ABSTRACT
We have previously proposed a process assessment method (known as IMPACT-A), based on a process reference model, for the assessment of hypermedia development processes. This reference model is used to guide the identification of suitable process quality attributes and subsequent assessment activities - or abstract tasks. These abstract tasks allow us to measure the selected process attributes. The abstract tasks should define those process entities which are to be assessed, what aspects of the development are involved (navigation, behavior, etc.) and what activities need to be carried out. Each abstract task can therefore be viewed as focusing on an abstraction of specific fragments of the development process. In other words, we have specific process patterns which are manipulated and evaluated by each abstract task.

A particular process pattern may, for example, involve those elements of the development where the underlying information structure is used, along with other inputs, in developing the high-level navigation structure. The pattern will show the inter-relationship between activities, resources, inputs and outputs, etc. The result will be a set of process patterns which can be used for both planning development projects and for carrying out assessment of existing projects. The paper will provide a basis for representing these process patterns, along with several typical patterns.

KEYWORDS: Hypermedia, development, process, methodology, evaluation, assessment, patterns.

INTRODUCTION
There has been a rapidly increasing interest in hypermedia and the Web, and as a result a growing focus on design methods, models and techniques for hypermedia applications. One area that has been at least partially overlooked is the process by which hypermedia applications can or should be developed. The work that has been carried out - such as work on design models (e.g. HDM [3]) and methods (e.g. RMM [5] and OOHDM [11]) - has tended to either assume a predefined process with little consideration of how the process can best be adapted to suit a particular problem, or has provided guidance on individual stages of development but not shown how these stages integrate into an overall cohesive pattern.

As an example, the authors of RMM explicitly state that it is only suited to a specific class of applications, but don’t provide any mechanisms for fitting the process to the domain. This stands in stark contrast to the recent emphasis on process in other disciplines, such as software engineering, and as exemplified by initiatives such as CMM [10] and SPICE [12]. The prevalent view in software engineering is well-summarized by these remarks:

Software development is the most communication intensive of all engineering processes. This unique software process characteristic suggests significant productivity gains are more likely to be realized through communication improvement rather than through technology. Communication effectiveness is a people issue controlled by organization structure, management approach, and the development environment. [6]

We would argue that hypermedia development is even more communication intensive than software development, due to the greater variety in team makeup and backgrounds in hypermedia: content experts, user-interface designers, programmers, graphics designers, etc. Diverse backgrounds, uncertain requirements, and emerging new technologies all add to communications overhead. To better support communication and creativity, we believe a defined process is
needed to systematize, or even automate, that part of the process that is commonly performed.

Another elements which typically makes hypermedia (and especially Web) development unique is the nature of development, release and maintenance cycles. Traditionally software systems have relatively well defined development and release cycles. The maintenance is typically carried out in well-structured increments. In contrast to this, Web sites have an initial development period up to the point where they are first released, and then further development is carried out in a complex combination of fine-grained and almost constant modifications, more substantial structural changes, and possibly even occasional complete overhauls and re-releasing. Again, this requires a sound understanding of the processes by which this development is carried out and managed.

To partially address these issues we have previously proposed a process assessment method [7] (known as IMPACT-A), based on a process reference model. This method focuses on the assessment of hypermedia development processes. A reference model is used to guide the identification of suitable process quality attributes and subsequent assessment activities - or abstract tasks. These abstract tasks allow us to measure the selected process attributes. The abstract tasks should define those process entities which are to be assessed, what aspects of the development are involved (navigation, behavior, etc.) and what activities need to be carried out. Each abstract task can therefore be viewed as focusing on an abstraction of specific fragments of the development process. In other words, we have specific process patterns that are manipulated and evaluated by each abstract task.

In this paper we provide a brief overview of IMPACT-A and the associated reference model. We then show how this model can be used to define possible process patterns. A particular process pattern may, for example, involve those elements of the development where the underlying information structure is used, along with other inputs, in developing the high-level navigation structure. The pattern will show the inter-relationship between activities, resources, inputs and outputs, etc. The process patterns which result will define approaches to developing and maintaining hypermedia, and can be used for both planning development projects and for carrying out assessment of existing projects.

BACKGROUND

Process Assessment and Modeling

Hypermedia development still has very few documented development processes or methods. By process, we mean the activities which are carried out, relationships between these activities, resources which are used, artifacts which are created, and the communications paths amongst the people involved in the project. Nanard and Nanard [8] discuss the need for a development process for hypermedia authoring in the context of the role that tools and prototypes play within the overall process. They mention that the process in hypermedia design is almost always overlooked, to the detriment of the development.

Even if we do not have a formal process to follow, hypermedia applications are developed, and therefore some process is being adopted, even if it is informal, ad hoc and undocumented. If this situation is to be improved then we need to begin to understand hypermedia development processes - which implies an ability to both model and assess processes. Unfortunately no comprehensive mechanisms currently exist for assessing the effectiveness, flaws, successes, and inadequacies of hypermedia development processes. It may be possible to use process assessment mechanisms from other disciplines such as software engineering. However by doing this we run the risk of misinterpreting aspects of the process. As was discussed in the introduction, there are sufficient differences between development of hypermedia and development in other domains to result in processes that are often substantially different.

Advocates of process modeling contend that, by explicitly modeling development processes, one can systematically analyse and hence improve the process and subsequently improve the quality of the end-product. Implicitly they assume that process improvement will lead to application improvement, an underlying assumption which is inherent in software process improvement approaches such as the Software Engineering Institute’s Capability Maturity Model (CMM) framework [10], and the SPICE architecture for process assessment [12].

This raises the question as to how we actually assess development practices. There has been a relatively large volume of research work focussing on the evaluation of interactive applications in general and hypermedia in particular. Probably most notable has been the work on usability - such as that by Nielsen [9] and Garzotto et al on SUE [4]. This work on evaluation has focussed almost entirely on evaluation of applications - rather than evaluation of processes. One exception is the work by Christodoulou et al [1] on a framework for the evaluation of application development systems. Even this work however does not directly consider the development process, but rather looks at the support for the development process provided by different systems and methodologies. In other words, it focuses on evaluating the method, and not the actual process when that method is applied in specific projects. Nevertheless it
does provide some interesting insights into those aspects which might be considered important in assessing actual practices. For example, the approach considers aspects such as support in the methodology for conceptual data model design and abstract navigational model design.

**IMPACT Process Assessment: Overview**

The IMPACT-A method [7] aims to identify quality attributes of the development process. Based on these attributes abstract tasks are specified which can be used to measure the attributes, and hence assess the process. IMPACT-A has two primary evaluation phases: the preparatory phase and the execution phase. In the preparatory phase a conceptual framework is created which can be used to guide the assessment in a systematic and comprehensive way. This phase includes three main components: the selection of a model, the definition of a set of process quality attributes and the specification of a set of abstract tasks for measuring the quality attributes. The preparatory phase is performed a single time for a specific class of applications and desired assessment goal(s). The design of the model, attributes and abstract tasks will be aimed at addressing particular issues. For example, if we wish to better understand the effort required to reliably develop an effective navigation structure, then we need to create a process model which captures those activities that consider or influence the navigation structure. We also need to consider attributes related to development effort.

The execution phase is carried out for each development project. The model is used to support an understanding of the development process. The attributes assist the developer (or assessor) in identifying those aspects of the development that need to be assessed. The abstract tasks provide a guide for how the attributes are to be assessed. The tasks are instantiated as specific inspection activities on the process. The results of these inspection activities are then interpreted and used to draw conclusions about the process or identify possible problems with the process.

Flaws in processes that can be identified by the IMPACT-A method include excessive communication with the client, poor understanding of how to develop applications for maintainability, and lack of ability to accurately estimate costs and development schedules.

**IMPACT Process Reference Model**

To guide the creation of project-specific models the IMPACT method utilises a reference model of the hypermedia development process. The purpose of the reference model is to provide a basis for identifying those elements that exist in any model of the development process and to provide a common terminology when discussing hypermedia development processes. This will also be important in describing and document process patterns. In the context of
assessment the reference model is used to support the identification of relevant process attributes. In project planning and process modeling, the reference model would usually be instantiated as either a prescriptive model (which guides the development) or a descriptive model (which describes what occurred).

The reference model is shown in Figure 2. This model is a composite of hypermedia concepts and process modeling concepts from other disciplines. The model contains three dimensions – a process entity dimension, a hypermedia dimension, and a time dimension. A process model that adheres to the reference model consists of a series of process entities within this three-dimensional space that have defined inter-relationships. The process entity dimension defines the classes of entities that can exist and the relationships that exist between entities of different classes. This dimension has been adapted from the Spearmint framework [13] for software processes and includes resources, activities, and artifacts. The second dimension of the model defines the hypermedia focus of the various model entities. These foci have been adapted from work on hypermedia design models such as HDM [3] and include: structure, navigation, behavior, and interaction (or interface). These foci can be viewed as a series of layers. Each process entity in the model can be perceived as existing in one or more layers. The final dimension in the reference model is time. Activities are carried out in a certain order, artifacts are produced and then utilised at certain times, and resources may only be available at specific stages of the development.

PROCESS PATTERNS

Based on the above observations it becomes possible to define patterns which re-occur in the hypermedia development process. These patterns show how to approach the development of an application (or part of the development) for specific situations – much as product or application patterns can provide patterns for aspects of the design or implementation of a hypermedia product.

We are currently undertaking a large-scale, and on-going analysis of a number of website development projects. We are using the IMPACT process reference model to guide the development of process models, and then analysing these models to identify commonalities. At present this work is only in its early stages and the process patterns included below are intended to illustrate only. (note: we have used the notation of [2]).

Partitioned Incremental Development Pattern

Name: Partitioned Incremental Development

Intent: To provide a basis for development of hypermedia / Web applications in an incremental manner, supporting progressive integration and delivery of components.

Motivation: As discussed above, much Web development and maintenance occurs in specified increments. The overall structure of the Website is designed, implemented and released. Subsequently, sub-components are developed and integrated into the site. Each of these components can be treated essentially as a separate development project, being developed largely independently from the other components provided the initial partitioning is carried out suitably.

For example, one project (with which we have been involved) involved the initial development of a website framework along with high-level pages and static version of many of the lower-level pages. A number of projects were then identified for automating the maintenance of these low level pages. The initial specification of requirements and partitioning of the development into sub-projects was careful to consider the interactions between these components (especially the cross-references and any functional or interface coupling). An example of a sub-projects in this case included converting pages relating to ‘upcoming events’ to a database driven system utilising dynamic creation of pages and delegation of updating the database to appropriate administration staff. Another project example was the inclusion of a comprehensive search facility. A further example was provision of a set of discussion groups, cross-referenced to content elsewhere in the site.

Applicability: This pattern is useful where the development of a website is under a tight timeframe, especially where there is pressure to obtain a web presence (or a modification of an existing presence) as rapidly as possible. In these circumstances, developing a full site is often impractical within the desired time constraints. Instead the basic framework can be developed and an initial site made available, whilst the various enhancements or additions to the site can be worked on independently. Further, it allows separate development teams to work on different aspects of the site, thereby facilitating parallel (and hence more rapid) development. It is most relevant for larger projects with
short-time frames – i.e. those requiring more than several members on the development team – as it facilitates parallel development.

**Structure:** Figure 2 shows the basic structure of the pattern. This includes an initial planning (and optional risks analysis) stage, followed by the determination of the overall site requirements which has an outcome a set of website requirements. These are used both in the development of the overall site, as well as the basis for partitioning the site ‘enhancements’ into a set of sub-specifications for each sub-project. Each of these sub-projects can then be carried out largely independently until the point where it is integrated into the overall website. Running in parallel with the whole process is the project management.

**Consequences:** This pattern provides an approach to managing short timeframes and parallel development in web projects. The benefits are more rapid development, earlier release of an initial site and better control
of the project through the use of well defined deliverable increments. The problems associated with the use of this template include the temptation to deliver an initial site with poor content or functionality, and problems in determining inter-relationships between the various increments.

**Other Patterns**

(*Other patterns to be added in final version of paper – especially patterns showing not just process activities, but focussing on other possible process elements. For example, a process pattern showing the relationship between different development artifacts, or even a ‘resourcing pattern’*)

**CONCLUSIONS**

This paper has proposed an approach to specifying patterns for the process of developing hypermedia and web site applications. This has been demonstrated by a simple example. Further work on creating a pool of process patterns is currently underway. This is carried out by initially creating process models of numerous projects, and then evaluating these models in order to look for similarities in the processes. These similarities can then be abstracted to provide a pattern. The actual projects which contributed to the pattern are analysed to determine the applicability and consequences of using the pattern.

This approach can potentially provide a much more structured approach to the development of hypermedia applications. It should allow us to more clearly define the development approaches which are best suited to particular problems. The potential results include more cost-effective development, an improved ability to manage the projects and higher quality applications.

**REFERENCES**