

**Some reflections on applying Workflow Technology
to Software Process**

Camilo Ocampo
Pere Botella

Report LSI-98-5-R

Some reflections on applying Workflow Technology to Software Process¹

Camilo Ocampo², Pere Botella

Departament de Llenguatges i Sistemes Informàtics

Universitat Politècnica de Catalunya

Campus Nord - Edifici C6. C/ Jordi Girona Salgado, 1-3

E-08034 Barcelona, Catalonia (Spain)

{cocampo, botella}@lsi.upc.es

<http://www-lsi.upc.es/~gessi>

January 1998

Abstract

Process Technologies (e.g., Workflow and Software Process) support co-operative work, where humans and Information Systems participate within an organisation. In the last two decades Workflow Management Systems have been demonstrated to be useful in the automation of Business Processes. We think Software Process is part of a more complex Business Process, and it is composed by several sub-processes, but with different granularity of detail. In recent years, environments have been created to automate Software Processes, but only few products are available today in the market, and they have shortcomings that can be addressed by Workflow Technology. Such technology can improve Software Process Technology leading to a Process Technology able to manage and automate other kind of Processes. The aim of this article is to reflect our ideas about the possibility of applying Workflow Technology to Software Process and vice versa.

1. Introduction

Different efforts have been made to manage and automate human work in several areas of the Software Engineering, as well as other technological and scientific fields. One extended approach to view human work is from a *process-centred* perspective, e.g., in *Transaction Management* [WS97], *Software Process* (SP) [FKN94], and *Workflow Management* (WM) [SGJ+96].

On the other hand, several disciplines focus their research from a different perspective: the human co-operation, i.e., *Human Co-ordination*, *Computer Supported Co-operative Work* (CSCW) [CACM91], *Human-Computer Interface*. We name both process-centred and human co-operation approaches as *Process Technologies* since they have a common issue: to support the co-operative processes among agents.

The aim of this article is to reflect our ideas about the possibility of applying Workflow Technology to SP and vice versa. We believe that the large experience on Workflow and the amount of tools available today could be used to extend and improve SP environments. Our position comes from the experience of using Workflow Tools in private companies, and preliminary hypothesis that we are trying to confirm with experiments on final student projects. We have also participated in a national project [OB96] developing a Workflow Tool.

¹ This work is part of and it has been partially supported by the project OBJECTFLOW, a co-operation with the *Centro de Cálculo de Sabadell* (CCS), a company from Spain.

² Supported by CONACyT, Mexico.

In this paper, we focus our attention on SP and WM. We think that applying some of the methodological and technological advances in both disciplines will improve the maturity of the Process Technology. It also could be translated to manage and automate any kind of processes.

We consider, as other authors [GS97] also, that a SP is a special case of a Business Process, but it has some particular issues that makes it different from traditional office processes. A SP necessarily implies high integration of tools (e.g., compilers, debuggers, CASE tools and so on), and high level of description of the task that compose the activities of the process. Modelling and management of processes in SP, have been treated with a fine granularity of description [BFGL94, BNF96, EG94], e.g., modelling the relation among tools. In WM the focus has been normally put in higher levels of abstraction with a more coarse granularity, i.e., processes are described as a set of steps that normally are considered as black boxes.

Regarding process support, research community on both WM and SP, have adopted different approaches ranging in a continuum from human to system orientation, but normally without covering conveniently this extent [SGJ+96].

The organisation of this article is the following: Section 2 presents co-operation issues related to the work among agents using computer support. Section 3 describes two Process Technologies, WM and SP, comparing their similarities and differences. Section 4 reviews the state of art in technological support for processes. Section 4 describes the different characteristics and the support that an organisation could need according to its process maturity level. Finally, we present our conclusions. A comparative table about WM and SP characteristics is presented in Annexe A.

2. Co-operation

One common issue present in the different approaches of Process Technology is the *co-operation* among agents (software or humans) working together to achieve a common goal within an organisation.

Co-operation is composed on four elements: *collaboration*, *communication*, *co-ordination*, and *coupling* [Yang95, CR97]. Collaboration is the creation and management of shared information. Communication is the exchange of knowledge and information among agents. Co-ordination is the management and focusing of efforts from a group or organisation of agents. Coupling is the use of compatible views of shared information.

From a temporal point of view, co-operation can be classified either as *Asynchronous co-operation* or *Synchronous co-operation* [EGR91]. In asynchronous co-operation, agents work at different times on the same information. In synchronous co-operation, agents work at the same time and place that could be a *virtual* one.

In practice, a process cannot be fully classified under either asynchronous or synchronous co-operation, because co-operative work is characterised by alternating periods of individual and joint work. Examples of such co-operative scenarios [AKT+96] are co-operative authoring, design for manufacturing, Software development, and Workflow applications.

Figure 1 shows the two opposing views in the role of the human in the process management, i.e., defining, enacting, and evolving the processes. On one hand, the process *emerges* as result of the co-operation among humans: they decide the number and order of the activities to be realised, or they assign the work to members of the team or other teams. On the other hand, human interaction is explicitly predefined and controlled by a process model. Changes in the interactions must be made through changes in the proper model. To determinate in which of the process views (emergent o predefined) is the agent placed, we should know who has the control: either the human or the model; and where the process evolution takes place: implicitly in the actions of the performers or explicitly in the Process definition. Another as-

pect regarding a process, is who does the work, humans or Software. Activities can be performed manually or automatically, in both cases, humans and Software can participate, but responsibilities and functionalities change depending on the type of activity. In manual activities, the human does the job normally assisted by a Software or a machine. In automatic activities, tasks are performed by a Software application or an Information System (IS) and the human intervention is minimal, e.g., providing some data input.

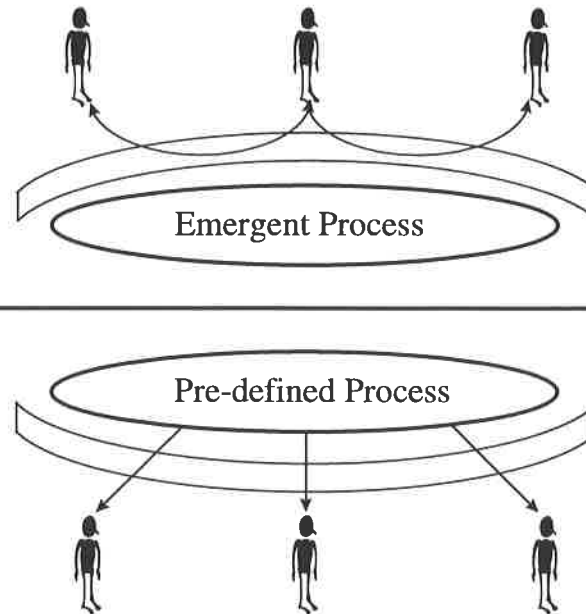


Figure 1. Human role in Process Management (Taken from [SGJ+96])

3. Process Technology

Process Technology supports co-operative processes where agents participate, which can be Software Systems or humans, co-operating to achieve a common goal within an organisation or a set of them, generally following a set of rules. WM, SP, and Advanced Transaction Processing could be considered Process Technologies.

One common issue of these technologies is the human agent and Software participation in processes. Often, they participate together performing activities, but playing different roles. Sometimes humans do the work supported by Software; other times, Software Systems perform a task and humans play a *secondary* role, e.g., entering input data. This kind of human interaction has also been named as *strong* and *weak* interaction respectively. The support for the process goes from human-oriented to system-oriented depending on the quantity of human or systems agents [GHS95].

Any process is composed of four basic elements [CKO92]: *agents*, *roles*, *artefacts* and *activities*. Agents could be humans or software systems, and they have responsibilities to accomplish an activity performing a role. Artefacts are produced from raw material or other products as result of completing activities or the whole process. Agent's work could be supported by tools.

A Process can be described from different perspectives, Curtis, Kellner and Over, mentioned the most common, which are *functional*, *behavioural*, *organisational*, and *informational* [CKO92]. Functional perspective represents how the process is performed by agents and which information is needed for them. Behavioural perspective describes when the agents perform their work, fulfilling a process element; and it represents the process flow. Organisational perspective shows where and which agents are assigned to each process element. Informational perspective describes items produced or manipulated by agents in each process

element. These perspectives have relationships among them, and it is difficult (if not impossible) to represent them with a single graphical language.

Modelling and automating a process is composed of three steps: *specification*, *implementation*, and *enactment*. In the specification phase, requirements are elicited to construct the different views of the process model. For the implementation, process model views are translated (automatically or manually) to a format that can be enacted by a computer. In the enactment phase, instances of the process are executed and the agents work co-operating and following the steps defined in the process model. From the moment that the process is enacted, deviations could begin to occur and they must be managed, then changes must be reflected in the process model. Different environments exist to support the three phases of the modelling and management of different kind of processes, e.g., SP or Business Processes (BP).

Next we describe WM and SP issues, commenting on those that are common in both disciplines, or could be important to cover some lack in Process Technologies.

3.1 Workflow Management

WM is an emerging discipline that has evolved during the last decade principally from Industry. More than two hundred commercial *Workflow Management Systems* (WMS) exist in the market. For this reason, and because their origin is far from the *scientific world*, many definitions have been given around WM and several buzzwords have been created.

To construct a reference framework and to try to define standards for interoperability among WMS, the *Workflow Management Coalition* was created in 1993 joining principal Workflow vendors, universities and users. Workflow Coalition's definition involves BP [WFMC96], but we think Workflow could be applied to any kind of process. Therefore, we define a Workflow as the *model* of a process, either real or *desirable*, i.e., WM is closely related to *Business Process Reengineering* (BPR) as an enabling technology to design how we really want a process to be performed.

WM is the automation of a set of activities (i.e., a process), during which information or *objects* are passed between both human and automated tasks, *sometimes* according to a set of procedural rules. Activities are very diverse with coarse granularity, since the focus is in the control and follow-up, without putting too much attention on the semantics of tasks.

A WMS is a Software System that supports the definition, enactment and management of processes. According to Worah and Sheth [WS97], WMS are being used in inter-enterprise and intra-enterprise environments to re-engineer, streamline, automate, and track, organisational processes involving humans and automated ISs.

For further readings, refer to the proceedings of the Workshop on Workflow a Process Automation in IS [Shet96].

3.2 Software Process Management

We define *Software Process Management* as the automation and control of the *modelling*, *assessment*, and *improvement* of the SP. SP is composed of three parts: *Software production process*, *Software meta-process*, and the *Software process support* [CFF94].

A *Process-Centred Software Engineering Environment* (PSEE) supports the definition and enactment of process models. Such processes are made by activities that are described in a very low level of detail, as well as relations between artefacts and supporting tools. Objectives of a PSEE are similar to those one of a WMS.

Only one technological approach is not enough to support all SPs, some activities are more human oriented than others. On the other hand, there are activities with more participation of

software agents. For example, requirement analysis processes need support for communication and collaboration between people that are involved in eliciting requirements. Coordination needed in such processes is only to keep track of the processes, recording and monitoring meetings and documents. Activities in the analysis process are not well defined, so creativity is expected from the participants. Processes like configuration management, are characterised by well defined, highly structured and well-understood activities that have been previously assigned to participants. Communication is less important in these kinds of processes.

Relevant issues in SP are tool interoperability, versioning, process evolution and deviation, assessment, improvement and modelling. These issues have received more attention than in WM.

For further readings, refer to the book *Software Process Modelling and Technology* [FKN94].

3.3 Issues in WM and SP

- **Integration/Interoperability.** Interoperability issues are related to applications used in the process. There are two kinds of interoperability: among WMS and between WMS and heterogeneous ISs and application.

WMS support BPs that involve any kind of ISs, therefore such ISs must interoperate among each other, sharing information, and the WMS accesses information from such systems. This natural requirement has always been present in WM, but it has not been satisfactorily resolved yet. Integration of ISs in WM is quite impossible.

There are two approaches on how PSEEs deal with ISs: the *integration* of tools in a unique environment, and the *interoperation* of heterogeneous and distributed systems and tools. Traditionally, PSEEs put more attention on tool integration than on interoperation.

Interoperability issues have been treated in disciplines like *Federated Databases* [SL90]. Characteristics of the component systems of such environments are *Heterogeneity*, *Autonomy*, and *Distribution*. WMS, and recently PSEEs, address *Heterogeneous*, *Autonomous*, and *Distributed* (HAD) Systems [GHS95].

The construction of an application to support management and enactment of processes has a bottom-up style, i.e., HAD component systems were designed before the construction of the application. Problems arise because several differences exist among applications, e.g., different data representation and models, different Operative Systems.

HAD Systems are autonomous because their participation in WMS or PSEEs is transparent to the user of such systems, i.e., they must behave as they were in a stand-alone mode of operation.

Nowadays processes tend to be more distributed every time. Due to new supporting technology (Java, CORBA, Internet), work could be geographically scattered and agents do not need to be in the same building or even in the same city. Current WMS use the Internet as a way to distribute the work (e.g., Metro from Action Technologies). In the same way, some prototypes of PSEEs use also the Internet to support processes.

- **Co-operation.** Not all the approaches developed in WM and SP to support processes, provide a suitable framework for co-operation, and if they support co-operation, they do not cover all the issues of co-operation: co-ordination, communication, and collaboration. One notable exception is Action Workflow.

Some prototypes of both PSEEs and WMS include tools or technology to support co-operation conveniently, e.g., APEL [DEA97], and the work done by Christie and Riddle [CR97].

- **Control.** Successful Workflow application has been developed providing more control over humans than enabling co-operation. Real processes have characteristics of both co-operation and control. Depending on the kind of business, process could need more control (e.g., car manufacturing) or co-operation (e.g., production of a newspaper).

SP has both features of co-operation and control. Software development is a process that involves agents co-operating to produce programs or IS, but also some degree of control must exist. This degree increases as the process and the organisation become more mature. We discuss this topic in Section 4.

- **Human role in process management.** This is one of the issues that differentiates between traditional WM applications and SP applications. Normally, WM applications have some activities executed by automatic agents, and others by humans. On the other hand, in a SP there are not activities³ completely fulfilled by software applications, at most some of their composing tasks can be fully automatic, e.g., to compile a file.
- **Transaction Management.** Work on Transaction Management and WM has been done during several years to address different issues of consistent and reliable Workflow execution [WS97]. WM applications (system oriented) involve huge amounts of data, managed by means of HAD systems. Traditional Transaction Models (TTM) with full ACID⁴ properties, are not appropriate to deal with these applications. TTM were designed for homogeneous and centralised Database Management Systems, for transaction of short duration and flat structure. Therefore, several Transaction Models have been designed to handle with these systems.

Human-oriented process applications also need transaction support, but with different type of models. Such models put more emphasis in the co-operative work than on the management of data.

Both types of transaction models have the same objectives: to define policies and techniques, so that in case of failures, the system is able to recover a consistent previous state or could execute alternate activities.

- **Process Models.** WM provides useful technology to do BPR, not only to model current processes, but to design ideal or optimal processes. Due to market pressure many companies have been pushed to change in a short time, the way in they do business [DM94].

Process Modelling Languages (PML) used in commercial WMS normally present limited views of the process. They tend to combine different perspectives in one graphical notation. Functional and behavioural views are well depicted, but organisational and informational ones are not. Regarding PSEEs, they have the same problem. New advances in PML combine paradigms integrating them in one language, e.g., *JIL* [SO97]; or combining different notations in one environment, e.g., *APEL* [DEA97]. Since Processes are software too [Oste87], they must be considered as subsystems within an Organisation System.

Different persons with different roles need specific views of the model, e.g., a Project Manager might need a general view of the process to validate it, a Software engineer needs to express all the relationships among artefacts and tools. Also, when processes are enacted, the Project Manager might need to follow up the process to detect delays or bottlenecks.

Another problem is the integration of different viewpoints from the same process. Different persons can have different perceptions about how the process is performed or how it must be performed. This issue is not treated at all in WM.

³ We consider that process is formed by either sub-processes or activities, and activities are composed of tasks.

⁴ Such properties are *Atomicity*, *Consistency*, *Isolation* and *Durability* [GR93].

- **Process Evolution and deviation.** Just after a process has been modelled and enacted, changes could appear. Therefore, such changes must be managed, reflecting them in the original model. Problems arise with managing instances of the model that are being enacted, since they were modelled with an old version of the model. This issue is being studied in SP and it has not been well treated in WM.

4. Technological Support for Processes

Sheth et al. [SGJ+96] maintain that technological support for processes is characterised in a spectrum as shown in Figure 2. At one end of the spectrum, processes are unstructured collections of activities developed in an ad hoc fashion for which there are not useful models of co-ordination. On the other extreme of the spectrum, work is highly structured and can be defined precisely before is executed and represented with formalisms or their execution can dynamically evolve following well-known rules. In practice many processes share characteristics of both ends of the spectrum.

Figure 2 also shows that while in practice processes are distributed along the continuum, technological support is punctual. There is good support for ad hoc activities, for instance agendas, group decision support systems, teleconferencing, e-mails. By the other hand, WMS (system-oriented) and PSEEs provide support to specify processes well structured and following predefined rules, and enact such processes by mean of representations interpretable by an engine. In practice, most of processes are distributed along the spectrum and need support for different degrees of task structure.

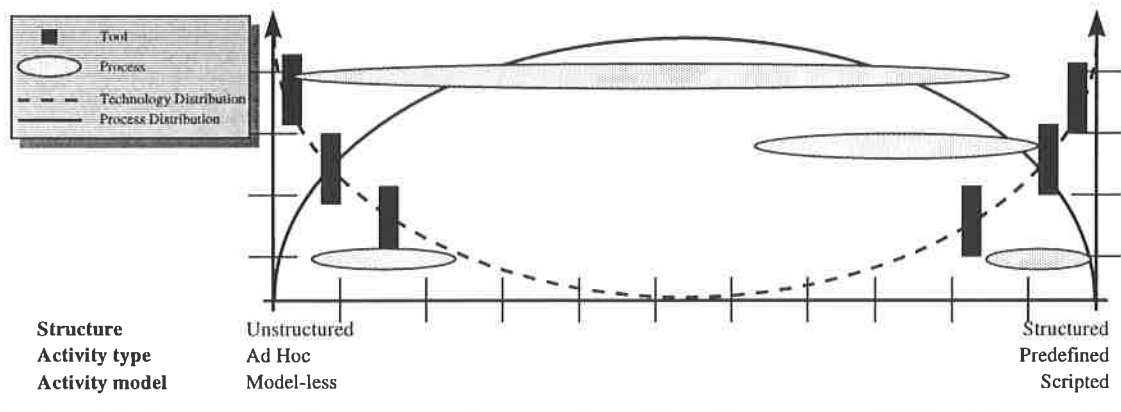


Figure 2: Technology spectrum for process support (taken from [SGJ+96])

5. Maturity levels & Process Technology

A SP could have characteristics of both ends of the spectrum described in Figure 2, e.g., at high levels, all activities of processes are predefined and modelled with a PML, but a low levels, it may be necessary to support informal and non defined meetings among the members

Process Maturity methods allow organisations to reach higher levels of quality [PCCW93]. Different kind of process support for the organisation is needed in every maturity level. At early levels, PSEEs must adopt a human-oriented point of view and provide services for effective collaboration and communication among human agents. At higher levels, PSEEs must keep the co-operation support, but they must *augment* their services covering also the system-oriented approach.

6. Conclusions

Process Technology goes beyond Software Engineering, and it involves the co-ordinating of different disciplines, in both the academic and industrial worlds.

Processes are composed of activities that share characteristics of asynchronous and synchronous co-operation, because work is characterised by alternating periods of individual and joint work. Technological support for processes must combine data and process-centric issues, since agents can be humans or Software Systems. Also, the more mature the SP within an organisation is, the more data intensive management is needed. Some emerging projects are developing tools to support co-operation beyond traditional process support, using key building blocks like CORBA and Java [CR97, DEA97].

SP is a subset of a WM, since it is a BP where the organisation's trade is to *make* Software products. Traditionally Workflow applications are focused on enterprises with general processes without considering the semantics of the activities. It is necessary for Process Technologies to be able to support processes with different level of detail, providing modelling and enactment services to different agents of the process.

SP agents must have freedom of creativity, but software control over them and the process is also necessary. Support for Co-operation must exist at all levels of process maturity. Besides, organisations with higher levels of process maturity, need environments providing support for control of the process, and support for advanced data management.

Perhaps, at the activity level (coarse granularity) of the process, it is necessary to have control over the agents, and at low levels of detail (task details) agents need co-operation facilities to respond to unexpected situations.

The experience gained over the years in both WM and SP could be applied to Process Technologies. Advanced Transaction Models has been developed to provide reliable process enactment. Process evolution and deviation have been studied in SP, and could provide solutions for this issue in WMS. Organisational modelling issues have been treated in WM with some degree of success, so it may be important to provide these results to PSEEs. Powerful PMLs are currently in the process of development.

Ideas expressed in this paper are result of the industrial experience of one of the authors, who participated in courses at Action Technologies in Alameda, California, and Template Software, in Fairfax, Washington. Also, they are part of the experience acquired by the authors from the OBJECTFLOW project [OB97] and the ongoing Ph.D. Thesis work. We want to validate such ideas with a set of experiments. Our intention is to model SPs described in scientific papers [FKN94] with commercial WMS finding advantages and shortcomings.

7. References

- [AKT+96] K. Aberer, J. Kilgermann, T. Tesch, J. Wäsch, and E. J. Nehold. *Transaction Models Supporting Co-operative Work - The TransCoop Experiences* -. In Proceedings of the International Symposium on Co-operative Database Systems for Advanced Applications (CODAS'96), Kyoto, Japan, December 1996.
- [BFGL94] S. Bandinelli, A. Fuggetta, C. Ghezzi, y L. Lavazza. *SPADE: An Environment for Software Process Analysis, Design, and Enactment*. In [FKN94].
- [BNF96] S. Bandinelli, E. Di Nitto, and A. Fuggetta. *Supporting Co-operation in the SPADE-1 Environment*. IEEE Transactions on Software Engineering, vol. 22, no. 12, pp. 841-865, December 1996.
- [CACM91] *Special Number on Collaborative Computing*. Communications of the ACM, vol. 34, no. 12, December 1991.
- [CFF94] R. Conradi, C. Fernström, and A. Fuggetta. *Concepts for Evolving Software Processes*. In [FKN94], Chapter 2.

- [CKO92] B. Curtis, M.I. Kellner, and J. Over. *Process Modelling*. Communications of the ACM. 35(9): 75-90. September 1992.
- [CR97] A. Christie and W. E. Riddle. *Defining and Supporting Collaborative Processes*. Speech Notes from Summer School on Software Process Technology, Principles and Practice, Promoter2 Esprit WG 21 185, Saint Pierre de Charreure, Grenoble, France, 30th June - 4th July 1997.
- [DM94] P.J. Denning and R. Medina-Mora. *Case study: George Mason University*. In *New Tools for New Times: The Workflow Paradigm*, Layna Fischer, editor, pages 235-251. Future Strategies Inc., Book Division, Alameda, CA, 1st edition, March 1994.
- [DM95] P.J. Denning and R. Medina-Mora. *Completing the Loops*. *Interfaces*, 25(3):42-57, May-June, 1995.
- [DEA97] S. Dami, J. Estublier, and M. Amieur. *APEL: a Graphical Yet Executable Formalism for Process Modelling*. In Special Issue of Automated Software Engineering, A. Fuggeta, editor, 1997 (to appear).
- [EG94] G. Engels and L. Groenewegen. *SOCCA: Specification of Coordinated and Cooperative Activities*. In [FKN94].
- [EGR91] C.A. Ellis, S.J. Gibbs, and G.L. Rein. *Groupware: Some Issues and Experiences*. Communications of the ACM, vol. 34, no. 1, January 1991.
- [FKN94] A. Finkelstein, J. Kramer, and B. Nuseibeh, editors. *Software Process Modelling and Technology*, Research Studies Press Ltd., Taunton, Somerset, UK, 1994.
- [GHS95] D. Georgakopoulos, M. Hornick, and A. Sheth. *An Overview of Workflow Management: From Process Modelling to Workflow Automation Infrastructure*. In Special Issue on Software Support for Workflow Management, Distributed and Parallel Databases. O. Bukhres and E. Kueshn, Eds., 3(2):119-153, April 1995.
- [GR93] Jim Gray and Andreas Reuter. *Transaction Processing: Concepts and Techniques*. Morgan Kaufmann, 1993.
- [GS97] V. Gruhn and W. Schäfer. *Software and Business Process Technology*. Tutorial Notes, ESEC/FSE 97, Zurich, Switzerland, 22-25 September, 1997.
- [PCCW93] M.C. Paulk, Bill Curtis, M.B. Chrissis, and C.V. Weber. *Capability Maturity Model for Software*. Technical Report CMU/SEI-93-TR-24, Software Engineering Institute, Pittsburgh, February 1993. Version 1.1.
- [SGJ+96] A. Sheth, D. Georgakopoulos, S. Joosten, M. Rusinkiewicz, W. Scacchi, J. Wileden, and A. Wolf. *Report from the NSF Workshop on Workflow and Process Automation in Information Systems*. Computer Science Department Technical Report, UGA-CS-TR-96-003, University of Georgia, October 1996.
- [Shet96] A. Sheth. Editor, *Proceedings of the NSF Workshop on Workflow and Process Automation in Information Systems: State-of-the-art and Future Directions*. Athens, Georgia, 8-10 May 1996.
- [SL90] A. Sheth and J.A. Larson. *Federated Database Systems for Managing Distributed, Heterogeneous, and Autonomous Databases*. ACM Computing Surveys, 22(3):183-236, September 1990.

- [OB97] C. Ocampo and P. Botella. *OBJECTFLOW: A Modular Workflow Management System*. Internal Report LSI-97-5-R, Department of Software, Technical University of Catalonia, January 1997.
- [Oste87] L.J. Osterweil. *Software processes are software too*. In Proceedings of the Ninth International Conference on Software Engineering. IEEE Computer Society, pp. 2-13, Washington, DC, 1987.
- [SO97] S.M. Sutton and L.J. Osterweil. *The Design of a Next-Generation Process Language*. LNCS 1301, Proceedings of ESEC/FSE 97. Editors M. Jazayeri and H. Schauer, Springer, Zurich, Switzerland, September, 1997.
- [WFMC96] The Workflow Management Coalition. *Terminology & glossary*, issue 2.0. Technical Report WFMC-TC-1011, June 1996.
- [WS97] D. Worah and A. Sheth. *Transactions in Transactional Workflows*. In S. Jajodia and L. Kerschberg, editors, *Advanced Transaction Models and Architectures*. Kluwer Academic Publishers, 1997 (to appear). Available in http://lstdis.cs.uga.edu/publications/pub_ALL.html
- [Yang95] Y. Yang. *Co-ordination for Process Support is Not Enough*. Proceedings of the Fourth European Workshop Software Process Technology, Lecture Notes in Computer Science 913, Springer-Verlag, April 1995.

Annexe A. WM and SP characteristics

	Software Process	Workflow Management
Goals	Modelling, enactment, assessment and improvement of Software processes production.	Modelling, reengineering and enactment of BP.
Theoretical foundations	Strong and diverse theories and formalisms.	Weak, practical solutions.
Process granularity	Fine, activities are very detailed.	Coarse, general activities.
Methodology	Process-centric.	Process-centric.
Human Interaction	Strong.	Varied, depending on the kind of product.
Agents	Normally Humans.	Humans and Heterogeneous Systems.
Co-ordination support	Limited.	Inherent.
Modelling Organisational Structure	Poor.	Supported with several degrees of detail.
Implementation Status	Several research prototypes and few commercial products.	Numerous commercial products and few research prototypes.
Applicability to non-software applications	Limited.	Extensive.
Models	Prescriptive.	Descriptive, for human-oriented WMS, prescriptive for system oriented workflows.
Temporality	Generally support for asynchronous modes	Both synchronous and asynchronous supported (depending on the WMS)
Space	Local and distributed modes.	Local and distributed modes.

Departament de Llenguatges i Sistemes Informàtics
Universitat Politècnica de Catalunya

Research Reports – 1998

- LSI-98-1-R “Optimal Sampling Strategies in Quicksort and Quickselect”, Conrado Martinez, Salvador Roura.
- LSI-98-2-R “Query, PACS and simple-PAC Learning”, J. Castro and D. Guijarro.
- LSI-98-3-R “Interval Analysis Applied to Constraint Feasibility in Geometric Constraint Solving”, R. Joan-Arinyo and N. Mata.
- LSI-98-4-R “BayesProfile: application of Bayesian Networks to website user tracking”, Ramón Sangüesa and Ulises Cortés.
- LSI-98-5-R “Some reflections on applying Workflow Technology to Software Process”, Camilo Ocampo and Pere Botella.

Hardcopies of reports can be ordered from:

Nuria Sánchez
Departament de Llenguatges i Sistemes Informàtics
Universitat Politècnica de Catalunya
Campus Nord, Mòdul C6
Jordi Girona Salgado, 1-3
08034 Barcelona, Spain
secrelsi@lsi.upc.es

See also the Department WWW pages, <http://www-lsi.upc.es/>