

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

5,300

Open access books available

132,000

International authors and editors

160M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Multi-Agent Models in Workflow Design

Victoria Jordan
West University of Timisoara
Romania

1. Introduction

Many researchers have focused on the intelligent agent and multi-agent systems approaches as alternatives for e-business and enterprise integration applications. Agent approaches and technologies were developed and used in e-business applications business process management in (Jennings et al., 1996), (Jennings et al., 2000), (Papazoglu, 2008) supply chain management in (Huhns & Stephens, 2001) and enterprise integration in (Pan & Tenenbaum, 1991), (Sikora & Shaw, 1998) and (Sikora & Shaw, 2002). During the 1990s were developed ERP and EAI, the last one, involves the integration of various enterprise applications so that they can share information and processes freely in (Linthicum, 1999).

Workflow systems, enable the automation of a business process, in whole or part, during which documents, information or tasks are passed from one participant to another for action, according to a set of procedural rules as in (Fischer, 2000), (Aalst & Akhil, 2003). Processes have associated resources, in terms of participants and invoked applications. In turn these resources may have extended descriptions in terms of the organizational model, and application environment respectively. Validation of the process usually requires the knowledge of a domain expert on the underlying business rules, business process logic and many types of local constraints in (Sadiq et al., 2004). Manual validation for any reasonably complex process models is complex and hard to achieve with an adequate dose of accuracy. Furthermore, it is unrealistic to assume that domain experts would also carry expert knowledge on process modeling. One can expect that workflow designers and domain experts would typically be disjoint but collaborating users of the process technology. The validation, like structural verification is not intended to provide guarantees on the correctness and/or completeness of the process model against business requirements. It is rather intended to support the process modeling activity, by providing smart tools for cross checking process and activity properties, and identifying potential conflicting and redundant elements in the specification, thereby boosting user confidence in the deployment of the process model.

The areas like inter-organizational cooperation and virtual enterprises require new solutions due to the high dynamics in their interrelations. Since the process perspective has been within the center of interest, workflow management systems (WFMS) have had a revival in the context of the development of distributed applications. Most of the commercially available workflow management systems do not offer sufficient flexibility for distributed organizations that participate in the global market. These systems have rigid, centralized architectures that do not operate across multiple platforms (Purvis et al., 2004).

Modeling processes and workflow for complex systems constitutes a challenge for designers. The resources that are used in systems are in limited amounts and these must be shared between the processes and workflow instances activities.

Due to the complexity of real systems are needed more and sophisticated models. The multi agent models are widely used in order to use in an efficient manner the information, based on the reasoning of intelligent agents. The roles and their usage play an important place in the models with agents in the design of workflows.

Our considerations refer to the need of a data base in that the workflow constituents: instances and activities must be represented in a data base and also their time evolution.

We intend to propose a model for resource representation and allocation and as conclusion propose considerations concerning the design of processes, workflows and workflow management systems.

2. Integration requirements

A single work system or a single actor within a work system does not have the necessary knowledge, resources, and capability to solve such complex problem as is the integration. As a general goal is to integrate different work systems in order to solve the problems that span multiple work systems, coordinating the goals, tasks and resources of multiple work systems in order to achieve the overall system goals.

The main integration requirements are: work systems integration, data integration and technology integration and in addition to providing task and decision support within individual work systems.

The work systems integration is concerned with the key organizational aspects of goals, tasks of business processes, actors and resources. Work systems integration, considered as being coordination involves structuring workflow, allocating tasks and resources to actors, sharing resources, managing interdependencies (or simply dependencies) between tasks, resources and actors, and enabling communication across multiple work systems. The coordination as managing dependencies between activities if there are no interdependencies among activities then is no need for coordination in the system, as in (Malone & Crowston, 1994).

The coordination perspectives: managing dependencies, managing workflow and managing communication among various work systems use adequate analysis and modeling techniques.

3. Previous works

Concerning the coordination, many models were developed. The dependency perspective, which presumes that coordination problems are caused in the first place by interdependencies among various organizational actors, tasks, and resources, and that an effective coordination system should address and resolve these mutual interdependencies. For the dependency perspective were developed many models between we mention: Actor Dependency Model (Yu & Mylopoulos, 1996) and Dependency Network Diagrams that allow representing the essential elements of inter-organizational relationships based on resource dependency theory (Tillquist et al., 2002).

The Role Activity Diagrams (RAD) are presented in (Ould, 1995), as a set of graphic notations for modeling business processes these were frequently used. The RAD technique

represents a business process into a set of interacting roles. Each of these roles and group activities together might be carried out by a person, group or machine i.e., an actor or an agent. Roles have constructs to depict concurrent or parallel behaviors. They act in parallel and communicate and synchronize through interactions. RAD is essentially a state-based modeling technique and actions and interactions of a role move it from the current state to a new state.

The Workflow Intelligent Business Object (WIBO) approach can model and implement a workflow with intelligence, autonomy, collaboration and cooperation (Fakas & Karakostas, 1999). The workflow is modeled as a collection of interacting business objects that are able to manage themselves. These objects belong to meta-level classes or types: process, role, actor and resources. eXchangeable Routing Language (XRL), based on XML, is a language developed to support seamless routing of inter-organizational workflow (Aalst & Akhil, 2003). Most workflow modeling techniques lack capabilities to capture different aspects of a workflow system such as data, process and organization (resources) in a single consistent view (Bajaj & Ram, 2002). In (Basu & Blanning, 2000) was proposed a framework that integrates the informational, functional, and organizational perspectives of the workflow in a single model and in (Bajaj & Ram, 2002) was proposed State-Entity-Activity-Model using set theory.

A major problem is that of the ontology in every domain (Kishore et al., 2004). In (Fox et al., 1998) was proposed an organizational ontology for enterprise modeling. An organization is a set of constraints on the activities performed by organizational agents. An agent plays one or more roles. Each role is defined with a set of goals that the role is created to fulfill and is allocated with proper authority at the level that the role can achieve its goals. Agents perform activities in the organization, each of which may consume resources and there is a set of constraints that constrain agent activities. An agent can also be a member of a team set up in response to a special task, possesses skills and has a set of communication links that specify the other agents in the organization with who it can communicate.

In the IBIS model presented in (Kishore et al., 2006), the business enterprise has a number of work systems each of which has been assigned certain goals to achieve that are derived from enterprise goals. The business enterprise can be a single business organization or it can be an extended, networked, or virtual organization or a strategic alliance consisting in a number of interacting partner firms who wish to integrate some of their work systems. Each work system contains multiple actors who are assigned sub-goals derived from work system goals, and who perform a number of tasks using some resources to achieve their goals, and in the process use or create some information objects. Accomplishment of actor goals contributes to the accomplishment of work system goals, which in turn contributes to the accomplishment of enterprise goals.

The above models and constructs include goals, roles, agents (actors), activities or tasks as components of processes, interactions, workflow, resources, data, and interdependencies. As was stated in (Kishore et al., 2006), the MAS paradigm provides an approach and suitable mechanisms for developing integrative business information systems to achieve the goal of creating an integrated enterprise.

In (Kishore et al., 2006), was proposed a conceptual framework for multi-agent-based integrative business information systems; was identified a minimal set of orthogonal ontological constructs in (Kishore et al., 2003) and (Sowa, 2000) that are central to the multi-agent-based integrative business information systems bounded discourse universe. In (Kishore et al., 2004) was given an approach of minimal ontological commitment and tried

to identify those ontological constructs that are essential for conceptual analysis and modeling of multi-agent-based integrative business information systems.

4. Workflow modeling

The commercial community defined and standardized workflow languages that at least hold the potential of interoperability. Typical workflow scenarios include document lifecycle management, internal application workflow and business process management. A number of engines have been developed which revolve around well established protocols such as BPEL, BPEL4WS, XLANG, WSFL and many others. The Figure 1 illustrates the considerations based on the meta-model defined in (WfMC, 1999).

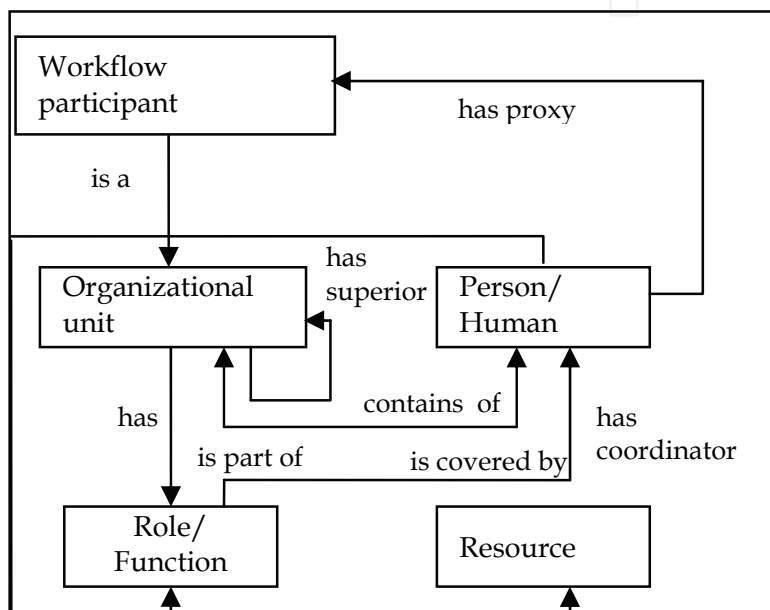


Fig. 1. Process instance state transitions meta-model

The activities that compose the process/workflow instances can gain various states and change each other during process instance. Activity state evolves as is illustrated in Fig. 2 (WfMC, 1999).

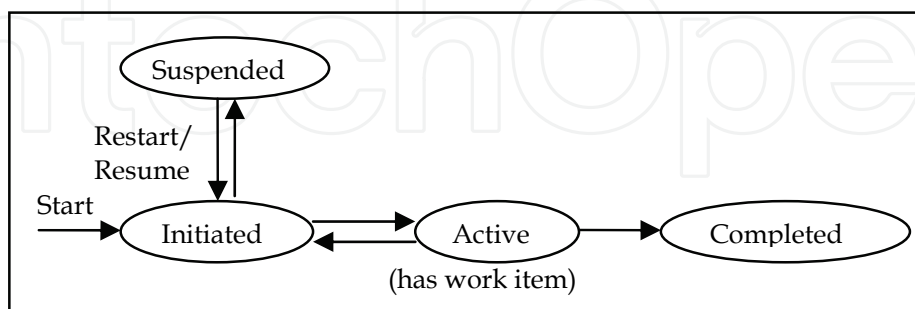


Fig. 2. Activity instance state transitions meta-model

In (Papazoglu, 2008) the Web Services are considered to be mandatory in conjunction with the specific languages (like BPEL4WS) based on numerous business process challenges such as coordinating communication between services, correlating message exchanges between parties, implementing parallel processing of activities, transforming data between partner

interactions, supporting long running business transaction and providing consistent exception handling. These languages use many protocols and standards that allow representing the Business process management both inside and between the organizations, as is shown in Fig. 3. Web Services are frequently used in Workflow design in order to allow an expressive dynamic of the real processes (Papazoglu, 2008). Web services are not implemented in a monolithic manner, but rather represent a collection of several related technologies illustrated in Fig. 3.

Management	Choreography			Business Processes
	Orchestration			
	WS-Reliability	WS-Security	Transaction	Quality of Service
			Coordination	
			Context	
	UDDI			Discovery
	WSDL			Description
	SOAP			Message
	XML			
	HTTP, IIOP, JMS, SMTP			Transport

Fig. 3. Standards used for Business Process Management

Using Enterprise Application Integration technologies (EAI) companies share their information in distributed information systems and try to automate their Workflow Management Systems (WfMSs). Workflows are commonly modeled as activity diagrams. This approach to workflow modeling is called Activity-Based Workflow Modeling (AWM). An activity diagram comprises a set of discrete actions and the relationships among them.

One fundamental characteristic of activity diagrams is that all actions are treated equal regardless of real differences in their nature. For example, some actions are user-initiated while others may be system-generated notifications or triggers. Yet other actions may be business rule executions, often referred to in the AWM literature as automatic actions.

Another approach to workflow modeling is Contextual Workflow Modeling (CWM) (Nguyen, 2009). CWM is centered on a modified finite state machine running strictly in the context of a data environment. The data environment isolates the state machine such that the only way to effect a change in the running status of the state machine is through modifications to the data environment. Instead of focusing on the actions as in the case of AWM, CWM is primarily concerned with how the outcome of an action impacts the data environment. CWM is a rule-based system where all workflow events, ranging from state activation to participant assignments, are driven by user-defined business rules. These rules operate on data drawn from the data environment and form the bridge connecting the data environment to the state machine. On the other hand, the combination of the current workflow status and the participant roles determines who has the privilege and responsibility to modify which part of the data environment. The CWM approach provides a uniform mechanism through which external actors, whether human users or information systems, interact with the workflow model. There is no artificial distinction of user actions versus system actions in how they impact the data environment, and consequently the workflow status. As a result, integration and interchangeability of the different actor types

are much more seamless than in the AWM approach. CWM represents a departure from the typical AWM approach. By switching the modeling focus from actions to states and their driving business rules and by deriving action requirements dynamically from the workflow data content, CWM puts tremendous expressive power in the hands of the business process architect, allowing him to capture nuances that are simply beyond the reach of AWM. Workflow models in CWM tend to be streamlined, with the different action types organized logically according to their semantics. The semantic structure of CWM enhances comprehension of complex processes by giving analysts a top-level overview of a workflow and allowing them to drill down into particular areas of the workflow as necessary. Workflow systems are driven by the process models, which describe the workflow process. Workflow patterns were deeply analyzed and some main features are given in (van der Aalst & ter Hofstede, 2002). Adaptive workflows have been discussed for many years and many people have described what should be done (van der Aalst, 2001). Were proposed techniques to manage adaptability and only a small number of actual implementations have been made that tackle some aspects of adaptability. Transferring running work cases to a new model is still a difficult issue. Some adaptability can be provided by manual transfer of tokens in the new process model (van der Aalst & ter Hofstede, 2002). In (Purvis et al., 2004) was developed an agent-enhanced WfMS, where the work associated with running a WfMS has been partitioned among various collaborating agents that are interacting with each other by following standard agent communication protocols. The proposed model is developed in JBees and uses a CPN execution tool.

The architecture of proposed model from (Purvis et al., 2004) is illustrated in the Fig. 4.

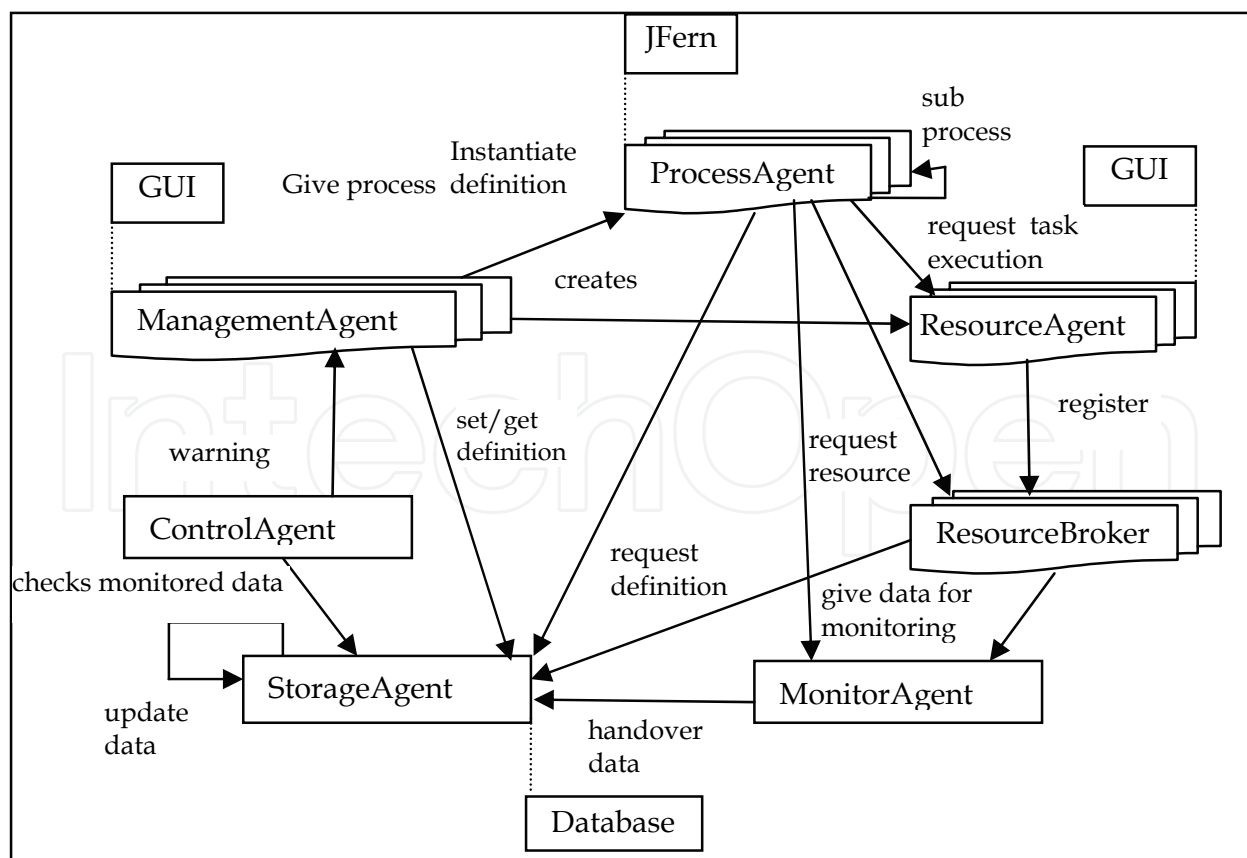


Fig. 4. The architecture of JBees

The flexibilities of workflow system enable to provide the support for distribution, adaptability, monitoring and controlling of processes. JBees supports the inter-organizational cooperation through the distribution of process.

5. Multi-agent systems

Multi-agent system can be considered a system that is composed of multiple autonomous agents and has the following characteristics presented in (Sowa, 2000), (Jennings et al., 1998) and (Aerts et al., 2002): (a) each agent has incomplete information or capabilities for solving the problem and, thus, has a limited viewpoint; (b) there is no global systems control; (c) data are decentralized; (d) computation is asynchronous.

An intelligent agent is autonomous, reactive, proactive, and social (Wooldridge, 2002). Agents are able to decide whether or not to execute an action after receiving requests (Yu & Mylopoulos, 1996).

In an agent-based system, agents use a common language with agent-independent semantics. There are different types of messages an agent can receive, and the agent is able to evaluate the received messages and decide what actions to perform. There are many agent communication languages between us mention KQML (Finin et al., 1995). Agent communication languages support several ways for agents to communicate, including direct communication, middle agent-mediated communication and broadcast.

Many coordination techniques have been developed in the intelligent agent and MAS literature. The coordination techniques include organizational structuring, metalevel information exchanging, multi-agent planning, contract net, negotiation, and market mechanisms depending on the domain where the agents are used. In (Kishore et al., 2006) is given a synthesis of developed technologies and their key constructs.

In (Yu & Schmid, 1999) is proposed a conceptual framework for agent-oriented and role-based workflow modeling. A workflow is viewed as a collection of agents interacting with others when they have interdependencies. The workflow is modeled a set of roles, which are further defined in terms of goals, qualifications, obligations, permissions and protocols. Interactions among roles are governed by protocols.

Roles are then assigned to agents based on the evaluation of qualification and capabilities. Once a role is assigned to an agent, the agent inherits the obligations and permissions specified in that role. Coordination of workflow is achieved by communication among agents. In (Sikora & Shaw, 1998) and (Sikora & Shaw, 2002) the enterprise information system is considered as consisting of multiple agents with different functionalities, and develop representational formalisms, coordination mechanisms, and control schemes necessary for integrating heterogeneous agents while meeting such performance criteria as overall effectiveness, efficiency, responsiveness and robustness. Based on a formal analysis, the same authors propose an agent-based framework for enterprise integration. In the framework, each process is modeled as an autonomous agent, which has its own well-defined goals and objectives. In (Jennings, 2001) and (Jennings, 2001) the business process is viewed as a community of negotiating agents. Was proposed the Advanced Decision Environment for Process Tasks (ADEPT) for conceptualizing, designing and implementing business process management systems.

Many of the following considerations are similar with those from (Kishore et al., 2006). In (Kishore et al., 2006), were presented conceptual similarities between integrated business enterprises and multi-agent systems in terms of their goal orientation (i.e., problem-solving

orientation). In an integrated enterprise, actors interact with each other in a number of work systems in a highly coordinated manner to achieve their own and shared goals, thereby contributing to the work system and thereby overall enterpriser goals.

A multi-agent system is similar to an integrated enterprise because it is essentially a community of autonomous and problem solving agents who interact in computer network systems in a coordinated manner to achieve their own and shared goals. Therefore, in terms of the goal construct, the MAS paradigm is suitable for modeling and developing integrated enterprise applications.

The MAS paradigm is suitable approach for modeling integrative business information systems because unlike any other paradigm, agents in the MAS paradigm come closest to mimicking the characteristics of human actors in business organizations: those of intelligence, adaptive problem solving behavior, sociability and autonomy. The actors within multiple work systems in an integrated enterprise need to be coordinated to manage and resolve interdependencies between tasks they perform and resources they use during the course of performance of their tasks. Interaction (or communication) among actors, and therefore among work systems, is a fundamental concept for integrating work systems as it is one of the basic means for coordinating activities of actors and for resolving resource dependencies in an integrated business enterprise.

Interaction is also a fundamental concept in the MAS paradigm as agents interact/communicate one another to share resources, check for task/goal accomplishment, check for availability of other agents for performing certain tasks, negotiate prices and timelines, subcontract tasks to other agents, decide upon future courses of action, etc. Activities (or tasks) are also fundamental to the MAS because both actors in human organizations and agents in multi-agent communities perform tasks in order to accomplish their own and organizational/system goals. The notion of workflow is also central to the MAS paradigms because a coordinated flow of activities (work) performed by various actors/agents and it is necessary for achieving organizational/system goals.

Information is obviously a key construct in MAS domains. A variety of information is required for actors in business enterprises to perform their assigned tasks in order accomplish their goals. Information exchanges also take place between actors within and across work systems in integrated business enterprises as part of their interactions to coordinate their activities and resources. The same is the case with multi-agent systems. Agents within MAS also use information to perform tasks and exchange information with other agents as part of their interactions. Therefore, from the perspective of interactions, activities, workflow and information, the MAS paradigm is quite appropriate for the modeling and development of integrated system applications.

The concept of resources is a very important construct in integrative business information systems. Resources within business organizations are always finite, and one of the key goals of any business organization is to optimize on resource requirement and utilization to contribute directly to the profitability goal of the organization. When resources are distributed across multiple work systems, improving resource utilization requires their effective coordination for solving resource dependencies and sharing resources across multiple work systems. Thus, resources of a wide variety (e.g., technical skills, documentation, plant and equipment, computer systems, utilities, consumable supplies, tools, etc.) need to be explicitly modeled in integrative business information systems. Resources are also a key construct in the MAS paradigm as agents contend for and consume finite computational resources such as memory, processor time, communication bandwidth,

etc. While the types of resources that agents in the MAS paradigm normally deal with are comparatively limited as compared to business enterprises, any type of physical or intellectual resource can be easily modeled using the MAS paradigm. Thus, the MAS paradigm appears to be suitable for modeling of integrated application systems from the perspective of resources as well.

The role has been used as a construct in many modeling techniques, for abstracting the tasks/activities that need to be performed by real human actors to achieve certain goals. These abstract roles are then assigned to real human actors who are expected to perform tasks (activities) included in these roles definition to achieve the goals the roles are designed to accomplish. While the MAS paradigm does not specifically deal with the notion of roles, it can be readily implemented by assigning abstract roles to intelligent agents within the MAS which are similar in a number of ways to human agents. Thus, from the perspective of roles as well, the MAS paradigm provides a suitable approach for modeling and implementation of integrated system applications. Also in (Kishore et al., 2006), was presented multi-agent-based integrative business information systems (MIBIS) and for it were proposed the following ontological constructs required for modeling systems in the MIBIS discourse universe: agent, role, goal, interaction, task, resource, information and knowledge. Every ontological construct was analyzed and motivated. Based on the above considerations, the MAS systems paradigm is suited for modeling and implementing a wide variety of integrative business information systems and also for workflow execution and modeling.

6. Agents and Roles in design of BPs and Wfs

The agent technology has been used in different ways. From a conceptual perspective workflows can be enhanced through the agent concept.

Agents offer a natural way to deal with open environments and are therefore of particular benefit for distributed systems. In some cases the agents fulfill particular roles that are required by different tasks in the workflow. In these cases the existing workflow is used to structure the coordination of these agents (Jennings et al., 2000), (Nissen, 2000).

An example of this approach is the work by M. Nissen in designing a set of agents to perform activities associated with the supply chain process in the area of e-commerce (Nissen, 2000). In other cases, the agents have been used as part of the infrastructure associated with the WfMS itself in order to create an agent-enhanced WfMS (Wang & Wang, 2002), (Stormer, 2001). These agents provide an open system with loosely coupled components, which provides more flexibility than the traditional systems. Some researchers have combined both of these approaches.

In (Purvis et al., 2004), an agent-based WfMS is used in conjunction with specialized agents that provide appropriate application-related services. We have taken the latter approach which provides sufficient flexibility required for a dynamic and adaptive system.

In (Reese et al., 2005) was developed a model that fragments the workflows for distributed execution with supporting protocols and architecture. This architecture is agent orientated, its formal basis is provided by Petri nets with a specific tool support. It allows the splitting of workflows into arbitrary fragments. These fragments, encapsulated by agents, are treated again as workflows and they can be executed at different locations using different workflow enactment systems, which are the conceptual platforms of the agents. Is provided a concept for a distributed and concurrent workflow management system, based on the FIPA compliant agent framework CAPA. The architecture is sketched in the Fig. 5.

Many specific techniques as input-process-output techniques, conversation-based techniques, and techniques based on role modeling, system thinking and system dynamics techniques, and constraint based representations techniques are used in BP modeling. Among these techniques, those based on role modeling have the advantage of supporting the well-known separation of duties principle (SoD) (Saidani & Nurcan, 2006).

The concept of role not only allows underlining the responsibility of each actor and reflects the organizational structure but also improves the understanding of the way responsibilities are achieved. Adopting role based methods to model BPs is useful, particularly if they are flexible enough to meet BP flexibility requirements, especially organizational, functional and operational requirements (Saidani & Nurcan, 2006).

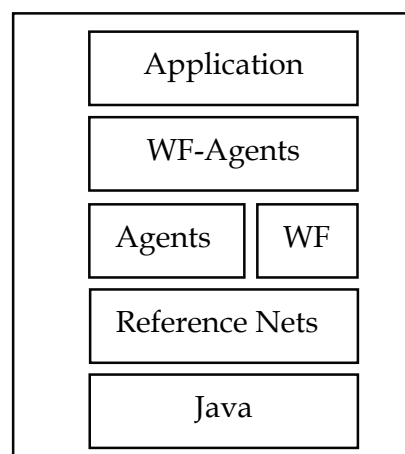


Fig. 5. The architecture of model proposed in (Reese et al., 2005)

The flexibility can be defined as the capacity of making a compromise between, first, satisfying, rapidly and easily, the business requirements in terms of adaptability when organizational, functional and/or operational changes occur; and, second, keeping effectiveness.

One of the main objectives of companies is to better and more quickly meet with the customer's requirements. In a changing environment, assuming that participants will always act as predefined is inaccurate, because it limits their autonomy and flexibility when changes make inapplicable some predefined conditions. The delegation concept is used in order to fulfill these requirements. Delegation is often defined as a substitution mechanism of all or a subset of the actor roles to one or more other actors. No actor can delegate a piece of role. However, in many cases an actor may want to delegate some missions from his/her role. What is more, in some cases, role-to-role delegation is needed. A flexible delegation model requires multiple forms of delegation, and supports flexible role, mission and operational goal level delegation.

The purpose of the separation of duties (SoD) is a policy to ensure that failures of omission or commission within an organization are caused only by collusion among individuals and, therefore, are riskier and less likely, and that chances of collusion are minimized by assigning individuals of different skills or divergent interests to separate tasks (Gligor et al., 1998).

In (Saidani & Nurcan, 2006) is detailed a model that allows to emphasize the flexibility in BP modeling and that will be sketched in the followings. The mission from (Saidani & Nurcan, 2006) is in our view the same as the task. Organizations can be viewed as being structured as networks of BPs in order to achieve their business goals. BP can be first analyzed in term of roles played by actors and holding tasks (missions). During the execution of a BP, actors

perform tasks (missions) that specify the responsibilities and the work included in swim-lanes in classical activity-oriented representation formalisms. During the execution of a BP, it is an actor who performs operations. Organization's roles and tasks (missions) are usually more static than actors and operations are. The central concepts are the role and the mission. A role is a semantic construct about which business rules and other concepts can be formulated. It can represent competency to realize particular missions, an engineer, and can embody authority and responsibility, a project supervisor.

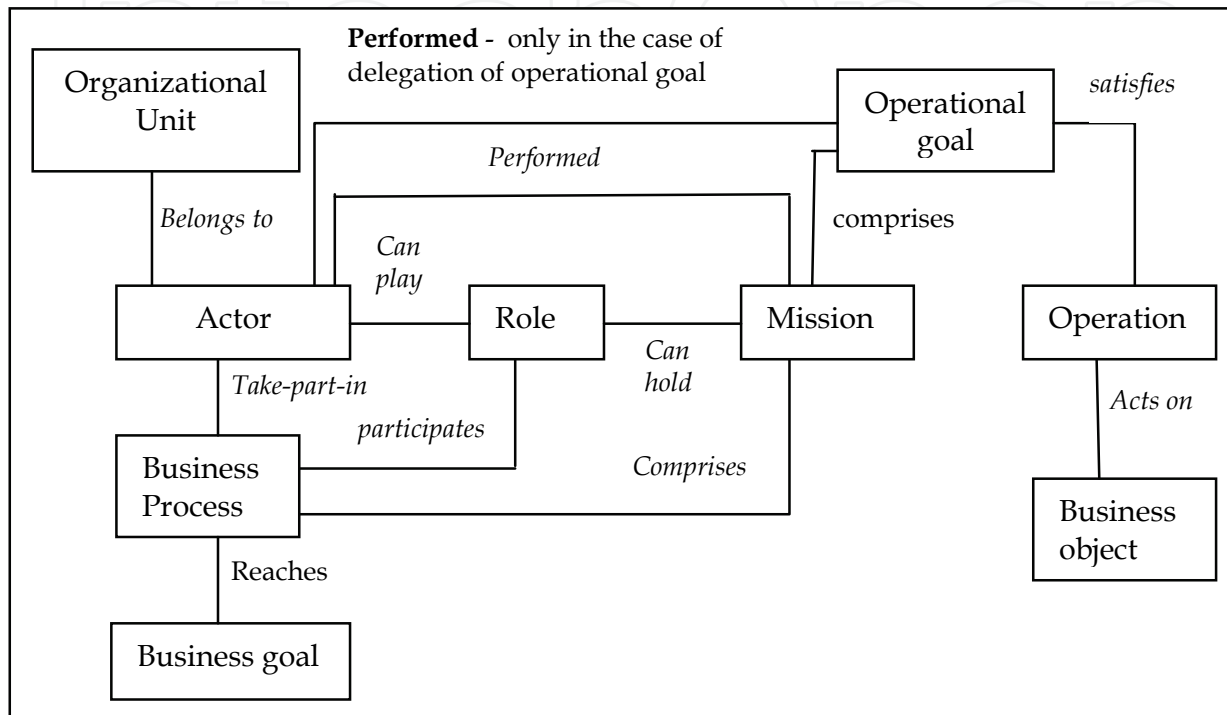


Fig. 6. The proposed model from (Saidani & Nurcan, 2006)

In Fig. 6 each actor belongs to at least one organizational unit and is assigned to appropriate roles based on his responsibilities and qualifications. The concept of task (mission) serves as a link between roles and operations: A task (mission) is defined as a collection of operational goals satisfied by achieving operations. A task (mission) can comprise several operational goals because it is not achieved performing straightforward and continuous operations without any interaction with other roles. The set of operations allowing a role (played by an actor during the process occurrence) to achieve an operational goal can be specified as an activity. The specificities of the model consist in the followings: (i) is defined the piece of responsibility of a role in the intentional level (operational goal), (ii) goes deeply in the specification of this operational goal (dealt with as a black box in usual workflow formalisms), and finally, (iii) specifies the operations which performance acts on the business objects and allows achieving the operational goal.

Constraints on the relationships defining flexibility and Constraints in delegation model are representative in flexibility management. In order to deal effectively with the BP flexibility, relationships between the concepts (Fig. 7) should be controlled to ensure the usability of the provided flexibility mechanism. The figure Fig. 7 illustrates the model.

As a major constraint on the relationships between the concepts is the constraint controlling separation of duties: separation of duties is a business technique trying to minimize fraud by

dispersing the authority and responsibility for an action over multiples actors. This can be ensured by defining mutually disjoint actor-to-role assignments with respect to sets of roles.

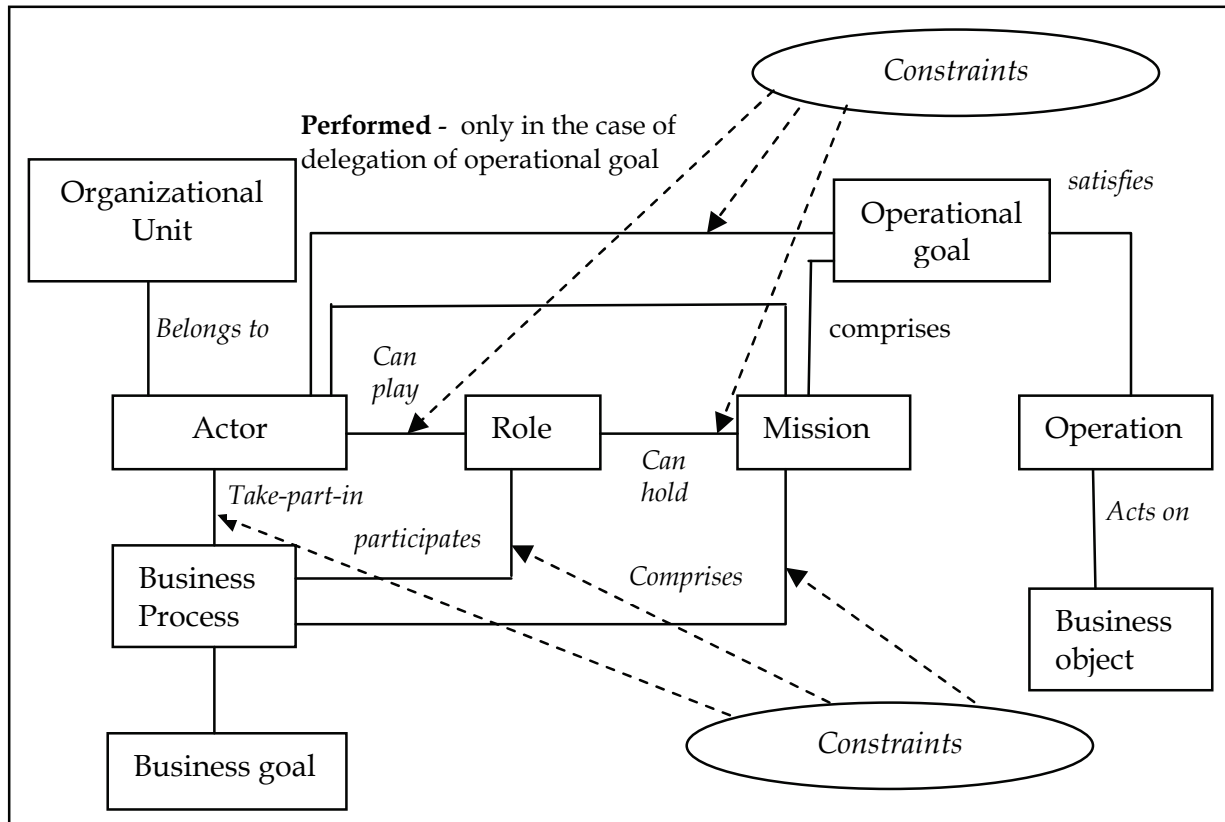


Fig. 7. Constraints for the control mechanism

Constraints can apply to actor-to-role, mission-to-role and operational-goal-omission assignments. They can limit, for instance, the number of members or missions of a role or the number of operational goals for a mission. Constraints can also apply to processes, and to talked-part-in and participates relationships associated with a BP. Constraints on BPs can limit the number of occurrences of a BP in which an actor can realize missions simultaneously. Constraints can precise if an actor or a role may participate to multiple business processes at the same time.

7. Reengineering failed workflows

In (Cicortas et al., 2006) and (Cicortas & Jordan, 2009a) was proposed a model that is based on a multi-agent system. The workflow and its constituents are preserved in a data base. Workflows especially for business processes are generally in a very high dynamic. So during the execution the state of the system and also of the workflow can change, or can fail. Whereas the system state is stable the workflow can evolve but while in some state the system changes in an unpredictable way the workflow will encounter particular cases (states) that were not developed. In such cases the workflow must be able to adapt to the new conditions. During the execution of an activity, of a workflow instance, a failure of the activity can appear. In this case the workflow will be stopped and its reply can be done, in the actual circumstances, or can be done only from its beginning.

As was stated in (Cicortas et al., 2006) we propose a model that allows using a multi-agent system that will be able to manage the failures in the execution of the workflow instances. For being able to do that, we need to develop some specific concepts and use an appropriate data bases. In (Cicortas et al., 2006) were given details that concern the activities and their results that eventually can be reused in the case of the reply of the workflow instance execution. For every activity it is needed some specific information that will be used in the case of a reply of the instance execution, after a failure. The information concerning the activity is placed into the Workflow Data Base (WfDB). The entity in this data base is the activity. Also for every workflow instance a specific data base it is needed, Workflow Instance Data Base (WfIDB), here the entity is the workflow instance. Due to the space of paper limitations some of the basic attributes of entities from the previous data bases are not given. The relation between the entities of WfDB and WfIDB can be easy imagined.

The Fig. 8 shows the proposed databases and the interaction of the agents with these data bases (Cicortas & Jordan, 2009a).

The multi-agent system (MAS) that is able to analyze the workflow instance that was blocked and based on its state to redesign the reminded part of the workflow instance that will be executed. This analysis must decide for every activity that its result can be or not can be used in the conditions that allow it, if we dispose for appropriate information. In a workflow, we have some activities whose result is in one of the following categories: the activity result can be retained and in future can be used, we can say it is "persistent"; the activity result is volatile in future, it does not be used, if we want the activity result in the future, we must replay it; the activity result has affected the real system state and the system state must be restored in the case if we want to replay this activity we must remove this "effect". If the workflow fails, then it fails due some activity that it failed.

The previous Workflow activities were executed and based on their result as previous analysis was done, we can find the activities whose results are reusable and are persistent. In the case of the replay of the Workflow we must make an analysis that: restores the activities that were affected in some way the system state, find the activities that have reusable and persistent results, reconstruct a new Workflow with the remained activities and reusable results. (The problem is that on every parallel branch we must have a reusable result). In the following we propose the steps that will allow constructing and using our proposed model:

- a. For every activity the designer must decide if it has remanent result and the way that allow identifying and storing it;
- b. For all activities it will specify if its result is volatile or the effect of the activity on the overall system, in the sense that it must be discarded (example: updating data bases with the cancellation of the activity effects).
- c. During the execution of a workflow instance will be recorded in a data base, the activities (their results if there is in the case of the remanence) as soon as these are executed.
- d. When a workflow instance fails due of an activity failure, then an analysis is done and it concerns the effects of activities on the system state; the last activities with remanent results will be retained. A mechanism for finding on every branch these remanent activities is required.
- e. After the previous step when the workflow instance must be replied, a new variant of the workflow instance is constructed as follows: as input locations that enter in a transition are the locations of activities with remanent results (as it was given in (Cicortas et al. 2006)).
- f. The new workflow instance that was constructed in the previous step can be started.

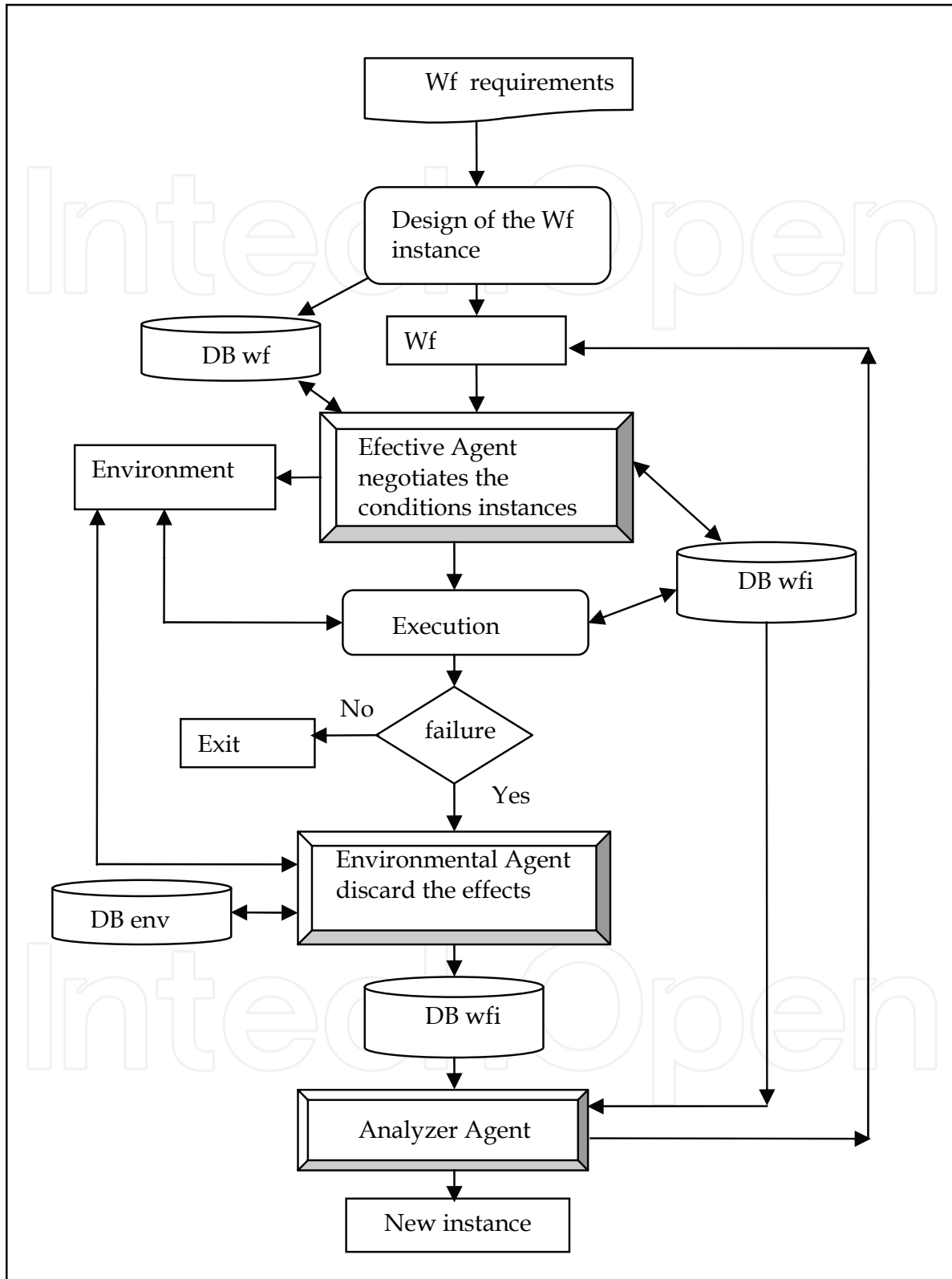


Fig. 8. Interaction of the agents with data bases

The agents of the proposed MAS act after the workflow was described as it was previous stated (i.e. a workflow description in that the activities were described).

In the case of a workflow instance failure the Analyzer Agent, make the analysis as follows: (a) all the activities on the every parallel branch are specified and their results are available; (b) for all the activities that affect the overall system state prepares their cancellation actions; (c) redesigns the new workflow instance.

The Environmental Agent executes the cancellation activities. The Effective Agent negotiates the conditions for the execution of the new instance of the workflow and launches it in execution. Decision for modelling is based on process complexity, failure of activities that can fail only the activity and “suspend” the process or fail all the process. The problem is to identify the reply of the process and its implications. Done to the process complexity the solution can be: restart the process from the beginning or reply the process from one of the previous “valid” states before its failure under some conditions. Needs for replies in a failure case are: (a) every activity must have specified its type concerning its result (persistent, affect the environment, insignificant); (b) during the execution every activity recalls somewhere (in a data base) its result and this result can be used depending of activity type in the future (if the activity is of reusable type). (Cicortas & Jordan, 2009b).

The Fig. 9 shows in short our proposal (Cicortas & Jordan, 2009a).

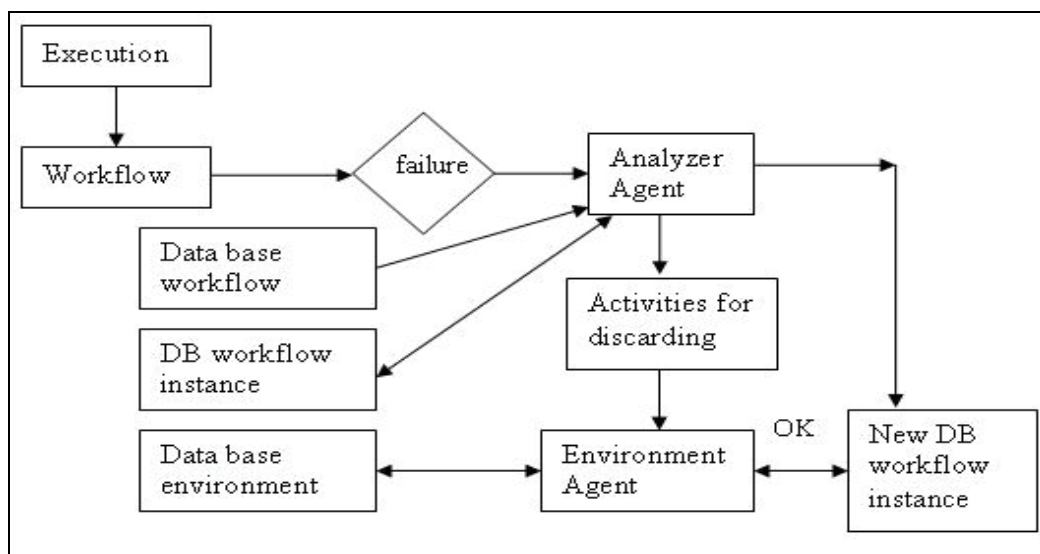


Fig. 9. The flow of actions

8. Conclusions

As was stated in the above considerations, we consider that as a main goal that argues and motivates the usage of MAS and the ontology constructs, are the following:

- the dependency modeling, that supports goal dependencies, task dependencies, and resource dependencies. Here we outline that the same role must execute many tasks and the role can be played by the same agent and at one time instant and we dispose only for one such agent (i.e., many tasks need in the same time instant the same role that can be played by the same agent). Some constraints concern the dependencies between the orders of tasks execution in this case. Here an intelligent analysis must be done and it justify the multi-agent concept;
- resource dependencies within and among roles and tasks can also be similarly captured and represented using the knowledge construct. The knowledge construct specifies

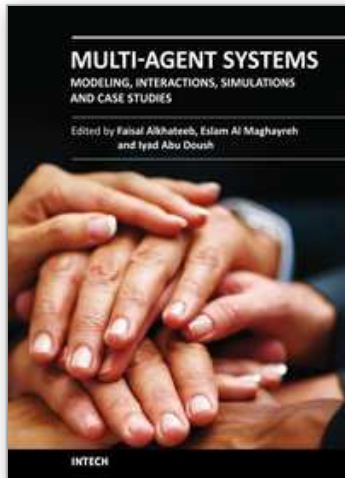
- which task performed by a particular role requires what resources. The knowledge construct can also be used to model the contention of resources and availability of resources can be signaled using messages and this message flow information (e.g., what message should be expected and from whom when a required resource becomes available) can be modeled using the knowledge construct;
- c. the concurrency is mandatory and it is one of the main requirements in the area. The agents that will be used act in an independent manner and in the same time they must cooperate in order to attain their goals and the system objectives;
 - d. new features are needed for the workflow engines that will allow the deal with the process dynamic and with the flexibility.

9. References

- van der Aalst, W.M.P. (2001). Exterminating the Dynamic Change Bug: A Concrete Approach to Support Workflow Change, *Information Systems Frontiers*, 3(3), pp. 297-317, (2001).
- van der Aalst, W.M.P. & ter Hofstede, A.H.M. (2002). Workflow Patterns: On the Expressive Power of (Petri-net-based) Workflow Languages. In: Kurt Jensen (Ed.): Proc. of the Fourth International Workshop on Practical Use of Coloured Petri Nets and the CPN Tools, Aarhus, Denmark, August 28-30, 2002, pages 1-20. Technical Report DAIMI PB-560, August 2002.
- van der Aalst, W.M.P. & Akhil, K. (2003). Xml-based schema definition for support of interorganizational workflow, *Information Systems Research*, Vol.14 (1), 2003, pp. 23-46.
- Aerts, A.; Szirbik, N. & Goossenaerts, J. (2002). The flexible agent based infrastructure for virtual enterprises, *International Journal of Computers in Industry*, Vol. 49, No.3, 2002, pp. 311-328.
- Bajaj, A. & Ram, S. (2002). Seam: a state-entity-activity-model for a well-defined workflow development methodology, *IEEE Transactions on Knowledge and Data Engineering*, Vol. 14, No.2, 2002, pp. 415- 431.
- Basu, A. & Blanning, R.W. (2000). A formal approach to workflow analysis, *Information Systems Research*, Vol. 11(1), 2000, pp. 17-36.
- Cicortas, Al.; Iordan, V.; Fortis, Al. & Fortis, F. (2006). Reengineering the Failed Workflows, *Annals of the Tiberiu Popoviciu Seminar, Supplement international Workshop in Collaborative Systems*, 2006, Vol. 4, ISSN 1584-4536, pp. 57-66.
- Cicortas, Al. & Iordan, V. (2009a). Multi-agent models in workflow design, Proceedings of 5th International Symposium on Applied Computational Intelligence and Informatics SACI'09, Timisoara, Romania, pp. 527-532, ISBN 978-1-4244-4478-6, IEEE CFP0945C-CDR
- Cicortas, Al. & Iordan, V. (2009b). Considerations on the Requirements for WFMS, Proceedings of the CSCS-17, 17-th International Conference on Control Systems and Computer Science, MASTS 2009, The International Workshop on Multi-Agent Systems Technology and Semantics, 26-29 May, Bucharest, Vol. 2, Ed. Politehnica Press, ISSN: 2066-4451, pp. 553-558.
- Fakas, G. & Karakostas, B. (1999). A workflow management system based on intelligent collaborative objects, *Information and Software Technology*, Vol.41(13), 1999, pp. 907-915.

- Finin, T.; Labrou, Y. & Mayfield, J. (1995). *Kqml as an agent communication language*, in: Jeff Bradshaw (Ed.), *Software Agents*, MIT Press, Cambridge, 1995.
- Fischer L. (2000). *The Workflow Handbook 2001*, Association with the Workflow Management Coalition (WfMC), 2000.
- Fox, M.S.; Barbuceanu, M.; Gruninger, M. & Lin, J. (1998). An organization ontology for enterprise modeling, in: M. Prietula, K. Carley, L. Gasser (Eds.), *Simulating Organizations: Computational Models of Institutions and Groups*, AAAI/MIT Press, Menlo Park CA, 1998, pp. 131–152.
- Gligor, V.; Gavrilă, S., & Ferraiolo, D. (1998). On the formal definition of separation-of-duty policies and their composition. In *Proceedings of the 1998 IEEE Computer Society Symposium on Research in Security and Privacy* (Oakland, CA, May), IEEE Computer Society Press, Los Alamitos, CA, 172–183.
- Huhns, M.N. & Stephens, L.M. (2001). Automating supply chains, *IEEE Internet Computing*, Vol. 5 (4), 2001, (Jul/Aug), pp. 90–93.
- Jennings, N.R.; Faratin, P.; Johnson, M.J.; Norman, T.J.; O'Brien, P. & Wiegand, M.E. (1996) Agent-based business process management, *International Journal of Cooperative Information Systems*, Vol. 2 and 3, 1996, pp. 105– 130.
- Jennings, N.R.; Sycara, K. & Wooldridge, M. (1998). A roadmap of agent research and development, *Autonomous Agents and Multi-Agent System*, Vol. 1 (1), 1998, pp. 7– 38.
- Jennings, N.R.; Norman, T.J.; Faratin, P.; O'Brien, P. & Odgers, B. (2000). Autonomous agents for business process management, *Journal of Applied Artificial Intelligence*, Vol. 14 (2), 2000, pp. 145– 189.
- Jennings, N.R. (2001). An agent-based approach for building complex software systems, *Communications of the ACM*, Vol. 44 (4), 2001, pp. 35– 41.
- Kishore, R.; Ramesh, R. & Sharman, R. (2003). Computational ontologies: foundations, representations, and methods, in *Proceedings of Ninth Americas Conference on Information Systems* (9th AMCIS), Association for Information Systems, Tampa, FL, 2003, pp. 3178– 3189.
- Kishore, R.; Zhang, H.; Ramesh, R. & Helix, A. (2004). Spindle Model for Ontological Engineering, *Communications of the ACM*, Vol. 47 (2), Feb. 2004. pp. 69–75.
- Kishore, R.; Zhang, H. & Ramesh, R. (2006). Enterprise integration using the agent paradigm: foundations of multi-agent-based integrative business information systems, *Decision Support Systems*, Vol. 42, 2006, pp. 48– 78.
- Linthicum, D.S. (1999). *Enterprise Application Integration*, Addison-Wesley, Boston, MA, 1999.
- Malone, T.W. & Crowston, K. (1994). The interdisciplinary study of coordination, *ACM Computing Surveys*, Vol. 26 (1), 1994, pp. 87–119.
- Nissen, M.E. (2000). Supply Chain Process and Agent Design for E-Commerce, In 33rd Hawaii International Conference on System Sciences, 2000.
- Nguyen, V. (2009). Contextual Workflow Modeling, <http://www.macronetics.com/docs/CWM%20Whitepaper.pdf>, accessed 10.03.09.
- Ould, M.A. (1995). *Business Processes-Modeling and Analysis for Re-Engineering and Improvement*, Wiley, Chichester, 1995.
- Pan, J.Y.C. & Tenenbaum, J.M. (1991). An intelligent agent framework for enterprise integration, *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 21 (6), 1991, pp. 1391–1408.

- Papazoglu, M. P. (2008). *Web Services: Principles and Technology*, Prentice Hall, ISBN 978-0-321-15555-9, 2008.
- Purvis M.; Savarimuthu, B.T.R. & Purvis, M. (2004). Evaluation of a multi-agent based workflow management system modeled using Coloured Petri Nets, In M. Barley and N. K. Kasabov, editors, PRIMA, volume 3371 of LNCS, pp. 206–216. Springer, 2004.
- Reese, C.; Ortmann, J.; Moldt, D.; Offermann, S.; Lehmann, K. & Carl, T. (2005). Architecture for Distributed Agent-Based Workflows. In: Proceedings of the Seventh International Bi-Conference Workshop on Agent-Oriented Information Systems (AOIS-2005). 2005; pp. 42–49.
- Sadiq, S.; Orłowska, M.; Sadiq, W. & Foulger, C. (2004). Data Flow and Validation in Workflow Modeling, *ACM International Conference Proceeding, Series; Vol. 52* archive, Proceedings of the 15th Australasian database conference, 2004, pp. 207–214.
- Saidani O. & Nurcan S. (2006). A Role-Based Approach for Modelling Flexible Business Processes, *BPMDs*, 2006, pp. 111-120.
- Sikora, R. & Shaw, M. (1998). A multi-agent framework for the coordination and integration of information systems, *Management Science*, Vol. 44 (11), 1998, pp. 65–78.
- Sikora, R. & Shaw, M. (2002). Multi agent enterprise modeling, in: C. Holsapple, V. Jacob, H.R. Rao (Eds.), *Business Modeling: A Multidisciplinary Approach Essays in Honor of Andrew B. Whinston*, Kluwer Academic Press, 2002, pp. 169–185.
- Sowa, J.F. (2000). *Ontology, Knowledge Representation: Logical, Philosophical and Computational Foundations*, Brooks/Cole, New York, NY, 2000, pp. 51–131.
- Sycara, K.P. (1998), Multiagent Systems, *AI Magazine*, Vol. 19 (2), 1998, pp. 79– 93.
- Stormer, H. (2001). AWA - A flexible Agent-Workflow System , In Workshop on Agent-Based Approaches to B2B at the Fifth International Conference on Autonomous Agents (AGENTS 2001), Montreal, Canada, 2001
- Tillquist, J.; King, J.L. & Woo, C. (2002). A representational scheme for analyzing information technology and organizational dependency, *MIS Quarterly* , Vol. 26 (2), 2002, pp. 91–118.
- Wang, M. & Wang, H. (2002). Intelligent Agent Supported Flexible Workflow Monitoring System, In *Advanced Information Systems Engineering: 14th International Conference, CAiSE 2002*, Toronto, Canada, 2002.
- WfMC, (1999). *Workflow management coalition terminology & glossary. Workflow Management Coalition. Document status*, Issue 3.0, 1999.
- Wooldridge, M. (2002). *An Introduction to Multiagent Systems*, John Wiley and Sons, West Sussex, England, 2002.
- Yu, E.S.K. & Mylopoulos, J. (1996). Using goals, rules and methods to support reasoning in business process reengineering, *International Journal of Intelligent Systems in Accounting, Finance & Management*, Vol. 5 (1), 1996, pp. 1– 13.
- Yu, L. & Schmid, B.F. (1999). A conceptual framework for agent oriented and role based workflow modeling, *Proceedings of International Bi-Conference Workshop on Agent-Oriented Information Systems*, Heidelberg, Germany, 1999.



Multi-Agent Systems - Modeling, Interactions, Simulations and Case Studies

Edited by Dr. Faisal Alkhateeb

ISBN 978-953-307-176-3

Hard cover, 502 pages

Publisher InTech

Published online 01, April, 2011

Published in print edition April, 2011

A multi-agent system (MAS) is a system composed of multiple interacting intelligent agents. Multi-agent systems can be used to solve problems which are difficult or impossible for an individual agent or monolithic system to solve. Agent systems are open and extensible systems that allow for the deployment of autonomous and proactive software components. Multi-agent systems have been brought up and used in several application domains.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Victoria Jordan (2011). Multi-Agent Models in Workflow Design, Multi-Agent Systems - Modeling, Interactions, Simulations and Case Studies, Dr. Faisal Alkhateeb (Ed.), ISBN: 978-953-307-176-3, InTech, Available from: <http://www.intechopen.com/books/multi-agent-systems-modeling-interactions-simulations-and-case-studies/multi-agent-models-in-workflow-design>

INTECH
open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2011 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](#), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen