

Workflows without Engines: Modeling for Today's Heterogeneous Information Systems

Cristina Chuva Costa

Instituto Politécnico de Coimbra/Centro de Informática e Sistemas da Universidade de Coimbra, Coimbra, Portugal

chuva@isec.pt

Paulo Rupino da Cunha

Centro de Informática e Sistemas da Universidade de Coimbra, Coimbra, Portugal

rupino@dei.uc.pt

Abstract

Workflow modeling has been able to handle requirements down to the point where the specification gets executed on a specialized engine. Present day information systems, however, pose new challenges. Data and decision logic are spread out across a collection of heterogeneous applications of different ages, provided by various vendors, supported on diverse technologies. "Running" a business workflow thus means orchestrating several applications that were not designed to work together. Middleware and other technological solutions can be used to engineer coordination and communication between the systems, but, at a higher level, modeling techniques must be adjusted so that their inputs and deliverables fit this new "engineless" execution environment. We describe an approach we have used to bring together generic information systems design and workflow-specific concerns to produce an architecture that easily maps into today's heterogeneous scenarios. Special care has been taken to ensure that end-users are able to contribute throughout the process.

Keywords: workflow, heterogeneous systems, requirements, modulation

Introduction

A series of phenomena have changed the nature of information systems. Instead of the traditional heavy reliance on custom programming, common off-the-shelf (COTS) software is now purchased to satisfy most business needs. On the other hand, companies have accumulated several

applications throughout the years, effectively creating a portfolio of systems that interact as business processes are carried out. Finally, the emergence of intranets provides means of leveraging the information stored in these various applications in a uniform and distributed manner.

This new reality means that business data and business logic are spread

Material published as part of this publication, either on-line or in print, is copyrighted by the Informing Science Institute. Permission to make digital or paper copy of part or all of these works for personal or classroom use is granted without fee provided that the copies are not made or distributed for profit or commercial advantage AND that copies 1) bear this notice in full and 2) give the full citation on the first page. It is permissible to abstract these works so long as credit is given. To copy in all other cases or to republish or to post on a server or to redistribute to lists requires specific permission and payment of a fee. Contact Publisher@InformingScience.org to request redistribution permission.

through dozens or even hundreds of autonomous applications “wired” together to meet the organization’s diverse needs. This heterogeneous collection of systems, supported on disparate technologies, provided by various vendors, at different times of the organization’s life, is the supporting infrastructure for the high-level business workflows. For instance, a request made by a client using the company’s call center may be recorded in the underlying customer relationship management system (CRM) from vendor A and forwarded to the relevant department to process it using the enterprise resource management system (ERP) supplied by vendor B.

The transfiguration of information systems into highly heterogeneous portfolios of applications rendered traditional design methodologies inadequate, since their deliverables – clean architectures of data and logic to be coded – ceased to be in sync with actual field reality (Avison & Fitzgerald, 1999; Baskerville & Fitzgerald, 1995). Workflow modeling faces similar challenges. Although high-level analysis and planning must still be conducted, this process must now acknowledge as input the “fragmentation” of information and processing logic across several heterogeneous applications, and produce results that can be deployed in these “engineless” environments.

We describe how we extended a recent information systems design methodology, conceived to handle heterogeneous portfolios of applications (Cunha, 2000), to include workflow modeling.

The remaining of this paper is organized in the following way: in section “Approaches to Workflow Modeling” we briefly review various existing approaches to workflow modeling, looking for practices or aspects that remain adequate when handling heterogeneous information systems. These aspects were considered when developing our proposal, which is presented in section “Taking on Heterogeneity”. The paper closes with a discussion of the results.

Approaches to Workflow Modeling

To avoid blindly proposing a new approach, we identified and reviewed several, quite distinct, workflow modeling techniques, to understand to which extent they could cope with the new reality of heterogeneous information systems. The detailed analysis has been published in (Costa & Cunha, 2002). In this section we merely summarize it and draw brief conclusions, pointing out which characteristics in each approach remain useful in today’s information systems.

Function Modeling Techniques

Although not originally intended to model workflows, dataflow diagrams (DFD) were adopted for this purpose by some authors (Sadiq, Marjanovic, & Orłowska, 2000). The main appeal of this approach is the effective way in which it represents data circulation and processing. Another similar technique – the integration definition for function modeling (IDEF0) diagrams (Air Force Wright Aeronautical Laboratories, 1981) – also uses simple graphical elements to represent the flow of information between activities. However, both types of diagram pose a serious handicap: in their original form they are unable to represent time, and need supplementary specification techniques to clarify control logic details. The notations also exhibit some limitations when applied to ill-structured situations and to scenarios with many interactions. Finally, DFDs and IDEF0 diagrams are hard to understand by end-users, whose knowledge of business process is fundamental to the correct specification of workflows.

Graphs and Petri Nets

Several authors suggested the use of graphs to model workflow processes (Aalst, 1995; Casati, Ceri, Pernici, & Pozzi, 1995). A proposal by Sadiq and Orłowska, 1996 includes different types of objects: flows, tasks, conditions and synchronizers. These objects enable the specification of

workflows by building and joining different constructs, such as ordering, alternative, exclusive join, concurrency, synchronization and iteration.

Different researchers preferred Petri nets over graphs (Peterson, 1997). The order in which the tasks are to be executed can be modeled by Petri Nets using constructs for sequential, conditional, parallel, and iterative routing. Despite its simplicity, Petri nets are not succinct and manageable enough to be useful in modeling complex business processes (Leyman & Roller, 1999). In order to surpass these difficulties, several extensions have been proposed in the last few years. Three well-known ones are color, time and hierarchy, to address, respectively, data, temporal aspects and large models. A Petri net extended with these characteristics is called a high-level Petri net (Aalst, 1999).

In their formal semantics, graphs and Petri nets emphasize the graphical representation of the sequence of workflow activities over the specification of the data. Modeling with graphs and Petri nets may, however, result in complex and less intuitive structures that, as it happened with DFDs and IDEF0 diagrams, inhibit an effective dialog with non-specialists.

The Language Action Paradigm

The Language Action paradigm (Medina-Mora, Winograd, Flores, & Flores, 1992) is quite different from the techniques described so far. It is based on the speech-act theory, and elects natural language as the tool for the planning and coordination of actions.

Using this approach, a business process is seen as a conversation between two actors (customers and providers of a service) (Medina-Mora et al., 1992) and it is represented as a closed-loop, ending with explicitly stated customer satisfaction (Carlsen, 1997).

The dialogs between customers and providers consist of four phases: “Proposal”, “Agreement”, “Performance” and “Acceptance”. In any of the four phases new actions can be added, such as clarifications or additional negotiation details related with the commitments between the participants (Medina-Mora, et al., 1992). This decomposition of dialogs creates a network of actions and commitments.

Language Action assumes that the participation of end-users and their expectations are extremely important in modeling workflow processes. This concern leads to the use of natural language, trading off formalism for easiness of communication. A potential pitfall of this approach resides in the assumption of a very explicit communication between all actors, which is not always achieved.

Unified Modeling Language Extensions

Since the Unified Modeling Language (UML) – used in software engineering to graphically depict the architecture of a system – established itself as a standard, some authors proposed adapting and extending it to encompass workflow modeling (Eshuis & Wieringa, 2001). One such example is the WIDE methodology (Baresi & Casati, 1999) that puts strong emphasis on the requirements phase and describes the business process using three perspectives:

Functional – represents the operational structures of the business processes by means of UML use cases, interaction diagrams, class diagrams and activity diagrams.

Organizational – models aspects related to roles, actors and organizational units (e.g. departments) involved in a business process. The organization, as a whole, can be described as a set of UML packages that cooperate between them, each having its own actors.

Business – models goals to be achieved by the business processes, decomposing them into sub-processes that are closely related to UML activities.

The UML diagrams used by the WIDE methodology represent data, control sequences and interactions. By explicitly identifying organizational units (such as departments) the WIDE methodology enables a global awareness over the context of the various existing flows. However, this methodology doesn't represent the workflows between those organization units.

Since WIDE makes use of UML, it is not particularly suited to engage in dialog with end-users, in view of the fact that its modeling elements are highly technical.

Use of Patterns

The pattern concept was originally introduced by the architect Christopher Alexander, to describe how several autonomous elements (such as "front porch" or "courtyard") could be joined together to create a harmonious building (Alexander, 1979). The idea was picked up in software engineering by (Gamma, Helm, Johnson, & Vlissides, 1994), to describe the context of common situations or problems, together with the algorithms to handle them.

The high level of abstraction associated with pattern descriptions is also an effective means of identifying, generalizing and modeling recurrent workflow scenarios in a manner independent from technology or specific of a particular application.

Currently, many workflow design patterns have been identified, ranging from fairly simple constructs available in any workflow language (such as sequence, parallel split or exclusive choice), to complex primitives (synchronizing merge, implicit termination or deferred choice) not yet supported by today's generation of workflow management systems (Aalst, Hofstede, Kiepuszewski, & Barros, 2000).

By holding solutions to common situations, patterns promote the reuse of knowledge. The analyst must, however, be familiar with the catalog of pre-identified patterns in order to match them to the situation at hand. The construction of such catalogs of solutions is also a significant endeavor.

Summing up

Recently, information systems have been transfigured from quite homogeneous software solutions into an entangled web of applications from different vendors that are used to support different parts of the business. Frequently, business workflows must be run on top of this heterogeneous multi-vendor, multi-technology, multi-era network, instead of on top of a "proper" specialized engine. The workflow modeling techniques we just briefly reviewed were introduced at different points in time, and shaped to provide answers to diverse challenges. Although present reality calls for new adjustments or proposals, the following characteristics, of the various approaches, remain important:

- The specification of information flow between activities (Function Modeling Techniques);
- The representation of the workflow control perspective (Graphs and Petri Nets);
- The participation of end-users in the requirements analysis (Language Action);
- The representation of the organizational structure (Unified Modeling Language extensions);
- The reuse of existing solutions (Patterns).

In the next section we will introduce a workflow modeling extension that we have added to a recent information systems design methodology (EISD) (Cunha & Figueiredo, 2001), already conceived to handle heterogeneous portfolios for today's applications.

Taking on Heterogeneity

In our effort to address workflow modeling over heterogeneous systems, we departed from a methodology, called Evolutionary Information Systems Design (EISD) recently conceived to cope with new challenges posed by the growing use of common-off-the-shelf software (COTS), the use of intranets and extranets, and the need to integrate and evolve legacy systems (Cunha, 2000; Cunha & Figueiredo, 2000, 2001). We extended it to encompass the specificity of workflow modeling. Although generic information systems design traditionally uses concepts different than those of workflow modeling (i.e.: data models and procedures or objects vs. activities, roles and documental flows), EISD narrows this semantic gap by using higher granularity concepts – entities and responsibilities – that closely model organizations' departments or teams and the services they are accounted for, respectively.

By systematically identifying and documenting all business entities and their interactions via responsibilities, this modeling approach affords an organizational model that blends in the information system. Full-scale applications, or ERP modules, such as financial, human resources or employee relationship management, may support the inner workings of a responsibility.

The EISD methodology uses business entity cards to document each business entity (consisting of a name, a list of its responsibilities and its client entities). On the other hand a responsibility card is used to detail information about each service: name, purpose, importance, main rules, exchanged data, business procedure to carry out and available information systems support are specified.

Figure 1 exemplifies how EISD would model a typical tendering system, with the relationships between all involved entities (identified with the bold designations and represented by large rectangles) and their responsibilities (smaller grey rectangles with their designation on the right side). Being external to the organization, supplier's responsibilities are not assigned an id.

As can be observed in Figure 1, the Marketing Team schedules a meeting with several Suppliers to obtain quotations for the desired items (R1). After analyzing the information, the Marketing Team asks for a final budget from suppliers that offered better conditions (R2). The Suppliers submit their proposals to the Administrative Services, which collect and deliver the proposals to the Executive Committee (R4).

The Executive Committee produces a ranked list of proposals (R6) and forwards it to the Administrative Services that subsequently send a notification of the acceptance to the selected Supplier. Finally, the chosen Supplier delivers the requested goods to the Marketing Team, which, after verifying conformance (R3), sends the invoice to the Administrative Services to process payment (R5).

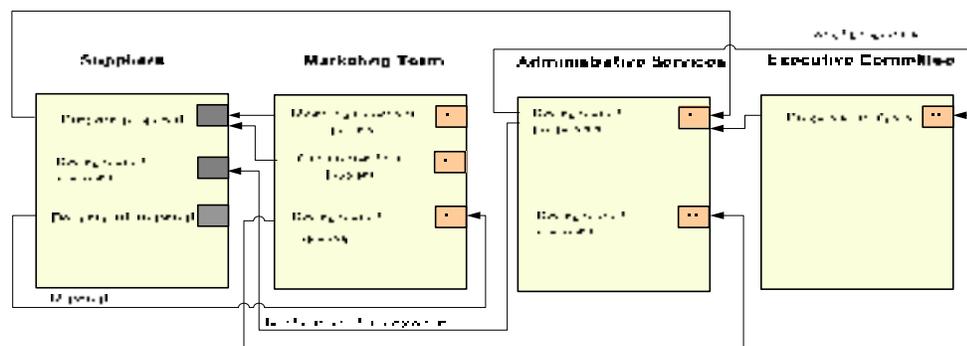


Figure 1. Acquisition process modeled by the EISD methodology

EISD doesn't have a specific technique to model all the complexities related with the "interaction protocol" (i.e.: the specification of how the information is supplied), but gathers all the information necessary to specify the data circulation between entities, typical of workflow modeling.

Considering the potential offered by the EISD, we developed a new workflow requirements approach that represents the documents circuits between entities/responsibilities and describes their business procedures. This extension focuses on the responsibility interface, through which interaction with an entity is possible.

After constructing the organizational model according to the original EISD methodology recommendations, it's possible to identify scenarios where workflow mechanisms should be included to specify in detail all the protocols between the responsibilities. We propose to address this issue in the workflow extension to EISD according to three perspectives:

- Intra-responsibility workflow - describes rules and document-related conditions inherent to a given responsibility.
- Document descriptions - provides detailed information about every document.
- Inter-responsibility workflow - describes the connections between responsibilities that cooperate in the execution of a business process and the documental circuit of that cooperation.

Intra-responsibility Workflow

Regarding the flow of documents inside a responsibility, five possible alternatives were identified:

- Documents created in a responsibility;
- Documents that end their flow in a responsibility;
- Documents for which the responsibility is an intermediary point in their overall circuit;
- Joining of documents in a responsibility;
- Splitting of documents in a responsibility.

To describe these cases, two forms were created that identify the responsibility being analyzed, using a title and a code.

The first form (Figure 2) describes the relation between each responsibility and each document it processes, namely, the responsibility position in each document flow (creation, terminus or intermediary point), the activities performed by the responsibility on the documents, the conditions associated with those activities and the way the responsibility handles the document (e.g., digital or paper format).

<i>RESPONSIBILITY NAME: RECEPTION OF GOODS</i>		<i>RESPONSIBILITY ID: R3</i>	
Document name	<i>Requisition form</i>		
Intermediary Points <input type="checkbox"/>	<i>Source</i>		
Creation <input checked="" type="checkbox"/>	Creation condition	<i>Issue a Request For Proposal (RFP)</i>	
	Responsible	<i>Marketing Team</i>	
Terminus <input type="checkbox"/>	Termination condition		
	Responsible		
	Source		
Actions to perform			
A ₁	<i>Check the delivery and sign the "Requisition form"</i>	Pre-condition	<i>All the goods specified in the RFP being delivered</i>
		Post-condition	
		Responsible	<i>Marketing Team</i>
		Destiny	<i>Supplier</i>
		Duration	<i>5 minutes</i>
		Maximum of duration	<i>15 minutes</i>
		Frequency of use	<i>1 time per month</i>
? _x		Responsible	
Procedure	<i>Paper forms are physically moved between entities</i>		
Observations	<i>Nothing to declare</i>		

Figure 2. Form describing the creation of the “Reception of goods” document

The simple case, depicted in Figure 2, illustrates how the document “Requisition form” is created in the responsibility “Reception of Goods” (“R3”) by the Marketing Team. The activities (represented by “Ax”) performed on the document are also described. If conditional activities existed, they should be written in the field marked “?x”.

The second form characterizes operations of joining or splitting of documents in a responsibility, identifying the responsible for that operation and describing the condition that originates it. In the example presented in Figure 3, several supplier proposals are joined in the responsibility “Proposal Analysis” by, the Executive Committee, to originate the document “Ranked list of proposals”.

<i>RESPONSIBILITY NAME: PROPOSAL ANALYSIS</i>		<i>RESPONSIBILITY ID: R10</i>
Junction <input checked="" type="checkbox"/>	Documents involved	- Model Design INC Proposal - 3LM INC proposal
	Resulting document	Ranked list of proposals
Disjunction <input type="checkbox"/>	Document's source	
	Resulting documents	
Responsible	Executive Committee	
? ₁ jd	All the fields necessary to originate the "Process of candidature" being filled in the two proposals	
Procedure	If the two proposals have all the information necessary to produce the process of candidature, this should be created by the executive committee	
Observations		

Figure 3. Form describing the joining of proposals into a new “Ranked List” document

This form is needed only for operations of joining or splitting documents inside a responsibility.

The joining or splitting operations originate new documents, new flows and also the circuit ending of the incoming documents. These modifications must be reflected not only in the form responsible for describing these operations (Figure 3), but also in the first form (Figure 2). For this reason, it's necessary to establish a relation between both forms used in the intra-responsibility perspective. The linkage between the two forms is provided by the “?xjd” field – the joining or splitting condition represented in the form of Figure 3 by “?1jd”. The “x” was substituted by “1” since the field in question was the first joining or splitting condition in that responsibility. This field should be used as the “Creation condition” or “Termination condition” in a form identical to the one in Figure 2.

It is possible to identify several responsibilities in a business model, like the one described in Figure 5. For this reason an index is attributed (represented by x) to the acronym “R”, used to represent the number of the responsibilities. The acronyms “A” (Activity) and “?” (Conditions) also have indexes associated.

Textual descriptions (in natural language) were used in the forms to promote more effective communication with end-users. The aim is to encourage their participation in the analysis of workflow requirements.

Description of Documents

The necessary information to characterize documents isn't restricted to items related with the specification of circuits and with the execution of activities, mentioned previously. It's also important to describe document's properties such as deadlines, formats, priority levels, passwords and attachments.

The form illustrated in Figure 4 was developed with the purpose of gathering all the information related with a specific document, by means of achieving a complete description of all the items relevant to its workflow requirements specification.

The replication of information that exists between the form responsible by the description of a document (Figure 4) and the forms that characterize the document's flow among organizations

(exemplified in Figure 2 and 3) is justified by the advantages of having in just one analysis tool all the necessary information to specify the workflow requirements of a document.

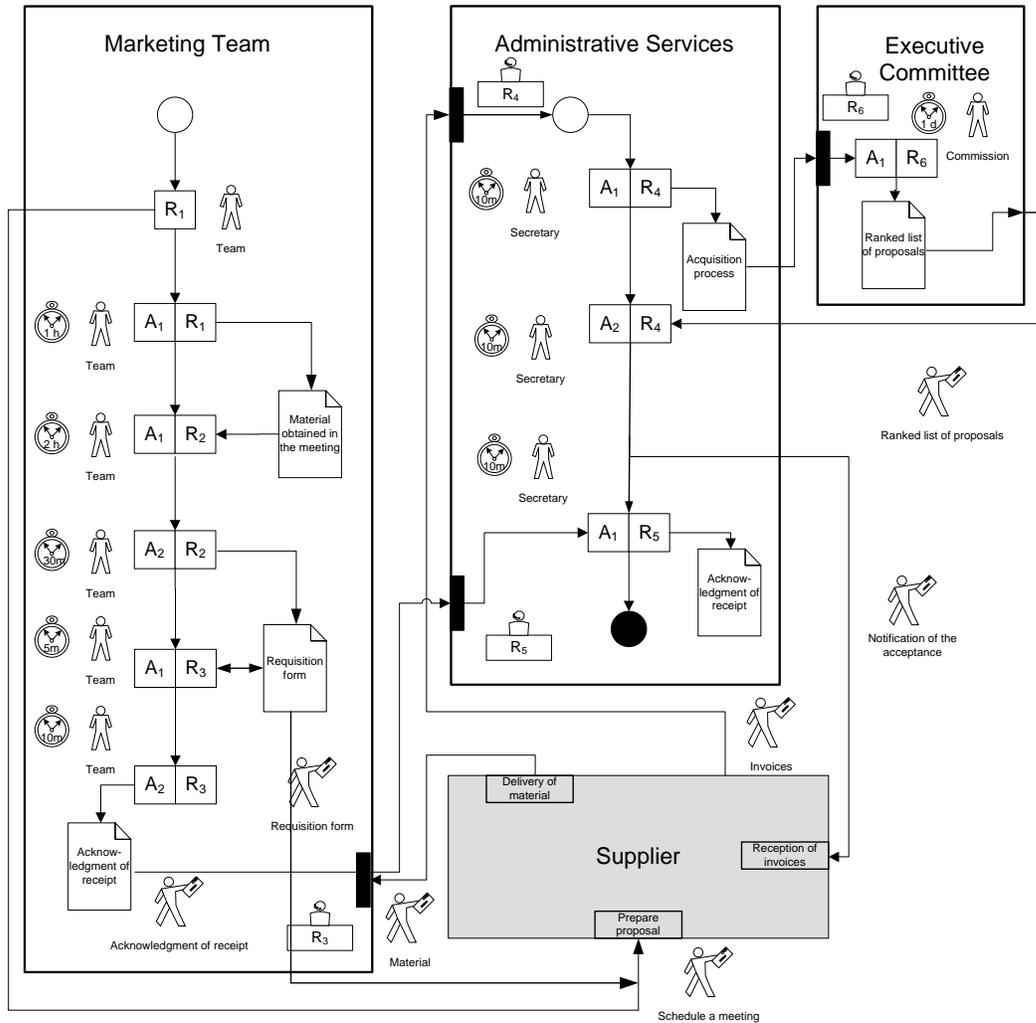
<i>DOCUMENT NAME: REQUISITION FORM</i>		<i>DOCUMENT ID: D1</i>
Date of creation	<i>08/06/2005</i>	
Document source	<i>Entity: Marketing Team Responsibility: Reception of materials</i>	
Intermediary point(s)	<i>Non existent</i>	
Document destiny(ies)	<i>Entity: Supplier (External entity)</i>	
Fields required	<i>All</i>	
Existing dependencies	<i>Non existent</i>	
Activities performed on the document	<i>Entity: Marketing Team Responsibility: Reception of materials ▪ Check the delivery and sign the "Requisition form"</i>	
Document conditions	<i>Creation condition - Issue a Request for Proposal (RFP) A₁ Pre-condition - All the goods specified in the RFP being delivered</i>	
Deadlines of the activities	<i>Supplier - 08/07/2005</i>	
Possible operations on the data	<i>Register and lecture operations</i>	
Document format	<i>Web pages</i>	
Attach	<i>Non existent</i>	
Priority of the document	<i>Median</i>	
Passwords	<i>Non existent</i>	
Access frequency and number of users	<i>Access numbers - 4 times per month Number of users - 2</i>	
Importance of the supporting information system	<i>Reduce the bureaucracy related with the reception of material</i>	
Informatics knowledge of the users that interact with the document	<i>Median</i>	
Description of how the support systems should represent the document behavior	<i>The Marketing Team may access the form "Requisition form" via an intranet, checking automatically the reception.</i>	
Observations	<i>The Marketing Team didn't show many interest in developing a support system to the document</i>	

Figure 4. Form used to describe a document

With this specification process it is possible to access in a straightforward and centralized way all the details that characterize a particular document.

Inter-responsibility Workflow

Tests and interviews with end-users revealed the necessity of representing graphically the flow of documents between entities, as well as the activities performed over the documents by the responsibilities involved in a business process. Figure 5 illustrates this representation for the previous example.



Legend

- R₁ – A₁ – Take notes about the proposal
 - R₂ – A₁ – Analyse quotations
 - A₂ – Make the invoice
 - R₃ – A₁ – Check the delivery and sign the "Requisition form"
 - A₂ – Register information in the "Acknowledgment of receipt"
 - R₄ – A₁ – Organize all the documents related with the proposal
 - A₂ – Process the decision of the executive commission
 - R₅ – A₁ – Register administrative information in the "Acknowledgment of receipt"
 - R₆ – A₁ – Create the ranked list of proposals
- | | | | |
|--|-------------------------------------|-----------------------|-------------------|
| - Access to the responsibility | - Document | - Flow starting point | - Entity |
| - Duration and responsible of the activity | - Send a document to another entity | - Flow ending point | - External entity |

Figure 5. Intra-responsibility form for the acquisition of materials

In this diagram, the entities from Figure 1 (large rectangles) are retained, together with their responsibilities (smaller, grey rectangles). Extra detail, regarding the inner workings of these elements is added. This perspective allows for perception over dataflow as well as control issues.

After the Marketing Team schedules a meeting with several Suppliers (external entity), the document “Material obtained in the meeting” is created and subsequently analyzed with the purpose of creating a “Requisition Form”, which will be delivered to the suppliers that offered better conditions. The Suppliers submit their proposals to the Administrative Services, which organize all the documents related with the proposal in a folder/document designated “Acquisition Process” and send it to the Executive Committee.

The Executive Committee establishes a “Ranked list of proposals” and delivers it to the Administrative Services that processes the decision of the Committee and sends a notification of the acceptance to the selected Supplier. Finally, the chosen Supplier delivers the requested goods to the Marketing Team, which, after verifying conformance and filling the “Acknowledgment of receipt”, sends the invoice to the Administrative Services that finish completing the fields in the “Acknowledgment of receipt” document.

The functionalities executed by each responsibility are represented independently of the technological solution adopted, providing the specification of requirements that may involve the integration of different workflow technologies, including the solutions used by external entities.

The symbols used in the diagram are simple, encouraging communication with end-users and reinforcing their participation in the specification of the requirements of a workflow management system.

All the analysis tools illustrated in section “Taking on Heterogeneity” should be used together, since they approach different perspectives of the specification of heterogeneous workflow requirements. To maintain the consistency between these tools the nomenclature used is always the same.

Conclusion

The extension we propose in this article entails a solution, which encloses in a single methodology the design of information systems and its workflow components, examining the adjacencies of these two worlds and, consequently, minimizing the effort expended on the specification of systems.

To achieve this proposal the most widespread techniques for the specification of workflows have been reviewed, in order to identify characteristics that should be retained. Those characteristics are: data and sequence representation, the importance of client participation in the analysis phase, the representation of document’s flows in the organization and the advantages of software reuse in the development of heterogeneous workflow management systems.

We departed from the EISD methodology because its underlying concepts were already close to those required by workflow. Its use of entities and responsibilities emphasized relationships and the encapsulation of functionalities. Furthermore, the collected information could easily be expanded to fully specify workflow management systems. As it happened with the original EISD, there is a separation between business process description and technological solutions that support them. This enables the use of Enterprise Application Integration techniques to articulate various heterogeneous systems on top of which the designed workflow must run.

References

- Aalst, W. (1995). *A class of Petri net for modelling and analysing business processes*. Eindhoven, Eindhoven University of Technology.
- Aalst, W. (1999). Interorganizational Workflows: An Approach based on Message Sequence Charts and Petri Nets. *Systems Analysis - Modelling - Simulation*, 34(3), 335-367.
- Aalst, W., Hofstede, A., Kiepuszewski, V., & Barros, A. (2000). *Workflow patterns*. Eindhoven, Eindhoven University of Technology.
- Air Force Wright Aeronautical Laboratories. (1981). *Integrated computer- aided manufacturing (ICAM) architecture - Function modeling manual (IDEF0)*. Ohio, Air Force Wright Aeronautical Laboratories.
- Alexander, C. (1979). *The timeless way of building*. New York: Oxford University Press.
- Avison, D. & Fitzgerald, G. (1999). Information systems development. In W. Currie & R. Galliers, *Rethinking management information systems*. New York, Oxford University Press: 250-278.
- Baresi, L. & Casati, F. (1999). *WIDE workflow development methodology*. Milano, Politecnico di Milano.
- Baskerville, R. & Fitzgerald, B. (1995). Beyond systems development methodologies: Time to leave the lamppost? In *Information Technology and Changes in Organisational Work, Proceedings of the IFIP 8.2 Conference*, UK.
- Carlsen, S. (1997). Conceptual modeling and composition of flexible workflow models. *Department of Computer Science and Information Science*. Trondheim, Norwegian University of Science and Technology.
- Casati, F., Ceri, S., Pernici, B. & Pozzi, G. (1995). Conceptual modeling of workflows. In *Proceedings of the 14th International Object-Oriented and Entity-Relationship Modelling Conference*, Gold Coast.
- Costa, C. & Cunha, P. (2002). Especificação de workflows num contexto de aplicações heterogêneas: reflexão sobre as técnicas existentes. In *Actas da 3ª Conferência da Associação Portuguesa de Sistemas de Informação*, Coimbra, Portugal.
- Cunha, P. R. (2000). Projecto de sistemas de informação para a realidade emergente: proposta baseada num modelo de carteira de soluções. Departamento de Engenharia Informática da Universidade de Coimbra. Coimbra, Universidade de Coimbra.
- Cunha, P. & Figueiredo, A. (2000). Information systems design under a different light. In *AMCIS*, California, USA.
- Cunha, P. & Figueiredo, A. (2001). Information systems development as flowing wholeness. In *IFIP WG8.2*, Idaho, USA.
- Eshuis, R. & Wieringa, R. (2001). A formal semantics for UML activity diagrams - Formalising workflow models. Twente, Department of Computer Science - University of Twente.
- Gamma, E., Helm, R., Johnson, R. & Vlissides, J. (1994). *Design patterns - Elements of reusable object-oriented software*. Massachusetts: Addison-Wesley.
- Leyman, F. & Roller, D. (1999). *Production workflow: Concepts and techniques*. New Jersey: Prentice Hall.
- Medina-Mora, R., Winograd, T., Flores, R., & Flores, F. (1992). The action workflow approach to workflow management technology. In *ACM Conference Supported Cooperative Work (CSCW'92) - Emerging technologies for cooperative work*, Toronto, Ontario.
- Peterson, J. (1997). Petri Nets. *Computing Surveys*, 9(3), 223-252.
- Sadiq, S., Marjanovic, O., & Orłowska, M. (2000). Managing change and time in dynamic workflow process. *The International Journal of Cooperative Information Systems*, 9(1&2), 93-116.
- Sadiq, S. & Orłowska, M. E. (1996). *Modelling and verification of workflow*. Queensland, University of Queensland.

Biography



Cristina Chuva Costa obtained her BSc. in Computer Science at the University of Coimbra in 1998 and her MSc, in the area of Workflow Specification, from the same University in 2002. She has been teaching classes of computing programming, systems analysis, computational architectures and simulation in the Computer Science Department of the Polytechnic of Coimbra since 1999.

Currently she is working on her PHD at the University of Coimbra. Her studies are related with the use and contribution of social theories, like Actor Network Theory, in the specification of requirements in information systems.



Paulo Rupino da Cunha is an Assistant Professor of Informatics Engineering and Information Systems at the Faculty of Science and Technology of the University of Coimbra, Portugal, where he lectures since 1993. He holds a Ph.D. (2001) and an M.Sc. (1997) in Informatics Engineering - Information Systems from the University of Coimbra. His main research interests centre on Information Systems Design, Quality Management and Information Systems, Design of Business Models and Strategy, Enterprise Application Integration and IT Investment Valuation. He is the Vice-President of Instituto Pedro Nunes, an Innovation and Technology Transfer organization providing specialized consulting, training and business incubation. He is the Scientific Director of the informatics volume of the Portuguese

encyclopedia Nova Activa Multimédia and a member of the board of the Portuguese Information Systems Association. For a period of three years, he was the elected Coordinator of the Informatics Engineering Chapter for the center region of Portugal of the Portuguese Engineering Association and for a two year term he held the Vice-Presidency of the Department of Informatics Engineering of the University of Coimbra. He also served as Deputy Director of the Informatics Lab at Instituto Pedro Nunes.

Paulo is the author of various papers. He serves in the Program Committee or as a reviewer for various international conferences. He is currently a member of the editorial review board of the International Journal of e-Collaboration. He was a member of the editorial board of the journal of the Portuguese Engineering Association.