LANCASTER
UNIVERSITY

Computing
Department

Managing Process Inconsistency using Viewpoints

Ian Sommerville, Pete Sawyer and Stephen Viller

ABSTRACT

This paper discusses why inconsistencies in software processes are inevitable and presents an approach to process analysis which helps discover and manage these inconsistencies. By analogy with viewpoints for requirements engineering which allow multiple perspectives on a software product specification to be managed, we have developed the notion of process viewpoints which provide multi-perspective descriptions of software processes. A process viewpoint includes a statement of focus or 'world-view', a set of sources of process information, a process description and a set of organisational concerns which represent goals or constraints on the process analysis.

We present a description and rationale of process viewpoints, discuss the process of applying process viewpoints for process understanding and illustrate the overall approach using an industrial example drawn from processes which are part of a safety-critical system development.

Managing Process Inconsistency using Viewpoints

I. Sommerville, P. Sawyer, S. Viller
Computing Department, Lancaster University, Lancaster LA1 4YR
{is, sawyer, viller}@comp.lancs.ac.uk

Abstract

This paper discusses why inconsistencies in software processes are inevitable and presents an approach to process analysis which helps discover and manage these inconsistencies. By analogy with viewpoints for requirements engineering which allow multiple perspectives on a software product specification to be managed, we have developed the notion of process viewpoints which provide multi-perspective descriptions of software processes. A process viewpoint includes a statement of focus or 'world-view', a set of sources of process information, a process description and a set of organisational concerns which represent goals or constraints on the process analysis.

We present a description and rationale of process viewpoints, discuss the process of applying process viewpoints for process understanding and illustrate the overall approach using an industrial example drawn from processes which are part of a safety-critical system development.

Introduction

Over the past few years, there has been a growing awareness that organisations can leverage significant productivity improvements by improving their business processes. Software development processes are one class of business process which have received particular attention and, since the early 1980s, a significant body of research into software processes and software process improvement has developed.

Very roughly, this research can be split into two main categories:

1. Fundamental research concerned with process modelling, enactment and support technology. This involves developing notations for representing processes and looking at how some automated support for these processes can be provided. Good summaries of research in this area are provided by [1] and [2].

2. Research concerned with process improvement, that is, the introduction of process changes to improve software productivity, quality, etc. This is related to general work on business process re-engineering [3] although software process improvement programmes tend to be evolutionary rather than revolutionary. The best known work in this area is that of the Software Engineering Institute and their Capability Maturity Model [4, 5] but other related work on maturity models for process improvement has also been carried out [6] [7].

To a large extent, these different aspects of process R&D have been parallel research streams. The fundamental process research has not really addressed the issues of how the research results can be applied to facilitate change and improvement. The process improvement work has taken a pragmatic approach to process description and is not dependent on structured or formal process notations. Proponents of this approach suggest that automation is not central to process improvements. Indeed, the SEI model discourages automation in the early stages of improvement. Its developers maintain that process enactment support can only be cost-effective once a disciplined and repeatable process has been put in place.

The work which we describe in this paper falls somewhere in between these areas of research. It was developed as part of a pragmatic process improvement project which was specifically aimed at discovering routes to requirements engineering process improvement [8]. However, to facilitate process elicitation for analysis and improvement, we decided that we needed a structured framework managing process descriptions which could cope with different types of process description and which, furthermore, reflected the organisational goals for process improvement.

Our studies of requirements engineering processes (and, previously, other types of software process [9]) revealed remarkable process inconsistency both within and across organisations. Inconsistencies in processes do not necessarily mean that there are process problems. They can arise for a number of good reasons:

1. Many tasks require software engineers to exercise professional judgement and experience. Different people may do the same thing in different ways, all of which may be effective.

2. Different processes with the same objectives may have developed in different parts of the organisation as a consequence of local factors such as tool and equipment availability, management style etc.

3. Depending on their particular interests, different people may focus on different aspects of a process. For example, a quality manager may be mostly concerned with process activities which affect product quality whereas a project manager will be concerned with

the scheduling and resources used by activities. They may selectively ignore parts of the process which are not of interest. This can result in inconsistent models of what the process actually is.

4. Processes vary depending on the type of system being developed and the participants in the process. This is particularly true of processes, such as requirements engineering processes, which are not solely technical but which involve significant interaction with customers and others outside the software development team.

The primary goal of our work was to facilitate process understanding and improvement. We needed a way of describing processes which would allow us to collect different perceptions of these process and reconcile the differences between them. We wanted to understand different process perspectives with a view to discovering the best practice and to identify areas where improvements seemed to be viable. Process inconsistencies were particularly important for two reasons:

- Areas of inconsistency may suggest process redundancy where activities are unnecessary. Fully or partially removing this redundancy may lead to process improvements.

- Areas of inconsistency can highlight particularly good practice. Where different processes for accomplishing some goal have evolved, the most effective of these processes may be selected for widespread dissemination across the organisation.

Most research on process modelling notations [10] has not really taken potential process inconsistency into account. Most of this assumes that a single model of the process can be elicited and agreed by all process participants. However, our experience suggested that this was very difficult to achieve for many software processes:

- The processes as seen by different process participants and stakeholders were often so inconsistent that no definitive model could have been produced. Where process descriptions from different participants could be integrated, the combined description was complex and virtually impossible to understand or validate.

- There was no single notation which all process participants and stakeholders were familiar and comfortable with. People wanted to describe 'their' process in their own way and we felt that imposing some, perhaps formal, notation on this would have been counter-productive. We believe that there is a lot to be said for Checklands 'rich pictures' [11] where processes are described using representations which are meaningful to the people involved in the process.

Consequently, we decided not to look for a single process representation but to allow the process to be represented in different ways reflecting different process perceptions. The process representation framework had to cope with inconsistency and support process improvement. Our previous experience of viewpoints for requirements engineering [12, 13] suggested that viewpoints might be an appropriate framework for encapsulating partial process knowledge. We decided to adapt the approach to requirements viewpoints which we had developed [14] for process representation.

Other researchers have come to comparable conclusions and have experimented with a multi-perspective approach to process modelling. The earliest work we are aware of was work carried out by Rombach in the late 1980s in development of MVP-L [15], a process modelling language which supported multiple views. This work has continued with a newer version of the language MVP-L 2 now available [16] . Verlage [17] has confirmed the needed for a multiple perspective approach to process modelling and presents a set of requirements for this approach. Turgeon and Madhavji [18] have also developed a multi-view approach to

elicitation with some automated consistency checking. This is based on their work on process elicitation and the methodological and tool support which has developed for this [[19] [20]. Previous work on managing process inconsistencies across viewpoints has mostly derived from comparable work in viewpoint-oriented software engineering. This relies on automatic or semi-automatic analysis of formal process descriptions to discover inconsistencies across these descriptions [21]. We have not followed this path because we found that practitioners were not interested in learning formal process representation languages. Furthermore, the complexity and subtlety of real processes made it almost impossible to produce formal process descriptions which were understandable by the people involved in the process. Rather, we have focused on developing a framework and an associated process which facilitates elicitation and process analysis by people rather than computers. This framework does not really on any formal representation and analysis but rather is a way of structuring process information to facilitate analysis and process improvement.

In the remainder of this paper, we present the process viewpoint model which we have developed, discuss the process of applying viewpoints for process analysis and present a small example of using the approach for the analysis of process descriptions in a safety-critical system.
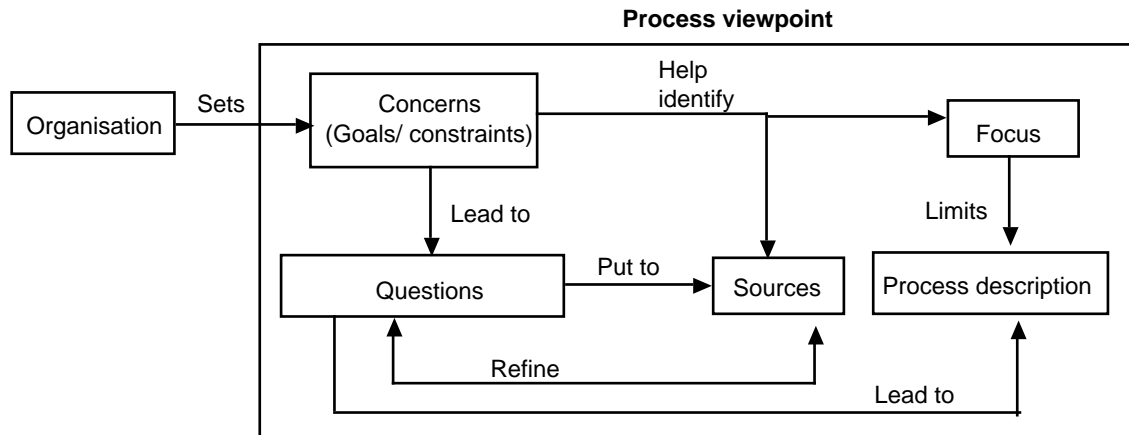
Process view points

The PREview approach to requirements engineering which we have developed [14] was explicitly designed for application in a range of industrial [22]. This viewpoint-oriented approach allows for different types of viewpoint (end-user, stakeholder, domain) to be accommodated within a single generic framework and provides a mechanism whereby business goals and constraints drive the requirements engineering process.

Given the need for process improvement to be driven by business goals, we attempted initially to use an almost identical viewpoint model for process analysis [23]. We found that this was not entirely appropriate so we derived a simpler model of a process viewpoints with 5 components as follows:

$$PV = \{name, focus, sources, process\ description, concerns\}$$

We discuss these components in more detail below but briefly the *name* identifies the viewpoint, the *focus* defines the particular viewpoint perspective, the *sources* document the source of process information, the *process description* is a description of the process from the perspective defined in the viewpoint focus and the *concerns* reflect business goals and constraints.

Figure 1 shows the relationships between these components of a process viewpoint. The shaded boxes are components of a viewpoint.

Process viewpoint

Figure 1 Process viewpoint relationships

The business concerns help identify what the focus of a process viewpoint should be and this restricts the process description. Questions which are derived from concerns are critical to the process of eliciting process information from sources and discovering process inconsistencies. Answers to these questions, which are put to sources, lead to the formulation of a process description.

A very simple example of a process viewpoint is shown in Figure 2. This is a viewpoint on a requirements review process. For simplicity, we have not actually included the process description here.

Name	Quality management
Focus	The requirements review process and how overall system quality may be influenced by that process. Product defects are being introduced as a result of requirements errors. Product development schedules are longer than they should be because of the need to detect and remove these errors.
Sources	project managers, quality managers, company standards,
Concerns	Time to market, product defects
Process description	*A description of the review process including inputs, outputs, activities, process participants and commentary on the process and its influence on system quality*

Figure 2 A process viewpoint

Now let us look at the different components of a process viewpoint in more detail and discuss why these have been included.

Viewpoint name

The name of the process viewpoint identifies it and gives the reader an indication of the likely perspective of the viewpoint. It may therefore be:

• The name of a role or department in an organisation such as 'Configuration management', 'Quality assurance', 'Customer', etc. This implies that the process description will focus on the process activities, inputs and outputs which are most important to that department or role.

- The name of a process characteristic which is of particular interest. This can either be a functional characteristic such as 'Process activities', 'Roles and actions' or can be a non-functional process attribute such as 'Repeatability', 'Performance', etc. A particular type of modelling notation such as a data-flow diagram or a Petri net may be particularly appropriate for describing the process.

Viewpoint focus

A viewpoint's focus is a statement of the perspective adopted by that viewpoint. This should normally include a statement of the parts of the overall software process i.e. the sub-processes with which the viewpoint is concerned, It may also include a statement of the organisational functions which are of most concern in the analysis of a process, a statement of the role of viewpoint sources or a statement of the particular type of model which will be presented.
Examples of focus descriptions might therefore be:

"Configuration management activities in the requirements engineering process"

"A system architect's view of the requirements engineering process"

"An entity-relational model of the documents used in the requirements engineering process"

We have found that explicitly defining the focus of a viewpoint is valuable for three reasons:

1. It helps to identify sources of process information.

2. It can be used in the development of organisational concerns (see below).

3. It can be used to help discover overlapping viewpoints (where conflicts are most likely) and gaps in the viewpoint coverage of the process.

The viewpoint focus may also have an associated rationale which is comparable to the notion of *Weltanschauung* or 'world view' in Software Systems Methodology [11, 24]. This rationale presents assumptions on which the viewpoint is based and helps the reader understand why the viewpoint has been included. Examples of rationale which could be associated with the above focus descriptions are:

"Our current configuration management process is not integrated with our requirements engineering process"

"System architects are normally consulted after the requirements have been defined and this can cause serious design problems"

"We need a formal description of the process entities to support improved configuration management"

Viewpoints need not be completely separate but may have overlapping foci. However, where there is a significant overlap, we recommend that the different viewpoints should be integrated into a single viewpoint.

Viewpoint sources

Viewpoint sources are an explicit record of where the information about the process has been collected. The most important sources of process information are usually:

1. The participants in the process

2. Management in the organisation where the process is being enacted.

3. Organisational process charts, responsibility charts, quality manuals, etc.

The list of sources connected with a viewpoint is useful because it provides an explicit trace to where the process information was derived. This allows the original sources to be consulted for possible problems when process improvements and process changes are proposed. Source information may be represented as names, associated roles and contact information if the sources are people, document identifiers and page references, WWW URLs, etc.

Process description

We do not mandate any particular notation for process description. Our experience showed that most people preferred informal process descriptions made up of informal diagrams and explanatory text. While these are more subject to misinterpretation that formal descriptions, we believe that this is more than compensated for by their understandability and flexibility in describing processes where exceptions are common. Of course, for some viewpoints which are concerned with particular types of process model, such as an entity-relational model, an appropriate formal or structured notation may be used.

Because of individual differences in process enactment, there may be alternative perceptions of a process presented by different sources in the same viewpoint. This is particularly likely where one of the sources is process documentation which defines the organisational perception of a process (or what a process ought to be) and another source is a process participant who can explain what really happens. If these differ very radically, they should really be separate viewpoints but where the differences are in the detail of the enactment, they can be accommodated within a single process description.

We could support this through a viewpoint inheritance mechanism. However, this leads to a viewpoint explosion and it is difficult to manage the large number of viewpoints which are then created. Rather, the differences can be accommodated by including a stable part and a variable part in the process description:

1. The stable part of a process description is the part of the description which is shared and accepted by all of the sources contributing to the process viewpoint.

2. The variable part of the process description highlights those parts of the process which exhibit variability and documents the different ways in which this variability occurs. In many cases, the variability manifests itself in the exception handling - different people cope with problems in different ways.

Describing processes using stable and variable parts is one way of tolerating inconsistency in process descriptions. As we discuss in the following section which describes the process of acquiring process descriptions, we try to reconcile inconsistencies as soon as they emerge but, if this is impossible, we simply leave them in the description. The inconsistency analysis which we also discuss later is then applied within the viewpoint as well as across process viewpoints.

The process description may be a hierarchical description with the process described at different levels of abstraction. At the top level, we recommend that the process description should fit onto a single page so that it may be understood as a whole. All or some parts of the process may then be described in more detail as necessary.

Concerns and concern decomposition

Process improvement should always be driven by the needs of the organisation enacting the process. To allow for this, we associate concerns with viewpoints where concerns reflect organisational goals, needs, priorities, constraints, etc. Concerns help to scope the analysis of current processes and proposed process improvements.

There are different types of concern which may be associated with process viewpoints:

1. *Understanding concerns.* These reflect the organisation's objectives for process understanding. The organisation may wish to understand a process to discover its relationships with other organisational processes, to define the process in a quality plan, to analyse the process for improvements, etc. Where an organisation has immature processes, understanding these processes is the first step towards process definition and, ultimately, improvement.

2. *Improvement concerns.* These reflect the objectives of the organisation as far as process improvement is concerned. At a very abstract level, these may be reduced time to process completion, reduced process costs, etc. However, as we discuss below, these have to be decomposed into realistically achievable goals.

3. *Constraint concerns* These are organisational constraints placed on the process or on the process improvement activity. They may limit the analysis or possible process improvements.

Concerns are not just another type of viewpoint. A viewpoint is an encapsulated partial process description; a concern relates the process description to the business needs of the organisation enacting the process. The concerns drive the process analysis so that proposed process changes and improvements contribute to the real needs of the business.
In practice, concerns are decomposed into a set of questions which are put to process sources. Therefore, if an understanding concern is *process definition*, then this may be translated into an abstract question:

> "What are our requirements engineering process activities"

If an improvement concern is process cost reduction, the most abstract question becomes:

> "How can the costs of the process be reduced"

However, these questions are so general and abstract that they are not particularly useful when analysing processes. Therefore, concerns are decomposed into sub-concerns and ultimately, into a specific question list which may be put to viewpoint sources.
As an illustration of this, Figure 3 shows how the above improvement concern might be decomposed. Notice that improvement concerns will almost always decompose to a mixture of understanding sub-concerns (what are we doing now?) and improvement sub-concerns (how can we do it better?). It is artificial to try to separate these as knowledge which can contribute to improvement emerges naturally as understanding of the process is developed.
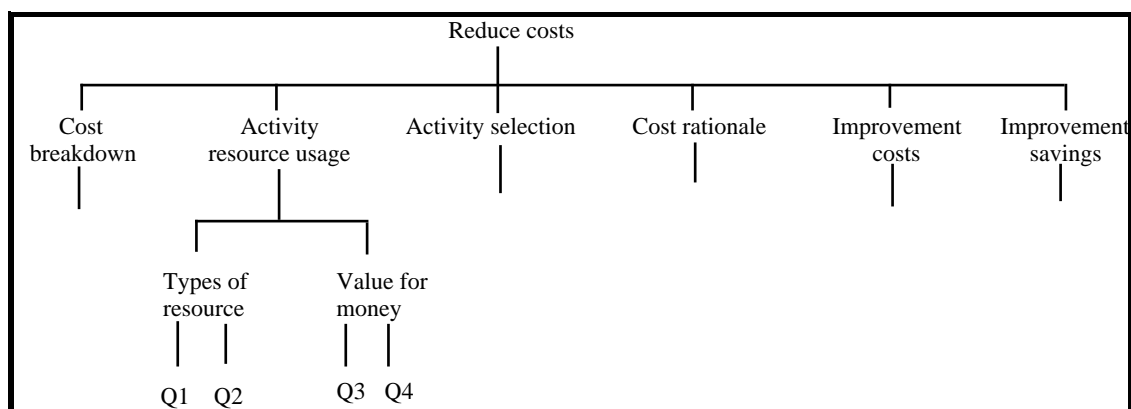
Figure 3 Decomposition of a cost reduction concern

Specific questions may be associated with nodes in this decomposition. In this example, we may have:

Q1.   What are the types of resource used in each activity

Q2.   How much of each resource is used in each activity?

Q3.   Does the output from the activity justify the resource input?

Q4.   Are there comparable activities which use disproportionate amounts of some resource?

 The formulation of questions helps identify the level of detail which should be included in a process description. The description must be constructed so that these questions can be answered. Therefore, for the above set of questions, the process description should identify the activities for which resource utilisation information is available. If this information has not been collected, process improvements cannot be evaluated. It implies that the project management process should be improved to collect more detailed process data.
As well as understanding and improvement concerns, organisations will also place constraints on both the process improvement process and the possible improvement proposals. These constraints may also be expressed as concerns and decomposed into questions which must be addressed during the stages of process analysis and improvement suggestion. Examples of concerns which are constraints might be:

1. *Budget* The budget available to the process improvement team is $XXXX.

2. *Existing tools and standards* Existing design notations such as SADT must be used to describe the system requirements.

3. *Training* Proposed improvements should require no more than Y days of additional training time per team member (or alternately Z days across the whole team).

These concerns are process requirements and process improvements which are proposed should be validated against these global requirements.
We know from experience of goal-oriented process improvement methods such as the ami method [25] that the decompositions of concerns (or goals) and the formulation of questions is not a simple process. In some cases, it may be easier to tackle this bottom-up rather than top-down. That is, a standard list of questions derived from other analyses may give clues to possible concerns and how these concerns may be decomposed. Once formulated, concerns and questions should therefore be made available to all subsequent process analyses.

Using process viewpoints in process analysis

The process viewpoint model which we have described is intended to help elicit and analyse information about processes with a view to subsequent process improvement. To support the use of the model, we have developed a process for process modelling and improvement which is shown in outline in Figure 4.
In the notation which we use here and in later figures describing processes, process activities are denoted in boxes. Dashed arrows linking boxes mean that there is a temporal relationship between these activities. It is not possible for an activity at the destination of an arrow to be completed until the activity at the source of the arrow has completed. However, the destination activity may start before the source activity has completed and the activities may be interleaved or may run concurrently. In Figure 4, the arrow linking the last box with the first box means that the process is cyclic and can be re-entered after a set of improvements have been proposed.
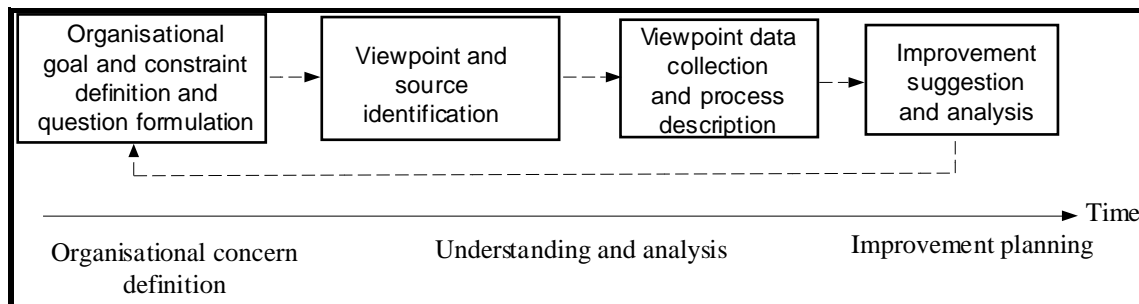
Figure 4 Process improvement with viewpoints

The overall process has 4 main phases:

1. *Concern definition* During this phase, the main business goals and constraints are identified. The people involved are the process improvement team, senior managers in the organisation and project managers. Concerns are decomposed into a set of questions as discussed in Section 3.5. A problem at this stage is ensuring that the concerns are realistic. Where organisations have immature processes, it is often only possible to set goals which will bring these processes under control. Process improvement goals such as reduced costs or elapsed time may not be possible.

   During the definition of concerns, it is usually necessary to develop a simple process description so that a basis for discussion is established. This may either come from existing documentation or from any process stakeholder.

2. *Viewpoint and source identification* Possible viewpoints and associated sources of process information are identified. We discuss this stage in more detail below.

3. *Data collection and process description* Information about the process is collected and the process is documented. Again, we discuss this in more detail below.

4. *Improvement suggestion and analysis* The processes as described in each viewpoint are compared and overlaps, inconsistencies and conflicts are identified in a review involving process participants and the process improvement team. Process inconsistencies and redundancies are the focus for improvement and may point to potential process modifications to select best practice or to reconcile these inconsistencies. All improvements are analysed against the identified concerns to ensure that they are consistent with business goals.

As the theme of this paper is process viewpoints and the support they provide for managing inconsistency, we will ignore the first and last stages in the process shown in Figure 4 and will concentrate on the middle two stages where viewpoints are applied.
Figure 5 is a more detailed description of these stages. In this figure, solid arrows between activities indicate data flow.
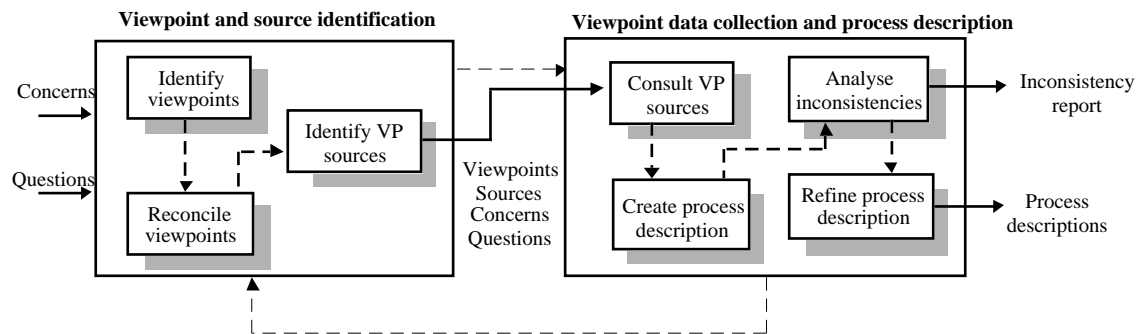
Figure 5 Viewpoint identification and process elicitation

The whole process is iterative and we recommend that it should begin as soon as some viewpoints have been identified. Once a viewpoint has been identified, some information about the process can be collected and this may then be used to help with the identification of further viewpoints.

Viewpoint and source identification

This stage of the process is concerned with identifying relevant viewpoints and the information sources associated with these viewpoints. Viewpoints and their sources are identified in an iterative way. In practice, these activities intermingle so that, a complete list of viewpoints and sources is not available until the end of the activity.

The inputs to this activity are concerns and associated questions. These questions may either elicit process details, discover rationale for process activities, discover information about the timing, duration and inter-dependencies of activities or the support for the process which is available.

The sub-activities involved in this stage are:

- *Identify viewpoints* This activity is concerned with identifying the most appropriate process perspectives which can contribute useful information about the process and representing these as viewpoints. As a starting point, viewpoints covering organisational standards, process participants, management and, where appropriate customers should be identified. There should be no restrictions on the numbers of viewpoints identified at this stage.

- *Reconcile viewpoints* To make the analysis practical, it is important not to have too many viewpoints. We have found that more than 5 viewpoints inevitably cause information management problems. In this activity, the focus descriptions of the different viewpoints are used to identify overlaps and areas where viewpoints may be merged. Where the focus descriptions are too broad and encompass extensive processes (e.g. all of software testing), we recommend that the scope of the viewpoint should be reduced and the focus description should be rewritten to be less ambitious.

- *Identify viewpoint sources* Viewpoint sources are information sources which can adopt the viewpoint focus. They may be people, documents, standards, domain knowledge, etc. These are usually identified by consultation with managers and engineers involved in the process.

The process of viewpoint identification may bring concerns to light which have not already been considered. Therefore, it may be necessary to revisit the previous stage to refine these concerns before moving on to elicit information about the process description.

Viewpoint data collection and process description

This stage of the process improvement process is concerned with understanding, analysing and describing the current process which is used. The general approach which we recommend is an incremental application of the steps described below for each identified viewpoint, refining the questions and descriptions as process information from viewpoint sources is elicited. That is, an initial set of questions to elicit process information is derived, viewpoint sources are consulted and a process description is proposed. The analyst then refines the questions and repeats this consultation and refinement process until all viewpoints have been covered.
The stages in this process are:

- *Consult process viewpoint sources* The analyst puts the questions derived from concerns to the viewpoint sources to discover process information. These questions may need to be refined for the specific viewpoint (e.g. by changing the terms used) depending on the background of the source. As well as the questions, of course, sources should be asked to comment on their processes. We have found that the best way to elicit information is to ask them to critique an existing process model which may be derived when concerns are established. The process description focuses the elicitation as stakeholders can point out where it is incomplete and differs from their actual process. Once more detailed process description has been elicited from one viewpoint, it may then serve as a basis for discussion about the process in other viewpoints.

- *Create process descriptions* Create and document a process description, taking into account the differences as seen by different viewpoint sources. Any notation may be used here but we have found that simple block diagrams supplemented by tables and natural language text are readily understood.

- *Analyse inconsistencies* This activity is concerned with analysing the process descriptions, to discover redundancy and inconsistencies. We discuss this in more detail in Section 4.3.

- *Refine process descriptions* The results of the inconsistency analysis are fed back to the process sources and, where appropriate, the process descriptions are modified. For example, where different people use different names for the same process, a single term may be agreed. Where inconsistencies cannot be reconciled, they are documented in an inconsistency report which is an input to the next phase of the process concerned with process improvements.

This activity is also likely to reveal problems with the identified concerns and questions. Some iteration may be required to redefine the concerns and the associated questions.

Inconsistency analysis

The process of inconsistency analysis is intended to discover and classify inconsistencies in processes. The processes are reviewed by a team including process participants and members of the organisational process improvement group and inconsistencies are highlighted. However, no decisions about process modifications are made at this stage. This happens in a subsequent process where the concerns are used to decide which process changes are the most effective way of contributing to the business goals.
This has been shown as a separate process stage in Figure 5 but, in fact, much of the work actually takes place during the elicitation of process information. Process elicitation for the

different viewpoints is a sequential process. Once a process description is available from one viewpoint, it may be used as an input to the next elicitation activity. During that activity, process stakeholders identify inconsistencies by pointing out how their view of the process differs from the view which is presented to them. In some cases, where inconsistencies are a result of misunderstandings (for example, where different terminology is used) it may be possible for the people involved to see immediately how to resolve the problem and the process descriptions are changed during elicitation to remove the inconsistency. In other cases, however, the inconsistency reflects a genuine difference and it is documented for subsequent analysis.

When process descriptions have been documented, the inconsistency analysis process can get underway. There are two fundamental stages in this process:

1. *Pre-review analysis* This is undertaken by a member of the process improvement team. He or she looks at the process descriptions to identify areas of similarity and difference. Inconsistencies which were identified during elicitation are an input to this. Terminology is always a problem and, ideally, a process glossary should be created. In practice, however, there may not be time to do this. The result of this activity is an agenda for the process review which lists the inconsistencies and the process fragments which must be discussed.

2. *Process review meeting* This is a review meeting which is comparable to a program inspection meeting where a process or process fragment is presented to the meeting and discussed by the meeting participants. Several different views of the same process may be presented if necessary. Each of the inconsistencies is considered in turn and classified as discussed below. The outcome of the meeting is an inconsistency report which is passed on to the next stage of the process improvement process.

Inconsistencies are classified as shown in Figure 6.

Inconsistency type	Explanation
Irrelevant	There is an inconsistency in processes as seen from different viewpoints but this has no practical effect on process efficiency. An example of this might be where different engineers use judgement to decide how to carry out a particular process and it doesn't matter which approach is used.
Necessary	The inconsistency in processes must be maintained because it has arisen as a consequence of some external factors which are outside the influence of the process improvement team. For example, teams in the same organisation working in different countries may not be able to resolve some process inconsistencies because they arise as a consequence of local laws
Reconcilable	These are process inconsistencies which appear to be reconcilable and where some standardised process can be developed. These often reflect process redundancy or omissions. Removal of redundancy may lead to process improvement. An example of this type of inconsistency is where the same process specified in different viewpoints seem to require different inputs. If the process operates efficiently, it may be possible to reduce the number of inputs to a minimum.
Improveable	These are inconsistencies where different sub-processes are used and where some of these sub-processes are clearly better than others. Process improvement may be possible by identifying the best practice and adopting this across all viewpoints.

Figure 6 Classes of process inconsistency.

A simple case study

In this section, we present part of a process analysis case study based on a real industrial process which illustrates the use of process viewpoints. In this case, our objective was to discover process inconsistencies and report them to the organisation who 'owned' the process. It was then up to them to consider what improvements might be possible.
The process concerned was the development process for a safety-related expert system which was being developed for a large client. The system involved the development of a 'safe' expert system shell and the instantiation of this shell with specific domain data. The system was to be formally specified and the implementation validated by correctness arguments (not a complete proof) against the specification. The organisation developing the system is a specialist in critical computing systems, is technically mature and is strongly committed to quality and quality improvement.

Concerns and questions

The first stage in the process analysis is to identify the organisational concerns which contribute to the process analysis. As this is a safety-related system, the most important of these concerns was *safety.* A critical business goal of the organisation was to ensure that the systems it developed were safe (and were known to be safe) and processes always had to take this into account. A further business concern in this particular case was *customer relationships*. While this should, perhaps, always be a concern, the customer in this case was a technically mature, large organisation and a likely source of future contracts. It was particularly important to ensure that good relationships were maintained throughout the project. The final concern identified was *skill utilisation*. The organisation developing the system is very highly skilled in a number of areas and processes had to be designed to utilise these skills.
To illustrate question derivation, let us consider the *customer relationships* concern. Some relevant process questions associated with this concern are:

1. What process and product documentation does the customer require?

2. What are the most important concerns (schedule, cost, standards compliance, quality, etc.) of the customer?

3. Will the customer participate in project management meetings?

4. Will a customer engineer be involved in reviews? If so, when will they need documents to be delivered?

5. Does the customer mandate the use of specific standards, tools or techniques?

Notice that these questions are not concerned with broad aspects of the process such as activities, inputs and outputs. Typically, questions derived from concerns ferret out significant process details which could be easily be missed by analysts. By applying these questions during process elicitation, we can discover if any aspects of the process are inconsistent with the business goal of customer satisfaction.

Viewpoint identification

The next stage of the process of analysis is to identify relevant viewpoints. The process we recommend involves identifying as many viewpoints as possible then reconciling them to fewer than five viewpoints for analysis. In this case, we will skip the first stage and simply discuss the final viewpoints used in the analysis. Three viewpoints were identified as significant in this case:

- A *project management viewpoint*
- A *quality management viewpoint*
- An *engineering viewpoint*

The scope of each of these viewpoints is defined by describing their focus. All viewpoints are concerned with the whole of the development process for the system so this need not be explicitly set out in the viewpoint focus descriptions:

1. *Project management* The project management viewpoint is concerned with the process as defined by those activities which are identified in the project plan and which are assigned specific resources and a schedule for completion.

2. *Quality management* The quality management viewpoint is concerned with those aspects of the process where the customer requires explicit evidence of validation activities and the conformance of process deliverables to requirements and standards.

3. *Engineering* The engineering viewpoint is concerned with the process which is actually enacted by the engineers involved in the system development.

In conjunction with the process of viewpoint identification, sources of information associated with the viewpoint should also be identified. The focus description may be used as a starting point for identifying information sources. In this case, once one source was identified, he helped in the identification of other possible sources.
For brevity, we will focus in the remainder of this discussion on the first two viewpoints above namely *quality management* and *project management*. Possible sources of information associated with these viewpoints are:

- *project management* - project plan, project manager, customer project manager, software developers

- *quality management* - project quality plan, organisational quality manual, quality manager, customer project manager

Of course, it is not always possible to get access to all sources. In this particular case, we had close links with the development organisation but no access to their customer. We were therefore unable to get any information from customer-based sources.

Process description

Once the sources have been identified, they are consulted and a process description is developed. The principal source for the project management and quality management viewpoint was the very comprehensive process documentation which had been produced to satisfy the quality requirements of the customer.
From the project management viewpoint, resources and schedules had been drawn up for 21 activities; from the quality management viewpoint, 36 explicit validation activities were identified. In each of these cases, there were logical groupings of activities so it was relatively straightforward to produce a more abstract model of the process.
This high-level project management model is shown in Figure 6. Again, the dotted arrows mean temporal sequence where a destination activity may start but may not finish before the source activity has finished. Where activities are vertically aligned, this means that they may (but need not) be carried out in parallel.
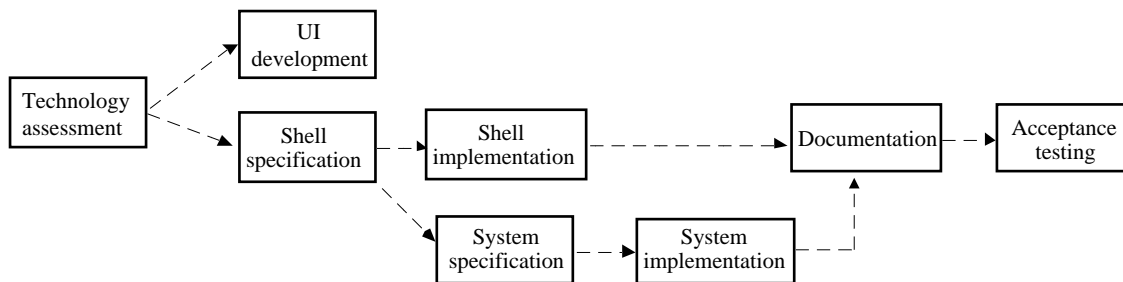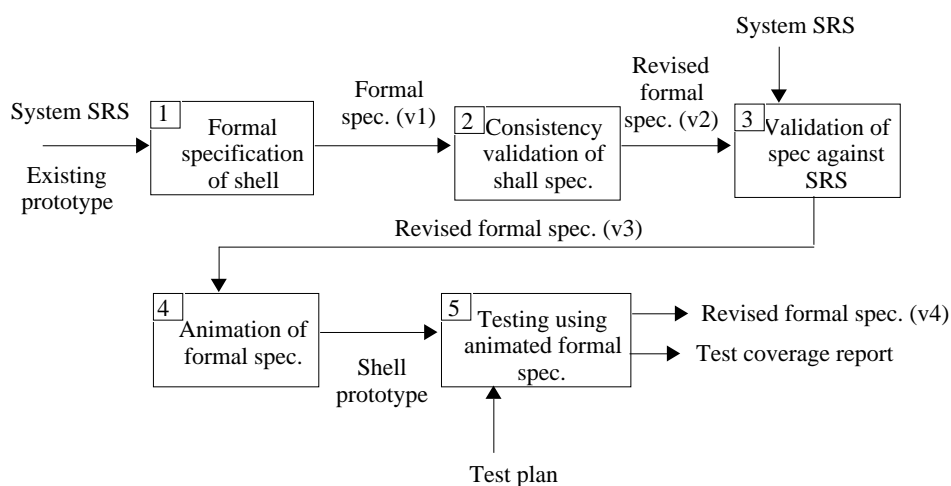
Figure 6 Project management model

From the quality management perspective, the high-level model almost matched this project management model. There were validation activities for each of the development activities in Figure 6. This model is illustrated in Figure 7.
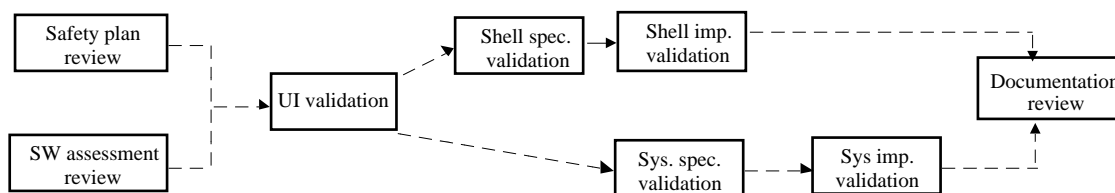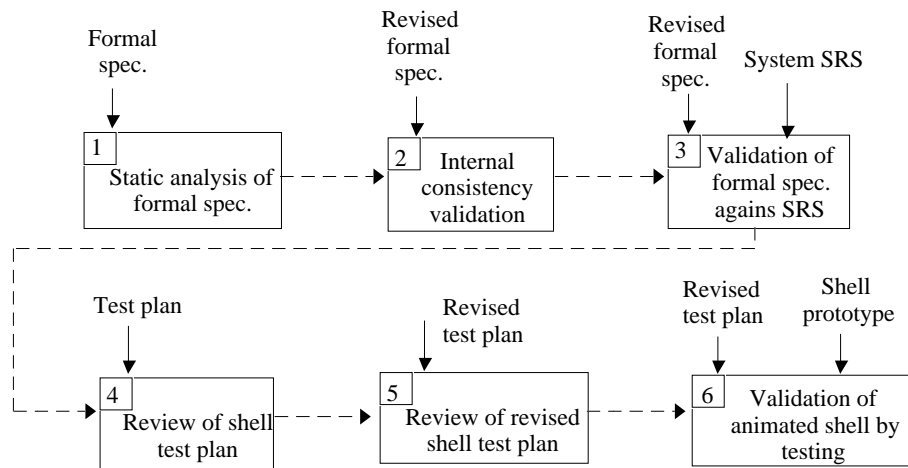

Figure 7 Quality management model

As you would expect from an organisation developing safety-related systems, at this level there are no significant process inconsistencies. However, we can see on the left of Figure 7 that there is a required review of the project safety plan with no corresponding activity in the project management model. This is almost certainly an accidental omission and consistency can be achieved by adding an additional activity to the project plan.
Further inconsistency analysis requires a more detailed look at the process. Let us look at the processes, inputs and outputs for the shell specification activity shown in Figure 6. This more detailed model is shown in Figure 8. Solid arrows linking boxes indicate data-flows. Note that the System SRS is a software requirements statement, written in English, which was produced by the customer for the system.

Figure 8 Project management model of shell specification

The comparable quality management model which identifies validation activities, inputs and outputs is shown in Figure 9. In this process, each validation activity produces a report and has a set of associated success criteria. These are not really inputs to the next stage (the inputs to validation are outputs from development activities) so are not shown as data-flows on this diagram.

Formal
spec.

Revised
formal
spec.

Revised
formal
spec.   System SRS

| 1 Static analysis of formal spec. | 2 Internal consistency validation | 3 Validation of formal spec. agains SRS |

Test plan

Revised
test plan

Revised
test plan   Shell
prototype

| 4 Review of shell test plan | 5 Review of revised shell test plan | 6 Validation of animated shell by testing |

Figure 9 Quality management view of shell specification validation

Inconsistency analysis

In analysing these more detailed models for inconsistency, we put the models side-by-side and ask a number of questions:

1. What is the correspondence between tasks for which resources have been allocated and identified validation activities? Are there any mismatches i.e. development activities with no corresponding validation activity or validation activities with no allocated resources?

2. Are all of the required inputs for validation specified as outputs in the project manager viewpoint?

3. Is the information provided complete i.e. is it clear which results from the development activities are inputs to the validation activity?

When we examine the process descriptions in the project management and process management viewpoint and put these questions to the models, we found inconsistencies in the process. We could not, in fact, classify these according to the inconsistency classification because our remit was process analysis and not process improvement in this case. All we could do was to highlight them in an inconsistency report. It is up to the organisation which 'owns' the process to decide if they are significant and what, if anything, to do about them. When we put the process fragments shown in Figures 8 and 9 together and examine them. we cam see that three validation activities have resources allocated to them and (somehow) these must map onto the six activities identified in the quality plan. When we looked at the processes in detail using the above questions, we found three possible inconsistencies:

1. The quality management view requires a static analysis of the formal specification (Activity 1) which is not part of the project management view. This could be accommodated in either activity 1 or activity 2 in the project management model or may be an omission.

It is not clear whether or not it is essential that the specification should 'pass' the static analysis with no significant errors or how the results of the static analysis are used in the internal consistency check of the specification. Is the static analysis report delivered to the customer or not? Resources for this static analysis do not appear to be explicitly allocated in the project plan.

2. There is no activity in the project management model where test planning is explicit and the test plan is not specified as an output of any activity. It is, however, an input to validation activities 4 and 5 which review the plan. As there are two test plan reviews, it is clear that, from a quality management viewpoint, significant effort should be devoted to test planning. Where will the resources for this come from?

3. The quality management viewpoint identifies inputs as 'Revised formal spec.' but it is not clear which version of these inputs should be used. That is, the project management model assumes a number of revisions of this specification (identified in the process as V1, V2, V3 and V4) but the quality management viewpoint does not explicitly state which of these are inputs to which validation activity.

The process of process analysis is continued for the other process fragments associated with the activities identified in Figures 6 and 7. We will not show this here as it simply involves applying the same process as we have described for the shell specification.

Overall, we identified a total of 14 inconsistencies between the processes as seen from the project management and quality management perspective. The majority of these seemed to be omissions and should be fairly easy to resolve. As we were not looking at different teams doing the same thing, we obviously did not find any examples of 'bast practice' which could be disseminated across the organisation.

Conclusions

This paper has described an approach to process analysis and improvement based on viewpoints where each viewpoint manages partial process knowledge. It allows inconsistent models of processes to be managed and provides a framework for analysing the inconsistencies with a view to subsequent process improvement.

The approach was developed as part of an industry-university collaboration and was carried out in conjunction with other work concerned with human and social influences on process reliability [26]. All the work has been strongly influenced by practical industrial requirements. We do not require the use of specialised notations for process description and, a novel feature of the approach is the use of concerns to relate process improvement and analysis to overall business goals. The process viewpoints approach has been applied to the case study described here and to some aspects of the requirements engineering process used by another of our industrial collaborators in the project. These revealed process inconsistencies and highlighted distinctions between the assumptions made by different process participants.

The case study described here demonstrated that it was realistic to apply our approach to real industrial processes although we must admit that this was a relatively small project with a limited number of stakeholders. We learned that it was important to use only a small number of viewpoints so that the elicitation process did not become too expensive and that concerns were a very valuable way of ensuring that process information relevant to business goals was not ignored. We do not yet know how the approach will scale up to the processes used in large software engineering projects.

Our current work with the process viewpoints approach is not, in fact, concerned with software processes but with more general business processes in the financial sector. We are

investigating how this approach to process analysis can be used in conjunction with structured requirements engineering techniques to help understand an organisation's requirements for business process support.

Acknowledgements

References

[1]   Finkelstein, A., Kramer, J., and Nuseibeh, B., ed. *Software Process Modelling and Technology*. Advanced Software Development Series, ed.^eds. J. Kramer. 1994, Research Studies Press Ltd.: Taunton, Somerset, England.

[2]   Fuggetta, A. and Wolf, A., ed. *Trends in Software Process*. 1996, Wiley: Chichester.

[3]   Hammer, M., "Reengineering Work: Don't Automate, Obliterate". *Harvard Business Review*, 1990. **July-August 1990**: 104-112.

[4]   Humphrey, W.S., "Characterizing the Software Process". *IEEE Software*, 1988. **5**(2): 73-79.

[5]   Paulk, M.C., Curtis, B., Chrissis, M.B., and Weber, C.V., "Capability Maturity Model, Version 1.1". *IEEE Software*, 1993. **10**(4): 18-27.

[6]   Haase, V., Messnarz, R., Koch, G., Kugler, H.J., and Decrinis, P., "Bootstrap: Fine Tuning Process Assessment". *IEEE Software*, 1994. **11**(4): 25-35.

[7]   Konrad, M. and Paulk, M. "An Overview of SPICE's Model for Process Management". in *Proc. 5th Int. Conf. on Software Quality*. 1995. Austin, Texas.

[8]   Sommerville, I. and Sawyer, P., *Requirements Engineering:A Good Practice Guide*. 1997, Chichester: John Wiley & Sons.

[9]   Sommerville, I. and Rodden, T., "Human, Social and Organisational Influences on the Software Process"*, in Trends in Software,* A. Fuggetta and A. Wolf, Editors. 1996, John Wiley and Sons: New York. p. 89-110.

[10]   Conradi, R. and Liu, C. "Process Modelling Languages: One or Many". in *Proc. 4th European Workshop on Software Process Technology*. 1995. Noordwijkerhout, The Netherlands.

[11]   Checkland, P., *Systems Thinking, Systems Practice*. 1981, Chichester: John Wiley & Sons.

[12]   Nuseibeh, B., Kramer, J., and Finkelstein, A., "A Framework for Expressing the Relationships between Multiple Views in Requirements Specifications". *IEEE Trans. on Software Eng.*, 1994. **20**(10): 760-73.

[13]   Kotonya, G. and Sommerville, I., "Requirements engineering with viewpoints". *BCS/IEE Software Eng. J.*, 1996. **11**(1): 5-18.

[14]   Sommerville, I. and Sawyer, P., "Viewpoints: principles, problems and a practical approach to requirements engineering". *Annals of Software Engineering*, 1997. **3**.

[15]   Rombach, H.D., *MVP-L, A Language for Process Modeling In-the-Large.*1991, University of Maryland Institute for Advanced Computer Studies: Technical Report UMIACS-TR-91-96, CS-TR- 2709.

[16]   Bröckers, A., Lott, C.M., Rombach, H.D., and Verlage, M., *MVP-L language report version 2*1995, Department of

Computer Science, University of Kaiserslautern, 67653 Kaiserslautern, Germany, 1995.: Tech. Rep. 265/95,.

[17]   Verlage, M. "Multi-View Modeling of Software Processes". in *Proc. 3rd European Workshop on Software Process Technology*. 1994. Villard-de-Lans, France.

[18]   Turgeon, J. and Madhavji, N.H. "A Systematic View-based Approach to Eliciting Process Models". in *Proc. 5th Euopean Workshop on Software Process Technology*. 1996. Nancy, France.

[19]   Madhavji, N.H., Holtje, D., WonKook, H., and Bruckhaus, T. "Elicit: A Method for Eliciting Process Models". in *Proc. 3rd Int. Conf. on The Software Process*. 1994. Reston, Virginia.

[20]   Heineman, G.T., Botsford, J.E., Caldiera, G., Kaiser, G.E., Kellner, M.I., and Madhavji, N.H., "Emerging technologies that support a software process life cycle". *IBM Sys. J.*, 1994. **33**(3): 501-29.

[21]   Easterbrook, S. and Nuseibeh, B., "Using ViewPoints for inconsistency management". *BCS/IEE Software Eng. J.*, 1996. **11**(1): 31-43.

[22]   Sommerville, I., Sawyer, P., and Viller, S. "Viewpoints for requirements elicitation: a practical approach". in *Proc. Int. Conf. on Requirements Engineering*. 1998. Colorado.

[23]   Sommerville, I., Sawyer, P., Kotonya, G., and Viller, S. "Process Viewpoints". in *Proc. 5th European Workshop on Software Process Technology*. 1995. Amsterdam, The Netherlands.

[24]   Checkland, P. and Scholes, J., *Soft Systems Methodology in Action*. 1990, Chichester: John Wiley & Sons.

[25]   Pulford, K., Kuntzmann-Combelles, A., and Shirlaw, S., *A Quantitative Approach to Software Management*. 1996, Wokingham: Addison-Wesley.

[26]   Emmet, L., Bloomfield, R., Bowers, J., and Viller, S. "PERE: Evaluation and Improvement of Dependable Processes". in *Proc. Safecomp'96*. 1996. Vienna.