

AN APPROACH TO INTER-ORGANIZATIONAL WORKFLOW MANAGEMENT IN AN ELECTRONIC INSTITUTION

Henrique Lopes Cardoso¹, Paulo Leitão², and Eugénio Oliveira¹

¹ *LIACC – NIAD&R, Faculty of Engineering, University of Porto
R. Dr. Roberto Frias, 4200-465 Porto, Portugal
{hlc, eco}@fe.up.pt*

² *Polytechnic Institute of Bragança,
Quinta Sta Apolónia, Apartado 1134, 5301-857 Bragança, Portugal
pleitao@ipb.pt*

Abstract: In a virtual organization, different business partners (individual organizations) cooperate in order to achieve a common goal. The coordination of the corresponding inter-organizational workflow is an important issue. In this paper an approach towards managing the operational embodiment of a manufacturing consortium is presented. This approach is conceptualized as a service within an Electronic Institution framework providing several agent-based services related with the formation and operation of virtual organizations. The behavior of the inter-organizational workflow management service is presented, modeled as an extension to a contract monitoring service. The paper also deals with the information exchange needs between these services and with the partners involved in a virtual organization contractual relationship.

1. INTRODUCTION

The main goal of the research project behind this paper is to develop an agent-based computational framework – embedded in an Electronic Institution (EI) – for supporting Virtual Organization (VO) composition and operation, by providing an appropriate set of services (Lopes Cardoso et al., 2005). We consider a VO as a temporary consortium of different organizations that “pool their resources to meet short-term objectives and exploit fast-changing market trends” (Davulcu et al., 1999). The EI is seen as a comprehensive framework that may effectively assist the lifecycle of a VO.

Institutional services include: assistance to the formation of VOs through negotiation mediation and contract templates; validation and registration of the agreed contracts (the EI acts as a trusted third-party); to monitor partners’ compliance to contractual commitments and impose correction measures.

Contracts representing the conjoint activity of a VO must take into account actions composing inter-organizational workflows that implement the desired cooperation. Also, in business relationships involving several different partners, the monitoring of contract-related activities is of paramount importance, in order

to respond in a timely fashion to unexpected events. Therefore, contract monitoring can evolve from a reactive perspective that waits for problems to arise, to a more proactive one where problems are anticipated and its potential damages avoided.

Manufacturing systems are complex, heterogeneous, stochastic and dynamic environments that can not be modeled as a linear function. They are complex being characterized by concurrency and parallelism, synchronization, resource sharing, asynchronous operations and non-preemptive operations, and heterogeneous comprising a variety of hardware and software applications. They are stochastic and dynamic environments, with certain resources becoming unavailable and additional resources being introduced at random times, new jobs arriving continuously to the system, new products being frequently defined and new regulations, such as quality and safety specifications, being regularly announced (Leitão and Restivo, 2005).

In this scenario, one crucial service provided by the platform is the supervision of the contract execution, demanding a close perception of what is going on at the shop-floor level, in an inter-organizational perspective. This service is addressed by the inter-organizational workflow management (IO-WfM)

agent. Inter-organizational workflow management is related to the cooperation of several distributed, autonomous and heterogeneous business processes, in order to achieve a common global goal (Cusati & DisENZA, 2000; Van der Aalst, 1999). That is, it comprises the gathering of business processes and the sharing of resources (information, human and machine).

The IO-WfM agent should also determine what happens to the inter-organizational workflow when the execution of a task is delayed or cancelled, possibly requiring a rescheduling of the planned workflow steps and a propagation of these adjustments to the dependent entities. When such situations arise, this service should provide feedback to the other components of the EI platform, allowing the exploitation of pertinent information that will be used in the future elaboration of contracts.

This paper intends to briefly describe the Electronic Institution platform and to focus on an approach to the inter-organizational workflow management agent, mainly discussing its behavior, interfaces and information exchanged.

The paper is organized as follows: first, section 2 introduces the Electronic Institution platform, presenting the main objectives and competences associated to each embedded component. Section 3 describes the behavioral model of the IO-WfM agent and section 4 describes its interfaces and relevant information supporting the inter-organizational workflow management. Finally, section 5 rounds up the paper with conclusions.

2. THE ELECTRONIC INSTITUTION PLATFORM

Human societies are governed by institutions providing services or regulating the way citizens interact. The same approach has been proposed as a means to regulate the interaction among software agents. The EI concept (Dignum and Dignum, 2001) represents the virtual counterpart of real-world institutions.

In our perspective, an EI is a comprehensive framework that provides a set of institutional services covering the formation and operation of VOs. One of the main roles of the EI is to provide trust by working as a third-party that enables partners to engage in (automated) business interactions. The provided services compose a coordination framework that assists the interaction of software agents representing different organizations or business units.

Fig. 1 depicts the main agent-based services provided by the EI. The figure omits typical services of an electronic market, such as white/yellow page support.

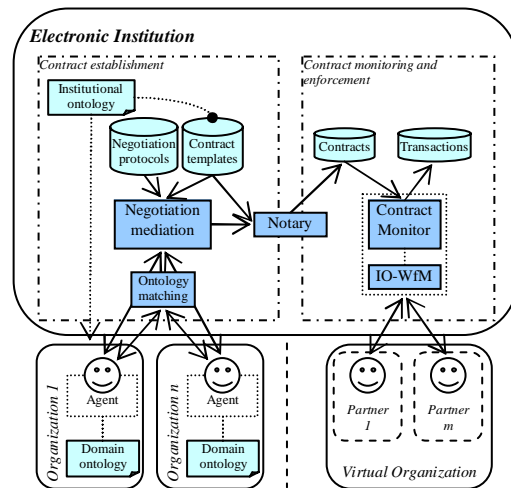


Fig. 1 – Services in an Electronic Institution (adapted from Lopes Cardoso *et al.*, 2005)

Negotiation mediation services are provided to assist the formation of a VO. This includes the utilization of appropriate negotiation protocols (such as Rocha *et al.*, 2005) and contract templates, which are instantiated with the outcome of the negotiation process.

When addressing open environments with no centralized design, it may well be the case that agents representing different organizations use different domain ontologies. In order to enable a meaningful negotiation, ontology matching services must be put into place (Malucelli *et al.*, 2005).

The mentioned services are used by different organizations, which can be seen as potential partners in a future VO. A subset of these, according to the outcome of a negotiation process, will become partners of a new VO.

Contracts resulting from successful negotiations are registered in the EI through a notary service, responsible for validating them according to institutional norms. The execution stage is assisted by providing services that monitor the carrying out of contractual commitments by each VO partner.

Although not shown in the figure, the EI is also responsible for the enforcement of contracts, by applying sanctions in case of non-compliance. Those sanctions may be predicted in the contract or institutionally defined. However, in order for such mechanisms to be effective, the EI also maintains reputation records that are updated with information about the performance of each contract participant.

The IO-WfM agent aims to supervise, in an inter-organizational perspective, the execution of the work-plan defined in each contract, interacting with the several partners responsible to execute the tasks. The main competences associated to the IO-WfM agent are: i) the synchronization of the inter-

organizational workflow, i.e. determine whether the process is ready to move to the next task, enforcing synchronization among individual workflows; ii) the re-direction of the workflow steps due to unexpected situations or unaccomplished tasks; and iii) to give feedback about the execution of the contract's work-plan in real factory plants, providing useful indications that may have a repercussion in redefining the content of future contracts.

In the next sections, the approach to the IO-WfM agent will be introduced and discussed.

3. MODELLING THE IO-WFM AGENT

The essence of contract is commitment: the assurance that others will, when the time comes, uphold their end of an agreement. Through contracts (sometimes called legally-binding agreements), parties expose themselves to legal sanctions for non-fulfillment of duties (Craswell, 2000). Contracts are composed of clauses that define each parties' obligations, together with associated sanctions in case of deviation. Typically, they also inherit default rules from legislative systems, which in our case are represented in an EI framework (Lopes Cardoso and Oliveira, 2005).

Monitoring and enforcing contracts means checking each partner's compliance to cooperation efforts, and acting accordingly. This approach consists of managing the relationships between different organizations at a contractual level. However, in some situations the business-value involved in an inter-organizational relationship is important enough to demand for a more pro-active approach. The management of such relationships can be pushed further to the business level, directly monitoring the activities that implement the cooperation as specified in the contract.

This paper focuses on the need to manage contractual relationships at the business level, addressing the creation and operation of an inter-organizational workflow. The management of the contract's workflow execution comprises mainly the creation of the work-plan, the supervision of the work-plan and the recovery from exceptions.

The functional and behavioral aspects of the IO-WfM agent are modeled using the Petri net formalism. Petri net is a formal modeling tool, graphical and mathematical, adequate to model, validate and verify the behavior of complex event-driven systems characterized as being concurrent, asynchronous and stochastic. In this work, the places represent the states of the system and a transition firing represents an activity. Two distinct types of transitions are considered (Colombo et al., 1997): *immediate transitions*, drawn by a thin bar, which fire in time

zero and model atomic activities, such as sending a message; and *timed transitions*, drawn by a thick bar, which have an associated delay time that specifies the amount of time that must elapse before the transition fires, modeling time consuming activities, such as the execution of an operation.

The behavior model of the IO-WfM agent is represented by the Petri net illustrated in Fig. 2.

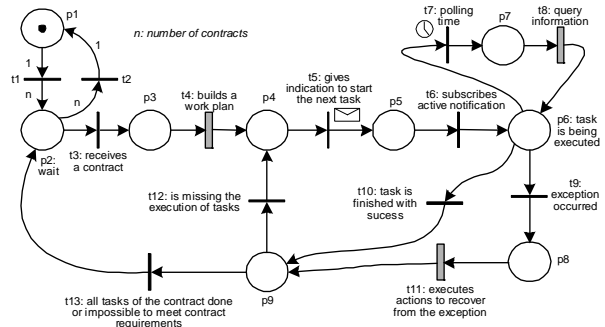


Fig. 2 – Behavior Petri net Model for the IO-WfM Agent

IO-WfM agent's goals are updated with the new contract workflow management request. The agent builds work-plans to achieve those goals. Those plans will be articulated with the different contractual partners involved, and may require revisions based on their execution progress. Meanwhile, the IO-WfM agent scans for new requests, being able to manage simultaneously the execution of several contract workflows. This way, the IO-WfM agent contains several threads of execution that are handled asynchronously in parallel, so that the execution of one thread doesn't block the execution of another thread. For example, it can execute simultaneously the monitoring of a contract workflow and the re-planning of another work-plan.

The first main function is related to building the work-plan from the information received about the contract from the Contract Monitor agent, modeled by the transition *t4*. For this purpose, an AND/OR based graph is built (Kruth et al., 1996) to represent the set of tasks and their precedence relationship. A node in the graph is one of the five different types: task, split-or, split-and, joint-or and joint-and. All paths following a *split-and* type node must be processed in any sequence, being necessary the execution of both paths. A *joint-and* type node is required to bring multiple paths back together after a *split-and* type node. Only one path following a *split-or* type node must be selected to execution, representing alternative operations. A *joint-or* type node is required to bring multiple paths together after a *split-or* type node.

During the construction of the work-plan, the IO-WfM agent elaborates a fine scheduling, adjusting the planned start and end dates of each task according to the real shop floor agenda of each partner.

After building the work-plan, the IO-WfM agent starts the synchronization of the inter-organizational workflow, by determining whether the process is ready to move to the next task, or not, according to the work-plan. The IO-WfM agent passes the control over the task to the partner entity responsible for its execution and waits for the end of execution. When the entity partner finishes with success the execution of the task (modeled by the firing of transition $t10$), the IO-WfM agent will determine the next task to be executed and notifies it to the correspondent entity partner.

If an exception that causes delays in the execution of the work-plan occurs (modeled by the firing of transition $t9$), the IO-WfM agent tries to recover from the disruption by implementing a set of actions, represented by the transition $t11$ and extended in the sub-Petri net model of Fig. 3. This workflow re-planning, in an inter-organizational perspective, aims to minimize the impact of the disruption, maintaining the terms defined in the contract.

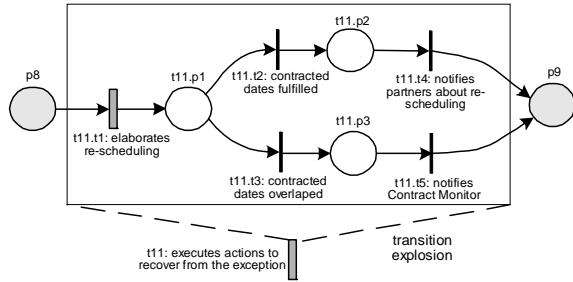


Fig. 3 – Actions to Recover from an Exception

This activity is mainly responsible for re-scheduling and finding a new plan for the workflow execution, if the deviation caused by the disturbance is in the scope allowed to the IO-WfM agent (i.e. the impact of the disturbance cause deviations that do not violate the terms of the contract). In case of success, the IO-WfM agent will notify the entity partners about the modification in the work-plan; otherwise, it will notify the Contract Monitor agent about the occurrence of the unrecoverable exception, which should then trigger the contract-predicted corrective measures.

4. IO-WFM AGENT INTERFACES

In this section, the interfaces and the relevant information exchanges that will support the inter-organizational workflow are identified.

The IO-WfM agent interacts with other components from the Electronic Institution platform, such as the Contract Monitor, and also with other entities, external to the platform but partners in terms of the contract execution. Fig. 4 illustrates the interfaces of the IO-WfM agent.

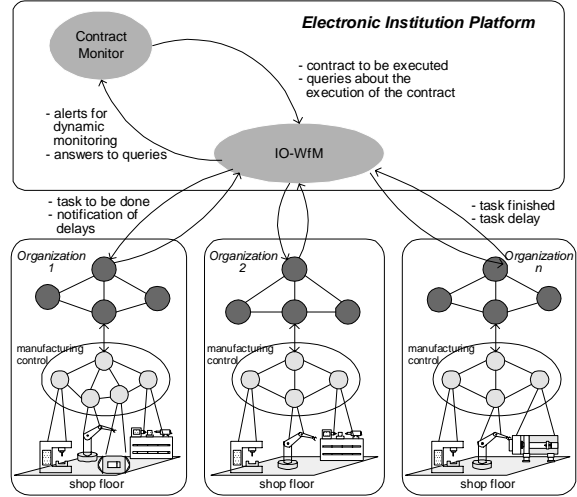


Fig. 4 – IO-WfM Agent Interfaces

The next subsections discuss deeply each one of the interfaces illustrated in the Fig. 4, describing the relevant information exchanged and that will support the inter-organizational workflow management.

4.1 Request to Coordinate the Contract Execution

As mentioned before, the Contract Monitor agent requests the IO-WfM agent to coordinate the physical execution of the work-plan resulting from the inter-organizational contract. For this purpose the Contract Monitor provides partial information about the contract, including task precedence information that will compose the work-plan to be managed.

The content of the message exchanged between the Contract Monitor and the IO-WfM agent has the structure defined in the following DTD (Document Type Definition):

```
<!DOCTYPE Manage-Workflow [
  <!ELEMENT Manage-Workflow (Contract,
    Task+)>
  <!ELEMENT Contract (#PCDATA)>
  <!ATTLIST Contract Id CDATA #REQUIRED>
  <!ELEMENT Task (Precedence*, Partner,
    StartDate, EndDate)>
  <!ATTLIST Task Id CDATA #REQUIRED>
  <!ELEMENT Precedence (#PCDATA)>
  <!ELEMENT Partner (#PCDATA)>
  <!ELEMENT StartDate (#PCDATA)>
  <!ELEMENT EndDate (#PCDATA)>
]>
```

In this work, a task is considered the elementary block of the contract that reflects the agreement between a partner entity providing services and an entity needing one of those services. It comprises a set of actions that should be taken to introduce value-added, such as the production or assembly of a product.

The IO-WfM agent transforms the received information by representing it in a more computational form, such as a work-plan graph. For

this purpose, precedence information and the start and end dates of each task are used. These dates were planned at the contract elaboration phase and are refined during the execution of the work-plan, according to the shop floor agenda of each partner.

4.2 Execution Synchronization

The supervision and synchronization of the work-plan requires the interaction of the IO-WfM agent with the several partner entities that are responsible to execute the tasks defined in the contract.

As illustrated in the Fig. 4, it is possible to group the set of these interactions in the following classes: notification to start the execution of a task, notification of task finished or delayed and queries of task status. The content of all messages uses the structure defined in the following DTD:

```
<!DOCTYPE Synchronization [
  <!ELEMENT Synchronization (Contract,
    Task)>
  <!ELEMENT Contract (#PCDATA)>
  <!ATTLIST Contract Id CDATA #REQUIRED>
  <!ELEMENT Task (Action, Reason?,
    NewStartDate?, NewEndDate?)>
  <!ATTLIST Task Id CDATA #REQUIRED>
  <!ELEMENT Action (#PCDATA)>
  <!ATTLIST Action Type (Start | End |
    Delay | Update | Query)
    #REQUIRED>
  <!ELEMENT Reason (#PCDATA)>
  <!ELEMENT NewStartDate (#PCDATA)>
  <!ELEMENT NewEndDate (#PCDATA)>
]>
```

The first type of interaction is used by the IO-WfM agent, according to the precedence graph of the tasks contained in the contract, to notify a partner entity that it can start the execution of the specified task. The content of the message exchanged between the IO-WfM agent and the partner entity, to notify the possibility to start the execution of a task, uses the action *Start* and requires the indication of the identification of the contract and the task.

After asking to start the execution of the task, the IO-WfM agent will wait for the result of its execution: *finished* if the task was executed with success or *delayed* if some problem occurred during the execution of the task causing its delay over the estimated due date.

In case of a task finished with success, the content of the message sent by the partner entity to the IO-WfM agent uses the action *End*. In case of delay, uses the action *Delay* is used, and the message specifies additionally the reason of the delay and the new estimated end date.

In this last case, the IO-WfM agent, based on the estimated end date proposed by the partner entity,

tries to find a new work-plan for the workflow execution that minimizes the impact of the disturbance, and notifies all dependent partner entities about the changes in the work-plan. The content of the message sent by the IO-WfM agent to the partner entity uses the action *Update* and provides information about the new start and end dates.

The IO-WfM agent can also query the partner entities about the status of execution of individual tasks, if these entities do not provide active notification services. For this purpose, it uses the *Synchronization* structure with the action *Query*.

4.3 Report of Execution

The IO-WfM agent should notify the Contract Monitor agent about the end or delay of a task execution, using an active notification form. The content of the message exchanged between IO-WfM and Contract Monitor agents uses the structure defined in the following DTD:

```
<!DOCTYPE Notify [
  <!ELEMENT Notify (Contract, Task)>
  <!ELEMENT Contract (#PCDATA)>
  <!ATTLIST Contract Id CDATA #REQUIRED>
  <!ELEMENT Task (Partner, Result)>
  <!ATTLIST Task Id CDATA #REQUIRED>
  <!ELEMENT Partner (#PCDATA)>
  <!ELEMENT Result (#PCDATA)>
  <!ATTLIST Result Type (End | Delay)
    #REQUIRED>
]>
```

The Contract Monitor agent can use this feedback information to update the confidence degree of the partner entity and to take appropriate measures as prescribed in the contract (such as applying sanctions) if the IO-WfM agent could not recover from the failure or delay.

The Contract Monitor agent can interact with the IO-WfM agent asking for details about the state of execution of the work-plan. The content of the query message sent by the Contract Monitor agent to the IO-WfM agent has the following structure:

```
<!DOCTYPE Query [
  <!ELEMENT Query (Contract, Task)>
  <!ELEMENT Contract (#PCDATA)>
  <!ATTLIST Contract Id CDATA #REQUIRED>
  <!ELEMENT Task (#PCDATA)>
  <!ATTLIST Task Id CDATA #REQUIRED>
]>
```

The IO-WfM agent evaluates if it has enough knowledge to answer to the query, and if not, it requests information to the lower-level entity partners, compiles the answers and sends back the requested information. The content of the inform message sent by the IO-WfM agent to the Contract

Monitor agent uses the structure defined in the following DTD:

```
<!DOCTYPE Report [
<!ELEMENT Report (Contract, Task)>
<!ELEMENT Contract (#PCDATA)>
<!ATTLIST Contract Id CDATA #REQUIRED>
<!ELEMENT Task (Partner, Status)>
<!ATTLIST Task Id CDATA #REQUIRED>
<!ELEMENT Partner (#PCDATA)>
<!ELEMENT Status (#PCDATA)>
<!ATTLIST Status Type (Allocated |
Delayed | Running | Finished
| Canceled) #REQUIRED>
]>
```

5. CONCLUSIONS

The business value associated with a VO (a consortium of different organizations) requires a close monitoring of an inter-organizational workflow. This way, possible problems can be anticipated and their potential damages minimized. Therefore, besides addressing the legal issues of an inter-organizational relationship, it is important to deal also with the operational concretization of such cooperation efforts.

This paper introduced an EI platform that provides services to support the VO composition and operation. One of these services is the IO-WfM, which interfaces with the supervision of the contract execution, providing a close perception of what is going on at the shop-floor level in an inter-organizational perspective. The paper presents an approach to this agent, modeling its behavior using Petri net models and describing its interfaces by defining DTD's for the information exchanged.

Our current efforts point to the inclusion of the described service in our EI prototype, which will include services such as those depicted in Fig. 1. Our implementation is based on JADE (Java Agent Development Framework).

In order to include the IO-WfM service, some issues will be further refined, such as the re-planning mechanism, and the integration with the (simulated) manufacturing control systems of partners, in order to obtain the feedback from the shop floor level.

ACKNOWLEDGMENTS

The work reported in this paper was supported by the FCT (Fundação para a Ciência e a Tecnologia) Project POSC/EIA/57672/2004.

REFERENCES

Colombo, A., Carelli, R., & Kuchen, B. (1997), "A Temporized Petri Net Approach for Designing,

- Modelling and Analysis of Flexible Production Systems", *The International Journal of Advanced Manufacturing Technology*, Springer-Verlag, **13**(3), pp. 214-226.
- Craswell, R. (2000). *Contract Law: General Theories*. In B. Bouckaert & G. De Geest (eds.), *Encyclopedia of Law and Economics*, Volume III: *The Regulation of Contracts*, Cheltenham, Edward Elgar, pp. 1-24.
- Cusati, F. & Discenza, A. (2000), "Supporting Workflow Cooperation within and Across Organizations", *15th International Symposium on Applied Computing*, Como, Italy, pp. 196-202.
- Davulcu, H., Kifer, M, Pokorny, L.R., Ramakrishnan, C.R., Ramakrishnan, I.V., & Dawson, S. (1999), "Modelling and Analysis of Interactions in Virtual Enterprises", *Proceedings of the 9th International Workshop on Research Issues on Data Engineering: Information Technology for Virtual Enterprises (RIDE 1999)*, IEEE Computer Society, pp.12-18.
- Dignum, V. & Dignum, F. (2001), "Modelling Agent Societies: Co-ordination Frameworks and Institutions", P. Brazdil, A. Jorge (eds.), *Progress in Artificial Intelligence: Knowledge Extraction, Multi-agent Systems, Logic Programming, and Constraint Solving*, LNAI 2258, Springer, pp. 191-204.
- Kruth, J., Detand, J., Zeir, G. V., Kempenaers, J., & Pinte, J. (1996), "Methods to Improve the Response Time of a CAD System that Generates Non Linear Process Plans", *Advances in Engineering Software*, **25**, pp. 9-17.
- Leitão, P. & Restivo F. (2005), "Experimental Validation of ADACOR Holonic Control System", V. Marik, R. Brennan and M. Pechoucek (eds.), *Holonic and Multi-Agent Systems for Manufacturing*, LNAI 3593, Springer-Verlag, pp. 121-132.
- Lopes Cardoso, H., Malucelli, A., Rocha, A.P., & Oliveira, E. (2005), "Institutional Services for Dynamic Virtual Organizations", Camarinha-Matos, L., Afsarmanesh, H. & Ortiz, A. (eds.), *Collaborative Networks and Their Breeding Environments - 6th IFIP Working Conference on Virtual Enterprises (PRO-VE'05)*, Springer, pp. 521-528.
- Lopes Cardoso, H., & Oliveira, E. (2005), "Virtual Enterprise Normative Framework within Electronic Institutions", M.-P. Gleizes, A. Omicini & F. Zambonelli (eds.), *Engineering Societies in the Agents World V*, Springer, pp.14-32.
- Malucelli, A., Palzer, D. & Oliveira, E. (2005), "Combining Ontologies and Agents to Help in Solving the Heterogeneity Problem", *Proceedings of the International Workshop on Data Engineering Issues in E-Commerce (DEEC'2005)*, IEEE Computer Society, pp. 26-35.
- Rocha, A.P., Lopes Cardoso, H., & Oliveira, E. (2005), "Contributions to an Electronic Institution Supporting Virtual Enterprises' Life Cycle", G. Putnik & M.M. Cunha (eds.), *Virtual Enterprise Integration: Technological and Organizational Perspectives* (pp. 229-246), Idea Group, Inc.
- Van der Aalst, W. (1999), "Inter-organizational Workflow: An Approach based on Message Sequence Charts and Petri Nets", *International Journal on SAMS*, **34**(3), pp. 335-367.