

# SUPPORTING BUSINESS PROCESS MANAGEMENT AND COORDINATION IN A VIRTUAL ENTERPRISE

---

L.M. Camarinha-Matos, C. Pantoja-Lima  
New University of Lisbon – Faculty of Sciences and Technology  
Quinta da Torre – 2825 Monte Caparica – Portugal  
Tel. +351-212948517 Fax +351-212941253 E-mail: cam@uninova.pt

*The availability of a modeling and management framework for distributed business processes is a major requirement for the implantation of the virtual enterprise paradigm. The business process creation, distribution, and execution requirements are discussed and a workflow-based modeling and execution approach and support tools are presented. Finally major results on coordination of such processes from two research projects are summarized and directions for further research outlined.*

## 1. INTRODUCTION

Coordination of activities is a major issue in virtual enterprises (VE) in order to ensure that the partial contributions of each VE member effectively add to the achievement of the common goal(s). Although cooperation is the fundamental characteristic of a VE, due to its distributed environment and the autonomous and heterogeneous nature of the VE members, cooperation can only succeed if a proper management of dependencies between activities is in place.

Coordination is the focus of attention of many research projects in this area along two main lines of development. A group of projects, such as VIVE (VIVE 1999), are mainly concerned with the characterization and modeling of the cooperative business processes (BP). Another group, as illustrated by PRODNET II (Camarinha-Matos and Afsarmanesh, 1999), without ignoring the BP modeling aspects, has concentrated efforts on the design and development of supporting infrastructures for VEs, i.e. the execution environment to support cooperative activities. There is now a need to combine the most important results of these two groups of projects. Modeling the business processes for a VE environment is certainly an important requirement but the actual execution of these processes requires a flexible supporting infrastructure.

This paper aims at presenting a discussion on some BP modeling issues and execution support requirements taking the PRODNET infrastructure as a baseline. In addition to this basic infrastructure further developments being investigated in the framework of the MASSYVE project are also considered.

## 2. BUSINESS PROCESSES MODELING AND EXECUTION

### 2.1 Basic Concepts

The activities carried out by a company in order to achieve a specific goal are usually organized in “clusters” of inter-related activities called *processes* (*BP - business processes*). They can be seen as a set of activities, rules and constraints specifying the steps that must be taken, and under which conditions, in order to accomplish a given goal. In other words, the composition of each process is designed in order to achieve a (partial) specific goal. Following a CIM-OSA-like *top down* view (AMICE, 1993), a business process can be decomposed into a hierarchy of sub-business processes and enterprise activities. The enterprise activities represent the lowest level in this decomposition, i.e. the basic building blocks the enterprise has to actually realize its processes.

For some authors the enterprise activities are the ones that actually require execution while the upper level processes represent logical aggregations of these executions. In this line, CIM-OSA considers that enterprise activities are supported by the Implemented Functional Operations. From a supervision point of view it makes sense to have a hierarchical execution system in parallel with the hierarchical decomposition of the BP. This is the approach followed in the HOLOS (Rabelo and Camarinha-Matos, 1994) and MASSYVE (Rabelo et al, 1999) systems. In these systems, although the leaves of the hierarchical decomposition of the BP (i.e. the enterprise activities) are carried out by the enterprise resources, temporary agents (named consortia agents) are created to take care of (supervise or coordinate) the execution of the higher level processes.

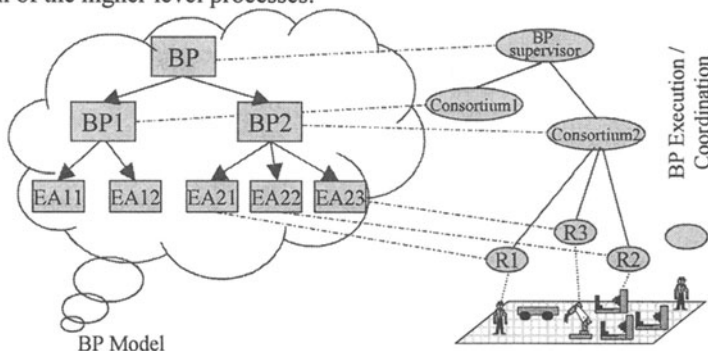


Figure 1 - HOLOS / MASSYVE BP execution framework

In the example of Fig. 1, R1, R2 and R3 are agents or collections of services representing different resources that are assigned to the execution of the enterprise activities EA21, EA22 and EA23 respectively. These three resources (service providers) constitute a temporary consortium for the execution of BP2. The agent consortium 2 is created to coordinate / supervise the execution of the BP2, i.e. to coordinate the execution of its constituting EAs.

When a business process is executed by a virtual enterprise parts of the decomposition of this BP (i.e. sub-processes) are assigned to different enterprises, becoming a *distributed business process* (DBP) or virtual business processes in the

WISE terminology (Alonso et al., 1999). The problem of the supervision or coordination of the BP at its various levels of decomposition becomes even more important in this context where the definition and enactment is not limited to a single organization but instead a set of autonomous, distributed and heterogeneous nodes have to cooperate. When properly “orchestrated”, the combination of the various processes taking place in the different members of the VE are expected to lead to the achievement of the global goal of the VE.

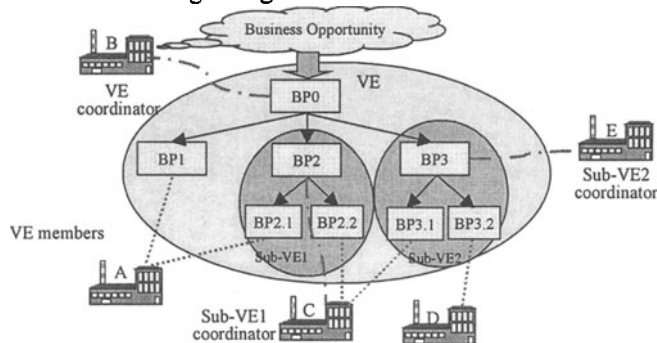


Figure 2 - DPB execution

The VE coordinator is responsible for the whole BP, while the VE members are responsible for the coordination of the sub-BPs assigned to them. A VE member may itself become a coordinator of a sub-VE inside the VE to perform its sub-BP. At the macroscopic level, the VE members play the HOLOS/MASSIVE's consortium role. Under this approach, temporary (sub-) consortia can be formed inside a VE. These sub-consortia are formed for the sole purpose of facilitating the coordination of activities involved in the related sub-business processes. Once a sub-business process ends, the sub-consortium “dissolves” and its members may become involved in other sub-consortia dynamically formed as the execution of the VE BPs evolves. For instance, the enterprise C (in Fig. 2) coordinates sub\_VE1 and is a member of sub\_VE2.

Several languages and formalisms have been used for BP modeling. Examples are IDEF3, MANIFOLD, UML, PIF, and workflow definition languages. In the VE area many projects, such as the case of PRODNET II (Camarinha-Matos and Lima, 1998), adopted a workflow-based approach due to the availability of experience with workflow systems in many enterprises and the standardization efforts promoted by the Workflow Management Coalition (WfMC, 1994).

## 2.2 Workflow-based modeling of BP

In the context of PRODNET II, the New University of Lisbon team developed a graphical language and an associated editor (LCF) that borrows many ideas from the workflow management systems, trying to be compliant with the WfMC Reference Model (Fig. 3) but including some adaptations to support the modeling of BPs in a VE environment. This modeling tool supports the following modeling primitives:

- Sequences of activities that might invoke supporting services or other sub-activities.

- Sub-workflow definition, as a mechanism to support hierarchical (nested) BP modeling.
- Data flow management for parameter passing when activating services / sub-activities, i.e. data that is essential for the process execution control flow. This is the explicit data exchange. A form of implicit data exchange is supported by a distributed information management module (see chapter 3).
- Splits and joins that can have the logical conditions AND / XOR.
- Simple and conditional transitions.
- Temporized and cyclic activities, providing the high-level coordination facility that is needed for instance in the case of monitoring contract clauses among VE members.
- Flexible configuration of catalogs of services and relevant data.
- Workflow instances and memory spaces. For each execution of a workflow model an instance is created with its memory space. The explicit data flow associated to an instance (relevant data) is only valid inside the memory space of that instance.
- Management of waiting lists. Each time an instance of a workflow model needs to wait for the conclusion of an external service it is put in a waiting list. Waiting lists are also used for instances waiting for temporized activities. Signals can be sent to the waiting lists manager to provoke changes in the status of workflow model instances.

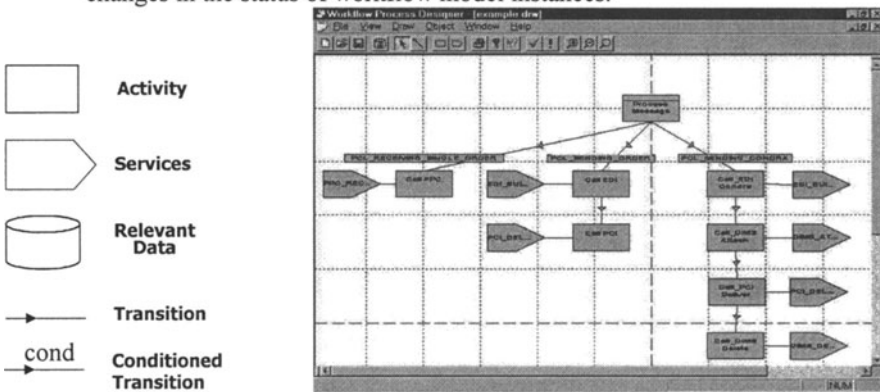


Figure 3 - a) Graphical process modeling primitives b) LCF graphical process model editor

The output of the graphical editor is stored as a WPD (Workflow Process Definition Language) file, following the WfMC standard syntax (WfMC, 1994). One example of a BP to produce a bicycle is represented in Fig. 4.

Hierarchical BPs can be represented using the concept of sub-workflow. For instance, the activity Pedal Mould Design Supervision in the example of Fig. 4 the can be supported by a more detailed business process as represented by Fig. 5. This figure also illustrates that this nesting process can proceed through several levels. In other words, the decomposition of processes into sub-processes is supported by the sub-workflow mechanism.

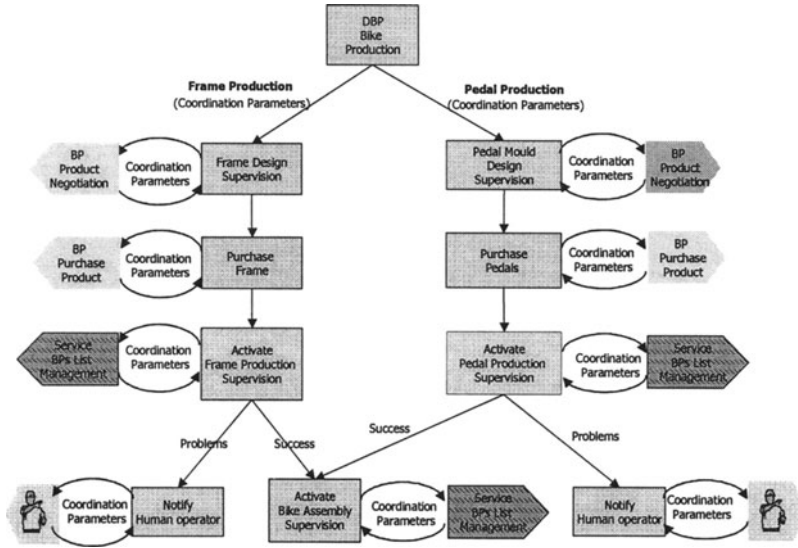


Figure 4 - Example of a BP

Sub-workflow models also provide a basic degree of *reusability* of a model, since a sub-workflow can be used several times in a workflow model, like a sub-routine. The more frequent tasks can be modeled as sub-workflow models and used as many times as necessary. This feature allows the creation of a library of sub-workflow models (templates) that represent the processes frequently performed.

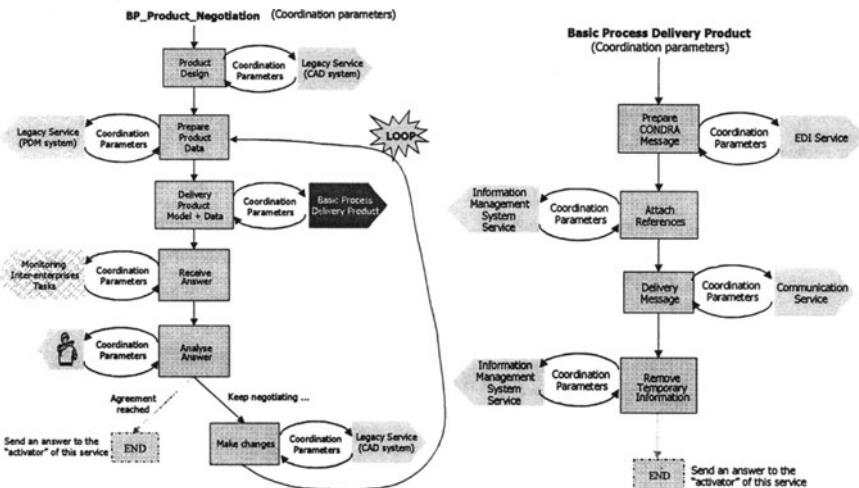


Figure 5 - Activity implemented as a sub-workflow

### 2.3 BP templates and allocation mechanisms

Assuming that the most common business processes have many similarities in different organizations, a set of templates can be made available to both the VE

coordinator and the VE members. A template BP can, however, be edited in order to represent the specific needs and autonomy of each VE and each VE member.

The experience with the PRODNET infrastructure showed that one of the elements of diversity is the set of available services (and resources) in each enterprise. Therefore it makes sense to consider two phases in the BP definition:

- Creation of the BP model (or edition of templates).
- Allocation of services / resources to the execution of the BP activities.

In order to facilitate the allocation phase, the BP modeling phase can specify the types of services (and interfaces) adequate to execute each activity. The identification of concrete services is made when specific instances of the BP model are created.

Besides the service types, the BP edition phase could allow the inclusion of other embedded knowledge that would facilitate the BP instantiation and execution. For instance, clauses for error recovery could be included in order to facilitate the run-time conflict resolution.

In the context of a DBP to be executed by a VE, the allocation of services can proceed in two phases:

- Assignment of BPs to VE members.
- Allocation of enterprise resource / services, by each VE member, to the activities of the (sub-)BPs assigned to the enterprise.

These functionalities require a change of the modeling tool developed in PRODNET II. Further developments in this direction, being pursued in the MASSYVE project, include the addition of a flexible scheduling mechanism to the VE members and services allocation steps.

In the MASSYVE (Rabelo et al., 1999) approach both the members of a VE and the resources inside each VE member are represented as agents. The allocation of a BP/EA to a specific agent is based on the contract-net / negotiation paradigm (Davies and Smith, 1983). Based on this paradigm, tasks are announced to potential candidate agents. Candidate agents, the ones that are able to fulfill the task requirements, submit bids to the contractor agent that decides to which one to assign the task.

One important aspect here is the conflict resolution. Conflicts may occur as a result of changes in the BPs specifications, changes in BP priorities, temporary failure or unavailability of resources, etc. Such situations may affect the overall BP execution and therefore the achievement of the VE goal. The negotiation paradigm provides a flexible mechanism for conflict resolution starting from the lowest possible level. The agent that faces / detects a problem will try to solve the problem at its level, trying to sub-contract another agent, and only in case of failure the problem is passed to the next level. A preliminary work in this direction can be found in (Klen et al, 1999).

## **2.4 Creation of BP**

One important question is to determine who is responsible for the creation of BP models and instances. The answer depends on the particular type of VE. Different VE organizations may consider different actors in this process and different coordination rules (Camarinha-Matos et al., 1998). Some possibilities are:

- **Centralized planning.** In a tightly integrated operation of the VE, the VE coordinator may plan the whole BP and send it to the VE members. In this case it is necessary to take into account the visibility rights of the VE members. Should a VE member see the whole plan or just the part it is responsible for?
- Another area to be explored is the application of *mobile agents* to facilitate the adaptation / optimization of the BPs to the conditions of the VE. For instance, an intelligent mobile agent can carry a macro plan (abstract BP definition) and detail it once it arrives at a specific VE member according to the specific conditions it finds there. One work in this direction, although applied to remote supervision / tele-operation, can be found in (Camarinha-Matos and Vieira, 1998, 1999).
- **Cooperative planning.** Another alternative is to consider a cooperative BP planning, by several VE members. In this case it is necessary to implement a shared planning space for the BP model design, a feature not implemented in the current version of LCF but that could be easily developed on top of the PRODNET distributed information management system (Afsarmanesh et al., 1998).
- **Hierarchical planning.** Finally, the typical case is one in which only the abstract model of the BP (the first few decomposition levels) is planned by the partner that identified the business opportunity (broker or VE coordinator). The level of detail of this model is just enough to allow the identification of the necessary partners / skills and main resources and the distribution of sub-BPs among these partners (Alonso et al., 1999).

Taking the example of Fig. 2, the VE coordinator could, for instance, decompose the BP0 into BP1, BP2, and BP3. Further decomposition of the BPs could be done by the VE members assigned to their execution. Once a VE member receives a BP, it can detail this BP according to the internal capabilities (skills, manufacturing processes, resources) of the company. For instance, the enterprise C could decompose BP2 into BP2.1 and BP2.2.

One question in this case is: should the VE members notify the VE coordinator of the actual decomposition they have performed? If the coordinator is informed, or if at least a simplified version of the decomposition is received, the monitoring activity performed by the coordinator can be simplified. For instance, at the current stage, in order to query a VE member about the status of a given process, like the status of a specific order, the querying member must be aware of the possible states considered by the internal ERP/PPC system of the queried member. For instance, in the PRODNET II demonstrator, the life cycle of an order inside the ERP/PPC system of each VE member is assumed to be known by the coordinator (Fig. 6).

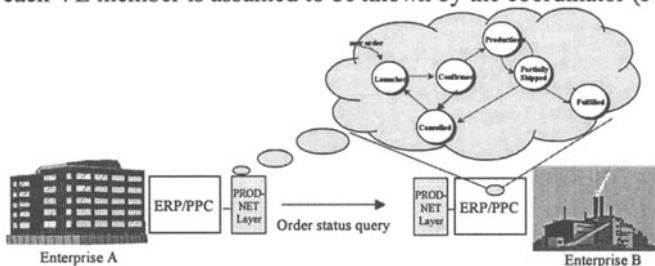


Figure 6 - To query an order status it is necessary to be aware of the order life cycle

If, in alternative, the VE members send a summary of the detailed decomposition of the BP to the VE coordinator a more flexible coordination approach can be achieved. This requires a normalized representation of the business processes (WPD, PIF, etc).

## 2.5 BP and contracts

There is a relationship between the contracts / cooperation agreements among the VE members and the BP definition. Contracts define the macro responsibilities of each partner, what is directly related to the identification of the BPs assigned to each partner and the execution supervision principles.

Contracts or cooperation agreements define the supervision rules to be applied during the VE operation. A contract may include rules about what will be monitored, the monitoring frequency, who has the initiative (push, pull) to send / request information, etc. For instance, if the contract states that enterprise A must send to enterprise B, every Friday evening, a report on the production status regarding some distributed BP, two workflow plans can be defined, one for each enterprise, in order to automate the fulfillment of this contractual clause. The workflow in enterprise A would include the activities necessary for gathering the production status data in the right time, formatting these data and sending the report to enterprise B. The workflow running in enterprise B would include the activities to monitor the arrival, in time, of the expected report and, in case it doesn't arrive, start some contingency procedure.

The agreements may also specify the levels of visibility, the conflict resolution mechanisms, and the liabilities. The area of contracts definition for VE is however an open issue requiring a considerable research and normalization effort.

In terms of future developments there is a need for an integrated tool for contracts definition / management and BP modeling and supervision.

## 2.6 Loosely constrained BPs

Once a VE infrastructure is available, more integrated cooperation forms can be supported. That is the case, for instance, in *concurrent* or *collaborative engineering* where teams of engineers, possibly located in different enterprises, cooperate in a joint project such as the co-design of a new product. A large number of computer supported cooperative tools are becoming widely available for synchronous cooperation. Some examples are teleconference, and chat tools combined with application sharing mechanisms. Considering the geographical distribution, the autonomy of the VE members, the local corporate cultures, and also the individual working preferences of the team members, it is likely that most of the activities will be carried out in an asynchronous way. In order to assure the proper progress in this loosely coupled environment it is necessary to implement some form of coordination of activities for these collaborative processes.

In the case of processes mainly executed by humans, rigid forms of control such as the workflow type or the CIM-OSA procedural rules sets are not adequate. People like to keep their freedom regarding the way they work. Product design, like any other creative process evolves according to a kind of "anarchic" flow. It is therefore necessary to also support *loosely constrained* sets of *business processes*. Another



aspect is the representation of *temporal interdependencies* among activities. For instance, in the case of the processes “Product Design” and “Process Planning”, although they can proceed with some degree of concurrency (i.e. process planning can start once a first draft of the product is made), Process Planning cannot finish before Product Design finishes. At least some details of the process plan definitely depend on the final commitments on the product model.

Several approaches to develop *flexible workflow* systems have been proposed (Heinl et al., 1999). One solution for the coordination flexibility was first introduced in the CIM-FACE system (Osorio and Camarinha-Matos, 1995) and later discussed for the context of virtual enterprises (Osorio et al., 1998). In this system, instead of rigid precedence rules, other types of relationships, inspired in the Allen’s temporal primitives, are possible: *start\_before*, *finish\_during*, *start\_after*, *finish\_after*, *do\_during*, etc. Other constraints such as pre- and post-conditions can be specified.

In order to support the coordination when the execution entities are human operators, probably resorting to some legacy tools (CAD, CAPP, PDM), some human front-ends (*Business Plan Assistants* in the CIM-FACE terminology) are necessary. These front-ends represent the human actors in the cooperative team and interact with the BP executing / coordination infrastructure. The coordination system is responsible for keeping track of the global execution status and to guarantee the pre-conditions required for any “intervention” of any actor in the common BPs. Complementarily in activities such as Concurrent Engineering there is a need for shared working spaces (implicit data sharing) and versioning to support co-design.

### 3. THE PRODNET COORDINATION INFRASTRUCTURE

A prototype VE execution system or support infrastructure for industrial virtual enterprises was designed and developed by the PRODNET II consortium (Camarinha-Matos et al., 1998), (Camarinha-Matos, Afsarmanesh, 1998). Although not all aspects mentioned in the previous chapter have been developed, the PRODNET infrastructure represents a flexible infrastructure covering most of the basic coordination needs. As proved by a comprehensive demonstration system, on top of this infrastructure it is possible to develop and evaluate further advanced coordination functionalities. The focus of this work was put on the needs of the small and medium size enterprises (SME), but the achieved results can be as well applied to large companies.

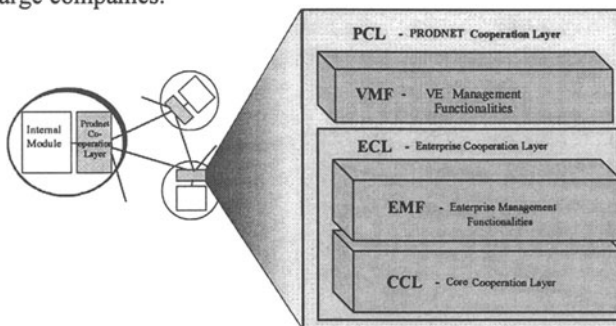


Figure 7 - PRODNET 3-level coordination architecture

Figure 7 shows the PRODNET architecture that comprises two main blocks: The PRODNET Cooperation Layer (PCL) and the Company's Internal Module.

The so-called Internal Module represents the various legacy systems and applications that run in the enterprise like a Production Planning and Control (PPC) system or Enterprise Resources Planning (ERP) system, a CAD system, a Product Data Management System (PDM), other engineering and internal tools.

The PRODNET Cooperation Layer (PCL) is the main component of the PRODNET infrastructure and is responsible for supporting all interactions between a company and its VE partners. From the activity/process coordination point of view, the PCL considers three levels of abstraction:

- *Core Cooperation Layer (CCL)*. The CCL is responsible for the basic interactions among VE members offering support for safe communications, exchange of business messages, sharing and management of cooperation information, federated information queries, etc. The main components of this layer are:
  - *LCM - Local Coordination Module*. This module is a kind of workflow engine, responsible for the execution or enactment of the workflow models that specify the desired cooperation behavior of the enterprise. This engine manages the execution of the activities in the various active workflows, providing mechanisms for the invocation of associated services (EDIFACT services, communications services, information management services, etc.) and for data flow management within the memory space of each workflow instance.
  - *DIMS - Distributed Information Management System*. The Distributed Information Management Subsystem, based on the federated database management paradigm, is responsible to model and manage all cooperation support information, while preserving the autonomy and information privacy of the involved enterprises (Afsarmanesh et al., 1998). In interaction with LCM, this module can also support the maintenance of a *process history* that can be used for auditing purposes or data mining towards process performance improvement.
  - *PCI - PRODNET Communication Infrastructure*. This module is responsible for handling all communications with the other nodes in the network assuring basic safety and authentication mechanisms.
  - *LCF - Local Configuration Functionality*. This module includes various configuration functions and the graphical workflow editor.
  - *EDI Module*. This module is responsible for receiving and formatting orders-related messages in EDIFACT format.
  - *STEP Module*. The STEP module's function is to handle the technical product data used within PRODNET. Ideally all product data should be exchanged in STEP format. The STEP services provided to PRODNET will allow the transmission and reception of STEP files that have been clear text encoded according to a defined schema.
- *Enterprise Management Functionalities (EMF)*. The EMF is responsible for coordination of the activities at the enterprise level. In other words, the EMF deals with coordinating the responsibilities of the enterprise towards the accomplishment of its assigned BPs or contracts with the VE and other VE-partners.

- *Virtual Enterprise Management Functionalities (VMF)*. The coordination aspects at the VE level are considered in the third level. The node playing the VE coordinator role will use this layer to monitor, assist, and modify the necessary activities related to the VE goal achievement. The VE Management Functionalities (VMF) resort to the services provided by the CCL and EMF of its node to communicate with the other nodes of the VE. Conflict detection and recovery (regarding task scheduling) (Klen et al., 1998) partners search and selection (Camarinha-Matos et al. 1999), are examples of functionalities included at this level.

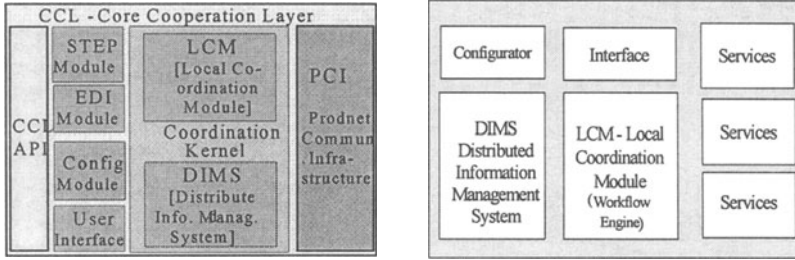


Figure 8 a) Core Cooperation Layer      b) General structure of a coordination layer

Each component of this architecture, namely CCL, EMF and VMF, has the same general structure as represented in (Figure 8.b).

Although in its current implementation only three levels are considered, the model can be easily generalized to any number of levels in order to cope with any BP tree. In fact, in addition to the VE coordination role, responsible for the global BP, other enterprises may assume the role of coordinators of sub-business processes that might be decomposed and performed by a sub-consortium of enterprises (as illustrated in Fig. 2).

Further details on the PRODNET architecture and developed tools can be found in (Camarinha-Matos and Afsarmanesh, 1999).

#### 4. CONCLUSIONS

The PRODNET project developed a generic infrastructure to support cooperation in a VE environment and therefore offering basic functionalities for distributed business process modeling and execution. Although further developments are necessary, the implemented tools and demonstration examples contributed to the identification and characterization of the main functional requirements for a VE execution system. The proposed workflow-based multi-level coordination approach represents a convenient tool to structure the various coordination needs in a VE.

Further developments towards a more flexible BP support are being pursued, namely in the framework of the MASSYVE and FETISH projects where the multi-agent approaches to dynamic allocation of processes, loosely constrained processes, and methods for BP planning are being investigated.

## Acknowledgments

This work was funded in part by the European Commission through the Esprit PRODNET II and the INCO MASSYVE projects. The authors also thank the valuable contributions of their partners in the mentioned projects consortia.

## 5. REFERENCES

1. Afsarmanesh, H.; Garita, C; Hertzberger, L.O. - Virtual Enterprises and Federated Information Sharing. *Proceedings of the 9th IEEE International Conference on Database and Expert Systems Applications", DEXA'98*, pages 374-383, Vienna, Austria, Aug 1998.
2. Alonso, G. et al, - *The WISE approach to Electronic Commerce*, <http://www.inf.ethz.ch/departement/IS/iks/research/wise.html>, Feb 15, 1999.
3. AMICE – CIM-OSA: Open Systems Architecture for CIM, 2<sup>nd</sup> revised and extended version, Springer-Verlag, Berlin, 1993.
4. Camarinha-Matos, L.M; Afsarmanesh, H. (Ed.s)– *Infrastructures for virtual enterprises – Networking industrial enterprises*, Kluwer Academic Publishers, Oct 1999.
5. Camarinha-Matos, L.M; Afsarmanesh, H.; Cardoso, T. - Partners search and quality related information exchange in a virtual enterprise, *Proceedings of APMS'99 – IFIP Int. Conf. on Global Production Management (K. Mertins, O. Krause, B. Schallock – Eds.)*, Kluwer Academic Publishers, ISBN 0-7923-8605-1, pp. 76-84, Set 99, Berlin, Germany.
6. Camarinha-Matos, L.M; Afsarmanesh, H.; Garita, C.; Lima, C. - Towards an architecture for virtual enterprises, *Journal of Intelligent Manufacturing*, Vol. 9, Issue 2, Apr 98, pp 189-199.
7. Camarinha-Matos, L.M.; Pantoja Lima, C. - A Framework for Cooperation in Virtual Enterprises, *Proceedings of DIISM'98 - Design of Information Infrastructures Systems for Manufacturing 1998*, Fort Worth, USA, May 98.
8. Camarinha-Matos, L.M.; Vieira, W. - Adaptive mobile agents for telerobotics and telesupervision, *Proc. of INES'98 – IEEE Int. Conf. On Intelligent Engineering Systems*, pp. 79-84, Vienna, Austria, Set 98.
9. Camarinha-Matos, L.M.; Vieira, W. - Intelligent mobile agents in elderly care, *Journal of Robotics and Autonomous Systems* (Elsevier), Vol. 27, N. 1-2, April 1999, ISSN 0921-8890, pp. 59-75.
10. Carriero, N.; Gelertner, D. – Coordination languages and their significance, *Communications of the ACM*, Vol. 35 (2), 1992, pp. 97-107.
11. Davies, R.; Smith, R. – Negotiation as a metaphor for distributed problem solving, *Artificial Intelligence*, Vol. 20, pp 63-109, 1983.
12. Goranson, H.T. – *Agile virtual enterprises: cases, metrics, tools*, Quorum books, 1999.
13. Heintz, P.; Horn, S.; Jablonski, S.; Neeb, J.; Stein, K.; Teschke, M. – A comprehensive approach to flexibility in workflow management systems, *Joint Conf. On Work Activities Coordination and Collaboration*, San Francisco, USA, 1999.
14. Klen, A.; Rabelo, R.; Spinosa, L.M.; Ferreira, A.C. - Integrated Logistics in the Virtual Enterprise: the PRODNET-II Approach, *Proceedings of IMS'98 – 5<sup>th</sup> IFAC Workshop on Intelligent Manufacturing Systems*, Gramado, Brazil, 9-11 Nov 1998.
15. Osorio, A.L.; Camarinha-Matos, L.M. – Support for Concurrent Engineering in CIM-FACE, in *Balanced Automation Systems*, Chapman and Hall, pp.275-286, 1995.
16. Osorio, A.L.; Oliveira, N.; Camarinha-Matos, L.M. - Concurrent Engineering in Virtual Enterprises: The extended CIM-FACE architecture, *Proc. Of BASYS'98 – 3<sup>rd</sup> IEEE/IFIP Int. Conf. On Balanced Automation Systems, Intelligent Systems for Manufacturing* (Kluwer Academic), pp. 171-184, ISBN 0-412-84670-5, Prague, Czech Republic, Aug 98.
17. Rabelo, R.; Camarinha-Matos, L.M. - Negotiation in Multi-Agent based dynamic scheduling, *Int. Journal on Robotics and CIM (Pergamon)*, Vol. 11, N. 4, Dez 1994, pp. 303-309.
18. Rabelo, R.; Camarinha-Matos, L.M.; Afsarmanesh, H. - Multi-agent-based agile scheduling, *Journal of Robotics and Autonomous Systems* (Elsevier), Vol. 27, N. 1-2, April 1999, ISSN 0921-8890, pp. 15-28.
19. VIVE – VIVE Reference model, 1999, [www.ceconsulting.it/VIVE/Results/default.html](http://www.ceconsulting.it/VIVE/Results/default.html).
20. WfMC - Workflow Management Coalition (1994) - The Workflow Reference Model - Document Nr. TC00 - 1003, Issue 1.1, Brussels Nov 29, 1994.