IDEA GROUP PUBLISHING
701 E. Chocolate Avenue, Suite 200, Hershey PA 17033, USA
Tel: 717/533-8845; Fax 717/533-8661; URL-http://www.idea-group.com
ITP5180

This paper appears in *Managing Modern Organizations Through Information Technology*, Proceedings of the 2005 Information Resources Management Association International Conference, edited by Mehdi Khosrow-Pour. Copyright 2005, Idea Group Inc.

A Comparative State-of-the-Art for Flexible Workflow Modeling

Selmin Nurcan
Université Paris 1 - Panthéon - Sorbonne, Centre de Recherche en Informatique, 90, rue de Tolbiac 75634 Paris cedex 13 France
nurcan@univ-paris1.fr

Awatef Hicheur
CNAM, 292 rue Saint-Martin 75141 Paris cedex 03 France, a_hicheur@yahoo.fr

ABSTRACT
Durring the early 90's, workflow technologies were the only ones to offer a transversal integration capacity to the enterprise applications. However, formalisms proposed for workflow specifications were almost systematically activity oriented. Most of the usual modeling formalisms, as well as standards defined by WfMC, are all based on organizational and operational concepts. Consequently, resulting process definitions have the advantage to be easily transformable in executable code but the disadvantage of being prescriptive and rigid. More recent works highlight requirements in term of flexible and adaptive workflows, whose execution can evolve according to situations that cannot always be prescribed. This paper presents the state of the art for flexible workflow management systems and criteria for comparing them. It also introduces a conceptual framework for flexible business process modeling.

INTRODUCTION
In all management challenges, information systems should be continuously adapted to changing business practices and needs. This can be achieved by developing process-centric solutions. The paradigm of 'Business Process Management' stresses the importance of integrating entire processes rather than simply integrating data or applications [6]. The process-oriented business management requires appropriate concepts to design business processes and their supporting IS. The aim is to define and control the organizational structures in a flexible way so they can rapidly evolve according to changing conditions.

Since the beginning of the application of the *Business Process Reengineering* [13] as a management method for transforming organizations, Workflow Management Systems (WFMS) have often been positioned as an appropriate technological solution to satisfy the objectives set by this management method. Workflow technologies allow integrating process islands at a high level so that they can collaboratively provide business solutions that each individual application is unable to provide. However commercial workflow solutions offer only limited evolution facilities.

The objective of the research in progress is measure the capacity of the studied modeling formalisms to *represent* various types of business processes and to *support their evolution*.

This paper is organized as follows: Section 2 discusses some limits of the current workflow technologies which offer an automated support to the enactment of business processes. Section 3 presents a survey on flexible workflow modeling. Section 4 gives an overview of our conceptual framework which allows us to describe the invariants (or the minimal definition) of business processes before specifying the manner of making them operational in particular organizational contexts.

AUTOMATED SUPPORT FOR DEFINING AND EXECUTING BUSINESS PROCESSES
Business processes can be roughly classified into two categories. The first concerns well-defined and -often- repetitive processes having important coordination and automation needs. The second category concerns ill-defined processes. The essential preoccupation with the latter is the information and knowledge sharing between the actors implied in the processes more than the coordination of their tasks. For many organizations, well-defined and ill-defined processes coexist and must be handled in the final business model [25].

According to [38], a *process definition* is "the representation of a business process in a form that supports automated manipulation, such as modelling or enactment by a WFMS. The process definition consists of a network of activities and their relationships, criteria to indicate the start and termination of the process, and information about the individual activities, such as participants performing them, IT applications supporting them, etc". This definition corresponds to a *prescriptive process model* in the sense that "*how* things must/should/could be done" should be pre-defined before the enactment of the *process definition.* By opposition, a *descriptive process model* aims at recording and providing a trace of *what* happens during the business process [12].

Several classifications have been proposed for workflow applications. The commonly used divides them into four classes, depending on the nature of the business processes they support and the value these processes have for the enterprise [2]:

- *Production workflows* involve repetitive and predictable business processes. They implement the core processes of the enterprise and incorporate accesses to various information systems. They form the closest category to the existing commercial WFMS solutions and the generic workflow product structure adopted by WfMC [38].
- *Administrative workflows* involve repetitive, predictable processes with simple task coordination rules and do not concern the core processes of the enterprise.
- *Ad hoc workflows* have no predefined structure. Workflow support is limited to communication mechanisms to route case data between workers and possibly some support for logging and state tracking. They are created to deal with exceptions, or where there is no set pattern for moving information among people. The coordination of the activities is controlled by human participants.
- *Collaborative workflows*, unlike the other categories, includes iterative tasks over the same step until some form of agreement has been made. It seems very difficult to model such a process using classical WFMSs since it is impossible to predefine the steps to follow. Most of the co-ordination is done by human participants.

Most of the existing workflow modelling formalisms are prescriptive ones. In terms of automated support for executing business process models, commercial WFMS and the underlying control flow models are useful for well-defined and repetitive processes (production and administrative). Nevertheless, they cannot be used for ill-defined business processes (ad hoc and collaborative) neither to deal with the dynamic modification of well-defined ones.

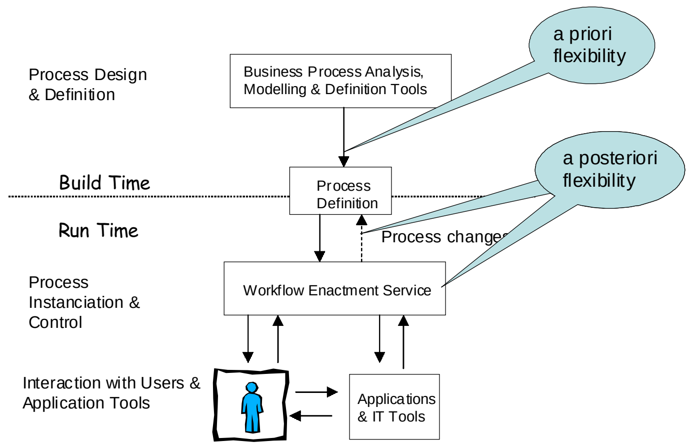
Few WFMSs (InConcert, Ensemble and TeamWARE flow) allow creating and modifying workflow models (process definitions) during their execution. Each workflow instance has a private process model and the problems rising from the modification of workflow models is thus avoided. Inconcert allows also initiating an empty process model. It supports the definition of workflow models by "discovering", i.e. by induction from process instances [34].

A SURVEY ON FLEXIBLE WORKFLOW MODELING AND EXECUTING

We studied several approaches proposed in the literature to deal with workflow flexibility and adaptability. Among multiple criteria we identified for comparison purposes, we present here only a subset of them: *nature of the flexibility*, *formalism* and *flexibility techniques*.

According to [37], WFMS may be characterized as providing support in three functional areas: Build-time functions, Run-time control functions and Run-time interactions with human users and IT application tools. The study of the literature allowed us to distinguish principally two kinds of **flexibility** depending on the capacity of dealing with change which might be incorporated in process definitions during build-time or run-time. Figure 1 illustrates the relationships between these functional areas. It also situates the capacity of any WFMS to implement the flexibility, in this well known schema of the WfMC.

Figure 1. WFMS Characteristics and Flexibility



Flexibility a posteriori or by adaptation allows to adapt (modify) the process definition or some of its instances during their execution. Approaches which offer only this kind of flexibility are based on prescriptive modeling formalisms. It could be considered that the resulting process definitions are not really flexible but rather adaptive or evolutionary. In fact, these approaches can not anticipate the capacity to change during the build-time. This is the most usual case found in the literature [5], [7], [9], [15], [18], [20], [23], [24], [28], [31], [32], [33], [35], [36]. Prescriptive modeling formalisms are well adapted to specify business processes which require high degree of control and prediction and for which the need for change remains an exception. This concerns *production workflows*.

Flexibility a priori or by selection is based on modeling formalisms which can offer the capacity to deal with the environmental change without any evolution of process definitions. This means that this capacity should be incorporated in process definitions during build-time. The process definition should be specified in a sufficiently flexible way so that *it will yield under the influence of the environment without breaking*. Accordingly, the workflow enactment service should be able to execute 'incomplete' specifications of process definitions. Therefore the enactment service depends on the user decisions for the selection of a process component (dynamic construction of the process instance) [21]; an execution path among several possible; a behavior to associate to a process component (actor, activity, resource …); a way-of-performing an activity [11]. The approach proposed in [21] is based on dynamic construction of instances by selecting components in a library. It offers only 'a priori' flexibility.

These two techniques are not mutually exclusive. Both are applied in [1], [8], [10], [11], [14], [16], [19], [22], [30]. For approaches using 'a priori' flexibility, it is recommended to offer the possibility of adapting a process instance when a not anticipated event happens and the system can not deal with it using its 'a priori' flexibility capacities.

The **formalism** defines the set of concepts allowing specifying process definitions.

An *activity-oriented* formalism allows prescribing the activities to be performed and their relationships regarding control and data flows which are pre-defined. This is the most usual specification formalism [1], [5], [7-9], [14-16], [19-24], [28], [31-33], [35], [36].

A *product-oriented* formalism stresses the business objects handled during the execution of the process, their flow among workflow activities [30], [34].

A *decision-oriented* formalism considers that users are 'knowledgeable' on those business processes which can not be entirely pre-defined using control flows among activities [10], [11]. Accordingly, he/she can select an alternative path during the enactment of the process instance or decide about the operations to perform to meet his/hers responsibilities.

An *intention-oriented* formalism allows addressing the essential question regarding business processes, "why", and to define alternative organizational solutions for a given business objective [30]. It permits also to develop user-oriented systems [29].

Flexibility techniques are meaningful for 'a priori' flexibility only. These techniques allow implementing *descriptive process definitions* which could be refined during the execution.

Late binding: process definition elements are considered as objects which behavior is defined during process execution [8], [11], [14], [16], [19], [21], [30]. This includes the association of activities to their most adequate implementation [10], [19], the selection of the resource better satisfying a given criteria or the selection of the best suitable actor for carrying out an activity according to some constraints [22].

Late modeling: some elements of the process definition are left open to innovation and creativity of process participants, especially in situations where the corresponding specifications can not be well identified during build-time [1], [10], [11], [30]. Some activities can be declared as mandatory while some others could be omitted during execution. It is also possible to specify activities and to leave to the participants the freedom of selecting the most appropriate precedence relationships during the execution of a process instance [19]. In [20], a participant can select the best implementation for an activity which was initially specified as a goal to be achieved. If none of existing ways-of-performing is appropriate, it is possible to dynamically create a new one.

In [19], several behaviors can be associated to an activity. It is also possible to not execute an activity which was declared as optional. The approach adopts late modeling in a controlled way, i.e. activities are specified but participants can modify their execution order. The approach presented in [14] allows selecting the behavior associated to a given activity in a dynamic way. If none of the predefined behaviors satisfies the current situation, it is possible to modify the workflow instance.

The case handling: This paradigm was proposed in [34] for supporting knowledge intensive business processes. It offers a good balance between the data-centered approaches of the 80's and the process-centered approaches of the 90's. Data and processes are not separated as in traditional WFMSs. Case handling focuses on what can be made rather than on what should be made. It leaves thus freedom to the participants of a process instance in a controlled way. The 'next' activity to be carried out is defined by the state of the process instance rather than by a predefined control flow. In this sense, the underlying modeling formalism is product oriented.

In [17], authors propose a way of processing flexible workflows based on a state driven view. This view focuses on changes that each activity introduces in order to move the process closer to its desired final state (i.e. to its goal). A process is considered as a trajectory in the space of its possible states. The process control is regulated by a set of activity planning rules classified into three categories: obligations, recommendations and prohibitions.

A CONCEPTUAL MODELING FRAMEWORK FOR FLEXIBLE BUSINESS PROCESSES

In order to deal with a wide range of business processes, we proposed in [25] a conceptual modeling framework offering at one hand the rigor necessary for modeling well-defined business processes, and at the other hand, the flexibility and adaptability required for ill-defined business processes. Let us remind that our vision of any organization is structured according to three layers of concern. The *objectives* of the organization are achieved by implementing *enterprise processes* whose are themselves supported by *enterprise information systems*. The two first layers focus on *intentional and organizational aspects of the enterprise*, i.e. the business objectives and how they are achieved through the co-operation of enterprise actors handling enterprise objects. The third layer focuses on *system aspects* i.e., application components that will support the processes and the actors of the enterprise.

The purpose of the *enterprise objectives layer* is to provide the intentional definition of business processes. The Map formalism used at this layer is *intention-oriented*. Business process definitions are described in terms of intentions to be achieved and strategies to be followed. A *business map* is a labelled directed graph with intentions as nodes and strategies as edges between intentions. It consists of a number of *sections* each of which is a triplet < source intention I_i, target intention I_j, strategy S_{ij}>.

A *business intention* expresses what the enterprise wants to achieve disregarding the considerations about who, when and where. A *strategy* is an approach, a manner to achieve an intention. The strategy, as part of the triplet $<I_i, I_j, S_{ij}>$ characterizes the flow from I_i to I_j and the way I_j can be achieved. The specific manner in which an intention can be achieved is captured in a section of the map. The business map contains a finite number of paths from *Start* to *Stop*; each of them is a *Business Process Model*. Therefore the map is a *multi-model*. A business map constitutes a strategic business plan. The approach suggests a dynamic construction of the actual path by navigating in the business map.

The Map formalism is also *decision-oriented* thanks to the navigation guidelines and their associated choice criteria. In fact, a major advantage of the proposed approach is the systematic way of dealing with enterprise modeling in terms of *knowledge modeling* used with a *process guidance* framework. A guideline has a signature defined as a pair <*situation*, *intention*>. A *situation* is a part of the product it makes sense to make a decision on. It indicates when the guideline can be applied [4].

The Map formalism provides a refinement mechanism allowing describing some complex parts of a business map in refined maps in lower levels of abstraction. Finally, non refined maps might be operationalized in the *enterprise processes layer* to capture organizational and operational properties of the business processes. The representation formalisms used at the organizational level might be classical activity-oriented modeling formalisms for *production* (or even *administrative*) *workflows*. Product-oriented formalisms are required for *collaborative* or *ad-hoc* workflows.

The Map formalism provides '*a priori' flexibility* since the navigation will be dynamically performed during the execution. It is also possible to modify a business map during its execution. The model offers thus also '*a posteriori' flexibility*. The evolution can concern intentions, strategies or the order of the execution of the intentions. The approach allows *late binding* and *late modeling*. Indeed, the choice of the business process chunk to be executed for the achievement of an intention is done during the execution of the business map. It is also possible to leave certain strategies open to later specifications (during the execution of the process).

The conceptual modeling framework described in [25] gives us the ability to describe, initially, the invariants of the organization in terms of objectives and strategies before specifying the manner of making them operational in a particular organizational situation. Similar *goal-driven modeling perspectives* can be found in a number of approaches such as [26], [3], [39], [27]. The concern is to establish a close relationship between the '*Whys*' and the '*Whats*'.

CONCLUSION

A clear representation of the business objectives simplifies the comprehension of the organizational changes and the evolution of the business model which results from these changes. The conceptual modeling framework which is overviewed in the previous section is useful for flexible workflow modeling. The formalism offers the ability to represent in the same process definition the well-structured process chunks as well as ill-structured or ad hoc ones. The framework is based on intentional modeling (why and what) of business processes before their organizational specifications (who, where and when). The purpose of the Map formalism is to define an integration layer for islands of business process chunks. The formalism offers the advantage of being able to represent in the same process model the well-structured process chunks as well as ill-structured or ad hoc ones.

The intentional view of the business represents the enterprise from the point of view of its objectives disregarding the considerations of the operational level. In fact, this view should be completed with the realization conditions of these objectives, i.e. taking in consideration the organizational and operational choices.

Our future work consists to formally define the concept of 'situation' used in the navigation guidelines of the Map formalism, and to extend the formalism with the concepts of (i) '*product*' -resulting from the process execution- and (ii) '*product state*'. Our aim is to tightly associate the *intention* concept of the Map formalism to the *product state* obtained by the fulfillment of this intention, describing thus more precisely the evolution of the status of a process during its execution.

REFERENCES

[1] Agostini, A. and De Michelis, G. (2000) A Light Workflow Management System Using Simple Process Models. In *Computer Supported Cooperative Work: The Journal of Collaborative Computing*, Kluwer Academic Publishers, Vol.9, No.3-4, August 2000.

[2] Alonso, G., Agrawal, D., El Abbadi, A. and Mohan, C. (1997) Functionality and Limitations of Current Workflow Management Systems, *IEEE Expert*, Vol12, No. 5, Sept/Oct 1997.

[3] Antón, A. I., McCracken, W. M. and Potts, C. (1994), Goal Decomposition and Scenario Analysis in Business Process Reengineering, Proceedings of the 6th International Conference on Advanced Information Systems Engineering (CAiSE '94), Utrecht, The Netherlands.

[4] Barrios, J. and Nurcan, S. (2004) Model Driven Architectures for Enterprise Information Systems, Proceedings of the 16th Conference on Advanced Information Systems Engineering, (CAISE'04), June 7-9, 2004, Riga, Lettonie, Springer Verlag (pub), 2004.

[5] Borghoff, U.M., et al. (1997) Reflective Agents for AdaptiveWorkflows, Second International Conference and Exhi-

bition on Practical Application of Intelligent Agents and Multi-Agents (PAAM'97) (London, UK, Apr. 1997), 405-420, Society Press.

[6] Burlton, R. T. (2001) *Business Process Management- Profiting from process*, SAMS Publishing, 2001.

[7] Casati, F., Ceri, S., Pernici, B. and Pozzi, G. (1996) Workflow Evolution. Proceedings of the 15th ER'96 international Conference, Oct 7-10, Cottbus, Germany, Springer Verlag Lecture Notes in Computer Science, 1996.

[8] Chiu, D.K.W., Li, Q. and Karlapalem, K. (1999) A Meta Modeling Approach for Workflow Management System Supporting Exception Handling, Special Issue on Method Engineering and Metamodeling, *Information Systems*, Pergamon Press, Elsevier Science, 24(2), May 1999.

[9] Dadam, P., Reichert M. and Rinderle S. (2003) Evaluation of Correctness Criteria For Dynamic Workflow Changes. Proceedings of the International Conference on Business Process Management (BPM '03), Eindhoven, The Netherlands, June 2003.

[10] Dellen, B., Maurer, F., and Pews, G. (1997) Knowledge Based Techniques to Increase the Flexibility of Workflow Management. Data and Knowledge Engineering, North Holland.

[11] Faustmann, G. (1998) Enforcement vs. Freedom of Action-An Integrated Approach to Flexible Workflow Enactment. Workshop on Adaptive Workflow Systems. Conference on CSCW, Seattle, USA.

[12] Gotel, O. and Finkelstein, A. USA (1996) An Analysis of the Requirements Traceability Problem. First IEEE International Conference ICRE'94, Colorado Springs.

[13] Hammer M. and Champy J. (1993) *Reengineering the Corporation: a Manifesto for Business Revolution*, Harper Collins Publishers, Inc., New York.

[14] Halliday, J., Shrivastava, S.K. and Wheater, S.M. (2001) Flexible Workflow Management in the OPENflow System, Fifth IEEE International Enterprise Distributed Object Computing Conference.

[15] Herbst, J. (1999) Inducing Workflow Models from Workflow Instances, European Concurrent Engineering Conference ECEC'99. Society for Computer Simulation (SCS).

[16] Joeris, G. and Herzog, O. (1999) Towards Flexible and High-Level Modeling and Enacting of Processes. M. Jarke and A. Oberweis, Editors, Proceedings of the 11th International Conference on Advanced Information Systems Engineering (CAiSE 99), Springer Verlag LNCS 1626. p. 88-102.

[17] Khomyakov, M. and Bider, I. (2001) Achieving workflow flexibility through taming the chaos, Journal of Conceptual Modeling , August 2001.

[18] Klein, M. and C. Dellarocas (2000) A Knowledge-Based Approach to Handling Exceptions in Workflow Systems. Journal of Computer-Supported Collaborative Work. Special Issue on Adaptive Workflow System, 9(3/4).

[19] Klingemann, J. (2000) Controlled flexibility in workflow management. 12th International Conference on Advanced Information Systems Engineering (CaiSE'00), Stockholm, Sweden

[20] Kradolfer, M. and A. Geppert (1999) Dynamic workflow schema evolution based on workflow type versioning and workflow migration, International Conference on Cooperative Information Systems (CoopIS'99), Edimburgh. IEEE Computer Society Press.

[21] Mangan, P. and Sadiq, S. (2003) A constraint Specification Approach to Building flexible workflows. In Journal of Resarch and Practice in Information Technology, vol. 35, No.1, February 2003.

[22] Manolescu, D. A. and Johnson, R.E. (1999) Dynamic object model and adaptive workflow, OOPSLA '99 MetaData and Active Object-Model Workshop November 2, 1999.

[23] Meng, J., Su, S.Y.W., Lam, H. and Helal, A. (2002) Achieving Dynamic Inter-organizational Workflow Management by Integrating Business Processes, Events, and Rules. Proceedings of the Thirty-Fifth Hawaii International Conference on System Sciences (HICSS-35), January 2002.

[24] Müller, R., Greiner, U. and Rahm, E. (2003): AgentWork: A Workflow System Supporting Event-Oriented Workflow Adaptation. Technical Report, University of Leipzig.

[25] Nurcan, S. (2004) Business Process Modeling for developing Process Oriented IT Systems, The "Business Process Management Tools and Technologies" track of the 2004 Information Resources Management Association International Conference- May 23-26, 2004, New Orleans, USA.

[26] Nurcan, S. and Rolland, C. (2003) A multi-method for defining the organizational change, *Information and Software Technology*, Elsevier. 45:2(2003).

[27] Ould, M. (1995), *Business Processes: Modelling and Analysis for Re-engineering and Improvement*, John Wiley & Sons, New-York.

[28] Reichert, M., Bauer, Th. and Dadam, P. (1999) Enterprise-Wide and Cross-Enterprise Workflow Management: Challenges and Research Issues for Adaptive Workflows. Proc. Workshop Enterprise-wide and Cross-enterprise Workflow Management: Concepts, Systems, Applications, 29. Jahrestagung der GI (Informatik '99).

[29] Robinson, W.N. (1996) Goal-Oriented Workflow Analysis and Infrastructure, NSF Workshop on Workflow & Process Automation, Athens, 1996.

[30] Rolland, C., Prakash, N. and Benjamen A. (1999) A Multi-Model View of Process Modelling, *Requirements Engineering Journal*, 4:4.

[31] Sadiq, S. (2000) On Capturing Exceptions in Workflow Process Models. In Proceedings of the 4th International Conference on Business Information Systems. Poznan, Poland. Springer-Verlag. April 12 -13 2000.

[32] Van der Aalst, W.M.P. (2001) How to Handle Dynamic Change and Capture Management Information: An Approach Based on Generic Workflow Models. International Journal of Computer Systems, Science Engineering, vol. 15, no. 5.

[33] Van der Aalst, W.M.P., Basten, T., Verbeek, H., Verkoulen, P. and Voorhoeve M. (1999) Adaptive Workflow: On the Interplay between Flexibility and Support. In J. Filipe and J. Cordeiro (editors), First International Conference on Enterprise Information Systems, 353-360, Setubal, Portugal.

[34] Van der Aalst, W.M.P., Stoffele, M. and Wamelink, J.W.F. (2003), Case Handling in Construction. , Automation in Construction, 12(3), 2003.

[35] Vossen, G. and Weske, M. (1999) *The WASA2 Object-Oriented Workflow Management System*; in Proc. ACM SIGMOD International Conference on Management of Data 1999, Philadelphia, PA, 587-589 , 1999

[36] Weske, M. (2001) Formal Foundation and Conceptual Design of Dynamic Adaptations in a Workflow Management System. HICSS 2001 management 34th Annual Hawaii International Conference on System Sciences (HICSS-34)-Volume 7, January 03 - 06, 2001.

[37] WfMC-TC-1003 v1.1 (1995), V 1.1. Workflow Reference Model, January 1995.

[38] WfMC-TC-1011 v3 (1999) Terminology & Glossary, February 1999.

[39] Yu, E.S.K. and Mylopoulos, J. (1996) Using Goals, Rules and Methods to Support Reasoning in Business Process Reengineering, *Intelligent Systems in Accounting, Finance and Management*, Vol. 5, 1-13.

Related Content