

Getting Some Perspective: Using Process Descriptions to Index Document History

Paul Dourish^{}, Richard Bentley[†], Rachel Jones[†] and Allan MacLean[†]*

^{*}Xerox Palo Alto Research Center
3333 Coyote Hill Road
Palo Alto CA 94304 USA
dourish@parc.xerox.com

[†]Xerox Research Center Europe
61 Regent Street
Cambridge CB2 1AB UK
{bentley,rjones,maclean}@xrce.xerox.com

ABSTRACT

Process descriptions are used in workflow and related systems to describe the flow of work and organisational responsibility in business processes, and to aid in coordination. However, the division of a working process into a sequence of steps provides only a partial view of the work involved. In many cases, the performance of individual tasks in a larger process may depend on interpretations and understandings of *how* other aspects of the work were conducted.

We present an example from an ethnographic investigation of one particular organisation, and introduce a mechanism, which we call “Perspectives,” for dealing with it. A “Perspective” uses the process description to provide an index into the history of a document moving through a process. Perspectives allow workflow systems to manage and present information about the execution of specific process instances within the general frame of abstract process descriptions.

Keywords: workflow, process modeling, process execution, visualisation, awareness.

INTRODUCTION

In the past few years, workflow technologies have become a common form of technology for supporting collaborative work. Workflow systems support the execution of regularised processes by dividing them into component tasks, mapping out the pattern of dependencies and relationships between these tasks, and then managing the execution of the individual tasks (and thus the progress of the overall process). We use the term *process description* to refer to the workflow system’s representation of the work process. Such a description is normally cast in terms of individual parts of the work (*tasks*) and relationships (normally, sequential relationships) between them.

We have been investigating the use of process descriptions in managing work and collaborative activity. In particular, we have been interested in resolving some conflicts and con-

traditions between rigid process formalisms and the more fluid and variable practices observed in real-world settings. This follows on from our earlier work on related issues with the Freeflow system [8].

In this paper, we are particularly concerned with the *intelligibility* of work. Although the mutual visibility and intelligibility of working activities has been a common focus of shared workspace technologies (in the form of awareness tools), it has gone largely unexplored in workflow systems. This may, in part, be because workflow technologies rely on the process description as a means to contextualise work. An implicit assumption behind workflow’s division of a process into a number of different tasks is that the accomplishment of these tasks is independent, so long as the correct sequence is maintained. However, in some cases, this is not true. In these situations it can happen that, in order to perform some task, someone may have to make informed judgements about the work that has been carried out elsewhere. In other words, the work of a particular task has to be made intelligible to the people carrying out other tasks.

We present an example of this sort of problem from a field study we have been conducting at a large British high street financial institution. For this organisation, the process provides a general framework for working activities, but the management of each actual instance of the work involves a great deal of communication of contextual information between people performing the different tasks. We use this example to motivate the design of a technical approach. We call our technique “Perspectives”. Perspectives use the process description as an index into the history of a document as it flows through the process, and allow a user to see the activity as it looked from another part of the process. The Perspectives technique is not rooted in one particular approach to workflow or another, but is a general mechanism applicable to a wide range of process-based systems.

We begin by discussing the field work investigations and the conduct of process-based work. We then step back to consider how workflow systems could be transformed to support intelligibility. We outline our approach, discuss some design features and trade-offs, and show how it has been embodied

in a prototype. Finally, we discuss the role for these sorts of technologies as a facet of workflow or process-based systems in general.

PROCESS-BASED WORK AT “THE SOCIETY”

In order to ground our investigations of process-based work in real-world understandings, we conducted a study of a major UK Building Society (roughly equivalent to US Savings and Loan), which we will call “the Society”. It has over three thousand employees nationwide, a central headquarters, about three hundred and fifty branch offices across the UK, and a number of regional and community offices. The Society offers a range of financial services including savings accounts, loans and insurance.

The work reported here is based on information from a number of sources. One is a “top-down” view of the organisation, gained from taking part in a one week intensive workshop at the Society's head office. This involved a number of senior members of the Society and a number of representatives from sales, marketing, technical and research arms of Xerox. The Society is a large customer of Xerox and the primary motivation for the workshop was to develop further the customer-supplier relationship by providing each organisation an opportunity to understand the other better. Xerox Research was involved for two additional reasons. The first was to help provide insights into the kinds of technologies and services currently in the research laboratories, and which the Society might expect from Xerox in the medium to long term. The second was to look for opportunities to learn more about longer-term requirements for organisations like the Society. Much of what we learned concerned the organisational structure and values, and the formal processes around which the Society was organised.

A second source of information is a “bottom-up” view of the organisation. This was derived from starting at the “end-nodes” of the Society, such as high street branches where business is generated. To provide a focus we concentrated on the Mortgage Application Process and “followed” it through the organisation, from the points where an initial application is received to where the cheque is issued if the loan is approved. We visited six branch offices, four Initial Approval Units (IAU) and two Final Approval Units (FAU) over a period of about nine months. Data was gathered from a combination of observation of work in progress, and from interviews with members of each department.

The Mortgage Application Process is an excellent example of a highly distributed, document-intensive process. The Society relied heavily on paper documents for carrying out the process. However, it was beginning to introduce new technologies to improve support for the process, which made this a particularly appropriate time to be working with them.

The Mortgage Application Process

The Mortgage Application Process is primarily about risk assessment. It entails building a file on each application. The

file is developed by checking and summarising information provided and gathering additional information. This is used to support assessment of the risk: if the Society lends this person the requested amount of money on this property, is it likely that the repayments will be made reliably and if they are not, is the property worth enough to safely cover the loan? Once a decision has been taken, the file becomes an accounting device to justify the decision. The process starts when a customer applies for a loan to buy a house.

The customer walks into a high street branch, discusses requirements with an adviser and fills in an application form. This form, along with supporting documentation, is sent along to the Initial Approval Unit (IAU) which takes input from a number of branches in the local community. The IAU takes the information provided by the customer at face value. They check the information provided and carry out a credit check for any signs of fraud. If there are no problems, they issue a preliminary offer to the customer and pass the application on to the Final Approval Unit. This unit carries out further, more detailed checks (e.g. getting input from third parties such as the employer and a valuer), and enters the information into the Society's central computer system. If all is well, they issue the formal offer and liaise with the customer's solicitor and their own head office to issue the loan to the customer at the appropriate time.

There may be deviations from this process depending on the precise case. For example, a large number of mortgage applications actually come from third party “introducers” rather than “direct” customers who walk into the branch. These introducers are other financial institutions or financial advisers who mediate in loan negotiations on behalf of their own clients, in return for a fee. A manager in charge of introduced business is typically associated with each IAU. Once the processing gets underway there is little difference between introduced business and direct business in terms of formal processes although there may well be some differences in the precise details of how it is dealt with.

Another example of deviation from the standard procedure applies in the case of a request for a loan that is greater than 75% of the value of the property. This has to go through additional checks (known as “SG referral”) to obtain a “Security Guarantee” which involves more stringent credit checks on the customer and additional insurance to cover the risk of the property value being insufficient to reimburse the Society should the borrower default on payments. In this case, the application has to go through the office of the Regional Underwriter. These deviations, then, introduce changes into both the documentary record of the process, and the nature of how that documentary record was constructed (and so, should be interpreted).

Since our aim in this paper is to focus on the coordination of work across process boundaries, we have only given a brief account of the activities performed by each participant in the

process. More information can be found elsewhere [20]. However, this should be sufficient for present purposes to give a reasonable feel for the kinds of activities involved and to ground the issues which this paper addresses.

Building a Case

A particularly interesting aspect of this process is the view of it as “building a case” for a loan. The Society regards itself as making the best effort to accept mortgage applications. From this perspective, then, the goal of the approval process is to make a convincing case that this loan is a safe one.

In many ways, this is in opposition to the view of mortgage handling as a “process.” From the process perspective, what is important is that each stage in the process is carried out, and that they are carried out in the right order. When we think of it as “building a case”, though, we can see that what is important is not simply that each step of the process is carried out, but precisely *how* each stage is conducted. When building a case, the goal is not simply to *collect* all the information, but to continuously *make sense* of it and assess it to determine whether or not a loan should be granted, and to justify that decision. While a process approach focuses on the structure of the work, this alternative view focuses on the “content.”

This emphasis on the assessment of information is crucial to the process, and so crucial to carrying out the work. However, as a result, problems emerge when we consider this process from the perspective of workflow technologies.

From one point of view, the work of mortgage approval at the Society *is* the work of assessment of the case and the information it contains. Even the process of collecting the information to build a case must be based on some sort of decision, formalised or *ad hoc*, about the persuasiveness of the case so far, and whether and what sort of information would serve to build it up. However, this assessment work cannot be captured conveniently in a process description. Process descriptions on their own are incomplete when the structure of the process, and the outcome of the work, depends so completely on the details of how each task is accomplished rather than what the task is or simply that it has been performed.

The way in which process-based systems divide the work into individual tasks may make the process of assessment *more* difficult to achieve. They isolate one task from another, routing the work artifact (and responsibility for working on it) from one person (and one computer workstation) to another. Process based systems are based on the premise that the execution of each task is largely an independent activity. However, in the mortgage approval task, the process of assessment is based on more than simply the information that might be placed on a form. What is critical in building a case is not the information itself, but how complete, trustworthy and compelling it is. Issues such as trust may depend on where the information came from, while it may be more or

less compelling depending on what other information is presented alongside it. So by splitting the work up into information collection tasks, the process system may actually make it harder to see the sort of contextual information on which an assessment needs to be based.

These issues are borne out by our field work observations. One of the things that was striking from our field work at the Society was the amount of time and effort spent contacting people in other offices, responsible for different parts of the process, to check on the current status of an application, or to ask for clarification. MacLean and Marqvardsen [20] point out that this communication is very much part of the daily routine even although it is not visible in the formal process descriptions, and is the sort of communication which is typically not supported in workflow systems. For some of these queries, improved information about how intermediate assessments were made and what information was used could help clarify issues that arise elsewhere in the process.

In general, then, our investigations at the Society highlight a gap between this form of work and process-based approaches such as those used in workflow systems. Process-based approaches typically divide the work into component tasks, based on the assumption that the actual activities comprising each task can be performed independently, as long as the form of the overall process is maintained (this being the role of the workflow system itself). What we see at The Society is an attempt to use a process description for work that evidently does not have this character; the content and context of the activities that comprise the individual tasks are all of considerable relevance to each other, and to the completion of the work as a whole. To support the work practices of the Society and other similar organisations, we need to be able to augment process-based approaches with support for seeing the context of each process instance.

MAKING PROCESS-BASED WORK VISIBLE

The fundamental issue that arises in our work at the Society is a need to make process-based work visible. Our study is not the only one to recognize the need to make work visible. Other work studies show how individuals make explicit aspects of their activities in order that other individuals can coordinate their work. In co-located work settings, it has been observed that activities are “made visible” within the work environment and can be interpreted by those familiar with the environment to understand the implications for their own work. Ethnographic studies demonstrate this in settings such as Air Traffic Control [17], office work [2] and City dealing rooms [15]. Making work visible, and maintaining an awareness of the work of others plays a central coordination role in these studies.

So, how do we make process-based work visible? Schmidt et al. [21] analyze existing computational notations within CSCW systems and evaluate them with regard to mecha-

nisms of interaction. They use Straus' term "articulation work" in describing the overhead of any cooperative working arrangement. They draw an interesting distinction in the locus of articulation work:

"Depending on the situation and the type of cooperative work, some of the articulation work is allocated to artifacts such as schedules, time tables, standardized office procedures etc. but in many situations the articulation is achieved through the interactions of the people involved in the cooperative work." (p. 155).

We use this distinction, between articulation work as a feature of interaction and as a feature of mediating artifacts, to review systems that address support for making process-based work visible.

ConversationBuilder [18] was developed as a support tool for providing active support for collaborative work activities. It contains "ActionSpace" windows, one for each conversation, which displays the state of that particular conversation. Different tools can be added to ConversationBuilder, such as an editor so that users can edit text and graphics to develop shared artifacts such as documents. Users can browse the system to find out how the various conversations they are involved in are related, show the relationships between the artifacts and check the versions of artifacts. Actions such as changes to the text are sent to all users. The system can say who else is looking at an artifact, who is editing it, which participants are active, etc. ConversationBuilder is based on a theory of conversation for action and, as a result, obligations are a major part of the system. ConversationBuilder offers many mechanisms to support the awareness of activities, but drawing on the distinction made by Schmidt et al., the mechanisms are focussed on supporting the interactions between people as distinct from providing artifacts that support articulation work.

In process support, some systems make the process description available which, though not intended as such, provides an artifact to support articulation work. For instance, Regatta focuses on enabling users to create and modify process definitions, even when the model is being enacted [24]. Regatta processes need to be defined in such a way that the average user can change and create process definitions. As a result, the ease of defining a process is important and Regatta includes a novel representation language, visual process language (VPL) to support this. The definition is available to users and so the process structure is visible, although originally this was intended for the purpose of changing the definition. Schmidt et al. refer to making the process definition explicit as useful in maintaining consistency during process rollback. Other workflow systems such as Mobile [16] have, similarly, sought to use the process description as a basis for flexibility in execution.

A process description serves to make the structure of the work visible; the content of the work however remains obscured. In the case of the mortgage application process, the content of the work matters. Some systems support easy

access to the content of the work at the time of enactment, however, they do not make the content visible to others. InConcert, for example, associates each task with a set of documents, where a document is any kind of abstract object that needs to be manipulated by the task [1]. Documents are associated with a task as part of the process definition and do not represent the actual documents used in a task. Eastman Software workflow ensures users have all the relevant information, applications and services they need at each step of a process, but again, the content of the work is not recorded or made visible to others [10]. WWAC [5] presents the history of the process in terms of the set of abstract decisions that led to this state, but these are divorced from the artifacts in terms of which the work itself is conducted.

Our goal is to find a way to represent the work that goes on around and within the process and to develop an artifact that makes it available to participants in the overall activity to support the coordination of their work. We will first give an outline of our technique, and then explore how it relates to existing approaches.

GAINING A PERSPECTIVE

The focus of this paper is a technique inspired by the examples described in our fieldwork which seeks to address the problems outlined above. Specifically, the "Perspectives" approach attempts to make visible the content of the work of individuals engaged in a workflow, with specific focus on capturing details of the context in which their work is achieved. The central idea is to provide access to this contextual information, thus allowing individuals to "gain the perspective" of others who have worked on the same case.

The Perspectives approach can be separated into two components; a means to capture contextual information as work takes place, and a means to organise this information in a manageable form. We address the first by recording details of relationships between documents and other resources used in the achievement of the work. We then organise this information according to the process description, such that relationship data are indexed relative to the tasks being executed when the relationships come into effect. We now look at these aspects in more detail.

Recording relationships

In our discussion of the Mortgage Application Process we saw how the work of the Society could be considered as "building a case" for a loan. This process of building a case entails constructing a "case file" which contains information relevant for assessing the loan in terms of level of risk. The information in the file is contained in documents, such as standard forms, letters of reference, and reports, which originate from a variety of different sources and are generated and added to the file at different stages in the process.

Transfer of the case file between individuals and departments is associated with moving between different stages of the process. Thus, as well as being a *product* of the process,

the case file is also the *vehicle* for conveying the results of a task, and thereby the state of a particular case, to those responsible for subsequent tasks. In a rationalistic view of the process, the case file should contain all the information required to perform the subsequent task before it is transferred, but we have already seen that this is not the case; information must often be clarified, interpreted and justified. The documents in the case file must be contextualised in order to do the work, hence the need for communication between those responsible for different parts of the process.

Resources

To provide a context for information flowing through the process, we seek to provide additional data on other documents and entities used in order to collect and process that information. These might be procedural manuals, notes maintained in the process of doing the work etc., that collectively contribute to the context in which a task is carried out. So, an entry on a mortgage application form detailing the applicant's monthly income might be augmented with a link to a letter from the applicant's employer providing the information; or a report on an applicant's mortgage history might be linked to documents provided by other Building Societies. The key is identifying the resources used in the course of performing a task, and making the relationships between these resources and the task accessible to others.

As defined in the Perspectives approach, "resources" are simply items that may assist someone in carrying out a task. They need not be documents, but could include people who possess relevant knowledge, printers which are pre-loaded with special, task-relevant forms, computer programs which generate information about other processes currently in progress and so on. Resources may be pre-specified as part of the process description, as specific items like procedural manuals or as place-holders which are bound to items when the process is executed (for example, a 'mortgage referrer' resource which, in each instance of the process, is bound to the name and phone number of the actual person who introduced the applicant to the Society). In the course of task execution, individuals can also attach any other items which they care to relate to a process instance (such as notes and annotations), and which may be of value in understanding how the work was done.

Relationships

There are a variety of ways in which relationships between tasks and resources can be represented and new relationships (or links) captured in the course of executing a process. We distinguish between the process of creating a link, and that of link typing. Both aspects may be performed by the system or the user, giving rise to a design space of possible solutions.

Link creation is concerned with how new relationships between documents, resources and tasks come into being. For relationships defined as part of the process description, we might consider this aspect as part of the process modeling

task. For more "ad hoc" relationships, however, which are defined as part of a single process instance, we need a way to capture links more dynamically. This might be accomplished by users creating links explicitly (e.g. by dragging documents on top of one another), or by the system determining that links exist (e.g. by collecting information about user activities, such as that a document was printed, or sent by email). On the other hand, link typing is concerned with what the relationships represent. An "untyped" link between documents would show only that "a relationship exists between these documents." By attaching link types, we could show that a document "is derived from that one", "was used as a source of information for that one", "is superseded by that one" and so on.

Within the space of possible solutions these distinctions present, the simplest mechanism would be to require users to explicitly create untyped links while a task is being executed. Explicit, user-performed typing could be added by associating place-holders with each task in the process description for potentially relevant pieces of information. So, for example, a form associated with a task could provide a "source document" place-holder associated with relevant items on the form, allowing users to associate information on the form with other documents. Alternatively, users could be allowed to annotate untyped links with text comments which describe the nature of the relationship.

There are clearly trade-offs implied by these options. Following Grudin [12], a balance must be struck between the generator of the information and those attempting to retrieve and make sense of it later. For example, while explicit user creation and typing of links imposes a burden on the information generator, it may increase its utility by forcing specification of exactly how the linked items are related. Similarly, implicit linking makes it easier for people at later stages in the process to find information that those earlier on might not have thought to record; while explicit approaches are more respectful of the (oft-observed) ways in which those performing the work will specifically *craft* the presentation of their activity for specific individuals and intended audiences. So, in different contexts, different approaches will be useful, and the Perspectives approach does not prescribe particular design choices for the creation of resource relationships. (We describe how one such choice can be realised in the following section, which discusses our implementation of a prototype application based on the Perspectives approach.)

Process-structured Case Histories

With this approach, as the execution of a process progresses, we envisage a general accumulation of relevant information in the form of relationships. The need is then to organise this information so that others can access it in such a way that it supports the contextualisation of the work as it was achieved earlier in the process.

To accomplish this, it would be possible to view the structure of documents and related resources as forming a semantic network, and to use this structure as the means of navigating the space of relationships. The process of adding relationships can be viewed as the continual construction of the network, and the case documents might be considered as entry points or 'root nodes'. In this form, however, we find this approach unsatisfactory. Over time, the pattern of relationships is not stable; someone performing one part of the process may see different relationships between items than another person. The pattern of relationships is likely to change over time, yet it is necessary to preserve relationship information from earlier in the process if we are to support viewing of the process from each participant's perspective.

We might address this by adding a "timeline" to show the pattern of relationships organised over time, from the start of the process to the end. The LifeStreams system [11] uses this technique to organise document collections. However we see drawbacks in this approach as an organising mechanism for our purposes. For example, should the process allow tasks to execute in parallel, the timeline might present a confusing picture as two or more perspectives are coincident and their related resources intermixed. It would also be difficult to draw comparisons between different executions of the process, as tasks may be of different duration, depending on the details of each case.

So, rather than introducing a linear timeline, we take a different approach which uses the process description itself to structure the history of resource relationships. When relationships are forged, they are indexed according to the task being executed at the time. The task-relative history which results is also a person-relative history, as tasks are typically performed by single individuals, and it is therefore straightforward to derive any one individual's 'perspective' on the work process. At the same time, while supporting the variability of relationship information across the process, this approach allows straightforward navigation of this information and comparison between different process instances at the task level.

Contrasting with Other Approaches

Amongst the existing approaches, we can see various ways in which work is made visible. One approach is to make the process model itself visible, so that working activities can be seen in their context, and so that the model can itself be an object of reflection and modification. This is the approach taken by Regatta [24], and that discussed by Schmidt et al. [21]. Another approach is to augment the process description with information about current activities, as Conversation-Builder does [18]. Perspectives offer a third option. Our approach gives access not only to current information, but also to historical information. The information we offer is not simply about the abstract process, but concerns specific *instances* of process execution. While we loosen the task model by allowing more information to flow between tasks,

we continue to capitalise on its value as a means to articulate and understand the work organisation.

PROTOTYPE IMPLEMENTATION

In order to explore these ideas in more depth, and to consider the implications for the design of workflow interfaces, we have embodied them in a prototype. Our prototype builds upon an existing document management infrastructure, which embodies workflow functionality as well as other features for personal and workgroup document management. The prototype extends this by allowing resources to be associated with tasks in process instances, and providing users with the means to visualise and explore them.

Infrastructure

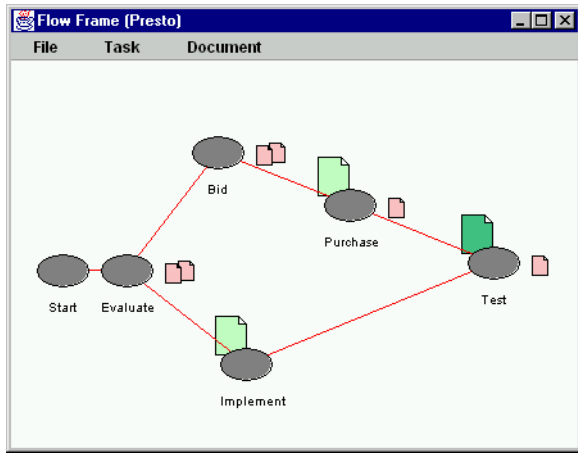
The infrastructure is a document management system developed as part of the Placeless Documents project under way at Xerox PARC. Placeless Documents is a system for personal and workgroup document management which is based on the use of document properties as a uniform mechanism for organising, viewing, controlling, grouping and interacting with documents [9]. Some of our experiments have looked at the use of document properties (and particularly active properties) to provide workflow's coordinative functions independently from the applications used to interact with the documents [19]. The prototype presented here forms part of an ongoing investigation of the relationship between these property-based document systems and workflow technologies. A number of features make Placeless Documents a particularly compelling substrate for a prototype implementing the Perspectives model.

The first is that the Placeless infrastructure integrates document content from a wide variety of sources, including everyday workstation files on local disks and remote file servers, as well as Web documents, database records, virtual documents, and so forth. In other words, it provides a uniform means of interacting with documents wherever they might be stored. Since task resources might include a wide variety of sorts of information, drawn from a wide variety of sources, this integrative function is valuable in an infrastructure for Perspectives.

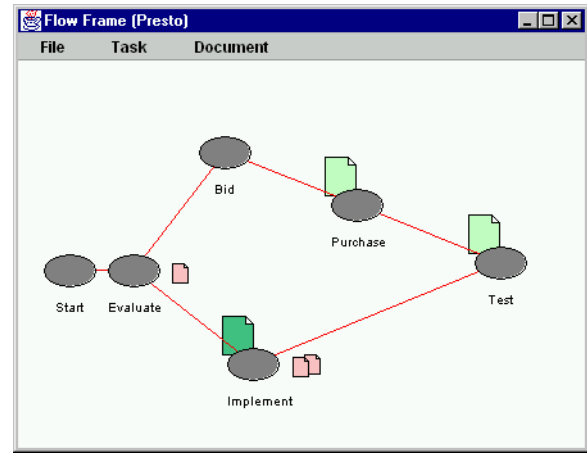
The second is that the notion of Perspectives fits very naturally with the idea of document properties. Document properties, in Placeless Documents, are arbitrary attributes of documents that are meaningful to users. The association between a document and a resource used in the work associated with that document fits naturally into a model where a document has a set of attributes that record information relevant to it. So, there is a close fit between the conceptual models at work.

Prototype

Figure 1 shows some examples of using the Perspectives prototype. There are five basic entities represented in the screen shots: *flows*, *tasks*, *process instances*, *process documents*, and *resources*.



(a.)



(b.)

FIGURE 1: As different process documents are selected, different task-related resources are brought into view.

A flow is a workflow representation of a process. There is a single flow represented in figure 1, representing a simple software deployment process. The flow is made up of a linked set of six *tasks*, which are steps in the process represented.

The flow represents the process in general. Specific instances of executing the flow (e.g. deploying a particular piece of software) are *process instances*. Each instance is associated with a document, called the *process document*. These are graphically represented in the interface as the larger document icons. Each process document is shown by the task that is its current execution state. So, in figure 1, there are three process documents, for three process instances, in the states Test, Implement and Purchase.

Finally, for each process instance, there are also resources that are associated with the different tasks in the flow. The resources are represented in the interface shown in figure 1 as the small icons. As we have seen, the interface shows all currently active process documents for the current flow. At any given moment, one of these documents is selected. When the user selects a process document, the resources for that instance are shown, beside the task for which they are relevant. So, in figure 1(a), the right-most document, associated with the task “Test”, has been selected (and is highlighted). This has caused the resources for that instance to be displayed. One of these is associated with task “Test”, one with “Purchase”, two with “Bid” and two with “Evaluate”. In figure 1(b), the user has selected a different process document, the one associated with task “Implement”. The resources for the first document have been hidden, and those for the second document have come into view; we can see two associated with the current task, “Implement”, and one associated with the previous task, “Evaluate”.

A Property-Based Approach

The property-based document system on which this is implemented offers both document and object system features [9].

This means it can seamlessly store both documents and representations of the process. It also offers other benefits.

First, since any document can have arbitrary properties attached to it, we can implement the Perspectives approach in a way that does not interfere with other document and application processing. While documents are linked into the system, they can still be accessed with whatever applications are appropriate; if a document is formatted using Microsoft Word, then double-clicking on it will fire up Word, and so forth. This level of integration is a critical feature in supporting our approach. Both the documents that serve as resources for a task, and the applications that operate over them, can exist outside our system. Implementing Perspectives means that we need to be able to reach beyond the boundaries of the process system itself.

Second, we can exploit the system’s ability to integrate a wide range of potential document object types. In our system, the same interaction model applies not just to documents, but to all objects. So, printers, people and other entities can be made uniformly available within the infrastructure. This means that the system can record non-document resources. It can also record historical information about people (or email addresses), active devices, and so forth. Since resources are more than simply documents, this level of flexibility is crucial.

Applying the Prototype to The Society

Since we drew upon the example of building a case at The Society to characterise the problems of understanding process-based work, we will now revisit that scenario as an illustration of how the prototype operates.

The Society, as we have explained, has an established process by which mortgage applications are handled. In our prototype, as with a conventional workflow implementation, this process is represented as a graph of tasks and transitions representing the various stages of handling and assessing a

mortgage application. Any given mortgage application is handled as an instance of this process. When the application arrives, it is entered into the system and identified as a process document. Being a process document is simply a property of the document; the property is attached to the document, causing it to appear in the browser.

At this point, being a process document creates a relationship between the document and the particular workflow representation with which it is associated (in this case, the mortgage application process.) Being a process document means that new operations are available on the document, such as moving it from one task to another, and so forth. These new features are available on the document concurrently with any others that are relevant for other applications.

Our prototype follows an “explicit link” model in which new resources must be directly associated with the document by the user. However, since our infrastructure can detect operations in the system, it may choose to prompt the user when they open related documents. So, for example, the first step may be to verify the correctness of information on the form. As other documents (e.g. phone lists, financial records) are called up to check information, the system can suggest that these be linked in as resources for the current task.

Any resources are associated with the current document and the current task. They appear as was illustrated earlier in the interface, as small icons beside the relevant task; and as the user moves from one process instance to another, the relevant resources are displayed.

As we have explained, the mortgage approval process at The Society is split across multiple sites. For a remote site, determining the strength of the case for a loan is dependent not simply on the fact information has been gathered, but the sources from which it was gathered, etc. So, for example, at a later stage in the process, it can become necessary to understand *how* it was that the information on the application form was verified. For example, if financial information (such as salary history) was gleaned from an internal database (because the applicant is a Society customer), then it can be more accurately and reliably determined than if it came from another institution, or from a telephone conversation with an employer. In a conventional implementation (and in the case of everyday work at The Society), this information must be determined by a phone call to the office where that stage of the process was carried out. Thus we see in the fieldwork a large amount of communication back and forth to clarify information about applications in process. In our prototype implementation, by contrast, the interface makes available not only the process document, but also the resources associated with its execution. So, when the process document is called up, the resources are displayed on the screen, associated with the relevant tasks, that describe the pattern of document relationships that constitute the context of execution of this particular process instance, including the related

documents by which the assessment was made; a database record from the Society’s own database, for example, or the transcript of an interaction with another institution, or a letter of verification from an employer and so forth.

Associating resource information in this way provides users with a means to access the contextual information that surrounds each specific instance of process execution. As we have seen from the fieldwork, this information is already critical to their work; what our prototype illustrates is how the Perspectives principle can be used to provide integrated access to this information as part and parcel of the work of carrying out the process itself.

DISCUSSION

In a previous paper describing the Freeflow system [8], we outlined a general philosophy of workflow technology which we were pursuing. Our goal was to look for ways in which the traditional mechanisms employed by workflow systems could be augmented so as to achieve a better parity between the structure of computationally-encoded processes and the fluidity of everyday working practices. We posited a number of ways in which this could occur, based around three dimensions of workflow technology: from work analysis to a process description, between the states of that description, and between the description and the work carried out. The approach we took in the original Freeflow paper concentrated on the second of these, and provided a richer model of process state so that we could reconstruct process descriptions in terms of dependencies between tasks rather than the order of their execution. This allowed users to engage in the work governed by the system in a more fluid manner, while still providing some computational leverage. The Perspectives approach described here constitutes an effort to explore the third of the dimensions outlined in that earlier paper. We had suggested there that another dimension of workflow technology to reconsider was that from the process description to the work itself. Instead of using the process description to *drive* the work of individuals, we could use it to *explain* those activities as they occurred.

Although the notion of Perspectives was driven primarily by the outcome of field work investigations in the Society, they fit directly into this idea of using the process description in an explanatory role. The process description provides the context in which to understand the activities that go to make up the work of executing a task. Perspectives are a mechanism for linking the work and the process description together so that this can be done.

In providing these sorts of functions within a workflow system, we are aiming to support three fundamental notions in the coordination of working activity.

The visibility of work. By recording and later re-presenting information about the accomplishment of particular tasks, our approach is aimed at taking details about the performance of individual tasks which would otherwise be

invisible and make them visible to others through the subsequent execution of the process. Making work visible in this way crucially supports the coordination of individuals engaged in the collaborative accomplishment of any activity. The concern with support for awareness in collaborative systems [7, 13], or field studies discussing the fine-grained coordination of activity in shared spaces where the activities of individuals are made mutually accessible [e.g. 15] are testament to the role of the visibility of work in achieving smooth coordination. Perspectives support the visibility of work primarily by recording information that would otherwise be lost.

The interpretability of work. It is not sufficient to make work visible; it must also be made interpretable. That is, in the cases where work is visible, the viewer must have enough context to be able to recognise the work for what it is, and to understand its import and significance, particularly for their own. The critical feature of the visibility of work, after all, is that it be visible *so that* it can contribute to the work in hand, and in particular, so that one person can see, in the activity of others, the consequences for his or her own activity. Bowers and Rodden [6] point to a similar kind of issue in emphasising the need for an understanding of the broader organisational context to ensure the success of a new system. Harper [14] describes a rather different situation in which desk officers responsible for writing certain kinds of reports regularly rotate around different responsibilities. Each time someone takes over a new desk it takes some time to work out what resources are the most important for the new role. Perspectives support the interpretability of work primarily through the use of the process description as a contextualising element, in its role as an index into the historical document relationship metadata. This kind of information should help someone else, whether responsible for another part of the same process, or taking on a new role in an established process, to make better sense of the sometimes cryptic information which has been derived from a variety of sources. The Perspectives approach preserves a record of these sources and resources and provides access to them.

The variety of work. When we see work described in terms of a set of predefined tasks with stated preconditions, outputs, activities and relationships to other people and other tasks, we can be misled into believing that any work which can be so structured must be, by definition, rote and homogeneous. The definition of working tasks often denies the variability of the work and hence the variety of means which must be employed to perform it. Our goal in supporting the visibility and interpretability of work through mechanisms like Perspectives is to provide workflow systems with a richer notion of the fluidity of work and with a means to support people engaged in its creative accomplishment.

The Status of Process Descriptions

While accepting the need to provide for the intelligibility of work in cases where it may be obscured by process represen-

tations, one might be tempted to wonder why we make recourse to those same process descriptions as a means to organise that information. Process descriptions are the source of all the trouble in the first place, surely? Perhaps we should remove them altogether?

This echoes the conventional mis-reading of Suchman's [22, 23] comments on the relationship between plans and the improvised sequential organisation of work. As Bardram [3, 4] has observed, Suchman's questioning of the status of plans is not an attempt to obliterate them altogether. Plans play a critical role in Suchman's account of situated action, as a resource for the organisation and management of activity. Plans may not be generative of action, but they are a fundamental means of reasoning about action, and in particular of reasoning about procedures. Similarly, as we have discussed, Schmidt et al. [21] discuss the role that artifacts such as process descriptions can play in the management of articulation work. Perspectives represent one technical response to this reorientation of abstract procedural representations as a tool for reasoning about work.

CONCLUSIONS

A commonly-observed problem with workflow or process-based systems is the way in which they lead to a decontextualisation of activity. Actions separated by the edges of tasks are invisible to each other, and the system eliminates individuals' abilities to manage the overall progress of work rather than the specifics of each individual task. This decontextualisation also leads to a homogenisation of the process execution – an assumption that it is performed in just the same way every time. We have presented a technical approach that tackles these twin problems.

Perspectives provide a means to “see the world as others saw it.” Perspectives is a mechanism which allows users of a process system to see the objects which are the focus of the process from the point of view of those who have carried out or are carrying out other parts of the process.

Perspectives make the details by which work is accomplished visible to others, and so provide for the mutual intelligibility of working processes. Our primary goal in doing this is to smooth the coordination of work across tasks in a workflow system. At the same time, it also enriches the model of work embodied by the model by acknowledging the variability of that work and the range of mechanisms that can be employed to accomplish it.

Perspectives exemplify an approach to workflow technology which proceeds from the assumption that the role of the system is not to *automate* the flow of work, but to *smooth* it, by providing users with the resources to better manage their own coordination. In this view, process descriptions are a resource for task coordination, rather than a prescription for process execution. We hope that techniques such as these can provide a bridge between understandings of the fluid

nature of everyday working practices and opportunities for technical system support.

Acknowledgments

We would like to thank Graham Button, Richard Harper, Pernille Marqvardsen, Kenton O'Hara, Kevin Palfreyman, James Pycock and Abi Sellen for their contributions.

References

1. Abbott, K. and Sarin, S. (1994). Experiences with workflow management: issues for the next generation. Proc. ACM Conf. Computer-Supported Cooperative Work CSCW'94 (Chapel Hill, NC), 113-120. New York: ACM.
2. Anderson, R.A. and Sharrock, W. (1993). Can organisations afford knowledge? Computer Supported Cooperative Work, 3(1), 143-161. Dordrecht: Kluwer.
3. Bardram, J. (1997). Plans as Situated Actions: An Activity Theory Approach to Workflow. Proc. European Conf. Computer-Supported Cooperative Work ECSCW'97 (Lancaster, UK). 17-32. Dordrecht: Kluwer.
4. Bardram, J. (1998). Designing for the Dynamics of Cooperative Work. Proc. ACM Conf. Computer-Supported Cooperative Work CSCW'98 (Seattle, WA), 89-98. New York: ACM.
5. Bose, P. and Zhou, X. (1999). WWAC: WinWin Abstraction Based Decision Coordination. Proc. Intl. Jnt. Conf. Work Activities Coordination and Collaboration WACC'99 (San Francisco, CA). New York: ACM.
6. Bowers, J. and Rodden, T. (1993) Exploding the interface: Experiences of a CSCW network. Proc. INTERCHI'93, (Amsterdam, April 1993), 255-262. New York: ACM.
7. Dourish, P. and Bellotti, V. (1992). Awareness and Coordination in Shared Workspaces, Proc. ACM Conf. Computer-Supported Cooperative Work CSCW'92 (Toronto, Ontario). New York: ACM.
8. Dourish, P., Holmes, J., MacLean, A., Marqvardsen P., and Zbyslaw, A. (1996). Freeflow: Mediating Between Representation and Action in Workflow Systems. Proc. ACM Conf. Computer-Supported Cooperative Work CSCW'96 (Cambridge, MA.), 190-198. New York: ACM.
9. Dourish, P., Edwards, K., LaMarca, A., Lamping, J., Petersen, K., Salisbury, M., Terry, D. and Thornton, J. (1999). Extending Document Management Systems with Active Properties. Xerox PARC working paper (submitted for publication).
10. Eastman Software. (1999). http://www.eastmansoftware.com/products/Workflow/pr_work_index.htm.
11. Freeman, E. and Fertig, S. (1995). Lifestreams: Organizing your Electronic Life. AAAI Fall Symposium: AI Applications in Knowledge Navigation and Retrieval (Cambridge, MA).
12. Grudin, J., 1987. Social evaluation of the user interface: Who does the work and who gets the benefit? Proc. INTERACT'87, 805-811. Amsterdam: North-Holland.
13. Gutwin, C., Greenberg, S. and Roseman, M. (1996). Workspace Awareness in Real-Time Distributed Groupware: Framework, Widgets, and Evaluation. In Sasse, Cunningham and Winder (eds), People and Computers XI: Proc.HCI'96 (London, August 20-23). Heidelberg: Springer-Verlag.
14. Harper, R. (1998). Inside the IMF: An Ethnography of Documents, Technology and Organisational Action. London: Academic Press.
15. Heath, C., Jirotko, M. and Luff, P. (1993). Unpacking collaboration: the interactional Organisation of Trading in a City Dealing Room. Proc. European Con. Computer-Supported Cooperative Work ECSCW'93 (Milan, Italy), 155-170. Dordrecht: Kluwer.
16. Heintz, P., Horn, S., Jablonski, S., Neeb, J., Stein, K. and Teschke, M. (1999). A Comprehensive Approach to Flexibility in Workflow Management Systems. Proc. Intl. Jnt. Conf. Work Activities Coordination and Collaboration WACC'99 (San Francisco, CA). New York: ACM.
17. Hughes, J.A., Randall, D. and Shapiro, D. (1992). Faltering from Ethnography to Design. Proc. ACM Conf. Computer-Supported Cooperative Work CSCW'92 (Toronto, Canada). 115-122. New York: ACM.
18. Kaplan, S.M., Tolone, W.J., Bogia, D.P. and Bignoli, C. (1992). Flexible, Active Support for Collaborative Work with ConversationBuilder. Proc. ACM Conf. Computer-Supported Cooperative Work CSCW'92 (Toronto, Canada). 378-385. New York: ACM.
19. LaMarca, A., Edwards, K., Dourish, P., Lamping, J., Smith, I. and Thornton, J. (1999). Taking the Work out of Workflow: Mechanisms for Document-Centered Collaboration. Xerox PARC working paper (submitted for publication).
20. MacLean, A. and Marqvardsen, P. (1998) Crossing the border: Document Coordination and the Integration of Processes in a Distributed Organisation. In Wakayama et al. (Eds): Information and Process Integration in Enterprises: Re-Thinking Documents. 109-124. Boston: Kluwer.
21. Schmidt, K., Simone, C., Carstensen, P., Hewitt, B. and Sorensen, C. (1993). Computational Mechanisms of Interaction for CSCW. Esprit COMIC Project Deliverable 3.1. ISBN 0-901800-30-9.
22. Suchman, L. (1983). Office Procedures as Practical Action: Models of Work and System Design. ACM Trans. Office Information Systems, 1, 320-328.
23. Suchman, L. (1987). *Plans and Situated Actions: The Problem of Human-Machine Communication*. Cambridge: Cambridge University Press.
24. Swenson, K.D., Maxwell, R.J., Matsumoto, T., Saghari, B. and Irwin, K. (1994). A Business Process Environment Supporting Collaborative Planning. Collaborative Computing, 1, 15-34. Chapman & Hall, London.