The Process Modelling Cookbook: Version 1.

Peter Kawalek (University of Manchester)

This document was created as part of a collaboration between the University of Manchester and British Telecommunications plc. It was issued in September 1991.

The cookbook presents a method for process modelling. It will be useful as a means of understanding, refining and managing work processes. It is intended to be especially relevant in any environment where human factors have an impact.

1.0 An Introduction To Process Modelling.

Here the intention is not to define process modelling but to explore its nature and suggest its importance.

1.1 The benefits of process modelling.

Process modelling can be a valuable activity in any environment where the processes taking place are not completely understood.

1.1.1 Benefits listed by Evans et al.

Evans et al [6] list eleven benefits that a process modelling exercise can bring;

- Greater visibility of processes.
- Structure of processes becomes evident.
- Identification of less well understood processes.
- Better understanding of important data.
- Identification of process weaknesses.
- Identification of controls.
- Increased resource usage information.
- Clarification of responsibilities.

- Tracking of data.
- Decreased training time.
- Greater people mobility.

1.1.2 Benefits listed by Rowles et al.

Rowles et al [14] note the increasing attention being given to "the quality of the processes used to create products rather than on the quality of the products themselves."[1] They account for this by listing the following benefits of process modelling;

- Process quality governs product quality.
- Process improves with experience.
- Focus on process allows better control.

1.1.3 Benefits according to Kellner.

Kellner [9] writes about the benefits to the software process that can be achieved through process modelling. He states: "Software process modelling has been advocated as a vehicle for understanding software processes, as well as improving the practice of software engineering and its resulting products. Software process modelling supports the goal of planned process improvements by providing a mechanism for:

- recording and understanding the current process, and
- evaluating, communicating, and promulgating process improvements."[2]

1.2 An end in itself.

The benefits listed above relate to issues such as understanding what happens, controlling what happens and the deployment of resources. The effectiveness of processes and the promotion of product quality through process quality emerge as central themes.

The realisation of the benefits in general is not dependent upon the provision of automated support for processes (although this may sometimes be desirable). Furthermore, not all of the benefits necessarily imply that a process modelling exercise will conclude by changing processes.

1.2.1 A source of knowledge about unstructured problem situations.

Checkland [3] identifies two kinds of problem[3] domain. There is the simple or 'structured' problem and the complex or 'unstructured' problem. The difference can be illustrated by reference to two dissimilar problems. The first is a car with a flat tyre. For this there is a simple and complete solution which is to fit a new tyre. This problem is therefore simple or structured in nature. The second is dissatisfac-

1. Rowles et al, Process Development Handbook, p9.

2. M.I. Kellner, 'Software Process Modelling: Value and Experience,' in SEI Technical Review 1989, pps 25/ 26.

3. Checkland's definition of the term 'problem' is quite sophisticated;

"A problem relating to real-world manifestations of human activity systems is a condition characterised by a sense of mismatch, which eludes precise definition, between what is perceived to be actuality and what is perceived might become actuality."

tion in Northern Ireland. This has no ready solution and therefore can be described as complex or an unstructured problem situation.

These examples are obviously very highly polarised. However, Checkland argues that most problems in the work environment are largely unstructured in nature. A ready inference of the eleven benefits listed previously is that process modelling can contribute understanding to an unstructured problem situation. It is therefore possible to identify process modelling as a source of knowledge about unstructured problem situations[1].

1.2.2 The answers and the questions.

Process modelling can therefore be described as a source of knowledge or a way of understanding what goes on in the work environment. It fulfils this description from two perspectives:

• A process model conveys important information about a process.

• A process modelling exercise raises the questions that are relevant to the achievement of knowledge about a work environment.

Therefore process modelling provides both the answers to questions about a work environment and defines the questions that should be asked.

1.2.3 Summary: An end in itself.

It is therefore argued that process modelling can be ‘an end in itself’ because it can accrue benefits even where there is no precondition relating to desired change or requirements. It is not dependent upon the automation of processes and may not always lead to the processes taking place being changed. Rather, it may satisfactorily conclude by deepening the understanding of a process, making the process more visible or clarifying responsibilities etc. Process modelling is a source of understanding about unstructured problem situations and reveals the questions that need to be addressed in the acquisition of knowledge.

1.3 A means to change.

Normally however, it is likely that a process modelling exercise will be undertaken as a means to change processes. This desire for change is unlikely to be expressed in terms as specific as a requirement for ipse type support but will more usually in terms of dissatisfaction with current processes and vague proposals for change.

1.3.1 To what ends can process modelling be put?

Process modelling can support change with a variety of motivations. As the benefits identified by Evans et al [6] indicate, it can be used to promote efficiency within an organisation. As a consequence of this the process focus has received increased attention from researchers concerned with quality management. For example, Mortiboys [12] emphasises the importance of developing “..a cost-effective quality management culture” based upon “controlling the process rather than controlling the people or the product.”[2] Rowles et al [14] also identify the efficiency and quality benefits that process modelling can bring to an organisation.

1. The definition of ‘problem’ adopted by Checkland means that the value of process modelling is not restricted to an arena where concerns are explicitly defined or voiced.

2. R.J Mortiboys, ‘Quality Management for the 1990s’ in ‘Managing Quality’ eds. B.G. Dale and J.J. Plunkett; pps. 39/40.

These benefits are obviously very important and as shall be shown, the cookbook has been designed to support them. However, it is the contention of the cookbook that process modelling is about developing understanding and knowledge of what happens in a work environment. Therefore it is not restricted to issues such as efficiency but has a wider vocabulary. It can be applied to support change with motivations such as increasing job satisfaction, aiding job rotation and decreasing the hours worked as easily as it can be applied to efficiency criteria. This wider vocabulary is accommodated in the cookbook.

2.0 An Introduction To The Cookbook.

The major features of the cookbook are the following;

• The construction and application of a sophisticated, multi-view graphical process model.

• The maintenance of a dialogue between the process modeller and those who actually participate in the process.

2.1 A summary of the cookbook method.

The structure of the cookbook is introduced in the following section. The method is summarised in Table 1.

Table 1: A Summary Of The Cookbook Method.

Phase.	Contents.	Options.
Representation	Dialogue with process participants core question is '**What actually happens?**'	Use the concept of work-groups to define who to talk to... Coarse work-groups (defined by job title). Refined work-groups (account for additional influences).
	Build a multi-view model cross-reference the views and end with a Quality Assurance review. Modelling notations are: **Person Centred Process Charts, Data Flow Diagrams, Role Activity Diagrams, Object Flow Diagrams.**	
Refinement	Process Analysis a. use knowledge gained through model building b. use expertise c. use process metrics *	
	Dialogue input core question is **'What do you think about the process?'** a. use output from previous dialogues b. renew / start new dialogues - 'the metric of the user's view' - reconciliation of different perspectives	Enrich the dialogue through cultural enquiry. Use techniques from SSM to achieve a reconciliation of perspectives.
	Implementing change model & document proposed refinements	
Process Management	Cyclical refinement the Refinement phase is cyclically applied to introduce evolutionary changes to the real world. The Representation phase is partially re-invoked to provide a cross-check of the model and the evolving reality.	The cookbook can contribute to the application of the Process Maturity Framework.

* A process metrics framework has not been developed for the cookbook.

The cookbook method is concerned with the construction of a process model and the application of this process model to the refinement of the real world process. Optional components allow the cookbook to be used for process management and to provide for specific circumstances of use. For example, the process maturity framework enables the cookbook to assist the development of process management structures.

2.2 The heritage of the cookbook.

The cookbook draws upon new work undertaken as part of the Manchester University/ British Telecom collaborative project. It also utilises existing research and techniques as outlined below.

2.2.1 Existing BT work.

A major contribution made by earlier British Telecom work is the tenet that "..to model a process comprehensively there is a need to make use of a number of different views."[1] The views prescribed are as follows;

• Functional - describing what activities are being performed.

• Behavioural - describing when and how the activities within a process are performed.

• Structural - describing the entities within a process and the relationships between them.

• Resource - describing who performs an activity and where it is performed.

•The following notations have been used by BT to represent each perspective:

• Functional - data flow diagrams.

• Behavioural - state transition diagrams.

• Structural - entity relationship diagrams.

• Resource - role/activity matrix.

The cookbook adopts a reformulated version of the multi-view approach which has been developed through associated case study work.

2.2.2 Soft Systems Methodology.

Checkland [3] cites Optner writing in 1965 as giving four reasons for the use of the concept 'systems.'[2]

• Many problems are seen to recur when they are looked at as problems of systems; hence solutions may be transferred.

• The systems view enables problem-solving to concentrate upon the **processes** by which things are done, rather than on (ad hoc) 'final outcomes.'

• "Systems may provide the objective standard by which problems can be organised for solution...From objective standards we may be able to gain greater insight to generalise on business phenomena."

1. Evans et al, Process Modelling Case Study, p.7.
2. Reference: S. Optner; Systems Analysis for Business and Industrial Problem Solving, Prentice Hall, 1965.

• Many business problems are ‘mixed’ in that they have both qualitative and quantitative attributes.

There is common ground shared by process modelling and the work of Checkland. It could be a useful, related research area to establish how close this relationship is. This is not attempted in the cookbook. However the cookbook has inherited some concerns and some procedure from Checkland’s Soft Systems Methodology. This is most notably in the areas of understanding complexity and promoting communication between participants.

2.2.3 Process Maturity Framework.

Humphrey [8] identifies five steps that organizations must take in order to improve their software capabilities:

• Understand the current status of their development process or processes.

• Develop a vision of the desired process.

• Establish a list of required process improvement actions in order of priority.

• Produce a plan to accomplish these actions.

• Commit the resources to execute the plan.

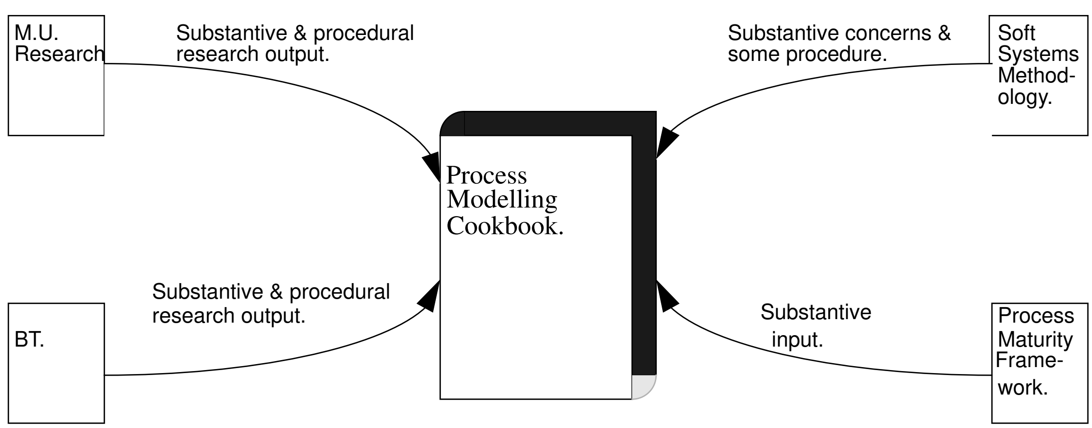
On the basis of these five steps a ‘maturity framework’ has been developed by Humphrey to characterise the status of an organization’s processes.

Whilst this was developed within an SEI project and is specifically oriented towards software processes, it is used by the cookbook as a means of classifying and developing the management of more general processes. Its use is an option and is not mandatory to the process management structure.

2.2.4 Summary: The heritage of the cookbook.

The heritage of the cookbook is summarised in Figure 1:

Figure 1: The Heritage of the Process Modelling Cookbook.



3.0 The Structure of the Method.

The structure of the method reflects the philosophy that underpins it. It will have two distinct phases reflecting concern for two domains.

3.1 Representation and Refinement.

The first of these, **Representation of the Process**, will be concerned with achieving a valid model of the real world process. Therefore in this domain, the method will address issues such as knowledge elicitation, process capture, the use of graphical techniques and the means of ensuring that the model is consistent with the real world. The second phase, **Refinement of the Process**, will use the model as a platform upon which understanding of the real world process can be built. It will then use this understanding as a basis for exploring ways of refining the real world process itself.

This structure of representation followed by refinement differs from most conventional methods. It is usual to start with requirements analysis and/ or some notion of a desired system. The Yourdon derived method adopted by Rowles et al [14], illustrates this concern by seeking to identify current deficiencies and prioritise goals very early on[1]. The cookbook differs by incorporating notions of desired change in its second phase, Refinement of the Process. The cookbook shall thereby establish knowledge about a process in place of uncertainty and on this basis be able to explore the means of and need for change. This accords with the tenet introduced above, that process modelling is in itself a valuable activity and can bring benefit to an organisation without a presumption that change will always result from the process modelling exercise.

This structure gives voice to the ‘wider vocabulary’ of motivations introduced in 2.2.1 above. The first phase is concerned with building up knowledge of a work domain. The second phase, Refinement of the Process, uses this knowledge as a basis for refining the process itself. Figure 2 shows how ‘problems, goals, inconsistencies, concerns etc.’ are input to this stage. This is where process participants have their say and encourage the process modeller to respond to what they deem important. Therefore, if efficiency issues are input the refinement phase will respond to them and use them as a guide for the refinement process. If job satisfaction issues

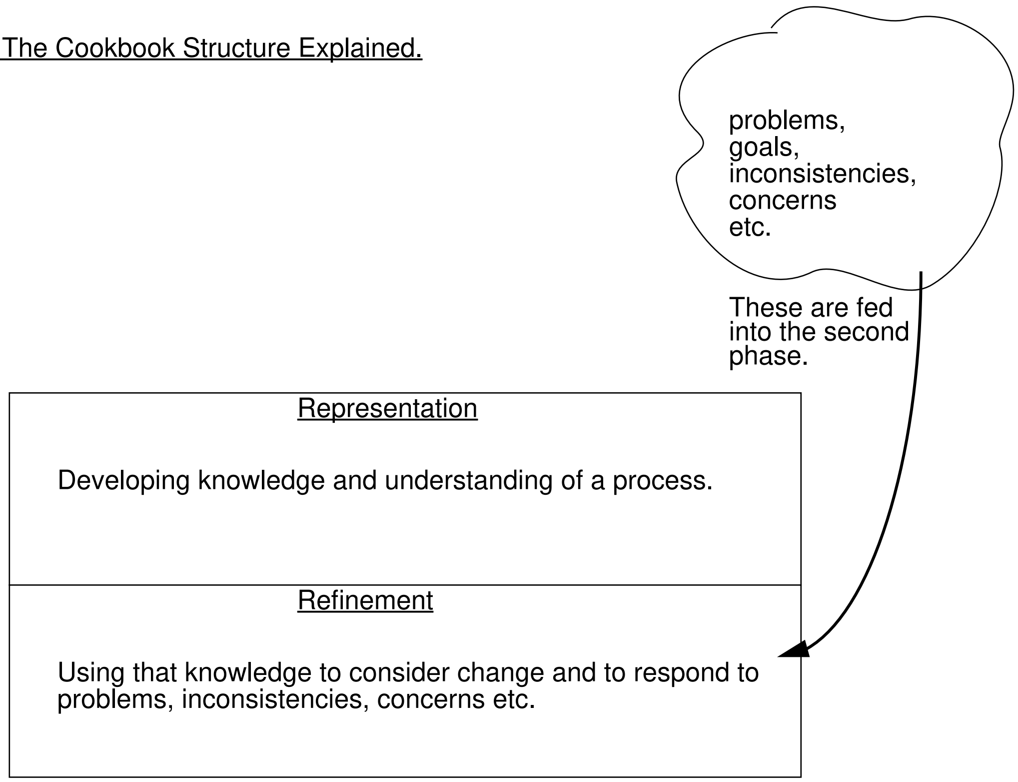
1. Whilst Rowles et al [14] adopt the Yourdon Structured Method, they state;

“Given the high degree of commonality in the approaches taken by different structured methods and CASE tools, we have no reason to suppose that it would not be possible to choose another combination of methods and tools to achieve the same purpose.” (p.25).

Broadly, the philosophical difference between the Process Development Handbook and the Process Modelling Cookbook can be described as follows. The former is oriented towards the development of executing systems using PSS. The latter is oriented around the benefits of the process modelling activity itself whilst retaining the option of developing executables where this is thought appropriate.

are input the refinement phase will respond to them and they will guide the refinement process.

Figure 2: The Cookbook Structure Explained.



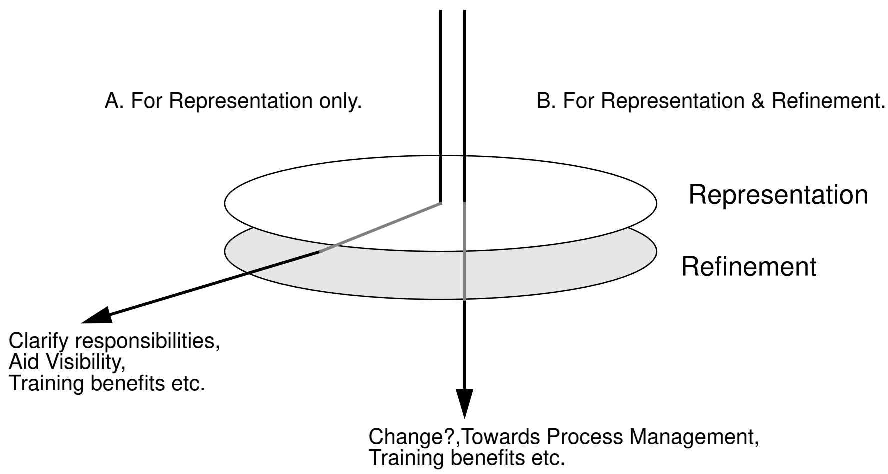
However, the method cannot of itself resolve the political issues noted by White [15]. Whilst the cookbook shall be flexible enough to respond to a variety of motivations for change, it cannot ensure that all perspectives are given equal weight. The political nature of the work environment may often result in certain motivations (e.g. management motivations) being voiced more loudly than others.

3.1.1 Representation: without refining the process.

As an extension to the philosophy that says process modelling can be an end in itself, it is provided that the first phase of the cookbook, Representation, may be used independently of the Refinement stage. This is possible because the output of the Representation phase is a developed process model of what happens in the given work environment. It may be useful where there is no wish to consider changing processes but where there is a need to clarify responsibilities, facilitate training or make processes more visible etc.

On this basis it is therefore possible to create a simplified paradigm of cookbook usage. Figure 3 describes two routes through the method (A & B). A third route, which uses the method for process management is introduced below at and described further at Figure 24.

Figure 3: Using The Cookbook.



3.2 Using the cookbook for process management.

Conventional structured methods are designed to be used once through with a system design or solution output at the end. The cookbook's concern for the work processes gives it a different orientation and a wider range of applications.

The maintenance of a cycle in the refinement phase enables the cookbook to assist process management. It becomes a vehicle for the investigation, monitoring and refinement of work processes. Thereby it responds to the need expressed by Mortiboys [12] for process control as a key to management quality.

4.0 Phase One: Representation of the Process.

The primary aim of this phase of the methodology is the construction of a complete and consistent model of **what actually happens** in the study environment. This model will be constructed using the multi view approach of Evans et al [6].

There are two work areas within the Representation phase:

• Dialogue with process participants.

• Multi-view modelling.

4.0.1 Highlights.

This section will show how the above work areas can be used to construct a graphical model of the work environment. This process will start with the modeller talking to participants. Work-groups may be defined to ensure that the modeller achieves a broad representation of participants in the dialogue.

Person centred process charts act as a structured notebook as the modeller develops the model. Data flow diagrams reflect a functional view and are created in a manner that ensures a process wide perspective. Role activity diagrams retain the person centred perspective established by the dialogue. They are used to group actions into natural roles and to allow the specification of conditions within a process. Object flows describe how defined classes of objects are used in the process. They can be mapped onto the data flow diagrams for a process wide perspective or the role activity diagrams for a more detailed person centred perspective.

4.1 Dialogue.

4.1.1 Background.

In presenting his Soft Systems Methodology, Checkland [3] gives great emphasis to the need for a comprehensive articulation of the problem situation. This cookbook reflects this concern by encouraging the process modeller to take as much information as possible from conversations with participants. Much of this information may seem irrelevant but will become useful as the context develops. Additionally, the information may be useful in that it acquaints the modeller with the climate and concerns of the study environment. In this way the process modeller takes on a role similar to that of ‘concerned actor’ Checkland assigns to the analyst.

The cookbook differs however, by requiring that the modeller obtains a core of information on ‘what the participant actually does.’ At this stage this should be the underlying concern of a conversation with a participant and the only thing that structures it. This core will increasingly become the focus for the modeller as the process of graphically representing what happens in the study environment develops.

4.1.2 Starting dialogue.

Upon entering a study environment, a process modeller will therefore be first concerned to talk to the participants. Very often, this will be a relatively straightforward to arrange. For example, in the RAG[1] case study the participants are relatively few in number and their contribution to the process is relatively visible. However this will not always be the case. Many processes will involve quite large numbers of participants and may be characterised by ‘fuzziness’ about contributions, involvements etc.This raises the question 'Who should I talk to? This question can be answered in terms of work-groups.

4.1.3 Defining work groups by job title.

In essence, the process modeller will need to ensure that all work-groups that participate in a process are represented in the dialogue. Through this the process modeller can protect against the danger of being 'swamped' by information whilst ensuring that 'information gaps' do not occur. Different work-groups have different viewpoints upon a process resulting from different organisational standpoints. For example in a theatre, bookings clerks, ushers, managers, cleaners, technicians, wardrobe assistants and actors will all have different viewpoints upon the process of putting on a performance. Therefore the modeller can start by sub-dividing the participants by department (where appropriate) and by job-title. This leads to a subdivision by **coarse work-group**. If the process is of considerable complexity it can be useful to incorporate this subdivision into a simple work group diagram or picture (e.g. a Venn diagram).

4.1.4 Refining work groups.

However, this dissection by job title may over simplify the number and character of work-groups within the process. For example;

• One booking clerk may be a trades union representative and have an altered perspective upon the process as a result.

• Among the group 'technicians' there may be very different jobs (e.g. lighting, sound, props) and very different concerns as a result.

• Wardrobe assistants may on the surface all do a very similar job. However, there may be some hidden factor which may be very influential. For example, it could be that only wardrobe assistants with over five years experience are allowed to purchase new stock. This may considerably affect the work that the senior wardrobe assistants do. Therefore, there may be a considerable schism in the work undertaken and in the feelings expressed within, apparently, a single work-group.

This means that the process modeller cannot always rely upon work-groups defined by job titles and may need to update the work-group diagram as more information is uncovered. Through this a sub-division into **refined work-groups** is achieved.

This is not intended to imply that all differences between participants matter to the process modeller. In this judgement is required. For example, the process modeller may not consider that the booking clerk who is also a trades union representative can contribute anything substantially different to any other booking clerk. However, it may be felt that the wardrobe assistants who have worked more than five years and have additional responsibility are worth considering separately to the more junior wardrobe assistants. Remember, the process modeller should orient the dialogue around the question 'What actually happens in the work environment?'

---

1. RAG is an acronym for Red Amber Green, a project management method developed and employed by BT. Projects are assessed according to cost criteria and given a red, amber or green status as a result. This status will determine the action taken on each project. For example, red status will mean that a project is reviewed immediately by a Project Action Forum.

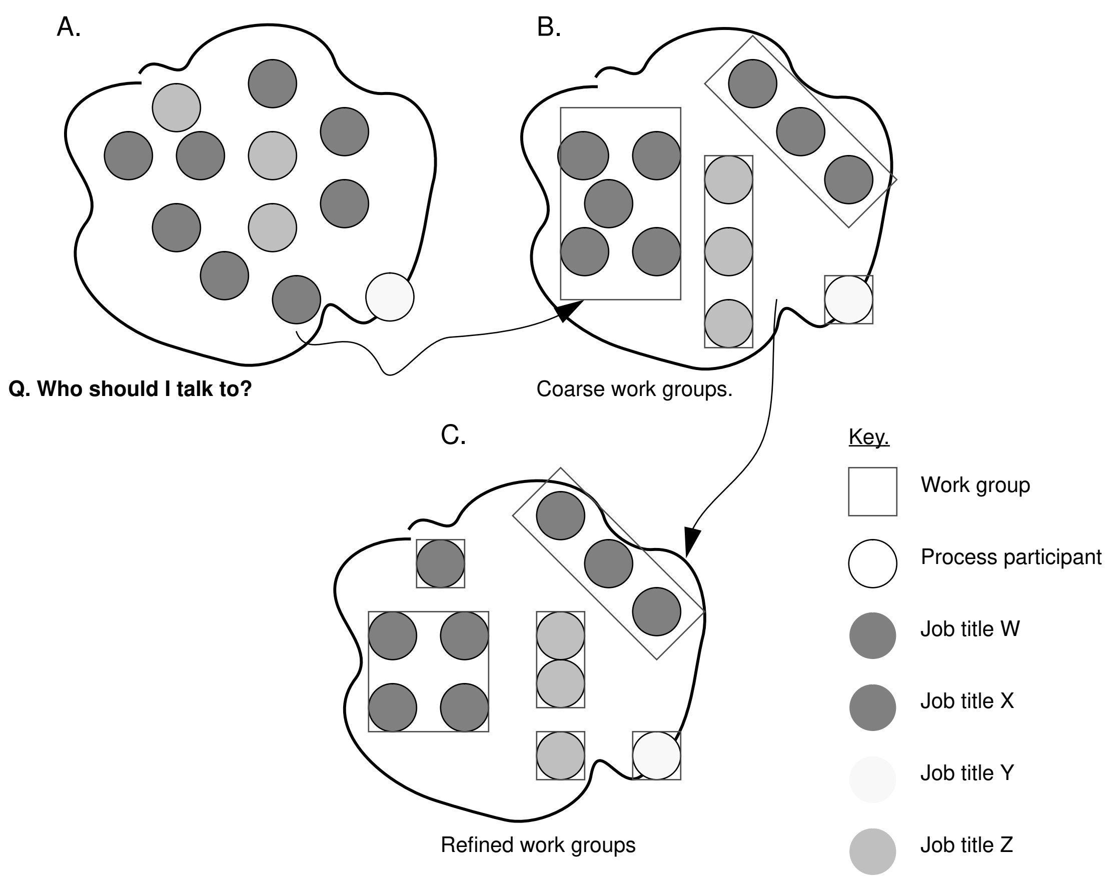
The RAG case study was used extensively in the preparation of this version of the Cookbook. A number of examples are quoted from it and some diagrams relate to it.

For an explanation of abbreviations see Appendix Two.

Whilst building up knowledge of the work-groups in a process, the modeller will be concerned by the question ‘How many participants from a single work-group should I talk to?’ Again, judgement is required. As a rule of thumb it can be suggested that more equals better whilst wishing to avoid the danger of taking in too much additional information.

The process of defining work-groups is summarised in Figure 4.

Figure 4: Defining Work Groups.



4.1.5 Scoping.

Scoping a process modelling investigation is a complex issue. It involves more than just the definition of a boundary of concern, although this is in itself difficult enough. It also involves the introspective scoping of what work is to be done within that boundary.

Within an organisation it will usually be difficult to separate one process from others. This is especially likely where human factors are an issue and, for example, one participant may have responsibilities relevant to the process model (e.g. RAG reports) and responsibilities irrelevant to the process model (e.g. project work).

By defining work-groups the process modeller naturally scopes the study. It provides a definition of a boundary (i.e. these are the relevant work-groups) and introspectively scopes within this (e.g. 'Within these work-groups I am not going to talk to everybody, just representatives from each.'). Therefore work-groups assist the process of scoping a study.

However, the scope of a study cannot be defined once only. The modeller must constantly review what is and is not being considered. For example, the RAG case study was originally scoped to include just the RAG process itself. However, it became apparent that there was some difficulty about the relationship between RAG and other project management processes. The boundary of the investigation therefore had to be reviewed and extended[1]. Therefore it is through a series of incremental and iterative decisions initiated by the definition of work-groups that the cookbook scopes its study.

An alternative to scoping through the use of work-groups is an adaptation of the Problem Definition Stage of SD-Scicion's CORE method. This involves interviewing a "senior member" of the organisation about the organisation's business, the organisational structure, the process structure, links to other organisations, relationships between processes, methods and techniques used, standards and procedures. Through this interview the process modeller will be able to develop a 'feel' for the organisational and process structure. It will then be easier to identify those roles within the process which are relevant to the investigation.

4.1.6 The role of official documentation.

It will be common for the modeller to be introduced to a study environment by a set of official documents and (written and verbal) support information. It is recommended that such information is for the most part set to one side in the earliest stages of a study. The modeller should instead try to take as much initial information as possible from the conversations with participants. This is because the concern at this stage is the discovery of how processes actually operate. Recommended procedures, official processes and environmental histories may be misleading.

Official documentation should only be used in the following ways:

• As a very brief initial guide to a study environment (e.g. RAG is a process for project control reflecting a red, amber and green grading system).

• As a way of filling in stubborn information gaps. These are likely to occur because of the unavailability of process participants, items repeatedly missed in conversations etc. However, all information gained in this way should be cross-referenced with a process participant who can say how a process operates in reality.

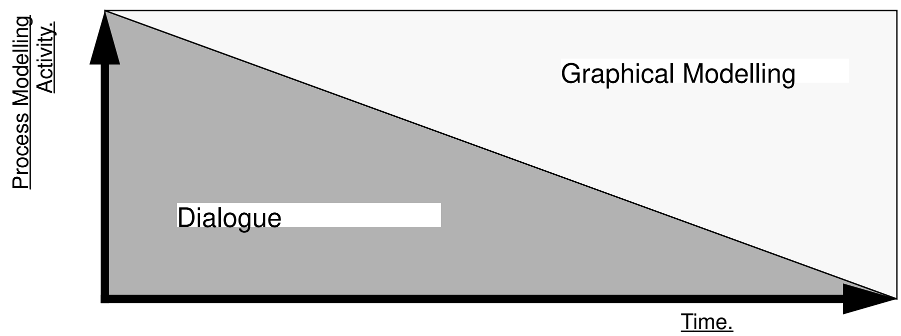
4.1.7 Maintaining a dialogue.

As the exercise progresses the modeller will become more concerned with the construction of a graphical representation than obtaining the information to model. However, there need not be a cut-off between the stages and it may be necessary to source information even when the modelling activity is very advanced. This process is simplified and summarised in Figure 5. The actual proportions of each activi-

1. It may not always be necessary to formally extend the study boundary. In some cases the modeller may need some knowledge of a related process (e.g. its outputs and aims) without requiring a complete process model of it.

ty taking place will in fact be determined by the modeller and the particular study circumstances.

Figure 5: A Simplified View Of The Representation Phase.



4.2 Multi-view modelling.

4.2.1 Four views of the process.

The cookbook adopts the following four modelling notations:

• Person Centred Process Charts.

• Data Flow Diagrams.

• Role Activity Diagrams.

• Object Flow Diagrams.

These reflect the following four views:

• Person Centred Process Charts - from the perspective of a single person, what activities are performed by that person using what objects.

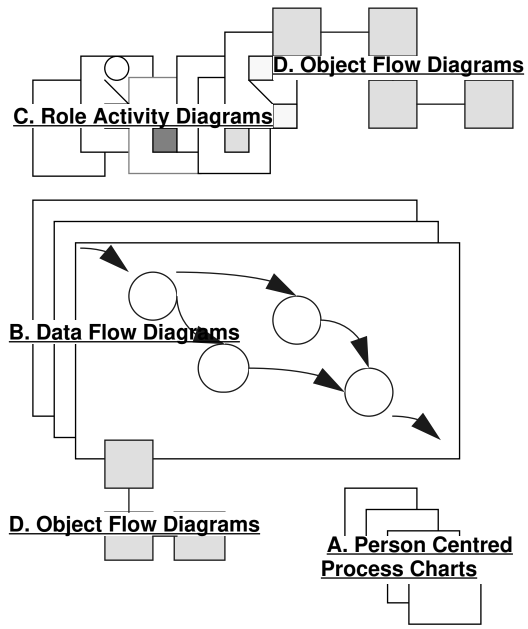
• Data Flow Diagrams - what activities are being performed.

• Role Activity Diagrams - when and how the activities within a process are performed.

• Object Flow Diagrams- what objects are manipulated by the process.

The role each modelling notation plays in the cookbook is introduced by Figure 6.

Figure 6: An Introduction To The Modelling Notations.



A. Person centred process charts provide a 'structured notebook' of activities, objects and roles.

B. Data flow diagrams document the activities that are performed from an overall perspective. They are not person centred.

C. Role activity diagrams show how roles are made up of logically connected activities. They can be enhanced to show the conditions under which activities are performed. The person centred perspective is adopted for the RADs.

D. Object flows can be superimposed over RADs or DFDs to show how objects are manipulated and used by the process.

4.2.2 Basic concepts.

The following concepts are common to the notations used in the cookbook:

• Activity/ Action - The terms activity and action are used interchangeably to describe a sub-component of an overall process. Therefore, in the language of the RAG case study, project management is the overall process whilst 'assess PMRs' and 'send invites to SPMs' are activities. The implication of this for data flow diagrams is that in this terminology they describe the flow of data between activities in a process rather than the flow of data between processes. There is no semantic difference between the two.

• Role - The term role is used to describe a logically linked collection of activities. Therefore any process participant has a potential one to many relationship with roles[1]. A role is not a single participant or a complete job description.

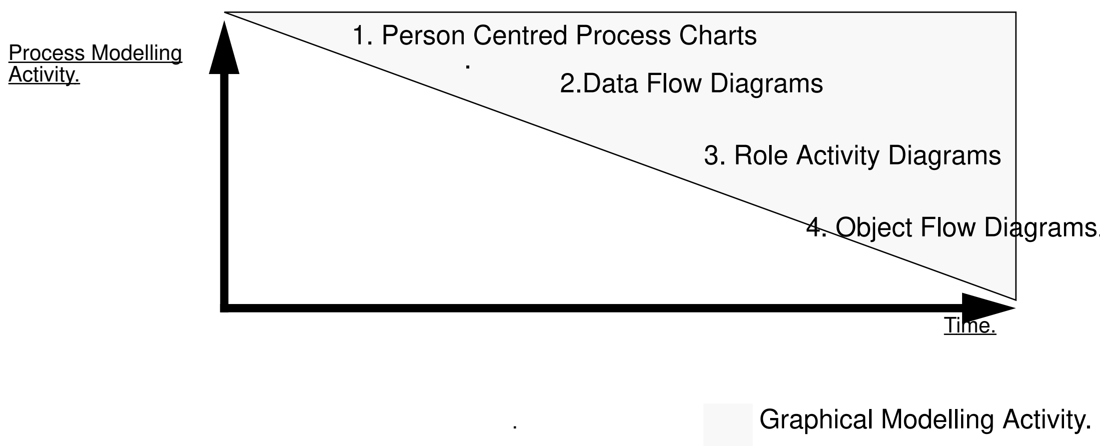
1. This is similar to the way in which roles are encoded in PML, an executable language developed for process modelling with PSS (Process Support System).

• Object - An object can be described as anything used by a process participant to facilitate the process. Therefore, in the RAG case study, the Project Manager's Report, the RAG database and email can all be described as objects. They can be further refined into three categories; read only resources, read write resources and tools. A resource is an object which is manipulated by the participant within an activity. If it is not changed by the activities then it is a read only resource. If it is altered in any way, then it is a read write resource (e.g. new information added, fields appended or removed, collated with resources of the same or a different type). A tool facilitates an activity, typically through provision of a medium for a resource (e.g. email, paper copy).

4.2.3 The graphical model building process.

The relative complexity of the multi-view model necessitates proper management. One way this is achieved is through an order assigned to the sequence in which the notations should be commenced. This is expressed in Figure 7.

Figure 7: A Simplified View Of The Ordering Of Graphical Notations.



This expression is simplified in that it does not properly display the way notations shall be cross referenced to ensure consistency. Three simple rules can help proper cross referencing:

• If the process is large it should be broken down into manageable units with a logical thread.

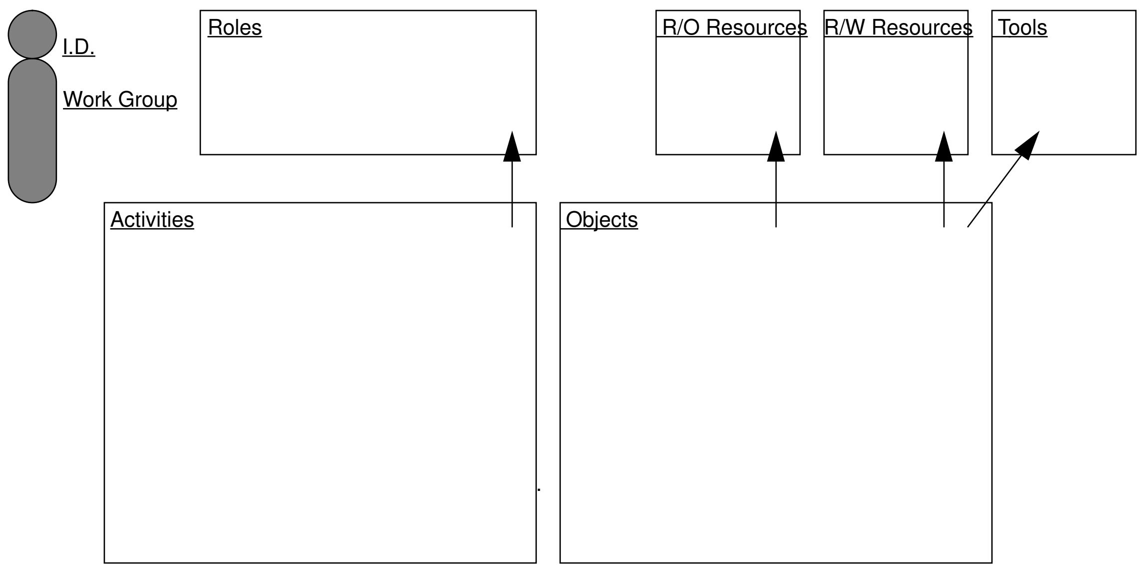
• The modeller should not attempt to complete the modelling of a unit in a single notation without starting work on its successor notation.

Much of the modelling work on different notations will take place in parallel. No view can be judged finished until it has been cross-referenced with the final view.

Other means by which the complex graphical model is managed include the use of person centred process charts and a Quality Assurance review.

4.2.4 The Person Centred Process Chart.

Figure 8: The Person Centred Process Chart Notation.



The person centred process chart enables the process modeller to record information gained from a participant in a manner that feeds into the rest of the modelling process. It provides an organised, matrix like format for the structuring of information essential to the model building process. It is therefore the easiest notation to start the modelling process with. Also, any single chart need not be 'completed' before modelling starts using the other notations. The modeller can return to a chart to enter new sections or to refine existing information. The chart is therefore analogous to a structured notebook into which the modeller records details from conversations with participants.

The chart is concerned with four questions. In likely sequential order these are;

1. What activities does the person perform?

2. What objects do these activities involve?

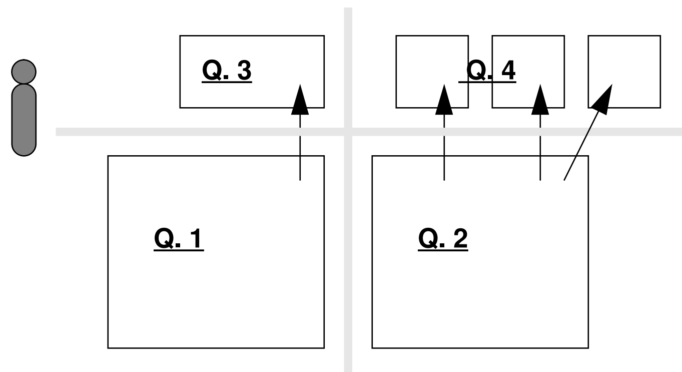
3. How do the activities map to roles?

4. How can the objects be classified?

A chart should be created for every participant interviewed by the modeller. By reference to the question 'What actually happens?', the modeller should be able to start answering questions one and two straight away. Questions three and four can later be answered by decomposition of activities into roles, and objects into object categories.

Figure 9 describes how the chart is used to record the answers to the four questions posed above.

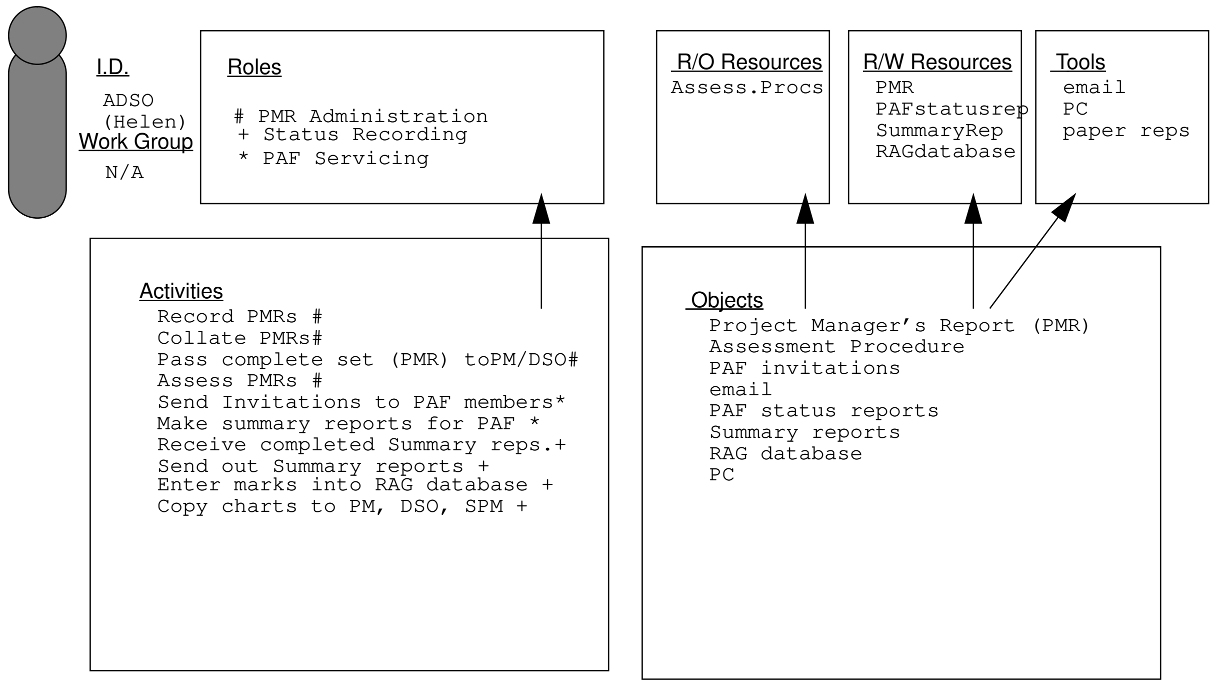
Figure 9: The Person Centred Process Chart Notation (2).



There are advantages to be gained from the modeller working to answer questions one and two for as many participants as possible before starting the data flow diagrams. This will enable the modeller to cross-reference information and where it is appropriate lessen the person centred perspective in favour of an overall process view. This in turn will help the modeller identify duplicity and contradiction in the charts and these can be refined accordingly.

Figure 10 shows a person centred process chart with all questions having been given consideration[1].

Figure 10: An Example Person Centred Process Chart.



4.2.5 Data Flow Diagrams.

The data flow diagram describes the functional view of the process or 'what activities are performed.' The notation is introduced at this point particularly to serve two ends:

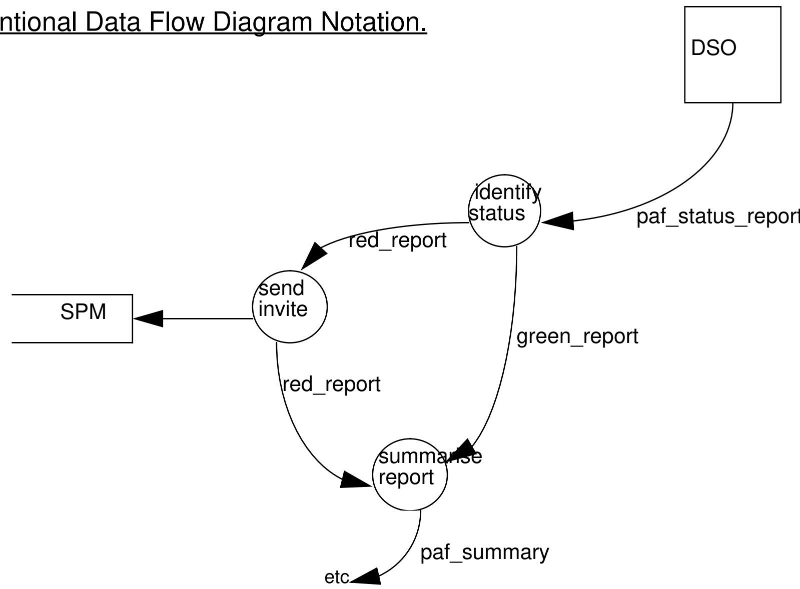
• As a medium to aid the communication between the process modeller and the process participants.

• To broaden the perspective of the process modeller from multiple instances of person centred views to an overall process perspective.

1. This example is not fully refined but is at a useful level of detail. For example, an activity such as "Make summary reports for PAF" could be sub-divided into "prepare summary reports" and "send summary reports." The general rule is to attempt to find a level that is accurate and understandable. For diagrammatic notations this may well be at a finer grain than for this chart.

The conventional data flow diagram notation as defined by Layzell and Loucopoulos [13] is suitable to support these needs. This is shown at Figure 11.

Figure 11: Conventional Data Flow Diagram Notation.



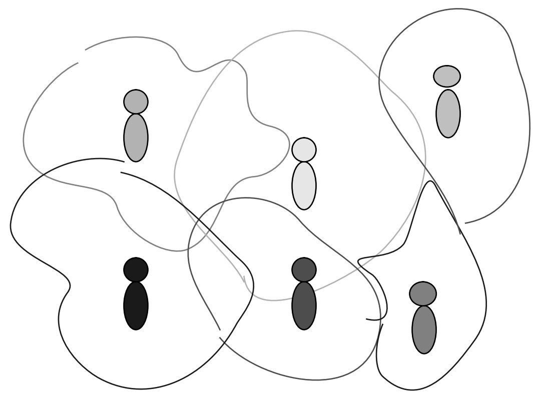
Ashworth and Goodland [1] state; "The data flow diagram is a meeting point of the analysts and users."[1] In other words, it is simple and precise enough to be used by the modeller to capture the model and it can also be used as an aid to interactive communication about the process. Therefore, the data flow diagrams are first utilised at a point in time when dialogue activity should still be considerable (see Figure 7).

A person centred perspective as used by the person centred process charts is an essential response to the need to acquire detailed information about 'what actually happens' in the process environment. The danger inherent to this approach lies in the fact that participants will always view a process from their particular standpoint within it. They cannot be relied upon to provide an overall process perspective, especially if the process is large or complex. This problem is summarised in

1. SSADM, A Practical Approach - ch3, p71.

Figure 12.

Figure 12: The Problem Of Perspectives.



Different process participants may only be able to see the process from the perspective of their particular work area. The process modeller needs to understand the process as a whole. It is not sufficient to rely on a series of partial perspectives that are not fully correlated into a whole.

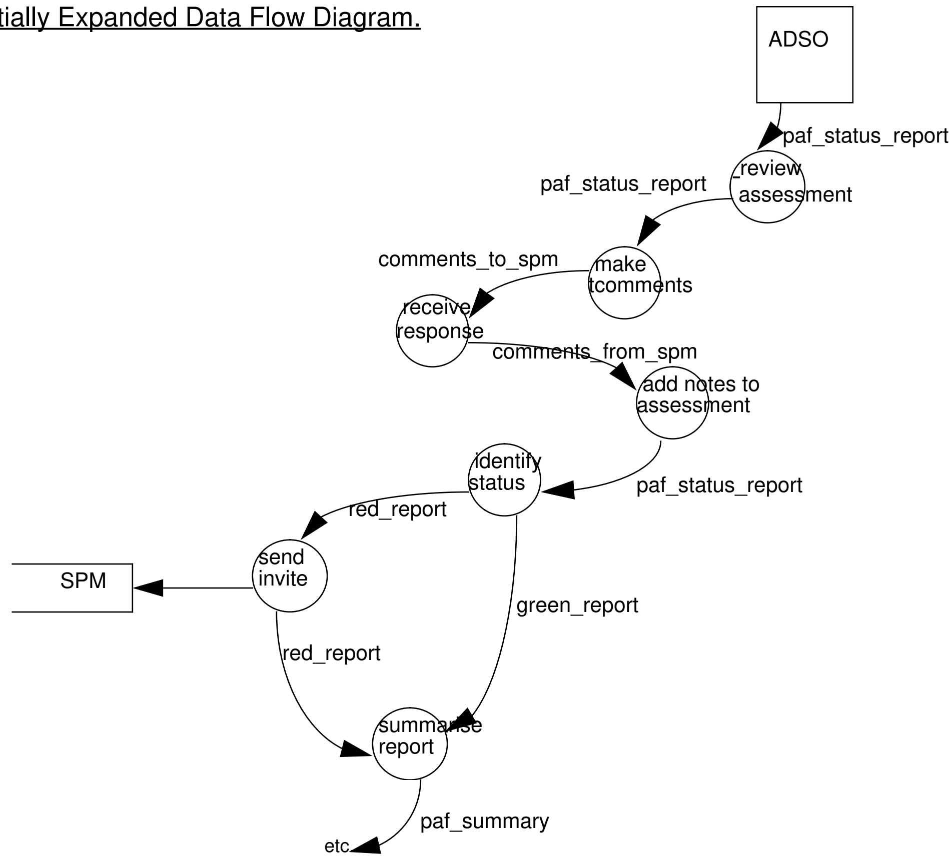
One response to this problem of perspective is the definition of work-groups. This enables the process modeller to ensure that the perspectives that are consulted are spread across the process.This is only a partial solution. There remains the difficulty of correlating these perspectives

This difficulty can be overcome by cross referencing activities from the person centred process charts and chaining them into logical sequences. Also, where a data source or sink is a participant in the process (for example Figure 11), there is a logical possibility of the expansion of the data flow diagram.

Figure13 shows an expansion of the process introduced in Figure 11. It is likely to be still incomplete because the incorporation of the ADSO as a data source suggests the possibility of expanding the diagram further (subject to the system boundary). Also, the passing of comments in the form ‘comments_to_spm’ and ‘comments_-

from_spm' implies that there may be another participants perspective yet to be incorporated in the diagram (namely the Sub-Project Manager or 'SPM').

Figure 13: A Partially Expanded Data Flow Diagram.



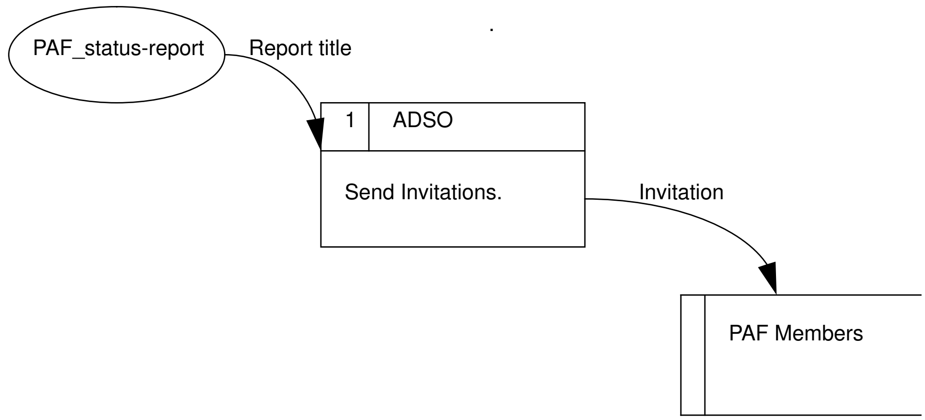
Note that with a broader process perspective the diagram invites questions to be asked from an overall standpoint. The following are examples of questions that might be asked from looking at the process in this way:

• What purpose does 'identify status' serve (given that the 'paf_status_report' is introduced earlier in the process?

• How are the 'comments_to_spm' and 'comments_from_spm' transmitted? What purpose do these transmissions serve?

A higher level interpretation of the data flow diagram can be added to the model to assist communication with users. This perspective level is useful also because the process of building it precurses the activity of defining roles for the role activity diagrams. The SSADM/ Yourdon/ Martin notation appraised in Evans [5] is selected for this purpose. This is shown in Figure 14.

Figure 14: High Level Data Flow Diagram Notation.



4.2.6 Role Activity Diagrams.

There are two main reasons for using RADs at this point:

• To provide a person centred, graphical notation that shows to an informative level of detail the roles that make up the process.

• To allow the description of conditions on the undertaking of actions within a process.

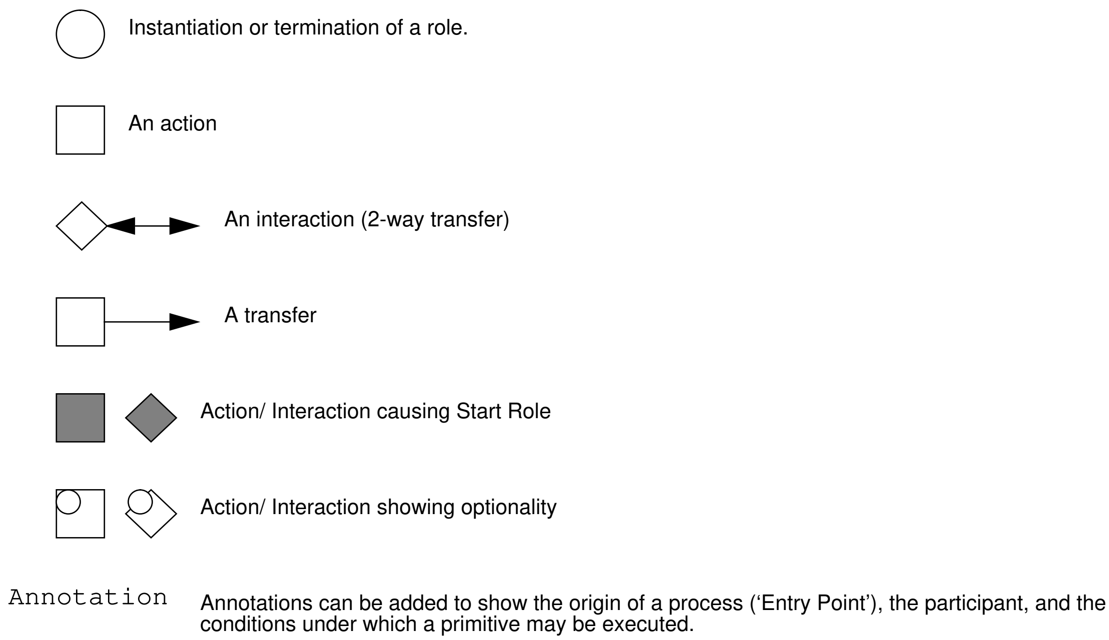
The person centred perspective should be maintained for the RADs. This will mean that the modeller will have graphical presentations of the overall process perspective (DFDs) and the person centred perspectives (RADs) when refining the process. The ability to switch between these perspectives will greatly enhance the modeller's understanding of the process and will enable the identification of how different participants contribute to the process.

Roles can be defined through the identification of logical groupings of activities in the person centred process charts. The modeller may find it easier to identify logical bindings by looking for short-term goals within the overall process goal. Short-term goals may be indicated by activities that a participant tends to do in a short time span. For example, the activities related to the role 'PMR Administration' in Figure 10 tend to be undertaken "together" (i.e. sequentially, in a short time span). The reason for this is that the participant (the ADSO) wishes to enable other participants (the DSO and PM) to get on with their work. Therefore, the goal in this case is 'to enable the DSO and PM.'

In general, roles should be defined so as to be logically cohesive and 'natural.' To this end the modeller should be prepared to refine the role definitions many times if necessary. Ultimately, it may be useful to check the role definitions with the relevant process participant. If the participant does not agree that the work can be divided into the areas identified by the modeller, then the modeller is probably wrong[1].

The notational primitives for the Role Activity Diagram used in the cookbook are as displayed in Figure 15[2].

Figure 15: Cookbook Role Activity Diagram Notational Primitives.



The following rules have been defined for the use of these RAD primitives:

1. Optional actions are possible within a role (an interaction can be one or more parts of an optional structure).

2. An interaction is a two way exchange between roles (e.g. 'request and receive comments' would be an interaction, 'send document' would not)[3].

3. Transfers are actions that involve one-way exchanges between roles (e.g. 'send document' would be a transfer). Their inclusion in a RAD is optional (i.e. where the modeller thinks it useful).

---

1. It may be helpful to think of the role definitions as they would execute on a PSS. If they are not 'natural' and 'straightforward' to the user, the PSS is likely to be unpopular (and redundant!). Similarly, if the role definitions used here are not 'natural' and 'straightforward,' they are likely to make the modeller's view of the process and the participant's view of the process disjunctive.

2. This 'Cookbook RAD' is a customisation of the primitives presented for the RAD in Maresh & Wastell [11].

3. Interactions can be decomposed into a series of transfers. However, this will not always be useful and may not accurately represent the dynamic nature of a two way exchange.

4. Interactions and transfers are in fact simply specialised actions. Notational rules which apply to actions, e.g. for the specification of conditions, will normally also apply to these specialisms.

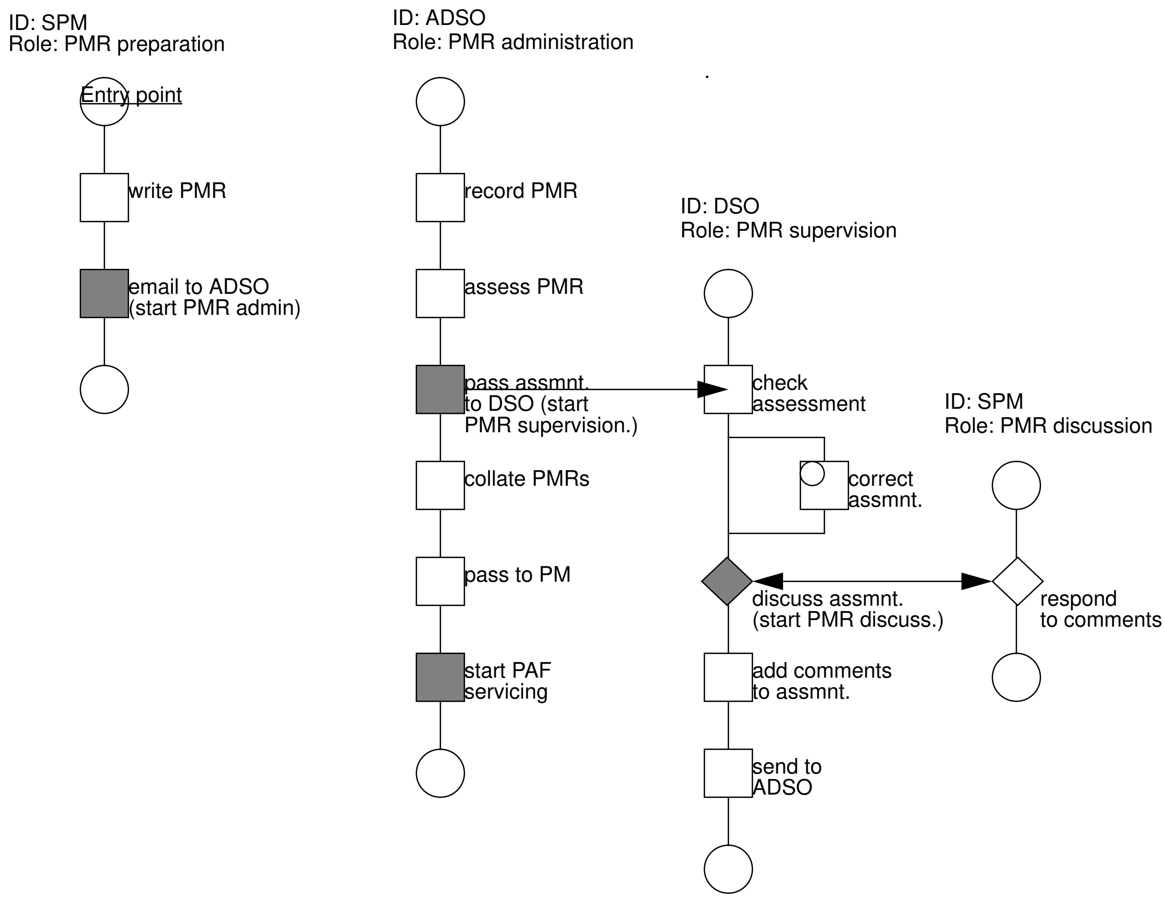
5. Start roles can be implicit (e.g. 'send document' could cause a new role to start) or explicit (e.g. 'start meeting support role').

6. Both actions and interactions can cause a start role. Logically, an interaction is limited to implicit start roles only.

7. Conditions can be added as required to show the circumstances under which a primitive may be executed.

Figure 16 shows a series of RADs created for the RAG case study:

Figure 16: RAD Example (1).



It is useful to note the following points in studying this example:

• There is an interaction between the 'PMR supervision' and 'PMR discussion' roles. Breaking down this interaction into a series of transfers (e.g. 'inform SPM of score,' 'request response from SPM,' 'provide response to DSO' etc.) would not confer any more information to the modeller. Indeed it may be more restrictive as the interaction may not always conform to an ordered pattern.

• There are a number of transfers shown in the example (e.g. 'send to ADSO' in PMR supervision role). Only one, 'pass assessment to DSO' in the PMR administration role is shown explicitly by the appendage of a directional arrow. This is shown as an example and transfers would normally only be made explicit when it is not obvious where their receiver is.

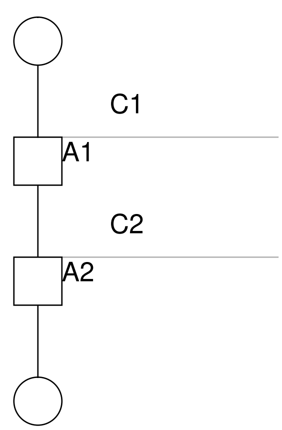
• All start roles except 'start PAF servicing' in the 'PMR administration' role are implicit. In order to make it clear in every case which role is started, a convention of providing bracketed confirmation has been adopted.

• A reasonably high level has been adopted for the RAD. Details necessary for process execution, e.g. how collections of reports are processed in each case, are not modelled at this level.

• A comparison with the DFD in Figure 13 will show that the actions do not map on to each other with thorough accuracy. For example, 'review assessment' (DFD) is described as 'check assessment' (RAD) and there is no action to 'correct assessment' in the DFD. This situation can be improved by the process of cross-referencing and mutual refinement previously described. The modeller should aim for maximum consistency between the two so that the perspective and notational differences can be most fully exploited.

Conditions can be added to RADs through the use of a horizontal bar connected to the top of an action primitive. This means that the action described can only take place when the condition is realised[1]. This is similar to the notational standard for State Transition Diagrams (appraised in Evans [5]), where conditions are shown on top of the bar with actions below and entity states described in rectangular boxes[2]. Figure 17 shows the standard notation for the addition of conditions to RADs.

Figure 17: RAD notation showing conditions.



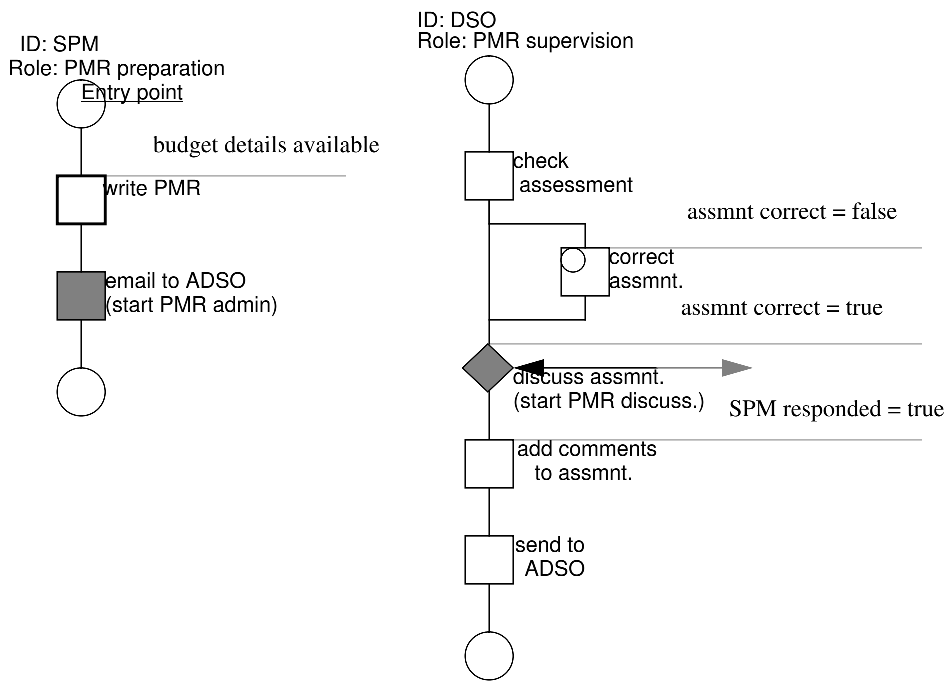
This means that A1 can execute only when C1 is fulfilled and that A2 can execute only when C2 is fulfilled. The top to bottom sequential ordering conferred by the RAD means that A2 cannot execute before A1 even if C2 is fulfilled before A1.

1. The condition is equivalent to a 'when' clause in PML.
2. Object flows can be used to describe entity states in the Cookbook.

There is no requirement that a condition be appended to an action in any particular case (this is high level modelling). The notational standard for conditions is an option for the modeller to use when it is thought useful to do so.

Figure 18 shows conditions appended to RADs:

Figure 18: RAD Example (2).



Boolean specifications of the kind used for the ‘PMR supervision’ role are potentially useful in ensuring rigour when mapping onto an analytical level where formal specification could be established[1].

4.2.7 Object flows.

Object flows are used to show how objects are manipulated by the process. They distinguish between resources which are used directly by the process and tools which facilitate the process. Whilst the cookbook concentrates upon resources, the concept of a ‘tool’ is very important. It allows a form of technology mapping to be built up which in turn can dictate how processes may be refined. Among the concepts introduced below are:

- Resources.
- Tools.
- ‘Read only’ and ‘read/ write’ state.
- Benign resources.
- Position.

1. This potential extension of the RAD notation is introduced in the Appendix.

Objects are defined by Macauley et al [10] as "..anything that a generic user manipulates, uses or handles in the course of work." This is a suitable definition for the purposes of the cookbook. Figure 10 shows how the general term 'object' can be broken down into three categories. Two of these categories are 'resource' categories whilst the third is a 'tools' or facilitating category.

Just what is a resource and what is a tool can be understood by examination of the relationship between them. An instance of a resource will always have at least one tool associated with it. A tool cannot exist without being associated with a resource.

A resource is an object which is input to the process or is created by the process with the purpose of achieving the overall process goal. Resources may often contain information essential to the process. For example, a design specification in a hardware manufacturing process is a resource. In the process of staging a play, a script is a resource. Resources may also be non-informative. For example, in an aircraft manufacturing process a wing is a resource (i.e. it is created by the process with the purpose of achieving the overall process goal).

A resource has state. It can either be 'read only' or 'read/ write.' A resource instance can move between these states at different stages within a process. To explain this the concept of 'position' is useful. Consider the Project Manager's Report (PMR) in the RAG case study. This is created by an action "write PMR" by the Sub-Project Manager early on in the process (see Figure 16). Therefore when associated with this particular action, or when at this particular 'position' in the process, the PMR is obviously a read/ write resource. Later on in the conduct of the process, the ADSO records the arrival of the PMR through the action "record PMR." The record is not made on the PMR itself which is not changed in any way. Therefore, at this position the PMR is read only[1].

Therefore read/ write state describes when a resource is created or has its form altered in any way. Read only state describes when a resource is used for some purpose but does not undergo any change. A resource can be described as 'benign' when it never has read/ write state throughout the entire process. The concept of position describes an association with a particular action.

Associated with each of the example resources documented above is at least one tool or facilitator. For example, a design specification may be associated with a facilitating PC. A script may be contained in a book. An aircraft wing may have many tools associated with it. For example, at one position it may be contained in a rack mount, at another it may be operated upon by a drill robot. Tools are needed to allow actions to be performed upon or using resources.

The classification of objects into categories can be initiated through the use of person centred process charts. The first action is to find the resources by looking through the actions list and noting the objects that are manipulated and used in each case. These can be entered into a general table marked 'resources' or subdivided further (as in Figure 10) to reflect the potential states of each[2].

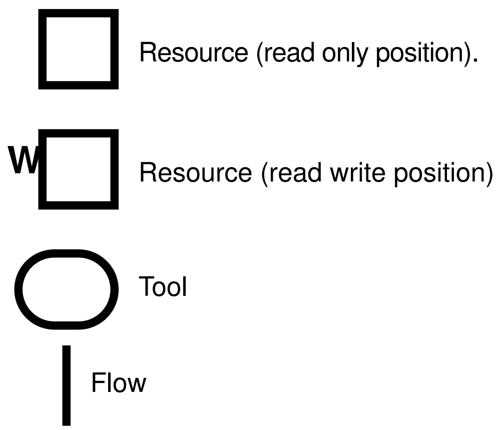
For each resource, the corresponding tool should be entered into a list. It may be important to map those areas of a process which are manual and those which have

1. Read/ write and read only states are here illustrated by reference to informative resources. These states are also useful to describe non informative resources. For example, an aircraft manufacturing process might include an action "measure wing." At this position the wing resource would be read only. There might also be an action "fit flaps to wing." Here the form of the resource is altered and so at this position the state is read/ write.

other technology dependencies. Without the capacity to do this it may be difficult to handle process improvements such as ‘allow X to write reports currently written by Y’ where Y writes all reports on a computer and X has no access to a computer[1].

The notational primitives of object flows are designed for use as overlays (either computer based or manual). They are introduced in Figure 19.

Figure 19: Object Flow Notational Primitives.



An overlay should be created for any object that the modeller wishes to display (it may occasionally be useful to map more than one object on a single overlay). The object can then be mapped over actions from a person centred perspective (RADs) or a process wide perspective (DFDs). Figures 20 and 21 illustrate these possibilities.

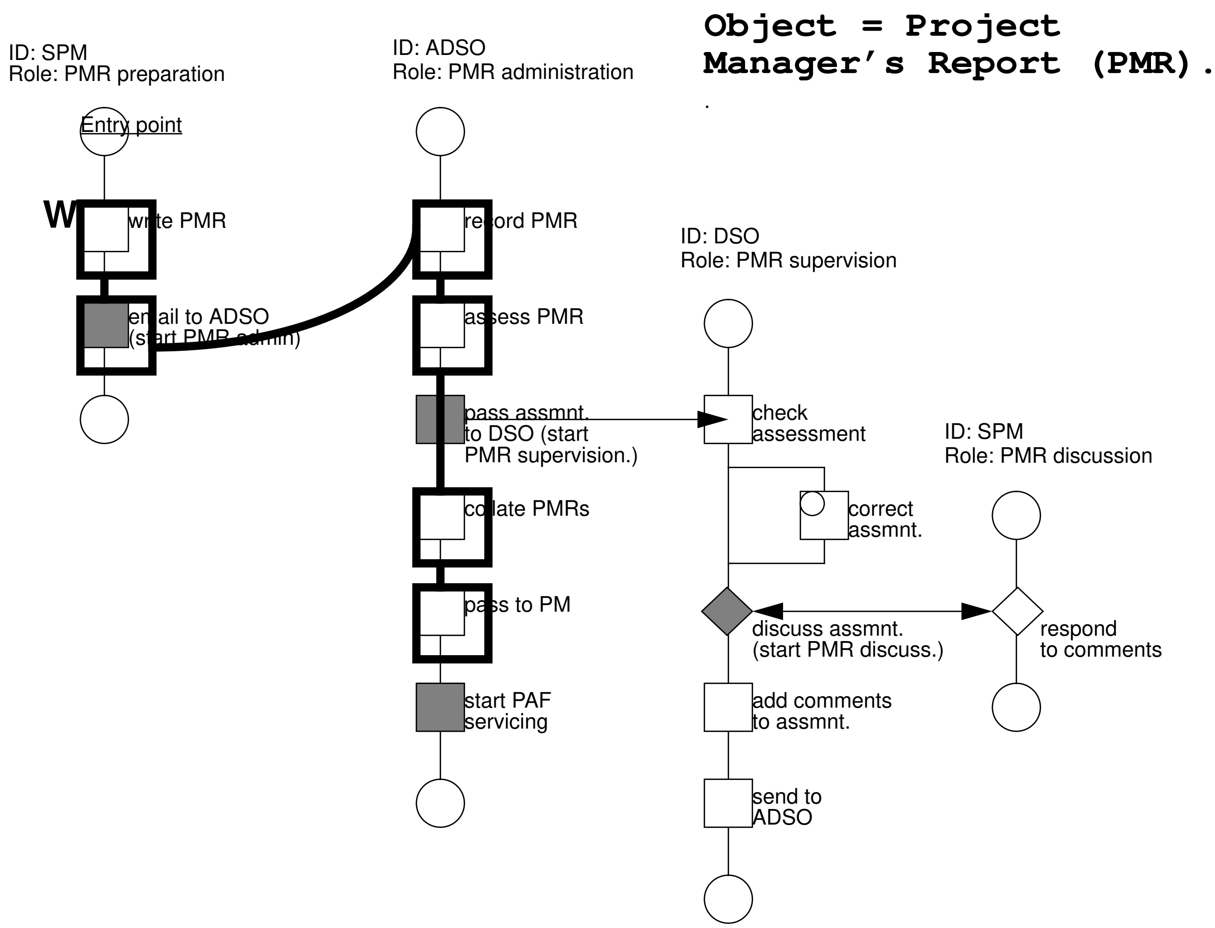
It is necessary to clarify what is meant by a flow. This is essentially a logical rather than a physical connective. A flow denotes the point where an object is next used in a process. For example, Clerical Officer X may use a shipping itinerary to make bookings. Clerical Officer Y may use the same shipping itinerary to pay costs. From a process perspective, the work of Officer X (bookings) must occur before the work of Officer Y (paying costs). Therefore, an object flow exists with the shipping itinerary flowing from bookings to paying costs. This is true even if physically separate copies of the itinerary are used by Officer X and Officer Y (e.g. duplicate photocopies).

2. Only a rough classification by state is possible using the person centred process charts. In Figure 10 the category ‘read only’ contains those resources which are read only through the entire process (i.e. benign) when seen from the person’s perspective. Those resources which have read/ write state at any position in the person’s activities are put into the read/ write category.

1. It will often be useful to identify computer tools at a software level. For example, it may be necessary to record that one participant exclusively uses WordPerfect whilst another uses only PCWrite.

Figure 20 displays a simple mapping of an object flow over a Role Activity Diagram[1].

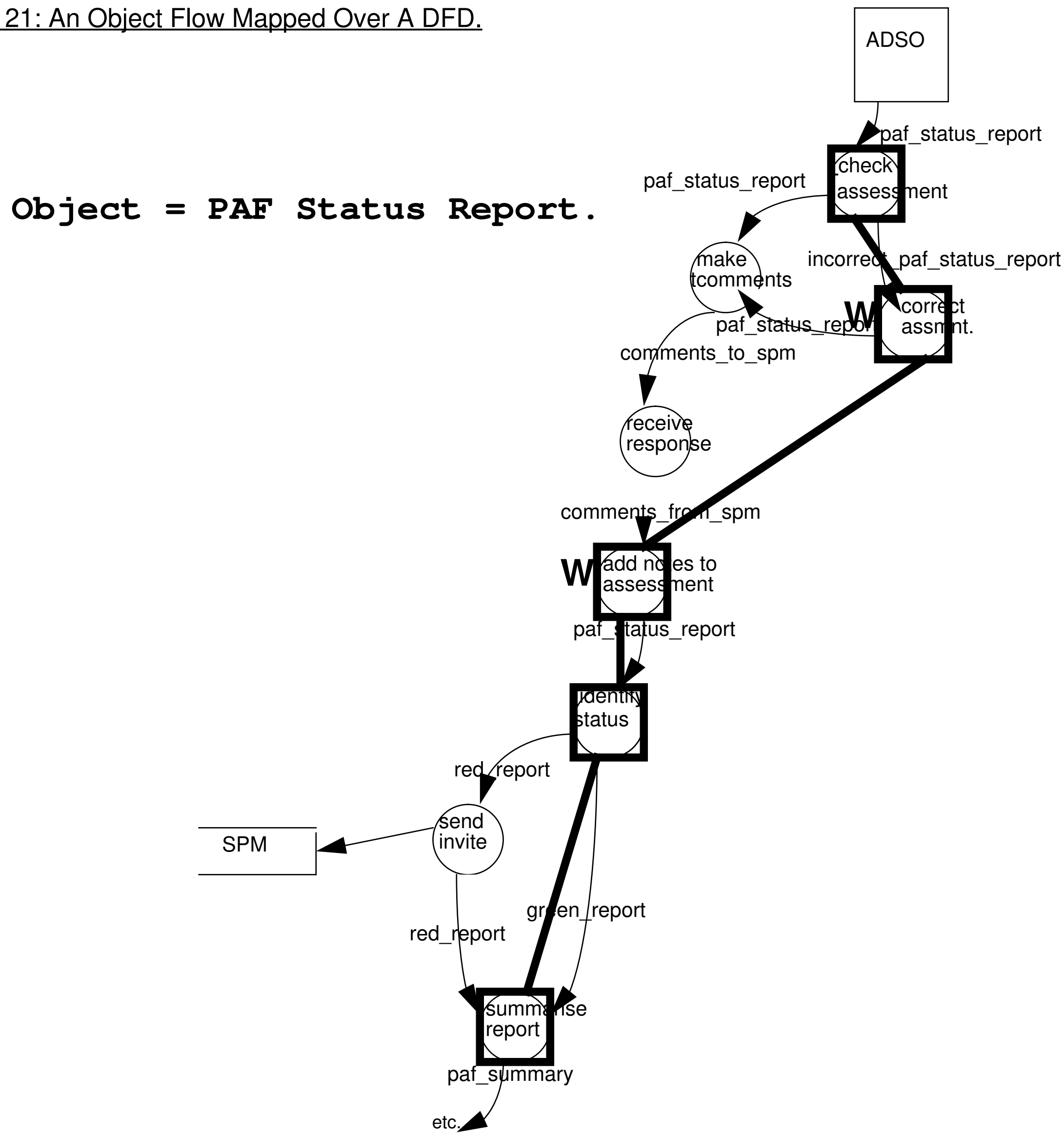
Figure 20: An Object Flow Mapped Over A RAD.



1. As with Figure 21 a little clarity is lost by mapping the object flow directly and not in the form of an overlay. Overlays are not essential but have the advantage of allowing the modeller to consider the process from a person centred (RAD) or overall perspective (DFD) and then super-imposing knowledge of how an object is affected.

Figure 21 displays an object flow mapped over a Data Flow Diagram.

Figure 21: An Object Flow Mapped Over A DFD.



The use of these notations in combination enables the modeller to assess how an object is manipulated by a process from an overall perspective.

4.2.8 The Quality Assurance Review.

Figure5 shows that the modeller will maintain a communication with the participants throughout the work on the graphical representation. It is likely that this liaison will diminish considerably as the model construction develops.

The aim of the Representation phase is to capture **what actually happens** in a process. To ensure that this has been achieved the modeller should undertake a quality assurance review. This will involve the process modeller selecting participants and describing, in as much detail as possible, how the participant does her/ his job.

The Representation phase concludes when a complete, consistent and correct multi-view model has been created[1].

1. There is an assumption that a single truth can be established. This may in fact be difficult! Different process participants may do the same tasks in different ways, or (worse) may be inconsistent about how a process operates. The orientation around the question 'what actually happens' and the provision of sophisticated modelling techniques should enable these problems to be overcome. It may be necessary to resort to participant observation and the modelling of alternatives within the same roles.

5.0 Phase Two: Refinement of the Process.

This is phase two of the method. The primary aim is to consider how the real world process might be improved. The model is used as a consistent reference point. Figure 2 shows that 'problems, goals, inconsistencies, concerns etc.' are input into this stage. These will often originate with the process participants. However, the process modeller will also have a pro-active role in identifying problems and will be responsible for the identification and implementation of all refinements.

There are two primary work areas within the Refinement stage:

• Process Analysis.

• Dialogue Input.

From whichever work area a potential refinement is sourced, it will need to be modelled and documented before it is proposed for the work environment itself. This is undertaken via a third work area:

• Implementing Change.

5.1 Process analysis.

In Checkland [] the systems practitioner undertakes 'systems thinking' in a detached environment. This approach has a parallel in the cookbook. In the Refinement phase, the modeller will explore the model that has been constructed and apply the knowledge that has been gained. This process of analysis can call upon three main resources:

• Knowledge arising from the development of the model.

• Process modelling expertise.

• Process metrics.

These resources are options. The process modeller need not apply all three but may rely on just two (one would probably be insufficient).

5.1.1 Knowledge arising from the development of the model.

In researching and constructing the model the modeller will have built up a very substantial knowledge of the process. The multi-view approach will have developed an understanding of how roles interact with entities, entities with processes etc. Furthermore, the process modeller will have substantial awareness of how participants perceive the process and how they interact with it.

Kellner describes a process modelling exercise as leading to "a substantial increase in understanding by participants and managers of the process being modelled, which is, in itself, highly valuable." He goes on to say that "carefully reviewing a model has been quite helpful in highlighting problems and areas of opportunity for process improvements."[1]

5.1.2 Process modelling expertise.

The expertise gained from previous process modelling exercises will allow the process modeller to apply intuition and experience to the analysis of the model. In

1. M.I. Kellner, 'Software Process Modelling: Value and Experience,' in SEI Technical Review 1989, p. 48.

this way certain generic characteristics of processes can be picked up. For example, the process modeller may identify certain process features or shapes which suggest inefficiency or inconsistency.

This is one way in which the cookbook recognises that the process modeller's experience and skill are resources which can be beneficially applied to a study environment. This will benefit the study environment and should help to ensure that the method is highly usable by process modelling experts.

5.1.3 Process metrics.

Process metrics are potentially a very valuable means of assessment of process characteristics and efficiency. A process metrics framework has not been developed for the cookbook. However, a number of techniques are already available for data flow diagrams. These are introduced and described in Benwood [2].

More broadly, Benwood describes metrics based upon project information which does not require the implementation of graphical process modelling techniques. Of particular relevance to the cookbook are the Project Profile Metrics. These are concerned with working practices, staff experience and staff turnover and could include information relating to the following;

- Whether or not documentation is reviewed.
- Which tools are used.
- The total number of staff working at some time on a project.
- The number of staff working on each phase of the project.
- The number of staff replaced during the development process.

These Project Profile Metrics are potentially a useful foundation for a process metrics framework for the cookbook. The metrics framework could then be developed to utilise an existing technique (s) for data flow diagrams plus new techniques for RADs and object flows.

5.2 Dialogue input.

In phase one of the cookbook when setting up a dialogue with the process participants, the process modeller was primarily concerned to uncover 'what actually happens' in the conduct of the process. In phase two the process modeller's orientation has altered and a new dialogue needs to be established around 'what the participant thinks of the process.' This will supplement any information relating to this question incidentally gained in phase one (see 4.1.1).

5.2.1 The metric of the user's view.

Dialogue inputs will enable the process modeller to draw upon the expertise of those who work in the study environment. Comments may be of the type 'Item X is always slow to come through, it holds me back,' 'I don't know why I have to sign this,' or 'this work is very tedious. What am I contributing to?' Alternatively there might be suggestions such as 'Wouldn't it be better if I sent the papers directly to management?' From this it can be seen clearly that the process modeller shall be able to respond to a variety of motivations such as efficiency and job satisfaction.

In doing so the modeller will be subjecting the process to the metric of the user's (or participant's) view. This appreciates Checkland's statement: "The component is the individual human being, and even if we depersonalize him as an 'actor' in a 'role' he will be an active participant in the phenomena investigated, attributing meanings and modifying the situation in a potentially unique way."[1]

HCI theorists and practitioners have been able to account for the success or rejection of computer systems in terms of the meanings ascribed to them by users. Process executions may be able to relate more closely to the user's meanings than traditional systems, but will still be subject to their influence. More particularly for the purposes of the cookbook, it is obvious that this system of meanings is not installed with the computer system but operates perpetually within the work environment. The task for the modeller is therefore to make the processes aware of this system of meanings and accountable to them whatever the level of technology.

5.2.2 Enriching the dialogue through cultural enquiry.

The value to the process modeller of dialogue input can be enriched through three types of 'cultural enquiry.' This is an adaptation of that devised for Soft Systems Methodology and described by Checkland and Scholes [4]. It is a particularly valuable way of identifying the meanings associated with a work environment. The levels of cultural enquiry are as follows:

- Analysis of the (process modelling) Intervention.
- Social System Analysis.
- Political System Analysis.

Checkland and Scholes report that for SSM "..it was found useful to think of an intervention in a problem situation as itself being problematical."[2] The implication of this is that the process modeller needs to understand the structure that underpins his/ her involvement in a work environment. This leads to analysis of the first type above. Structurally the intervention can be divided into three participants. The first participant 'client' is the person or persons who caused the modelling exercise to take place. By identifying the client, the process modeller will be able to make the client's reasons for causing the exercise to take place, explicit. This in turn makes the modeller able to draw upon or override these reasons in an explicit and therefore potentially accountable manner. The second participant in the intervention is the 'would be problem solver.' This participant may also be the client. As 'would be problem solver' the participant is whoever wishes to do something about the problem situation in general. Through the identification of this participant, the process modeller will be able to make the problem solver's concerns and reasons explicit. Again, this makes the modeller able to draw upon or override these concerns in an explicit and therefore potentially accountable manner.[3]

Checkland and Scholes say of Social System Analysis that ".the nature of the social system is not likely to emerge in response to direct questions. Direct questions will probably receive as responses the official myths of the situation."[4] Therefore, for

1. Checkland, 'Systems Thinking Systems Practice, 'ch 3, p 69.

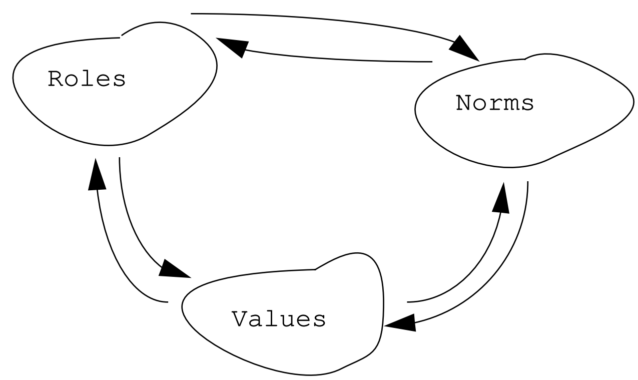
2. Checkland & Scholes, 'Soft Systems Methodology in Action,' p45.

3. This is different to SSM. For the problem solver Checkland and Scholes state "..the intervention had better be defined in terms of their perceptions, knowledge and readiness to make resources available." Appreciating the potential for conflict between client/ problem solver and problem owners, the cookbook makes the problem solver's concern's explicit so that whether or not the process modeller accepts the dominance of these concerns is also explicit (to the modeller at least).

4. Checkland & Scholes, 'Soft Systems Methodology in Action,' p49.

the second level of cultural enquiry they recommend that a file is opened "mentally or actually" and that the inferences relating to the social system in every interview or conversation are recorded in it. What this means is that the modeller will be concerned with recording details of the social system according to a simple paradigm of it. This paradigm, described in Figure 22, is made up of roles, norms and values. A role is a social position which may be defined institutionally (e.g. 'project manager,' 'engineer') or behaviourally (e.g. 'reliable team member'). Norms are the expected behaviours which characterise the role. Values are the "..beliefs about what is humanly 'good' or 'bad' performance by the role holders."[1] The paradigm is constantly updated and never completed by the process modeller as new inferences arise. This is useful in that the process modeller can chart the social currents and meanings that act within the work environment and influence it and perceptions of it.

Figure 22: SSM Paradigm For Social System Analysis.



"Each element defines, and is defined by, the others."

Roles: A position in the social system which can be defined institutionally or behaviourally.
Norms: Expected behaviours in a role.
Values: Local standards about what is good or bad performance by the role holders.

Political System Analysis is built upon an understanding of politics as a process by which differing interests reach accommodation. The key to the analysis is the question 'What are the commodities through which power is expressed in this situation?' Examples of likely commodities in this sense are access to information, membership of a work-group, intellectual authority, personal charisma, association with other process participants and the authority to write reports or minutes. The recognition of the commodities of power possessed by a participant will enable the modeller to appreciate how influential that participant is upon a process. Indeed it may be that the participant has no influence upon the process and yet may have a unique and valuable insight into its operation. Therefore Political System Analysis will enrich the modeller's insight into the nature of a process, the way it is changed, and the perspective from which participants talk about the process.

1. Checkland & Scholes, 'Soft Systems Methodology in Action', p.49.

5.2.3 The reconciliation of differing perspectives.

The net result of this dialogue input will be that the modeller has access to a wide range of expressed feelings, suggestions and concerns. On this basis the modeller may select and apply a suggestion to the implementing change work area (and hence to the process itself). Alternatively the modeller may formulate new refinement proposals. However, there are likely to be several layers of complexity attached to such decision making and there may be reasons why the modeller chooses not to adopt this role of arbiter over all concerns. For example, ideas and suggestions may contradict each other. Such contradictions can arise been different layers of an organisation (e.g. manager vs worker) or within the same layer (e.g. manager vs manager). Choosing which of the contradicting concerns should be given preference is unlikely to be an easy task. The modeller will be helped if the cultural enquiry has been carried out. For example, there may be reasons why opinions of the person identified as the client (in the analysis of the intervention) are given most weight by the modeller[1].

The modeller may choose to go further and subject contradictory concerns to a process of reconciliation. This can be achieved by adopting techniques from Checkland [3]. The modeller can bring a group of participants together and ask them to separately define their ‘root definitions’ of the process[2]. These definitions can be discussed openly[3] and an agreed composite can be assembled. This agreed composite can be contrasted with the process operation as it is understood now. This provides a new forum for the introduction of concerns and suggestions which can again be openly discussed. Through this the modeller can delicately encourage a reconciliation of perspectives.

5.2.4 Summary: The new dialogue.

Thus the new dialogue will be oriented around a new question, ‘What do you think about the process?’ Existing work-group definitions can be maintained for this, but the modeller need not necessarily speak to the same participants. The output from this dialogue may be of the type ‘..this work is very tedious. What am I contributing to?’ or may be suggestions such as ‘Wouldn’t it be better if I sent these papers directly to management?’ The value of this dialogue can be enriched through cultural enquiry.

5.3 Implementing change.

When the process modeller identifies a possibility for change in the process, this refinement should be modelled across all four views. This will involve amending the existing model according to the modelling order described in 4.2.3. Consistency checks should be made between the four views. Comparisons should then be made between the amended process model and the original. On the basis of these comparisons the modeller should clearly document how the refinement is likely to affect the real world process.

1. And if this is the case, the cultural enquiry will make the modeller aware of his/her own reasoning & decisions in this respect.

2. i.e. what they think the process is about.

3. Open discussion does not imply equal discussion. People play roles according to social and political circumstances. Again, cultural enquiry will help the modeller account for this.

If a refinement then appears to contribute to the process environment, the modeller should suggest its implementation and provide the documentary forecast of how the refinement will affect the real process.

6.0 Process Management.

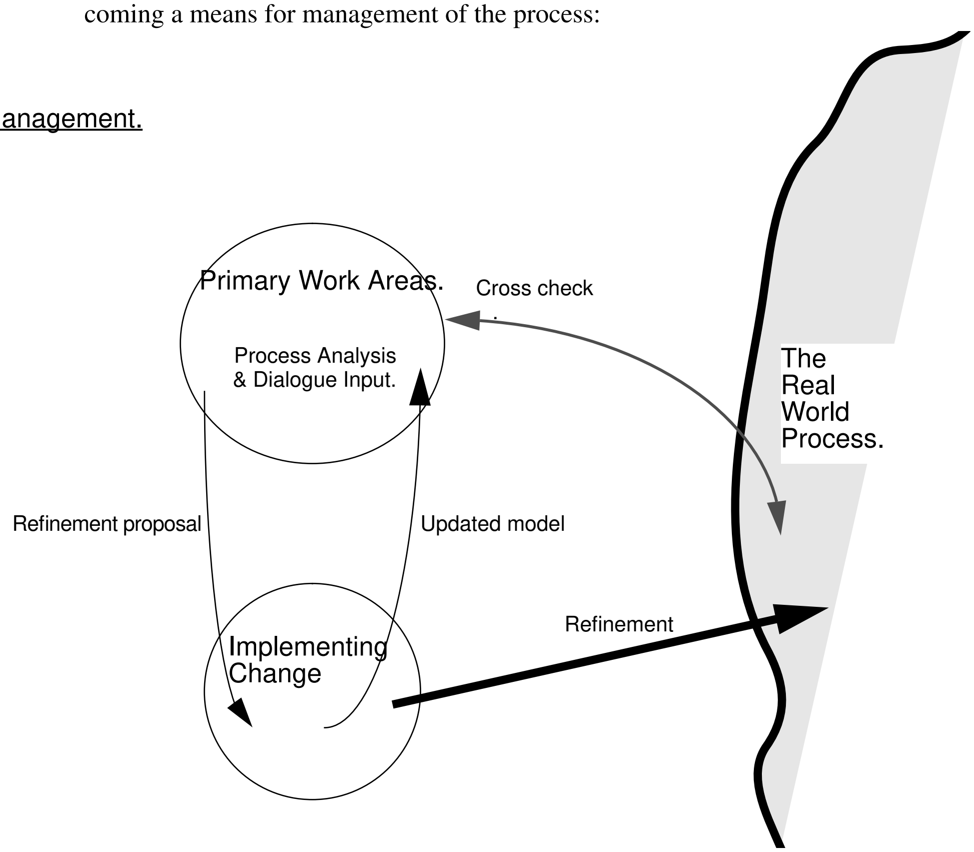
By applying the work areas of the refinement stage cyclically, the cookbook can be extended to provide a means for process management. The maintenance of this cycle necessitates the re-introduction of features from the Representation phase.

The process maturity framework is an optional structure for process management. It has been adapted from Humphrey's original [7], to apply to processes more generally than just within the SEI remit and to fit alongside cyclical refinement of the processes themselves. It is likely to be useful when the process is characterised by uncertainty, when efficiency is the overriding or primary goal and there are critical change issues determined by the requirements for the products themselves. Therefore, the process maturity framework is likely to be most useful when considering software processes or other complex products.

6.1 Cyclical refinement.

Figure 23 describes how the Refinement phase can be made cyclical thereby becoming a means for management of the process:

Figure 23: Process Management.



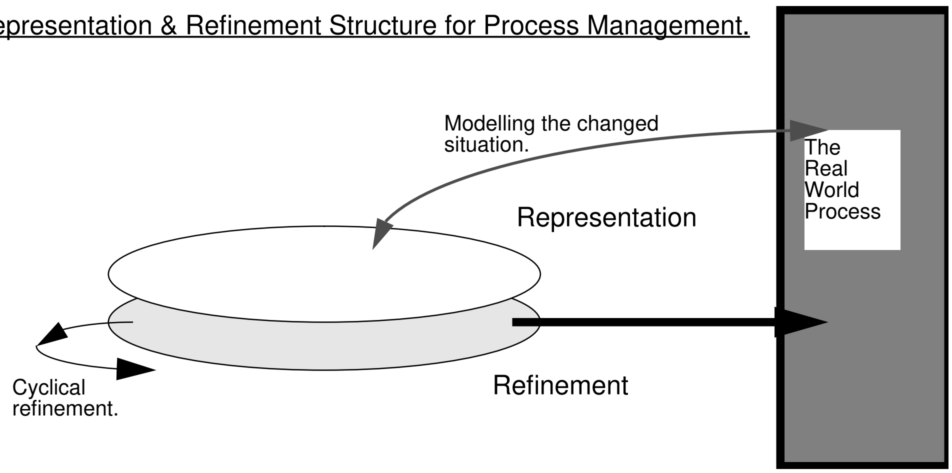
This works in the following way. Any single refinement proposal is a product of process analysis and/ or dialogue input (known collectively as the primary work areas). The refinement proposal is modelled in the Implementing Change work area and its likely impacts are documented. If the process modeller is satisfied with the proposal it can be applied to the real world process. The residual of the Implementing Change work area is then an updated model of the process. This can be re-applied to the primary work areas. Hence, the modeller can analyse it again and the participants can input their feelings about the refined process.

As the model is updated so the real world process should evolve to accord with it. This means that in theory the process model and the real world process should always be in mutually consistent states. However, when dealing with processes of any complexity this is unlikely to be the case. The model and the reality will evolve differently for a number of reasons. These may include participants' reactions to process refinements and unexpected impacts of change[1]. As a consequence, a cross-check to ensure consistency is required.

6.1.1 Cross checking; model to reality.

The need to cross check the model and its referent reality requires a partial re-invocation of the Representation phase. This is concerned with 'what actually happens' in the work environment. Figure 24 represents this.

Figure 24: Using the Representation & Refinement Structure for Process Management.



It is not sufficient to simply model the part of the process that was affected by the refinement. The knock-on effects of a process change are not necessarily inclined to be local or easily visible. Therefore, it is necessary to maintain a surveillance additional to a local cross-check. Where this is likely to be difficult, for example if the process is large, then a good case is made for the adoption of the process maturity framework.

In other cases a cross check can be provided by re-modelling the work processes of the work-group local to the refined part of the process. This should take place a significant time after the refinement has been introduced and when the participants report that they have 'settled down' with the new refinement or otherwise 'done as best they can' with it. Once again dialogue should initiate the modelling process.

A wider surveillance can be accomplished at the same time by selecting other parts of the process to re-model. These could be partially selected at random but are likely to involve different work-groups.

6.1.2 Using the cycle for management.

Through the application of this cycle, the cookbook becomes a means for the management of processes. In particular it is a source of information about the state of a process, it allows for the simulation and control of change to a process, and thereby

1. Unexpected impacts of change will be minimised by thorough application of all stages of the cookbook method.

provides for the evolution of a process. In short, by maintaining the cycle throughout the lifetime of the process the manager gains the degree of control of the process necessary to enable quality management techniques to be thoroughly applied.

6.2 Process maturity framework.

The maturity of the process can be advanced by use of the process maturity framework. This has been adapted from Humphrey's original [7]. It is concerned with the refinement of the **management of the process** rather than directly with the process itself. It is therefore different to the cookbook in its orientation and its likely usefulness is noted above at 6.0.

There are five levels to Humphrey's [7] process maturity framework. They are as described below.

1. Initial - This level "could properly be called ad hoc, and it is often even chaotic. At this stage the organization typically operates without formalized procedures, cost estimates and project plans." [8]

2. Repeatable - "The organization has achieved a stable process...by initiating rigorous project management of commitments, costs, schedules and changes." [8]

3. Defined - "The organization has defined the process as a basis for consistent implementation and better understanding." [8]

4. Managed - "The organization has initiated comprehensive process measurements and analysis, beyond those of cost and schedule performance." [7]

5. Optimizing - "The organization now has a foundation for continuing improvement and optimization of the process." [8]

The knowledge gained during the Representation phase should enable the modeller to classify the process according to this paradigm. Organisations at the first four levels can be advanced through the institution of devices recommended by Humphrey. In many cases, use of the cookbook will in itself provide the device needed for a particular advancement.

6.2.1 The initial level.

Organizations at the initial level can be improved by the institution of basic project controls. The following will be needed for the advancement of most organizational processes:

- Project Management.
- Management Oversight.
- Quality Assurance.
- Change Control.

These project controls can be instituted through the following devices:

- A plan should be developed to control schedules and resource commitments for all but the simplest projects.
- Management approval and review should be instituted for all plans.
- An assurance organization charged with ensuring work is done the way it is supposed to be done. This should generally be of "about 5 to 6 percent the size of the

development organization" and possess "an independent reporting line to senior management" [7].

• Change control requires that where changes have to be made they are introduced in a managed, orderly way. Use of the Refinement phase of the cookbook will ensure that this is the case.

6.2.2 The repeatable level.

Organisations that have reached the repeatable level of maturity have established commitment control but, according to Humphrey, are unlikely to be able to cope with changes in circumstances. Three examples of changes that represent the highest risk to organizations at this level are:

• New tools and methods that affect how the process is performed and thus destroy the relevance of the 'intuitive historical base' upon which the processes are founded.

• The requirement to develop a new product will nullify the experience upon which the existing process has been built.

• Major organizational changes are unsupported and can be very disruptive. For example, a new manager has "..no orderly basis for understanding what is going on and new team members must learn the ropes through word of mouth."

•An organisation at the repeatable level can be advanced to the defined process through the following:

• The establishment of a process group. For software organisations Humphrey recommends that the process group should be about 1 to 3% of the size of the development organisation. Where the product is less complex and more easily specified, it is reasonable to assume that a figure at the lower end of this range will be sufficient. In general, the establishment of a process group will mean that at least one person has a full-time assignment to work on the process rather than the product. If the cookbook is being used for process management, then by implication this condition will have been met.

• The establishment of a process architecture that describes the technical and management activities required for proper execution of the process. This can be achieved by model building in the Representation phase.

6.2.3 The defined process.

An organisation at the defined level has, according to Humphrey, "...achieved the foundation for major and continuing progress." However, he states that: "As powerful as the defined process is, it is still only qualitative: There is little data to indicate what is going on or how effective the process really is."[1] The institution of process metrics is required to advance an organisation to the managed level. For more information about how this can be done see 5.1.3 and Benwood [2].

Overall, Humphrey specifies the following requirements for the advancement of an organisation at the defined level:

• The establishment of a minimum set of process measurements in order that the quality and cost parameters can be identified for each process step.

1. W.S. Humphrey, 'Characterizing the Software Process: A Maturity Framework', IEEE Software March 1988, p76.

• A process database should be set up and maintained. Cost and yield data should be central to this thereby facilitating process quality and productivity analysis.

• Resources should be assigned to ensure that the database is maintained to a high level of quality and advice is given to those who will benefit from the data.

• A quality assurance group should track each project against its quality plan and inform senior management when targets are not being met.

6.2.4 The managed process.

At the optimizing level, an organisation will focus upon the optimization of its processes through quantitative examination. In order to achieve this, it is required that the metrics established at the managed level are tightly defined and adequately supported. Thus, whilst the organisation at the managed level will have already secured substantial quality benefits, the organisation at the optimizing level will continue to realise these benefits in a cost-effective and maintainable fashion.

Therefore to advance from the managed process level an organisation satisfy two requiremnts:

• The automatic gathering of process data should be supported to improve upon the ommissions and errors likely to occur where process data is manually collected.

• This data should be used to analyze and modify the process.

The satisfaction of these requirements is likely to be a complex matter. Some progress towards their satisfaction has been achieved by Benwood in ‘Process Quality Report’ and ‘Measurement Requirements for a Process Support Tool’[1].

1. Refreences:

H. Benwood, ‘Process Quality Report,’ Support Technology Focus Project, British Telecommunications 1991.

H. Benwood, ‘Measurement Requirements for a Process Support Tool,’ Support Technology Focus Project, British Telecommunications 1991.

6.2.5 Summary: the process maturity framework.

The use of the process maturity framework is summarised in Table 2:

Table 2: The Process Maturity Framework.

Level.	Features additional to the previous level.	How the cookbook contributes...
5. Optimizing	1. Automated & maintainable process metrics.	
4. Managed	1. Process metrics. 2. Resources (staff & database) to collect & apply these metrics.	The Refinement phase incorporates process metrics (1).
3. Defined	1. A process group. 2. A process architecture that describes the technical & management activities required for the execution of the process.	Using the cookbook for process management implies (1). Model building in the Represent-ation phase can contribute (2).
2. Repeatable	1. A plan. 2. Management approval & review of plans. 3. An Assurance Organisation. 4. Change control.	The Refinement phase of the cookbook ensures change control (4).
1. Initial		

7.0 Appendix One: Further Development Of The Cookbook.

The following are among the possible developments of this cookbook:

• The development of a framework for process metrics.

• The enhancement of the relationship between the cookbook and the Process Maturity Framework or a similar (new) structure.

• Exploration leading to the achievement of automatic code generation from the cookbook's multi-view model. This will enable a process enactment to be coded directly from the model for execution on a technology such as PSS. Processes will then be capable of being automated 'naturally' from their analysis. Currently, the most advanced technology of this type is probably that developed by Rowles and described in [14]. This uses a CASE tool and a PML code generator to produce enactments from a graphical model.

• The development of an analytical level of modelling specifically to assist automatic code generation. This could conceivably include logic proofs for consistency.

• The development of tool support for the graphical techniques. In particular thew person centred process charts could be related to a database that ensures the consistency of the other modelling formalisms.

• General developments such as the development and enhancement of optional features to allow the cookbook to be less prescriptive and offer more choice to the modeller.

8.0 Appendix Two: RAG Case Study Abbreviations.

The following abbreviations relate to the RAG case study which was undertaken as part of the development of the Cookbook:

ADSO : Assistant Divisional Support Officer.

DSO : Divisional Support Officer.

PAF : Project Action Forum (also used to denote reports intended for this forum, e.g. ‘paf_status_report.’).

PM : Project Manager.

PMR : Project Manager’s Report.

SPM : Sub-Project Manager.

9.0 References.

[1] C. Ashworth, M. Goodland; **SSADM - A Practical Approach.** McGraw-Hill Book Company (UK) Ltd, 1990.

[2] H. Benwood; **Process Measurement Report.** Support Technology Focus Project, British Telecommunications 1991.

[3] P. Checkland; **Systems Thinking Systems Practice.** John Wiley & Sons, 1981.

[4] P. Checkland, J. Scholes; **Soft Systems Methodology In Action.** John Wiley & Sons, 1990.

[5] I. Evans; **Process Modelling Techniques.** Support Technology Focus Project, British Telecommunications 1991.

[6] I. Evans, K. Folan, N. Jussab, P. Kawalek; **Process Modelling Case Study.** Support Technology Focus Project, British Telecommunications 1991.

[7] W.S. Humphrey; **Characterizing the Software Process: A Maturity Framework.** IEEE Software, March 1988.

[8] W.S. Humphrey; **Managing The Software Process.** Addison Wesley 1989.

[9] M.I. Kellner; **Software Process Modelling: Value and Experience.** SEI Technical Review, 1989.

[10] L.A. Macauley, C.J. Fowler, M. Kirby, A. Hutt; **USTM: A New Approach to Requirements Specification,** Journal of Interacting with Computers, Butterworths, vol2 no1, 1990.

[11] J. Maresh, D. Wastell; **Process Modelling And CSCW: An Application Of IPSE Technology To Medical Office Work.** in Diaper et al (eds) Proceedings of INTERACT '90, Elsevier, 1990.

[12] R.J. Mortiboys; **Quality Management for the 1990s** in B.G. Dale, J.J. Plunkett (eds); **Managing Quality.** Philip Allan, 1990.

[13] P. Layzell, P. Loucopoulos; **Systems Analysis and Development**, 3rd Edition. Chartwell-Bratt, 1989.

[14] J. Rowles, P. Butler, J. Searles; **Process Development Handbook.** STL/608/00149 Issue 1, STC Technology Ltd 1991.

[15] P. White; **Process Modelling and Computer Supported Cooperative Work**. IOPT/35 Issue 1, IOPT 1991.

10.0 Bibliography.

C. Ashworth, M. Goodland; **SSADM - A Practical Approach.** McGraw-Hill Book Company (UK) Ltd, 1990.

H. Benwood; **Process Measurement Report.** Support Technology Focus Project, British Telecommunications 1991.

P. Checkland; **Systems Thinking Systems Practice.** John Wiley & Sons, 1981.

P. Checkland, J. Scholes; **Soft Systems Methodology In Action.** John Wiley & Sons, 1990.

P.B. Crosby; **Quality Is Free.** McGraw-Hill Book Company, 1979.

B.G. Dale, J.J. Plunkett (eds); **Managing Quality.** Philip Allan, 1990.

I. Evans; **RT7313 Case Study: High Level Process Model.** Support Technology Focus Project, British Telecommunications, 1990.

I. Evans; **Process Modelling: A Method.** Support Environment Focus Project, British Telecommunications 1990.

I. Evans; **Process Modelling Techniques.** Support Technology Focus Project, British Telecommunications 1991.

I. Evans, K. Folan, N. Jussab, P. Kawalek; **Process Modelling Case Study.** Support Technology Focus Project, British Telecommunications 1991.

B.G. Glaser, A.L. Strauss; **The Discovery Of Grounded Theory: strategies for qualitative research.** Aldine De Gruyter, 1967.

A.W. Holt; **Diplans: A New Language for the Study and Implementation of Coordination.** ACM Transactions on Office Information Systems, Vol. 6, No. 2, April 1988.

W.S. Humphrey; **Characterizing the Software Process: A Maturity Framework.** IEEE Software, March 1988.

W.S. Humphrey; **Managing The Software Process.** Addison Wesley 1989.

A. Hutt, N. Donnelly, L.A. Macauley, C.J. Fowler; **Describing The Product Opportunity: A Method Of Understanding The Users' Environment**. in Diaper & Winder (eds) People and Computers III, Proceedings of the BCS HCI specialist group, Cambridge University Press, 1987.

M.I. Kellner; **Software Process Modelling: Value and Experience.** SEI Technical Review, 1989.

P. Layzell, P. Loucopoulos; **Systems Analysis and Development**, 3rd Edition. Chartwell-Bratt, 1989.

S. Linkman, J. Searles, P. Westmacott; **A Process Model Development Method.** IOPT/ 16 Issue1, IOPT 1991.

L.A. Macauley, C.J. Fowler, M. Kirby, A. Hutt; **USTM: A New Approach to Requirements Specification,** Journal of Interacting with Computers, Butterworths, vol2 no1, 1990.

J. Maresh, D. Wastell; **Process Modelling And CSCW: An Application Of IPSE Technology To Medical Office Work.** in Diaper et al (eds) Proceedings of INTERACT '90, Elsevier, 1990.

G. Morgan; **Images Of Organization.** Sage Publications, 1986.

D. Nicholls; **Introducing SSADM - The NCC Guide.** National Computing Centre, 1987.

J. Rowles, P. Butler, J. Searles; **Process Development Handbook.** STL/608/00149 Issue 1, STC Technology Ltd 1991.

SD Software Technology Centre; **CORE - The Method.** Software Technology Centre Pembroke House, Pembroke Broadway, Camberley, 1989.

H.T. Smith, P.A. Hennessy, G.A. Lunt; **The Activity Model Environment: An Object-Oriented Framework For Describing Organisational Communication.** Proceedings of the 1st European Conference on CSCW, EC-CSCW 1989.

A. Sutcliffe; **Jackson System Development.** Prentice Hall International (UK) Ltd 1988.

P. White; **Process Modelling and Computer Supported Cooperative Work**. IOPT/35 Issue 1, IOPT 1991.

P. White, D. Wastell; **IPSE Technology As A Basis For Implementing CSCW Systems**. Proceedings of INTERACT 1991, Elsevier.