Flexible Support of Work Processes – Balancing the Support of Organisations and Workers

John Krogstie and Håvard D. Jørgensen

Norwegian University of Science and Technology and
SINTEF Telecom and Informatics, Norway
{jok,hdj}@informatics.sintef.no

1 Introduction and Motivation

Today's organisations are characterised by dynamic, only partially understood and error-prone environments. Thus systems must be developed, which support truly dynamic organisational processes, often involving co-operation across traditional organisational boundaries.

The limited success of workflow management systems (WMS) has partially been attributed to lack of *flexibility*. Consequently, flexible workflow is a hot research topic. Most work within this area looks at how conventional systems can be extended and enhanced in other words how static workflow systems can be made *adaptive*. Most of this work recognise that change is a way of life in most organisations, but a basic premise is still that work is repetitive and can be prescribed reasonably completely.

On the other hand, a lot of dynamic organisational activity is best described as emergent. In those activities the process model and structure is unclear at the start and emerges during enactment.

Traditional ERP systems tend to be quite inflexible, hardly adaptable at runtime, and primarily support the organisationally agreed processes. Existing workflow management systems have typically been focused on dealing with exceptions and have thus offered some support for adaptive processes. These types of systems, however, have typically overlooked emergent processes, which seem to encompass an increasing part of organised activity. An alternative approach is to support *active* models as a general technique for increasing the flexibility of computerised information systems for co-operative work support. What does it mean that a model is active? First of all, the representation must be *available* to the users of the information system at runtime. Second, the model must *influence the behaviour* of the computerised system. Third, the model must be *dynamic*, users must be supported in changing the model to fit their local reality, enabling tailoring of the system's behaviour.

On the other hand, only supporting the emergent work style of the individual knowledge worker is probably at times inefficient, because routine parts of the work can be prescribed and automated, and because sharing of explicit process models facilitates co-ordination, collaboration and communication between multiple parties. Thus there is a need for a balance between prescription and emergent representations.

K.R. Dittrich, A. Geppert, M.C. Norrie (Eds.): CAiSE 2001, LNCS 2068, pp. 477–481, 2001.

The panel will describe and discuss different approaches to more or less flexibly support organisational, group and personal work processes, applying novel approaches including active models.

2 Position Statements of Panel Members

Stefan Jablonski (Stefan.Jablonski@informatik.uni-erlangen.de)
Universitaet Erlangen-Nuernberg

Flexible Support of Work Processes

We divide our contribution into three position statements. From our perspective they span a good amount of the issues related to the three topics

- flexibility
- work processes and
- workflow management.

The main theme throughout my position statements is that the need for work process support does not ”automatically” connect to ”workflow management”. Therefore, we strongly advice to investigate first whether an application is suitable for workflow support at all. Not before the answer is ”yes” it should be investigated how this support can be tailored to specific requirements of an application scenario; this investigations directs to the issue of flexibility for workflow management systems.

My three position statements are:

Uncritical application of workflow management

Our first position is that the workflow management techniques are applied uncritically. People do not sufficiently analyse application scenarios in order to find out whether it is really worth to be supported by workflow management. There is a huge range of different sorts of work processes. Some of them should never be supported by workflow management like emergencies in an hospital environment. This does not say that in such a case there might be something like an after-treatment that asynchronously maps the emergency to a workflow.

Wrong workflow models

Another problem we are facing very often is that workflow models are just wrong. They do not reflect the real application scenario but do impose processing constraints onto an application scenario that are not justified. Mostly these undesirable effects stem from the fact that workflow models are not expressive enough to allow adequate modeling.

Missing flexibility in workflow management approaches

Just this third position statement refers to the workflow management approaches. It is a fact that flexibility is not supported sufficiently by actual workflow approaches. We propose two mechanisms to introduce more flexibility to workflow management: flexibility by selection and flexibility by adaptation.

Paul Johannesson (pajo@dsv.su.se) Department of Computer and Systems Sciences, Stockholm University/KTH

Explicit Goal Representation as a Means for Flexible Process Modelling and Enactment

Current software systems for supporting process management, such as workflow management systems, are effective for predictable and repetitive processes. However, they are typically unable to adapt to a dynamic environment where unexpected situations have to be managed. It is possible to distinguish among four different types of exceptions that can occur during the execution of a process, [Eder95]: *basic failures*, which are failures of the supporting software system or its environment; *application failures*, which are failures of the applications that are invoked by the software system; *expected exceptions*, which are predictable deviations from the normal flow of the process; *unexpected exceptions*, which are mismatches between the actual execution of a process and its definition in the software system.

Unexpected exceptions are caused by process changes that could not have been anticipated at design time, e.g. a new government regulation. Thus, unexpected changes are the most difficult ones to handle in workflow systems. One approach for addressing the problem of managing unexpected exceptions is to explicitly include goals in process specifications and workflow systems – not only events and activities. Goals are typically more stable in a process than the events and activities that are carried out to achieve the goals. Therefore, including goals in process specifications will provide a stable frame around which activities can be ordered. A process specification will then consist of a collection of goals and for each goal a tentative and adaptable structure of activities to be carried out for achieving the goal. For this approach to work, it is required that the implemented workflow system makes goals visible to the user. Thus, goals are not only used in the early requirements specification phases – they should also become explicit in the application logic of a system as well as in its user interface. In this respect, the approach is similar to the philosophy behind Tropos, [Mylopoulos00], where intentional concepts are used in late software development phases.

[Eder95] Eder J, Liebhart W. The Workflow Activity Model WAMO. In: 3rd International Conference on Cooperative Information Systems. University of Toronto Press. 1995, pp. 87-98

[Mylopoulos00] Mylopoulos J, Castro J. Tropos: A Framework for Requirements-Driven Software Development. In: Information Systems Engineering, ed. Brinkkemper S. et. al., Springer 2000, pp. 261-274

*John Krogstie (*jok,@informatics.sintef.no) Norwegian University of Science and Technology and SINTEF Telecom and Informatics, Norway

From Supporting Organisations to Supporting Knowledge Worker in Organisations

Information and workflow systems are traditionally made to primarily support the goal of an organisation, and only secondarily the users of the systems. When the goals are well understood, this is a sensible approach, and production workflow and ERP systems are a good investment. On the other hand, a larger proportion of work is

today done by what is termed symbolic analyst, whose work resources and work products are mainly symbol-structures. The main work pattern of these persons is knowledge intensive projects often in dynamically networked organisations. Symbolic analysts such as consultants, reporters, and researchers will typically have many tasks going on concurrently, and will be interested in a lot of different information there and then, much of which can not be anticipated fully beforehand. There might be a need for learning on the fly, but also for capturing interesting situations to feed back to the organisation supporting knowledge management and learning processes for a larger part of the organisation. Although many of their processes are emergent and creative, a large proportion of the time spend by these so-called knowledge workers, is on routine, administrative tasks. We also predict that many of these workers would like to be supported also in their more complex personal processes involving e.g. governmental agencies, banks, and insurance companies using the same system if possible, as the distinction of work and private life is blurred. This is similar to how many people nowadays use PDAs, where the calendar contains both private and professional tasks and appointments.

These developments have opened the need for emergent workflow systems where modelling is viewed as an integral part of the work, performed by the process participants, as they are the only actors who have sufficient knowledge of the process. The focus is on unique cases, especially knowledge intensive projects. Emergent models, as other active models, depend more on the individual, social, organisational and situational usage context than routine procedures modelled by software engineers or process designers. When process modelling and enactment happen in parallel, we cannot assume complete models. These challenges cannot be met by algorithms alone. Instead we must involve the users in resolving incompleteness and inconsistencies. In fact, it may not even be the case that emergent process models should be enacted by software, the interpretation of the models might be carried out by the users instead. Some challenges for emergent workflow systems for knowledge intensive project work are:

- Modelling must be done also by process participants, not just by external process experts. The end users must be motivated to externalise and share their knowledge about the process. We thus need to increase the users' benefits of keeping models up to date.
- Users don't want to start from scratch. In order to reuse modelling effort and learn from past experience, we need mechanisms that enable the harvesting of past models into reusable templates.
- How can we support co-ordination based on an emergent process model? We view enactment as "co-ordination by automated sequencing of tasks", and this should be combined with more flexible CSCW approaches.

Although the individual knowledge worker typically would prefer to be supported across his range of processes with the same tool, there are as indicated many differences between emergent workflow support and the traditional static and adaptive approaches. From this, it can be argued that combining conventional and emergent workflow into one system should not be a main objective. Though they share the core of active support based on process models, adaptive and emergent

workflow is perhaps best kept in different systems and integrated through standards from the OMG, WfMC and IETF?

Matthias Weske (mathias.weske@acm.org) Eindhoven University of Technology

Flexible Workflow Management: Adapt and Bend, but Don't Break!

The dictionary tells that an artefact is flexible if it can ”easily be changed to suit new conditions”. This general definition of flexibility holds well in the workflow context, where artefacts correspond to workflow specifications and applications, and changes are the means to cope with new conditions of application processes, imposed by changes in their business or legal environments. Hence, it is not surprising that flexibility has been a major motivation for workflow management from its beginning: By explicitly modelling application processes consisting of activities, their causal and temporal ordering and the technical and organisational environment of their execution, the structure and embedding of a given application process can be changed conveniently. In particular, changes can be applied by modifying workflow specifications rather than by changing computer programs, which hard-coded application processes in the pre-workflow era. Hence, traditional workflow technology is well suited for modelling and controlling the execution of a wide range of application processes.

However, as turned out in recent years, traditional workflow technology does not suffice for advanced applications in today's dynamic and competitive environments in commerce, public administration, and in science and engineering. In these settings it is increasingly unlikely that an application process is modelled once to be executed repeatedly without any modifications. Hence, more advanced flexibility mechanisms to facilitate workflow modification at different instants with different scopes have to be developed. Recent work on flexible workflow management has generated a considerable body of literature, focusing on a variety of aspects. One of the hardest problems in this context is dynamic adaptation, i.e., the adaptation of running workflow instances to new workflow schemas. This functionality is required in settings where workflows are typically long running activities and where changes to the process environment occur frequently.

Besides conceptual research issues like correctness criteria for dynamic adaptations, flexibility requirements also raise a number of interesting workflow management systems design issues For example, the strict separation of a workflow's build time and its run time as found in many traditional workflow management system is not adequate for supporting dynamic adaptations effectively; interleavings of a workflow's build time and its run time are required and, thus, have to be facilitated by flexible workflow management systems.

Once the scientific results are consolidated and put into practice, the second interpretation of flexibility the dictionary tells can also be fulfilled: An artefact is flexible if it can ”easily be bent without breaking”, an interpretation which may be even more appropriate in the workflow context.