

A reference framework for process-oriented software development organizations

João M. Fernandes¹, Francisco J. Duarte²

¹Dep. Informática, Universidade do Minho, Braga, Portugal;
E-mail: jmf@di.uminho.pt

²Blaupunkt Auto-Rádio Portugal, Braga, Portugal;
E-mail: francisco.duarte@pt.bosch.com

Received: 30 December 2003/Accepted: 22 April 2004

Published online: 29 July 2004 – © Springer-Verlag 2004

Abstract. In this paper, a proposal of a generic framework for process-oriented software development organizations is presented. Additionally, the respective way of managing the process model, and the instantiation of their processes with the Rational Unified Process (RUP) disciplines, whenever they are available, or with other kind of processes is suggested. The proposals made here were consolidated with experiences from real projects and we report the main results from one of those projects.

Keywords: RUP – Process-oriented organizations – Software development process – Business modelling

1 Introduction

The development process for software must be oriented towards the client's needs. Systems using new technologies and concluded in time and in budget may not have a full utilization, if, for example, the client's requirements were not understood and included in the final application.

It is commonly accepted that visual and semi-formal information models, like those specified in the Unified Modeling Language (UML), lead to a better communication among software engineers and clients. UML allows the visualization, specification, construction, and documentation of the components of a software system [1]. It is also important to adopt processes, like RUP (Rational Unified Process), that use UML with added-value to the final client.

Currently, organizations are moving from a hierarchical structure, sub-divided by departments where the activities are performed by collaborators with a limited view of the organization, to a model where multi-skilled

teams run horizontal business processes that cross the organization, and may affect suppliers and clients. Thus, to develop software systems for those organizations two aspects must be addressed: (1) the development process must be adequate and controlled; (2) the platform where the organizational processes will run must be modelled and considered, both in the software development and the target organizations.

In this paper, based on a framework for generic organizations, we present a proposal of a reference framework for process-oriented software development organizations. We also propose the way of managing the processes within the framework and their instantiations with RUP disciplines (formerly designated workflows), whenever they are available.

The structure of the paper is the following. In Sect. 2 some related work is presented, namely a description and comparison of RUP and OPEN (Object-oriented Processes, Environment and Notation). Section 3 introduces the concept of a process-oriented organization that focus its activities towards the needs of the clients. The relevance of processes in organizations is discussed in Sect. 4. In Sect. 5 we propose a generic framework for a process-oriented organization and in Sect. 6, based on that framework and in the RUP disciplines, we detail it in order to describe organizations that develop software. The Business Process Modelling is explained in Sect. 7. In Sect. 8, we report on some of the lessons learned from applying the proposals made in this paper to an industrial case study. Finally, conclusions close the paper in Sect. 9. The main scientific contributions of the paper are described in Sects. 5 and 6.

2 Related work

Process modelling is different from other types of modelling in computer science, since many of the phenomena

This paper represents a significant revision and extension of the work published in the Proceedings of the OOPSLA 2003 Workshop on Process Engineering for Object-Oriented and Component-Based Development [6].

modelled must be enacted by a human instead of a machine. In fact, the definition of “process” should include not only descriptions of phases, activities, tasks, and techniques, but also issues related with human resources, technology, and the life-cycle model to be used [15]. Several proposals exist for defining processes within organizations that develop software. In this paper we restrict our discussion to RUP and OPEN.

RUP [19, 20] is a software process that identifies the activities needed to map user requirements into a software application. RUP is accepted to be a generic/parameterizable process that can be adapted for a wide range of different contexts, namely organizations with distinct Capability Maturity Model (CMM) levels of maturity, different skills and tools, and unequal number of team members. With RUP Builder and Plug-ins, it is possible to select a base RUP configuration, applying plug-ins and other elements to choose the relevant process components to the project at hand.

In a software development organization, RUP can be used as a systematic approach to assign tasks and responsibilities to its members. The main aim of RUP is to construct quality software that meets the requirements of the stakeholders, within a typical engineering context (i.e. taking into account, for instance, the restrictions on human resources, the schedule, and the budget).

RUP can be used in all the five ways proposed in [3]:

1. To facilitate human understanding and communication;
2. To support process improvement;
3. To support process management;
4. To automate process guidance;
5. To automate execution support.

RUP SE, an extension of RUP, to address system development, provides a broader view on the development, tackling projects that include not only software but also hardware development and that require specification of worker roles [2].

The OPEN Process Framework is a public domain framework for building processes to develop high-quality software-intensive applications and object-oriented business models [8]. The OPEN Process Framework provides a template for generating flexible processes within the development of software systems and taking into consideration the schedule and the budget.

In [18] the best practices for developing software are identified and a basis for the assessment of three processes (Catalysis [4], OPEN, and RUP) is presented. Their main conclusion is that RUP is presented in a very management oriented way, while OPEN is described with a software engineering bias. The level of detail of RUP is particularly aimed at managers in contrast to software managers who would benefit more from UML descriptions. OPEN is a rich and well-documented process, being of use to both the manager and the software engineer.

A general comparison of RUP with OPEN and a detailed evaluation of both processes from a project management view point is available in [13]. The same authors also present a qualitative evaluation on RUP and OPEN in [14]. The comparison is centred around issues related to the process architecture and underpinning meta-model, the concepts and taxonomy utilized and the support for project management. They conclude that the meta-level architecture of RUP leads to problems in terms of the lack of support for an iterative development and a strong dependence on the use case model. OPEN combines the adaptability to construct a process to the specific needs of a specific domain and to adapt the process continuously to particular projects. OPEN provides a more solid support in the area of cross-project suites of application developments and maintenance and it offers more extensively metrics and quality issues.

One of the most common problems with any process is its usage in a given project, namely when project members have to decide which process components to use [15]. Although RUP is criticized for being a heavy process [16], we adopted it because it is well-documented and supported by computer tools. Additionally, RUP can be easily adapted to several scenarios, namely by tailoring in which artefacts can be modified or deselected. For example RUP was already customized to conform with the agile methods’ philosophy in small teams [17] and was also extended with other techniques to address documentation needs [27].

3 Process-oriented organizations

The concept of a process-oriented organization is a way of focusing the activities of an organization towards the clients’ needs [11]. These activities are oriented to and validated by the clients. The necessities of a client must be satisfied efficiently and with quality. Reengineering, and its process-orientation, must be applied to anticipate change and not as a corrective procedure when bad business indicators occur.

The typical pyramids-like hierarchies constitute a real obstacle to satisfy the clients, since the functions and skills are distributed among several departments. It is quite common that there exist, for example, a department responsible for acquiring raw material, another for production, and another one for dispatching the products. Each of these departments has internal aims, which may create inter-departmental conflicts due to their divergences and, at the end, threaten the clients’ needs. Thus, in traditional organizations (Fig. 1a), the clients’ needs usually collide with the internal interests.

A process-oriented organization tackles these conflicts by creating structures that assume for themselves the clients’ needs. Thus, as illustrated in Fig. 1c, a process that incorporates all the needed functions is created and whose objectives match the clients’ ones. Similarly,

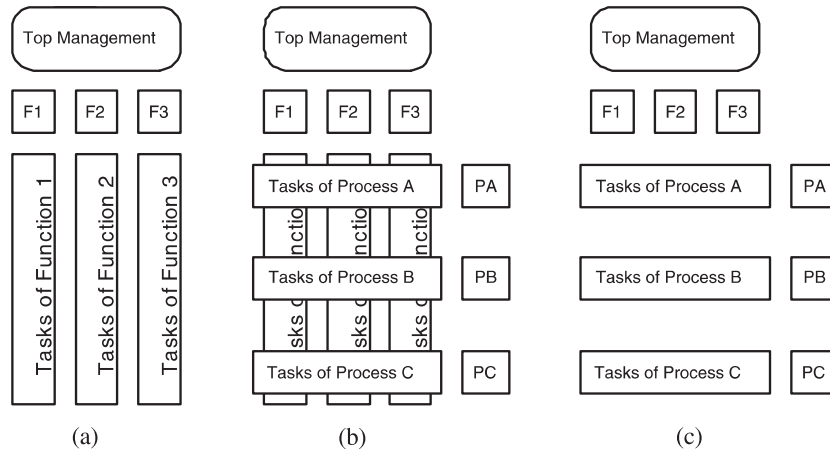


Fig. 1. Functional and process-oriented organizations

in functional organizations with integration of processes (Fig. 1b), the goals of the processes should have priority over the departments' ones.

The existence of an organization makes only sense if it provides utility to the clients. Four principles can shape the role and structure of an organization [11]:

1. The mission of an enterprise is to create added-value to its clients;
2. The processes of an enterprise are the main source for creating added-value to its clients;
3. The success comes from excellence;
4. The superior performance of a process depends on having the right persons to execute it and on settling the right working environment.

In client-oriented and, consequently, process-oriented organizations, it is mandatory an easy adaptation to changes, so that the clients' needs are continuously satisfied. This favours the continuous improvement of every aspect of the enterprise, being it process-, product-, or organizational-related.

Information technologies are among the principal factors to permit a process-based restructuring of a given organization [33]. The development of a software application for organizations of this kind must consider their process framework. Thus, the software engineering processes must take into account the organization structure. With this model, the application becomes more useful to the organization, and maintenance is facilitated since no major modifications and adaptations to the process framework are needed.

Software is often the realization of the business rules [26] and, whenever these rules change, the software must also change accordingly. Nevertheless, the mere computer-based implementation of the business processes that already exist in an organization does not show all the advantages that that implementation could offer [9]. It is desirable that, in parallel with the implementation activities, some effort is undertaken in redefining and optimizing the way work is done. The performance of an

organization will get better if both the automation of the old activities/processes, and the constant improvement of its business processes are considered.

The validation of process improvements must always take into account the business perspective and must be evaluated according to the creation of value to the client of that process [33]. The use of object-oriented (OO) techniques to conceptually model an organization and its processes permits the analysis of information systems and the re-engineering of the business processes to become a unique activity [21], and requires a common set of artefacts [9]. The adoption of the OO paradigm and UML, for modelling both the organization and its information system, is supported by RUP.

4 Processes in organizations

Business processes can be viewed as a set of related tasks, executed to achieve a given business goal. Within the business process, human resources, raw material, and internal procedures are combined and synchronized towards a common objective. An enterprise can be viewed as a group of competence centres, which share the same goal [11] (Fig. 2).

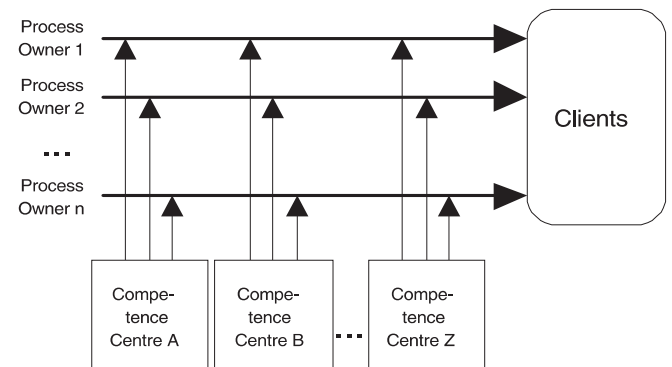


Fig. 2. Generic structure of a process-oriented organization

For each competence centre, a coach must be attributed, whose mission is to manage the members and provide training to them, and to assure the uniformity of technologies related with his centre. The coach is also supposed, when requested, to assign members of his centre to the processes being created. The owner of a process is responsible for the success of his process, according to the clients' objectives.

In this type of organizations, the members are no longer dependent on an hierarchical chain of command. Each member must have expertise in one area where he performs his functions. For that, he is helped by a coach to improve his capacities/skills and by a process owner to coordinate his activities with those of the other team members.

The career within these organizations does not follow the typical promotion to higher positions in the hierarchy, thus reducing the problem of transforming a good technician into a bad chief. Instead, an element progresses within his area of competence, and he does not need to become a chief or manager.

5 Generic framework for process-oriented organizations

Generically, an organization exists to supply a set of products or services to its clients, with added-value for itself. To achieve this, the organizations need to execute a set of internal activities. Actually, organizations do not exist isolated, but are part of markets where other organizations can be their competitors, suppliers, clients, or partners. The wealthy creation, within private organizations, or the supplying of products or services with social impact, within non-profit organizations, follows the same underlying principle: the fulfilment of the clients' needs. Thus, these needs and the expectations of the clients must be considered a vital element for defining the internal structure of an organization.

A process inside an organization can be viewed as a set of activities that has as inputs a set of services and/or materials and has as outputs also a set of services and materials. This view must be oriented towards the necessities of the client and the creation of added-value. This implies that the clients' requirements must always be considered, both in the design and in the performance of the system.

In an organization, there are other processes rather than those that provide added-value to the clients. The existence of different types of processes are necessary, to assure, for example, the strategic planning for the organization, the recruitment of the human resources, or the fiscal duties. As illustrated in Fig. 3, these processes are instantiated in Management and Support Processes.

Within an organization, the management by processes requires a structure that differs from the typical functional hierarchy. It is mandatory to synchronize the processes among them and to fulfil the strategic objectives

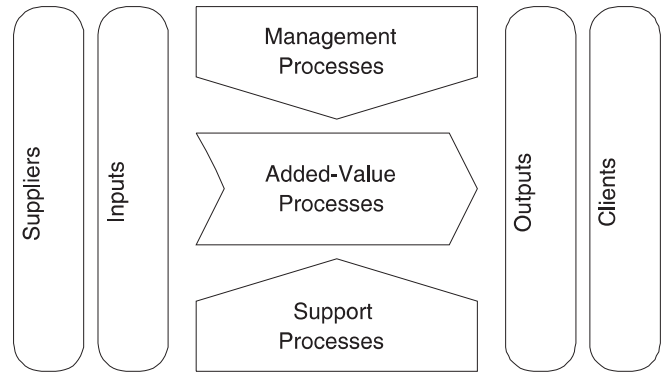


Fig. 3. Framework for an organization's processes

of the organization. Thus, for a process-oriented organization, a structure with the following elements should exist:

Process Management Top Team: This team includes the top managers and all process owners. Its mission is to revise all the processes according to the strategic objectives of the organization, to analyze the effectiveness of the process-oriented management (by changing, for example, the process owners), and to decide about unsolved problems at the processes' interfaces. This team is supposed to meet every year.

Process Godfather: One godfather, which must be a top manager, should be designated for each process. His mission is to help and instruct the process owner, to decide when there is a problem of interface among processes, to determine the strategic orientation of the process (e.g. to decide that the development process of a given product has higher priority over another one), and to assure that the process is uniform within the organization, i.e., that the development process of a new product is similar to the development of other products.

Process Owner: For each process, an owner is needed. He must have know-how on managing processes and persons, but also competency in the areas associated with the process. His mission is to lead the process' multi-disciplinary team, being also responsible for the process design, measurement, improvement, and efficiency. The process owner and the godfather should designate the members of the multi-disciplinary team. This team must be constituted by elements with proven expertise on the several sub-processes and activities. Its cardinality must allow an effective management: lack of elements may provoke absence of knowledge and capacity of intervention, while excess of elements may cause delays on the decisions, difficulties in scheduling meetings, or overlapping of knowledge areas.

Multi-Disciplinary Team: This team must be created for each added-value process, since they represent the most important processes for the clients. For management and support processes, and considering the organization size and its strategic objectives, multi-disciplinary teams may also be established. This decision must be taken by the process owner and godfather. The mission

of this team is multiple: to monitor its process, to define and analyze the key indicators and the process objectives, to ensure that the process documentation is updated, to decide when and how to use improvement teams and to coordinate them, to manage the process execution teams (e.g. to define the number of teams, the criteria of segmentation, the members validation). This team must meet in a weekly or monthly basis.

Execution Teams and Team Leaders: These teams and their leaders represent the instances of a given process [30]. Therefore, during the execution of a process, some teams will use it with a specific focus. For example, for a given production process, one team may be responsible for producing parts for industrial clients, while other team may produce them for individual clients. This partitioning must be managed by the process' multi-disciplinary team, but the team leader is supposed to suggest its composition. Each leader must obey the definitions and use the indicators of the process which he is responsible for. It is possible that some processes have only one execution team, whose members are the same as those of the multi-disciplinary team, but representing different roles. The execution teams must meet each week, since they daily execute activities and tasks.

To align a process-based organization with its strategic objectives, it is crucial that the goals are based on the organization's mission and vision, and also on its principles and values. Based on those strategic objectives and in the business plan, the priority when deciding the key processes within the organization can be perceived.

We propose the following pragmatic procedure, to introduce the processes in an organization:

1. Define the processes: identify the collection of processes in an organization and decide on their relative importance;
2. Define the sub-processes and tasks: each process is decomposed into sub-processes, until it is reached a level of detail where only primitive activities remain;
3. Identify the interfaces among processes;
4. Assign the owners and godfathers to the processes and the members to the multidisciplinary teams;
5. Define the goals of the processes;
6. Define the key indicators;
7. Define the processes instances' team leaders and executing teams;
8. Measure and monitor the processes;
9. Execute corrective actions;
10. Review and continuously improve the processes.

The identification of the key processes within an organization (step 1 in the previous enumeration) must be accomplished with the help of some established criteria. Some of those criteria are:

- Identify if the process traverses several functions; if this is not the case, we might just have an activity or a task;

- Verify if the process is measurable and is integrated in the business plan; otherwise, the process must be terminated, since it is not aligned with the organization's strategy;
- Verify if there is a focus towards the clients and their requirements; when this does not happen, the process may be inefficient;
- Determine the importance of the process for the final quality of the product;
- Determine the importance of the process for the fulfilment of the organization's mission;
- Verify if the process is multi-disciplinary; if that is not the case, we may be in the presence of an activity or a task;
- Find out the importance of the process to the success of the organization.

Based on these criteria, we can select which processes of a given organization will have a process-oriented management and the activities and tasks they will be composed of. Some processes, activities, or tasks can be eliminated if they do not add any value to the clients, neither to the organization. These eliminated (or redefined) processes, activities, and tasks and their respective consequences in terms of reorganization and impact in human resources are the essence of re-engineering [11].

Besides this general framework (Fig. 3), inside each process category (management, support, and added-value), there are several processes that define in detail the types of activities that exist inside organizations (Fig. 4a). The business processes are designed with the aim of bringing value to the organization. This value must be quantified to serve as a basis to trace the process state and possible improvements. Some general indicators for each process were proposed [23]:

1. Productivity: relation between the output value and the input cost;
2. Added Value: the output value minus the input cost;
3. Cycle Time: difference between the begin and the end times of running a process instance;
4. Queue Size: medium size of the queue with the requests to execute the process;
5. Quality Indices: number of defects as a percentage of the process instances already executed.

In the management category (top of Fig. 4a), three generic business processes are proposed:

- Global Strategy (GS): Within this top-level management process, potential clients are evaluated, the innovations are planned, and the general policies for the enterprise, such as its mission, its vision, its principles, its values, and its long-term objectives, are defined. Additionally, the change in the organization to run effectively the processes must be considered;
- Policy Deployment (PD): The organization's policy and short-term objectives, usually defined in an annual basis, are unfolded and delivered to the competence centres and processes;

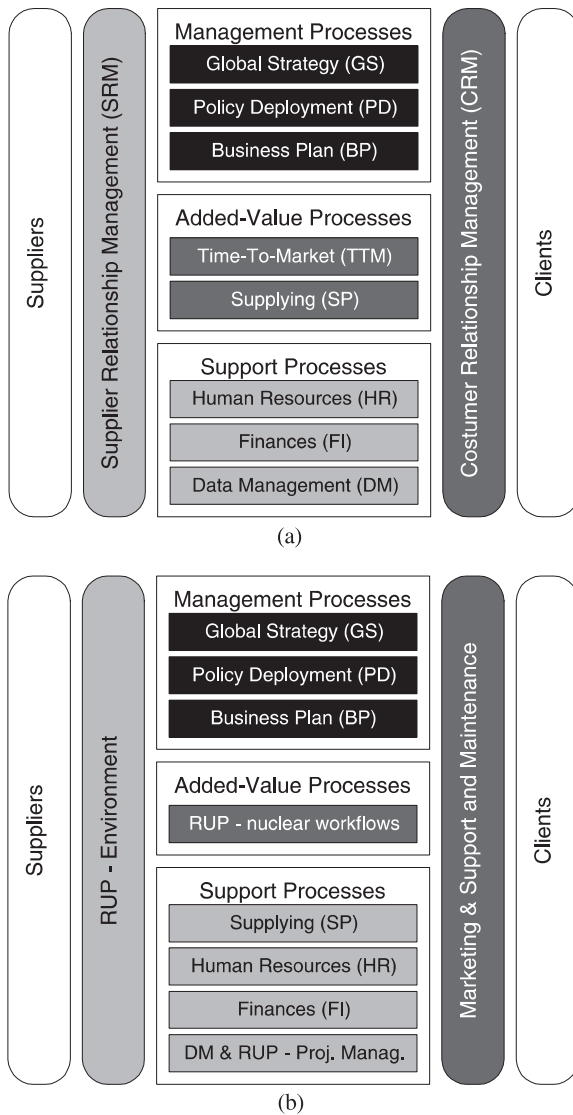


Fig. 4. **a** General composition of the processes within a generic organization; **b** General framework for the processes of organizations that develop software

- Business Plan (BP): The planning and the budgeting of the activities and their distribution among processes and competence centres are defined. Normally, these activities are planned with an annual scope. The activities for controlling the execution of the plan are also established.

In the added-value category (centre of Fig. 4a), there are three general business processes:

- Time-To-Market (TTM): This represents the development process associated to a new product and the environment needed to produce it. Usually, this process occurs between the discovery of a business opportunity in the market and its complete development;
- Supplying (SP): This process occurs normally after the TTM for the same product, and starts with a request from the client, or with a decision of production, and ends with the deliver of a product to the client

with the fulfilment of the required schedules and quality levels;

- Customer Relationship Management (CRM)¹: This process includes activities such as marketing, after-sales service, management of clients' accounts, and generally all activities that allow clients to be satisfied with the products or services supplied to them.

In the support category (lower part of Fig. 4a), we suggest four generic business processes:

- Supplier Relationship Management (SRM)²: This process includes activities such as the selection of the materials' suppliers, the execution of audits to verify the quality, or the establishment of agreements for just-in-time delivers;
- Human Resources (HR): This process includes activities related to hiring people, salaries management, and generically all issues associated with the well-being and qualifications of the organization's collaborators;
- Finances (FI): All the activities related to fiscal issues belong to this process;
- Data Management (DM): This process includes activities associated to data quality and the indicators of the organization, the security of data, the priority of computer related projects and their budget control, data-warehousing and data-mining.

It is important to indicate that if the processes are described in great detail, it will be difficult to implement them. In fact, processes must be simple, so that they can be followed and updated [34]. The idea is to maintain the processes as generic and abstract as possible, to allow their instantiation to a broader class of organizations.

The emphasis and priority, and also the specific activities of each process, depend primarily on the type of organization where they will run, and secondly on the strategic objectives of the top management. The activities related to the continuous improvement, to the quality and environment management (including certification tasks), are all examples of activities that tend to be confused with processes. These activities are part of all processes, such as those of Fig. 4a, and should not have a separate management as that followed for the management, added-value and support process categories.

6 Framework for organizations that develop software

Since the process framework presented in Fig. 4a describes a generic organization, it can also serve as a basis for modelling an organization that develops software

¹ Although the CRM process is mainly seen as an interface process, it was also included in the added-value category, since client-oriented organizations must address the customer needs.

² The SRM process is mainly classified as an interface process, but it was also included in the support category, because it generates inputs for added-value processes.

(Fig. 4b). Being a more specific type of organization, it allows us to add more details.

In this section, we analyze and compare the specific aspects of the model in Fig. 4b with the one in Fig. 4a that was in its origin, highlighting those topics that differentiate them, namely indicating how the RUP disciplines can be used. This constitutes the major contribution and innovation of this paper. In organizations that develop software, we propose the business processes to have the following meaning:

- Any type of organization needs management processes. Thus, the processes named Global Strategy (GS), Policy Deployment (PD), and Business Plan (BP) are equivalent to those of any other organization, although we must take into account the particularities of the software market, such as the rapid changes in technology and the competition in worldwide markets.
- Since software is an intangible product, it is obvious that no raw materials are needed to produce it. For organizations that develop software the SRM process is instantiated in RUP's Environment discipline, since it furnishes the working environment (e.g. development tools), which will be used by the teams, and the development guidelines to be followed by the teams within the organization.
- The Supplying (SP) process consists essentially in creating copies of an application. In contrast with more traditional industries, where it represents probably the most important process, in software, due again to its intangible nature, this is a trivial process. Usually, the kind of materials used to support it, CDs, DVDs, and printed manuals, implies that external suppliers are hired. The outsourcing of this process comes from the fact that it is considered to be secondary for an organization that develops software. Therefore in this kind of organization, this process is a support one.
- The Time-To-Market (TTM) process is mapped into the set of RUP's core disciplines: Business Modelling, Requirements, Analysis and Design, Implementation, Test, and Deployment. This set of activities, or sub-processes, run in parallel for the same development project [29]. In our opinion, this is the most critical process for an organization that develops software.
- The Human Resources (HR) process for software factories is the same as for other types of organizations. We must however point out that software development requires highly-specialized people, being their hiring a critical issue for the success of the organization. It is impossible to produce quality software without skilled people.
- The Finances (FI) process is the typical fulfilment of the fiscal obligations, which is common to all types of organizations.
- The Customer Relationship Management (CRM) process is divided in two sub-processes: Marketing, and

Maintenance and Support. This ensures that, when a software application is delivered to the final clients, its life-cycle does not end, but instead continues with this process, incorporating changes and corrections and providing training to the users, while the application is being used by the clients. The Marketing assumes, in this case, a form similar to that observed in other types of organizations.

- The Data Management (DM) process, in what concerns the data related to RUP's core disciplines, is mapped into the RUP's discipline 'Project Management'. In this discipline, some activities lead to the production of indicators of the project status. Its existence is the foundation to take decisions based on facts, related to the advance of the project and also to adjust and improve the software development process.

To reach the highest CMM levels, the constant improvement of the development processes must be part of each process, instead of being a single autonomous process. CMM level 3, which is considered the minimum when discussing about the software process [12] is reachable with RUP if it is extended according to the proposals made in [22] or in [32].

RUP's core disciplines, that implement the added-value process in Fig. 4b, are sub-divided in activities, which can be viewed as sub-processes. The description of those sub-processes is made with UML activity diagrams (an example is presented in Fig. 5), complemented optionally with other type of diagrams, such as interaction diagrams, and business object diagrams. This represen-

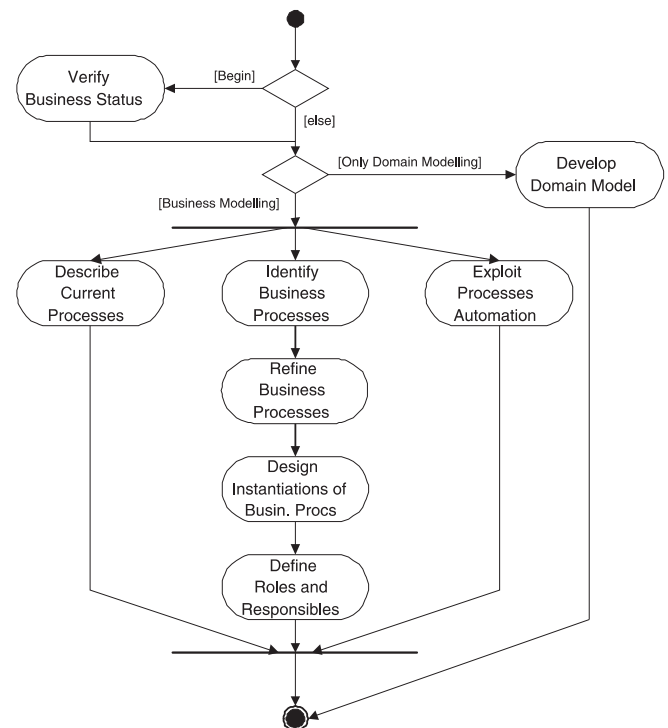


Fig. 5. Activity diagram in the business modelling in RUP

tation is also valid for all other processes of a generic organization, as illustrated in Fig. 4a. Whenever an organization that develops software executes a development process, the TTM process will be executed. Since we are proposing this process to be implemented by the six RUP's core disciplines, it implies that the discipline 'Business Modelling' will also be executed. This discipline strongly recommends the use of activity diagrams (and other complementing diagrams) for modelling purposes. Therefore, the target organizations, as proposed in Fig. 4a, will be modelled by a collection of these diagrams. In parallel, within the organization that develops software, since the discipline 'Business Modelling' is a sub-process of the TTM process, it can also be modelled by activity diagrams (Fig. 5).

7 Business process modelling

In this section we concentrate on 'Business Modelling', because of the six RUP's core disciplines, it is the one most directly related to the clients' needs. During the development of software, all the stakeholders must have a common understanding of the business processes that exist in the target organization. This reality is not circumscribed to the obvious organizational information systems, but can also include real-time and embedded systems if they affect the business processes.

If the development of applications does not take into account the current business processes (or those to be implemented), the result will be probably unsuccessful. This may be caused by the fact that the end users are not using correctly the application, since it does not support directly the activities under their responsibility. Within RUP, the business processes model is obtained in the 'Business Modelling' discipline (Fig. 5).

The main activities of this discipline are centred around the identification, refinement, and realization of the business processes and in the definition of the roles of people associated to the business. Each role in this RUP's discipline (e.g. the agent that executes some activity within the organization that develops software) has under its responsibility the execution of several activities that will have as deliverables several artefacts (Table 1).

It is important to note that the activities of Table 1 are at a higher level of detail than those of Fig. 5. For example, the activity 'Refine Business Processes' includes the activities 'Structure the business use case model', 'Review the business use case model', and 'Detail business use cases'.

Among all the activities and their respective artefacts, only some are mandatory. This flexibility permits the configuration of RUP, so that it can be adapted to a specific project executed in a given organization. Thus, taking into account the options made when choosing the artefacts, the following set allows modelling business processes [29]:

- **Business Vision:** This artefact captures the goals of a particular business modelling activity, stating what is to be modelled and the reasons for it. It also serves as an oracle to all future high level decisions.
- **Business Glossary:** In this artefact all business terms and expressions are kept. They are necessary for a good understanding among all project stakeholders.
- **Business Rules:** The business rules correspond to policy statements and conditions that should be fulfilled, from the business perspective. They are similar to systems' requirements, but they focus on the business core, expressing rules related to business, but also its architecture and style. Its modelling must be rigorous, being one possibility the usage of the Object

Table 1. Roles, activities and artefacts for business modelling in RUP

Role	Activity	Artefact
Analyst of the Business Process	Verify target organization	Business rules
	Establish and adjust objectives	Business use case model
	Capture the vocabulary of the business	Business glossary
	Find business actors and use cases	Business object model
	Maintain the business rules	Business vision
	Structure the business use case model	Supplementary business specification
	Define the business architecture	Target organization verification Business architecture
Reviewer of the Business Model	Review the business use case model	
	Review the business object model	
Designer of the Business	Detail business use cases	Organizational units
	Find business workers and entities	
	Define the automation requirements	
	Detail business entities Detail business workers	

Constraint Language (OCL) as specified in UML [24]. Alternatively, using a natural language in a structured form is also admissible, since even though there are clear advantages in using a more formal approach, the understanding by all the stakeholders is usually paramount. In our case, the business rules are modelled with structured English, using the following constructors [25]: (1) IF; (2) ONLY IF; (3) WHEN; (4) THEN; (5) ELSE; (6) IT MUST BE ALWAYS VERIFIED; (7) IT WAS CORRECTLY COMPLETED.

- **Business Use Case Model:** The main goal of this artefact is to show how the business is being perceived and run by all stakeholders. This is achieved by modelling the business processes and their interactions with external parties, based on use case diagrams (with stereotypes for business use cases and business actors) [5]. The business processes are associated to a discipline that needs to be modelled, and that specify how added-value is created for the business actors. This modelling can be supported by activity diagrams, possibly extended with the representation of organizational units interfering in the business process and with the distribution of the activities by those organizational units. This allows a better understanding about ‘who is doing what’.

8 Case study

The proposals made in this paper are being introduced in Blaupunkt/Braga, in what concerns the organization’s structure and the adoption of a RUP-based process and UML. We next report on some of the lessons learned from one of the projects that were conducted at Blaupunkt/Braga.

The project, entitled ‘Premium Wage’, consists on the development of a software application to calculate the payment of extra money to employees, based on their pro-

ductivity [7]. This project was classified as critical, since it may have important social and behavioural impacts on the organization, if the amount is badly calculated or if it is impossible to explain how it was obtained. This mechanism was introduced with the objective of improving the organization’s overall productivity and to return the excellence to the workers.

Besides its criticality, the business process is also complex due to its dependency from other processes. In this case, the payment of a premium depends on three main factors: individual absenteeism, quality of the products made in the employee’s line, and individual performance. The first two sub-processes were extended in order to support new functionalities. For the third, a complete reengineering was carried on. Finally, for the premium wage calculation a new process was designed, modelled, and implemented.

In the project, the selected development process is extensively used and we evaluate the capacity of the process to cope with complex organizations. The final application requires the complete re-engineering of some business processes, the extension of functionalities into new business processes, and finally the design and implementation of a new business process and its supporting application.

8.1 Business modelling

The technology-planning horizon for big companies is now a synthesis of software engineering and process engineering [31]. Additionally, the success of a project depends heavily on the correct perception of the business process to be modelled. Taking into account these two aspects, the RUP’s Business Modelling discipline assumes a critical role in the software development process and therefore a special attention is paid to it in this section.

This discipline can generate the artefacts presented in Table 2, and has the following objectives:

Table 2. Artefacts in business modelling

Artefact	Used	Motivation
Business Rules	yes	Rigor in describing a new business process
Business Use Case Model	yes	First description of the functionalities and the organization’s business actors
Business Glossary	no	The business terms are common to the target and software development organizations (they are sub-organizations of the same organization)
Business Object Model	yes	Realization of the Business Use Cases
Business Vision	no	The business vision is common to the target and software development organizations
Supplementary Business Specification	no	The Business Use Case and the Business Object Models are sufficient
Target Organization Verification	no	The target organization is perfectly known by the developers. The current processes are modelled by the Business Use Case Model – Current Situation
Business Architecture Document	no	The details present in the Business Use Case Model and in the Business Object Model are sufficient
Organizational Units	yes	Mapping the business process functionalities on the target organization’s structure

- To understand the structure and dynamics of the organization where the system will be executed;
- To comprehend the current problems of the target organization and to identify potential improvements;
- To assure that clients, final users, and developers have a common understanding about the target organization;
- To capture/deduct the requirements of the system necessary to support the target organization.

The procedure for configuring this discipline in this case study was also adopted for describing the other RUP's disciplines.

RUP can be parameterised and used both in small and complex projects and we next discuss what artefacts were produced. For this parameterisation to occur, it is necessary to choose which artefacts to use and their level of detail. On Table 2, we show which artefacts of the Business Modelling discipline were used in the project and the arguments considered to decide on its usage. This choice was validated by the quality assessments. Thus, both the subset of used artefacts and also its degree of detail can not be anticipated with rigor, but must be selected based on experience and knowledge of the development team in relating the characteristics of each project with the functionalities offered by the artefacts.

The criteria to fulfill this choice are related with:

- Characteristics of the project itself (e.g. criticality of the modelled business processes, type of target organization);
- Characteristics of the organization that develops software (e.g. team size, level of knowledge about internal rules);
- Temporal restrictions. Since resources are limited in engineering projects, it is always necessary a balance between the quantity and detail of the produced artefacts and the deadlines for implementing the project.

The produced artefacts result from a set of activities that occur inside those disciplines. In this case study, we identified the need for the artefacts to represent two distinct situations in terms of business: one part of the project represents reengineering activities of some business processes, while the other part represents the introduction of a new business process.

In several diagrams (e.g. Business Use Case Model), the standard UML is augmented with the stereotypes defined by RUP, thus allowing the creation of RUP-like artefacts.

8.2 Other relevant techniques

Besides the artefacts and the techniques suggested by RUP, we believe that it is also important to document other issues associated with the development of a software application:

- Classification of requirements according to the FURPS+ approach [10];

- Checklist to assess the project status in the Quality Verifications that evaluate the transitions between the process phases (inception, elaboration, construction, transition);
- Planning and analysis of the required capacity to support and maintain the system during the productive phase [28].

8.3 Results analysis

We now analyze the positive and negative points associated with the parameterisation of RUP to implement the application under development. The main positive issues were:

- RUP, but especially the artefacts of the Business Modelling discipline, had a good performance in modelling the business processes, since they served as a basis for the artefacts at lower levels;
- The UML artefacts, with the stereotypes suggested by RUP, were a very efficient communication medium among all the stakeholders;
- The artefacts generated by RUP, namely the source code, have an easy utilisation for future systems that need them, allowing thus the reuse of code;
- The development process standardisation in the organization that develops software was initiated to adopt the usage of RUP;
- RUP proved that it can adapt to the needs of organizations that develop software and specific projects;
- The set of RUP's disciplines allows to cope with the complexity and ensures that no important aspect related to the software development is disregarded, diminishing the risk of failure.

We also identified the following main negative aspects:

- The usage of RUP and UML must be accompanied with training for the developers;
- The need to create new roles to execute the software development process, such as the business process designer or the tools expert, forces internal reengineering.

8.4 Lessons learned

With this project, the lessons learned were the following:

- It is very important that all stakeholders are aware of the tasks related to the use of a new software product. In this way all stakeholders, as suggested by RUP, should know and validate all the requirements.
- The mapping of use cases into classes was accomplished via the following methodology [5]: (1) Transform system use cases into classes of 3 categories: Interface, Control, Data; (2) From the full set of classes generated, only the meaningful ones (depending of the description and use case realisation) are kept; (3) Aggregate the similar remaining classes (e.g. database-related classes); (4) Create associations between the

groups of classes. From the remaining class groups, a mapping into a 3-tier architecture was inferred. The class groups, Interface, Control, and Data, were distributed into the three tiers, Presentation, Business, and Data, respectively. This method provided a mechanism that led to a time improvement in reaching the final architecture for the system.

- In matrix organizations, some problems, external to the clients and to the software development teams, arise just because of the existence of departments. When the departments are among the stakeholders for a project, they block the software development, because it will model business processes that can threaten the existence of tasks inside departments or even the departments as entities.
- The RUP usage, even in the first projects, is not itself a delay factor to reach the final solution.

9 Conclusions

In this paper, we have presented a generic framework for process-oriented organizations, which serves as a foundation to model organizations. This framework is adequate for modelling target organizations. Based on this framework, we have also proposed a process framework for organizations that develop software, the corresponding way of managing it, and the instantiation of their processes with RUP disciplines. The concepts and ideas were exemplified in one case study, where the usage of UML and RUP, to develop software for a process-oriented organization, were experienced.

The modelling capabilities of a graphical modelling language, such as UML, and the understanding that it offers to all the stakeholders is a crucial factor to avoid communication and interpretation errors and to improve the solution's utility and correctness.

The existence of a process, within an organization that develops software, that ensures control over the activities being carried out and guarantees the quality of the final solution is of major importance. However, more important is the continuous adaptation and change of the process to match changing circumstances.

Whenever the target organization can be modelled as a process-oriented organization, with the structure depicted in Fig. 4a, RUP can handle its modelling and the transformation of the business processes into computer applications supporting them. RUP can be used for systems already in execution, developed initially without a process-oriented perspective, allow their integration in a new system, by re-engineering the business process, through the business use case diagrams (and their respective use case realizations) for the current and future situations.

Acknowledgements. This work was partially funded by Fundação para a Ciência e a Tecnologia (FCT) and Fundo Europeu de Desenvolvimento Regional (FEDER), under project “METHODES:

Methodologies and Tools for Developing Complex Real-Time Embedded Systems” (POSI/37334/CHS/2001).

References

1. Booch G, Rumbaugh J, Jacobson I (1999) The Unified Modeling Language User Guide. Addison-Wesley
2. Cantor M (2003) Organizing RUP SE Projects, RationalEdge, July
3. Curtis B, Kellner MI, Over J (1992) Process Modeling. Communications of the ACM 35(9):75–90
4. D’Souza DF, Wills AC (1998) Objects, Components, and Frameworks with UML: The Catalysis Approach. Addison-Wesley
5. Fernandes JM, Machado RJ (2001) From Use Cases to Objects: An Industrial Information Systems Case Study Analysis. In: 7th Int. Conf. on Object-Oriented Information Systems (OOIS '01), Springer-Verlag, pp 319–28
6. Fernandes JM, Duarte FJ (2003) A Reference Model for Process-Oriented Software Development Organizations. In: González-Pérez C, Henderson-Sellers B, Rawsthorne D (eds), OOPSLA 2003 Workshop on Process Engineering for Object-Oriented and Component-Based Development, pp 31–42
7. Fernandes JM, Duarte FJ (2004) Using RUP for Process-Oriented Organisations. In: Bomarius F, Iida H (eds), 5th Int. Conf. on Product Focused Software Process Improvement (PROFES 2004), Lecture Notes in Computer Science 3009, Springer-Verlag, pp 348–62
8. Firesmith D, Henderson-Sellers B (2002) The OPEN Process Framework: An Introduction. Addison-Wesley
9. Fowler M (1997) Analysis Patterns: Reusable Object Models. Addison-Wesley
10. Grady R (1992) Practical Software Metrics for Project Management and Process Improvement. Prentice Hall
11. Hammer M (1996) Beyond Reengineering: How the Process-Centered Organization Is Changing Our Work and Our Lives. Harper Collins
12. Henderson-Sellers B (2000) The OPEN Framework for Enhancing Productivity. IEEE Software 17(2):53–8
13. Henderson-Sellers B, Dué R, Graham I, Collins G (2000) Third Generation OO Processes: A Critique of RUP and OPEN from a Project Management Perspective. In: 7th Asia-Pacific Software Engineering Conf. (APSEC'00), IEEE CS Press, pp 428–35
14. Henderson-Sellers B, Collins G, Dué R, Graham I (2001) A Qualitative Comparison of Two Processes for Object-Oriented Software Development. Information and Software Technology 43(12):705–24
15. Henderson-Sellers B (2003) Method Engineering for OO Systems Development. Communications of the ACM 46(10):73–8
16. Hesse W (2003) Dinosaur meets Archaeopteryx? or: Is there an alternative for Rational’s Unified Process? Software and Systems Modeling 2(4):240–7 DOI 10.1007/s10270-003-0033-y
17. Hirsch M (2002) Making RUP Agile. In: OOPSLA 2002 Practitioners Reports. ACM Press
18. Hull MEC, Taylor PS, Hanna JRP, Millar RJ (2002) Software Development Processes – An Assessment. Information and Software Technology 44(1):1–12
19. Jacobson I, Booch G, Rumbaugh J (1999) The Unified Software Development Process. Addison-Wesley
20. Krutchen P (1999) The Rational Unified Process: An Introduction. Addison-Wesley
21. Machado RJ, Fernandes JM (2002) Heterogeneous Information Systems Integration: Organizations and Tools. In: Oivo M, Komi-Sirviö S (eds), 4th Int. Conf. on Product Focused Software Process Improvement (PROFES 2002), Lecture Notes in Computer Science 2559, Springer-Verlag, pp 629–43
22. Manzoni LV, Price RT (2003) Identifying Extensions Required by RUP (Rational Unified Process) to Comply with CMM (Capability Maturity Model) Levels 2 and 3. IEEE Transactions on Software Engineering 29(2):181–92

23. Marshall C (2000) Enterprise Modeling with UML: Designing Successful Software through Business Analysis. Addison-Wesley
24. Object Management Group (2001) OMG Unified Modeling Language Specification version 1.4. version 2001.03.00
25. Odell J (1998) Advanced Object-Oriented Analysis & Design Using UML. Cambridge University Press
26. Pressman RS (1997) Software Engineering: A Practitioner's Approach. 4th edition, McGraw-Hill
27. Priestley M, Utt MH (2000) A Unified Process for Software and Documentation Development. In: IEEE Prof. Comm. Society Int. Prof. Comm. Conf. and Proc. of the 18th Annual ACM Int. Conf. on Computer Documentation: technology & teamwork, IEEE Educational Activities Dept., pp 221–38
28. Ramaswamy R (2000) How to Staff Business-Critical Maintenance Projects. IEEE Software 17(3):90–4
29. Rational Software Corporation (2003) Rational Unified Process: Product Overview, August
30. Scheer AW, Nüttgens M (2000) ARIS Architecture and Reference Models for Business Process Management. In: van der Aalst W, Desel J, Oberweis A (eds) Business Process Management, Models, Techniques, and Empirical Studies, Lecture Notes in Computer Science 1806, Springer, pp 376–89
31. Smith H, Fingar P (2002) Business Process Management: the Third Wave, Meghan-Kiffer Press
32. Smith J (2000) Reaching CMM Level 2 and 3 with the Rational Unified Process. WhitePaper, <http://www.rational.com/products/whitepapers/100416.jsp>
33. Spurr K, Layzell P, Jennison L, Richards N (1994) Software Assistance for Business Re-Engineering, John Wiley & Sons
34. Ward RP, Fayad ME, Laitinen M (2001) Software Process Improvement in the Small. Communications of the ACM 44(4):105–7



João M. Fernandes is an assistant professor at the Department of Informatics, Universidade do Minho (Braga, Portugal). He received a Lic. degree in Informatics and Systems Engineering in 1991, a M.Sc. degree in Computer Science in 1994, and a Ph.D. degree in Computer Engineering in 2000, all from Universidade do Minho.

His Ph.D. thesis, entitled “An Object-Oriented Methodology for Embedded Systems Development”, addresses the usage of object-oriented concepts, namely UML, to analyze, design, implement, and test embedded systems. From Sep/2002 until Feb/2003, he was a post-doctoral researcher at the TUCS Embedded Systems Laboratory (Turku, Finland). His research interests focus on Embedded Software, Hardware/Software Co-Design, Methodologies for System Development, Software Modelling, Software Process and Management, and History of Computing. For more information, consult his website at <http://www.di.uminho.pt/~jmf>.



Francisco J. Duarte is responsible for the Information Systems group at the Department of Informatics, Blaupunkt Auto-Rádio Portugal (Braga, Portugal). He received a Lic. degree in Informatics and Systems Engineering in 1993, and a M.Sc. degree in Informatics in 2002, both from Universidade do Minho. His M.Sc. thesis, entitled “Process-Oriented Software Engineering”,

addresses the usage of object-oriented concepts and processes, namely UML and RUP, to model process-oriented organizations and to develop software systems for those organizations. His research interests focus on Software Engineering Processes, Business Process Management, Business Modelling, and Requirements Analysis.