, S., Lu, J.: Dynamic Service Matchmaking Among Agents in Open Information Environments. ACM SIGMOD Record 28(1) Special Issue on Semantic Interoperability in Global Information Systems (1999) 47–53

[Tate, 1998] Tate, A.: Roots of SPAR - Shared Planning and Activity Representation. Knowledge Engineering Review 13(1) (1998) 121-128

[Tate, 2000] Tate, A.: <I-N-OVA> and <I-N-CA> - Representing Plans and other Synthesized Artifacts as a Set of Constraints. In: Proc. of the AAAI-2000 Workshop on Representational Issues for Real-World Planning Systems, Austin (2000)

[Tsvetovat et al., 2000] Tsvetovat, M., Sycara, K., Chen, Y., Ying, J.: Customer Coalitions in the Electronic Marketplace. In: Proc. of Agents 2000 Conference, Barcelona, Spain (2000)

[Unruh and Nodine, 2000] Amy Unruh and Marian Nodine. Industrial-Strength Conversations. In: Proceedings of Agents-2000 Workshop on Agent Communication, 3 June 2000, Barcelona

[Uschold et al., 1998] Uschold, M., King, M., Moralee, S., Zorgios, Y.: The Enterprise Ontology, Knowledge Engineering Review 13(1) (1998)

[Wagner et al., 1999] Thomas Wagner, Brett Benyo, Victor Lesser, and Ping Xuan. Investigating Interactions Between Agent Conversations and Agent Control Components. In Agents 99 Workshop on Conversation Policies, 1999.

[Walsh and Wellman, 1998] William E. Walsh and Michael P. Wellman. A market protocol for decentralized task allocation. In: Proc. of the Third International Conference on Multi-Agent Systems, pages 325--332, 1998.