

**ESPRIT IV - Technologies for Business Processes  
Project 26.786**

# **Proflex**

**Professional Management of Flexible, Customer Driven,  
Responsive Processes**

---

## **Final Report**

Authors: CLRC - RAL

Status: Draft

Date: 23 June 2000



## **Executive Summary**

This report describes the **Proflex** flexible process support tools resulting from a research and development project partly funded by the EU Esprit IV programme.

The software tools allow SME's to flexibly characterise their business processes, and support delegation of tasks to staff in the organisation, reminding managers when tasks have, and have not been completed. The tools are more flexible than others in the market which require a complete, detailed description of the processes before they can be first used. The **Proflex** approach only requires those parts of the process to be defined which a manager wants to be, and others can be clarified later as desired.

The **Proflex** software client uses the MS-Exchange server and interacts with the MS-Outlook client to provide the integrated business solution to flexible control of business processes.

The benefits to SME's of using this software have been evaluated and are reported in detail.

It is planned to market the **Proflex** software tools as a commercial software product.



## Contents

1	INTRODUCTION	1
2	MOTIVATION	2
3	BACKGROUND	3
4	OBJECTIVES	5
5	DEVELOPMENT METHOD	6
6	REQUIREMENTS	8
7	ARCHITECTURE OVERVIEW	13
7.1	Project Manager	13
7.2	Intray	13
7.3	Store	13
8	MODULE DESIGN	15
8.1	The <b>Proflex</b> Manager	15
8.1.1	Activities	15
8.1.2	Project data	16
8.1.3	Skeletons	17
8.1.4	Enactment	17
8.1.5	Monitoring	17
8.2	The Actor's Inbox	18
8.3	The Background Agent	20
9	EXAMPLE APPLICATION RUN	21
10	EVALUATION	23
11	CONTRIBUTION TO THE STATE OF THE ART	25
12	BENEFIT TO THE PARTNERS	26
12.1	User Partners	26
12.2	Technical Partner	26
13	PROJECT MANAGEMENT ISSUES	28
14	RESULTS AVAILABILITY AND THEIR EXPLOITATION	29

<b>14.1</b>	<b>Project Results</b>	<b>29</b>
<b>14.2</b>	<b>Exploitation Methodology</b>	<b>29</b>
<b>14.3</b>	<b>Competition</b>	<b>30</b>
<b>14.4</b>	<b>Exploitation Window</b>	<b>30</b>
<b>14.5</b>	<b>Pricing</b>	<b>30</b>
<b>14.6</b>	<b>Products</b>	<b>30</b>
<b>14.7</b>	<b>Consortium exploitation objectives</b>	<b>30</b>
<b>14.8</b>	<b>Exploitation partner</b>	<b>31</b>
<b>14.9</b>	<b>Business Plan</b>	<b>31</b>

## Figures

Figure 1: The marketplace for process support tools .....	4
Figure 2: <i>Proflex</i> Overview.....	14
Figure 3: The <i>Proflex</i> Manager.....	16
Figure 4: Linking two activities using the PM .....	16
Figure 5: Inbox with new activity .....	18
Figure 6: The Actor form - <i>Actor</i> tab sheet .....	18
Figure 7: The Actor form - <i>Results</i> tab sheet.....	19
Figure 8: A newly loaded project .....	21
Figure 9: The Activity Form in use to revise an activity .....	22
Figure 10: The updated state of an activity after revision .....	22
Figure 11: <i>Proflex</i> positioning in the marketplace.....	25





# 1 INTRODUCTION

---

This is the final report on the **Proflex** project to develop software tools to support the Professional Management of Flexible, Customer Driven, Responsive Processes which was partially funded under the EU Esprit IV programme through grant 26786 from 1998-2000.

The purpose of this report is to provide those interested in the project, or the project topic with enough information to decide if the project results require detailed investigation by contacting the project partners. Details of those participating in the project are listed in Appendix 6 to assist such further contact. Other Appendixes also list publications and documents produced in the project that may be available from project partners to aid in such investigation.

The user organisations participating in this project are two automotive suppliers, the Belgium sheltered workshop Waak situated in Kuurne and the German sheltered workshop Lebenshilfe Werkstatt in Munich together with Alcatel IKO Kabel from Sweden, an important cable producer. The supplier, Rutherford Appleton Laboratory from the UK is developing the technical solution for **Proflex**. With the exception of IKO all partners have already been successfully co-operating in other relevant ESPRIT projects (e.g. We-Do-IT, HICOS). IKO has a long record of co-operation with Waak.

The overall project objective is to increase the competitiveness of the user organisations, especially the sheltered workshops, and of the technology suppliers by elaborating business best practice processes for the support of flexible, customer driven, responsive processes and finally for the sheltered workshops to retain and create work for their people with disabilities.

The report is structured to provide background motivation and requirements for flexible process management tools, the architecture and results produced in the project to meet those requirements, and the benefits that arise from using such tools as evaluated during the project, and as the user organisations in the project saw them.

## 2 MOTIVATION

While there are a growing number of structured workflow systems to support the routine high volume business processes typical of large companies, SMEs do not have the same level of support for the smaller, more volatile processes which typify their work. SMEs, and in particular supply chain SMEs, depend more heavily on their front office processes. These processes, dealing with a small number of major customers, are normally carried out by highly skilled senior management using their knowledge and intuition as much as, or more than, pre-defined procedures to achieve their goals. In many cases, major problems are caused by the need to respond within short timescales to changing customer requirements.

The **Proflex** project was conceived directly as a result of the experiences of our user partners in managing such processes. One such process, initiated by a customer requiring a prototype component to be developed, was examined in detail and the basic system requirements and a potential outline architecture were established. This became the basis for the project proposal.

This one example taken from the daily set of actions of a production business shows the need to cope with:

- high importance to the business - a successful prototype can mean a big production order, while a failed bid can mean not only loss of an order but a marked decrease in customer confidence;
- coping with "impossible" customers' requests (e.g. asked to supply prototype even when delay on delivery from suppliers exceeds time available); giving rise to continuous adaptation of the process and need for continuous control;
- often long elapsed time;
- frequent bursts of activity over short time periods;
- many actions, of varying urgency;
- need to tie incoming information with the current process;
- need to re-define the process at short notice, in both minor and major ways;
- need to delegate and monitor aspects of the process to staff.

The process itself thus needs to be explicit and very flexible, as changes in customer needs can mean major re-writes of the project plan. The focus here is on responsive procedures that cannot be completely structured from the very beginning, but need to be amended depending on the actual state of internal tasks and external events. One of the main problems at this phase is the design and control of the process by the process owner, either on management or on team level, monitoring progress, ensuring deadlines are met (or shifted), and assessing the impact of external events, such as a change of customer requirements.

The above mentioned example of a flexible business process shows that the organisation lacks those additional flexible tools, based on standard technologies and providing compatible interfaces, to enable them to easily monitor and progress the process. No existing system, or set of systems, could cope with all the process support requirements.

### 3 BACKGROUND

The user requirement is for a tool supporting tasks where the co-ordination of many people in several organisational units, usually under tight time constraints, is concerned. There are three main software categories that are relevant: workflow management systems, project management software, and computer-supported cooperative working (CSCW). No one of these satisfies the requirements of the project's users, but each has something to contribute:

- a) **Workflow:** strong focus on business process
- b) **Project management:** good representations of project *activities*, esp. with respect to time
- c) **CSCW:** for co-operation issues, social/organisational as well as technical

#### Business Process Support Systems - Workflow Management

It is inevitable that many large organisations, and a number of smaller ones, have sought IT support for the execution of the business process itself - workflow systems. One definition of a workflow system is: "Software designed to improve business processes by providing the technology enabler for automating these aspects of the workflow:

- routing work in the proper sequence;
- providing access to data and documents required by the individual work performers;
- tracking all aspects of the process execution."

There are a very large number of workflow management systems on the market. A number of distinct classes of workflow systems can be identified:

- Production:** designed to automate business-critical transaction-oriented processes (e.g. loan arranging, insurance claims, accounting).
- Collaborative:** for business-critical processes that are not transaction-oriented (e.g. technical document production, software development).
- Administrative:** software to automate administrative work driven by paper forms (e.g. expense claims, purchasing, budgeting).
- Ad hoc:** provide routing and tracking of routine office work that is based on unstructured information (e.g. information distribution, review/approval).

Possibly the major drawback of the workflow system installation and maintenance is the relation to the supporting IT infrastructure; much time and effort is required to provide a satisfactory interface to existing IT systems, which can often constrain the effectiveness of the business process they are intended to support. One impact is that changing business processes, while possible at the process definition level, becomes difficult due to the inherent inflexibility of the IT tasks to be undertaken as part of the process.

#### Project Management Software

Project Management Software generally supports the process of planning, monitoring and evaluating projects. In the planning phase, worksteps are related to required effort and to human or technical resources requested in order to achieve the objective of the workstep. Parallel worksteps may as well be defined as subsequent ones, where the respective effect on scheduling, resource and cost planning is taken into account. Different views on the data and functions are available supporting the planning process. The following is a list of functionalities generally covered by products on the market:

- the related time horizon and deadlines of the project as a whole or of single worksteps are displayed in GANTT-Diagrams,
- the procedure of tasks to be performed can be taken from PERT-Diagrams,
- the critical path is automatically calculated and visualized, as well as buffer times,
- availability of resources is taken into account providing resource specific calendar functions
- excessive load of resources is automatically detected, and consequent adaptation of the planning is supported.

The monitoring phase requires a permanent update of the progress of every single workstep and its related cost. The effort required to keep the project plan up to date in order to allow evaluation of

usually time and cost critical projects may thus be fairly high. The project manager needs to collect the complete information on used resources and related cost and pass this information on to the PM software. This task of gathering and entering the data, needed to form the basis for continuous evaluation, is not supported by PM software.

Keeping track of the proceeding and collection of relevant data for project evaluation is thus a task where the project management is not supported by PM software, since respective views are only available when the information has been entered manually. The integration of workflow and CSCW technologies with PM aspects thus provides a much more powerful environment for the project manager under severe pressure.

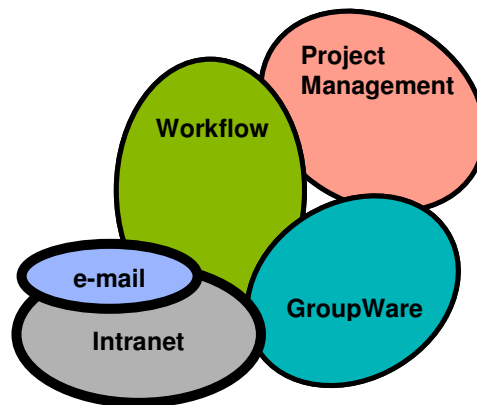
## Groupware

Groupware is software that facilitates people working together, helping overcome location and time differences. From the perspective of support for flexible processes, some of the key features of groupware are:

- Viewing and working on the same information as colleagues, simultaneously.
- Ease of sending information to colleagues, via email, and of relating series of messages to each other, via discussions.
- Access to information from any location in the company.
- Archiving of shared materials, including current status, previous modifications, etc.

Lotus Notes and Microsoft Exchange are the dominant groupware systems currently on the market. They focus on the sharing and distribution of information, but only provide rudimentary facilities for managing 'process'. Even where routing is provided, this is achieved by individual scripts attached to "documents", forwarding the document to pre-specified individuals.

The current marketplace can be summed up in the following diagram:



**Figure 1: The marketplace for process support tools**

As shown by the gap between the areas, no existing system provides all the facilities to handle flexible processes adequately. Although each of the different systems (Workflow, Project Management and Groupware) are being extended towards each other, narrowing the gap between them, it is unlikely that they will fully address responsive customer-driven processes due to the underlying requirements of their base area.

## 4 OBJECTIVES

---

The main objective of the project was to provide effective IT support for such responsive, highly dynamic and complex processes. These processes are completely customer driven. There are three main areas where support had to be provided:

- Determining, selecting and scheduling actions, planning subsequent activities.
- Executing the process.
- Keeping control of the process.

Given the initial customer contact, the activities to be undertaken have to be **determined, selected, and scheduled**. The initial activities can usually be easily identified, but the **planning for subsequent activities** is likely to be rather sketchy. While there are some things in common between different processes, there will be sufficient differences that each case will essentially be unique. In spite of this longer-term vagueness, the process still has to be **"executed"**, activities delegated to the appropriate staff, information collected and progress monitored, especially against tight deadlines.

One of the main problems at this phase is the **control of the process**, knowing how it is progressing, ensuring deadlines are met (or modified) and assessing the impact of external events, such as a change of customer requirements. Looking at each of the phases identified, the major sub-objectives for **Proflex** were to:

- support the planning/scheduling of activities, based on a rough template that is fleshed out/changed as the process progresses;
- support the enactment of the process, including delegation and co-ordination of tasks, delivery of necessary information to colleagues;
- keep track of progress, with particular regard for deadlines;
- allow "on-the-fly" changes to the process, without losing control;
- encourage development of a set of "standard" templates or skeleton processes.

The software was to be a highly graphical tool running on desktop/portable PCs. Different views of the process would be supported, depending on the changing requirements of the process owner, e.g. hierarchical view of the process structure, a flowchart-like view for process flow design and a list detailing all activities. The tool would interface to and coordinate existing project/process management applications at the user sites rather than replace them.

The user groups addressed by **Proflex** were on the one hand middle management staff who are usually responsible for determining, selecting and scheduling actions, as well as keeping control of the resulting process. On the other hand end users carrying out a process that is run by **Proflex** will work with the system, either on the design level to detail the local worksteps, or on the execution level, using the integrated tools. The monitoring functionality should be usable by every end user to keep track of the progress of projects in which they participate.

The **Proflex** system had to:

- build upon existing tools (e.g. a Groupware infrastructure),
- add its own value-adding components (e.g. graphical design tool, runtime component for executing responsive workflows, library for process skeletons) and
- be usable by non-IT staff.

## 5 DEVELOPMENT METHOD

The **Proflex** system development is very much an evolutionary process. There are constant iterations around the design, implement, test and review cycle. Feedback on the implementation informs, where necessary, further design or re-design. Given our delivery platform it was important to prototype various aspects of the **Proflex** system to ensure an optimum design. At regular intervals the potential **Proflex** end users have the opportunity to test and evaluate the embryo system. The potential end users of the system have also been closely involved in specifying and documenting the user requirements and taking those requirements into the next stage of the design process.

The **Proflex** project recognised the need for a development methodology to help structure the development ensuring that user requirements were met. Although an early decision was made to use UML (Unified Modelling Language) this alone was not enough. There are several development methodologies around which use UML and we chose to modify one such methodology, the Unified Process.

The Unified Process has four main sections: requirements capture, analysis and design, implementation and test. There are also four phases during the project lifetime; inception, elaboration, construction and transition. Elements from the four sections are undertaken during the various phases. Each phase contains one or more iterations around the cycle comprising the four sections. In **Proflex** we took a prototyping approach which means there is an even distribution of design and implementation for successive iterations. At the end of each section, and sometimes before then as well, there was consultation between the users and developers to ensure that the decisions made in that section were agreed by both groups.

The Together/J tool was used to support the whole analysis, design and implementation process by the technical partners. It provides support for all the UML diagrams which were required and has good support for reverse engineering code into class and sequence diagrams.

A glossary was kept up to date with definitions of key words and phrases within **Proflex** which helped all partners to have a common understanding of the key terms.

**Requirements capture** was a highly interactive process between the users and technical partners. The initial requirements document (Deliverable 1.1) was expanded and refined (Deliverable 1.3.3). Requirements were gathered by the user organisations talking to appropriate members of their staff, some of whom were participating in the project. In-depth discussions were held between the user organisations and technical partners to refine and clarify the requirements found. Feedback from the users about the current implementation and proposals for new functionality were captured as detailed requirements for each release of the system.

### Analysis and design

The System Architecture gives compact end-user oriented overview on the basic architecture of **Proflex**. From this, IT-literate users should be able to acquire the necessary background for a fuller understanding of the system and therefore be in a position to best evaluate and exploit **Proflex**.

The Architecture document (Deliverable 2.1) is complemented by the Detailed System Design documentation (Deliverable 3.1), giving a system oriented description of **Proflex** intended for implementers, maintainers and developers. The Architecture document specifies the existing software, which the system is built on. We choose Microsoft Exchange and Outlook98 as the tools to store the projects and to communicate between the managers and the actors. This was primarily because the user organisations required a system, which would fit with their existing software base.

The analysis and design, after producing the architecture for the system, explored a representative sample of use cases (guided by the process diagrams in the requirements document) and extracted the main objects which are part of the high level design. These were then translated into classes and interactions by developing sequence diagrams, comprising the lower level design. As each new major

piece of functionality was developed the same process was undertaken to design the required enhancement to the system.

### **Implementation**

The implementation is guided by the coding standards adopted by the project (Deliverable 4.1.1). The overall delivery schedule was agreed early in the project (Deliverable 1.3.2) and modified as required during the project.

The main implementation language is Java. Visual Basic is also used by the activity forms used by Outlook. As Microsoft barely supported the Java interface to Outlook, we had many problems during the implementation, some of which required us to rework the design. In the BackgroundAgent we had to use C++ in places in order to be able to receive events from Exchange at all. The graphic view of the **Proflex** Manager uses an ActiveX control called AddFlow, which can be used run-time royalty free.

The following tools were used to support the implementation: Together/J, VisualSourceSafe as the shared project repository, Visual J++ and Visual Studio (for the C++ development),

The implementation is currently at version 3. It requires additional software to be installed for it to run. The implementation is split into a **Proflex** client for the manager, a **Proflex** client for an actor and a **Proflex** server. The **Proflex** server requires Exchange Server 5.5 to be installed on the machine. The **Proflex** Manager (client) requires that Outlook 98 is installed on the client machine. The actor client requires Outlook 98 is installed on the client machine.

Productisation is an important part of our implementation. Versions of the release have been produced in English, French, German and Swedish. In order to make the installation procedure simpler two install shields were built. These do as much of the installation as possible, including installing the software. Installation of Outlook forms is not included, as it was not found to be possible within the install shield in the time available.

### **Testing**

The test section of the project verifies the entire system. First each use case was tested separately to verify that its participating classes worked together correctly. Then the system as a whole was tested with use-case descriptions as input to these tests. The version of the system was then delivered to the user organisations for installation and further testing. More specific details of the testing strategy for **Proflex** can be found in Deliverable 1.3.2.

Results of the users tests of one version informed part of the implementation and refinement for the next version.

## 6 REQUIREMENTS

The **Proflex** system evolved out of the need for a solution to the many problems of managing flexible processes. The requirements for the system also evolved during discussions and walkthroughs and continued to evolve following the early system deliveries. What follows is a summary of the requirements for the **Proflex** system using a series of roles and the operations that should be available to each.

The users of the **Proflex** system can assume various roles. These have been identified as:

- Manager
- Actor
- Viewer

In the manager role the **Proflex** user can create projects and design or re-design activities within these projects. Activities can also be added or re-designed in an existing project. The manager can also send out activities for enactment. As a manager it is very easy to slip into the role of viewer when first reviewing an existing project, checking which activities have been completed or started and noting those that may have been rejected. When the **Proflex** user takes action they assume the role of manager again.

The actor role is provided for the **Proflex** user who receives the activities to perform and has the ability to accept, start, complete, delegate or reject the activity. A user in the actor role may be allowed to take on the manager role for a certain activity and hence be able to refine the activity and create sub-activities that can be sent out for action to someone else.

The viewer role provides monitoring and managing information to both the Manager and the Actor. In addition the notion of the Line Manager is considered as one of the users of the **Proflex** system in the viewer role. The viewer typically needs to be able to check the state of the projects and activities and to obtain the answer to the question "Where do I have problems?".

These three roles have been identified so as to provide a means of logically grouping functionality.

### **Proflex User: Manager - Overview**

The first functions are those at a high level that the manager role will need in order to manage a **Proflex** project. The process begins with opening the **Proflex** Manager and deciding whether it is a new project that is to be worked upon or an existing one. If it is an existing one then that project needs to be selected and loaded into the **Proflex** Manager. Otherwise an empty, new project is created. Some of the actions that are suggested for a project may be able to be performed without the project being loaded. For example if a relationship with a customer goes sour one might want to list all projects relating to that customer and suspend them all. These actions are not included separately as any action that can be performed on a selected project can be performed on a loaded project.

So with either a new project started or an existing one loaded there is then the list of the main things that can be done with a project. The first category is concerned with working *within* the project, i.e. on the activities within the project. This is expanded in its own process map, following. The second category concerns working *on* the project as a whole and includes actions such as configuration, suspension, re-starting, cancelling".

When current work on or within the project has been completed for the time being the project is closed away. Another project can now be opened, a new project created or one can leave the PM.

"Delete Project" is required for deleting and removing all information about the project from the system. Once a project has been deleted the PM is empty and so a new project can be created, an existing project loaded or the PM can be exited

Instead of closing a project one can close and exit **Proflex** Manager or with no project loaded just exit the **Proflex** Manager. This is for when work on the projects has for the time being been completed.



## Proflex User: Manager - Work on activities

The set of actions that fall within the category of working within the project all relate to activities. An activity can typically be added, configured and sent out. Once an activity is sent out it will only go if all the required information has been provided and the conditions specified are met. The configuration of an activity includes describing the activity, setting the required actor, giving the expected results and providing the required timings. Dependencies can also be set.

Activities may be suspended. Results need to be saved and work stopped. When the time comes a suspended activity can be re-started.

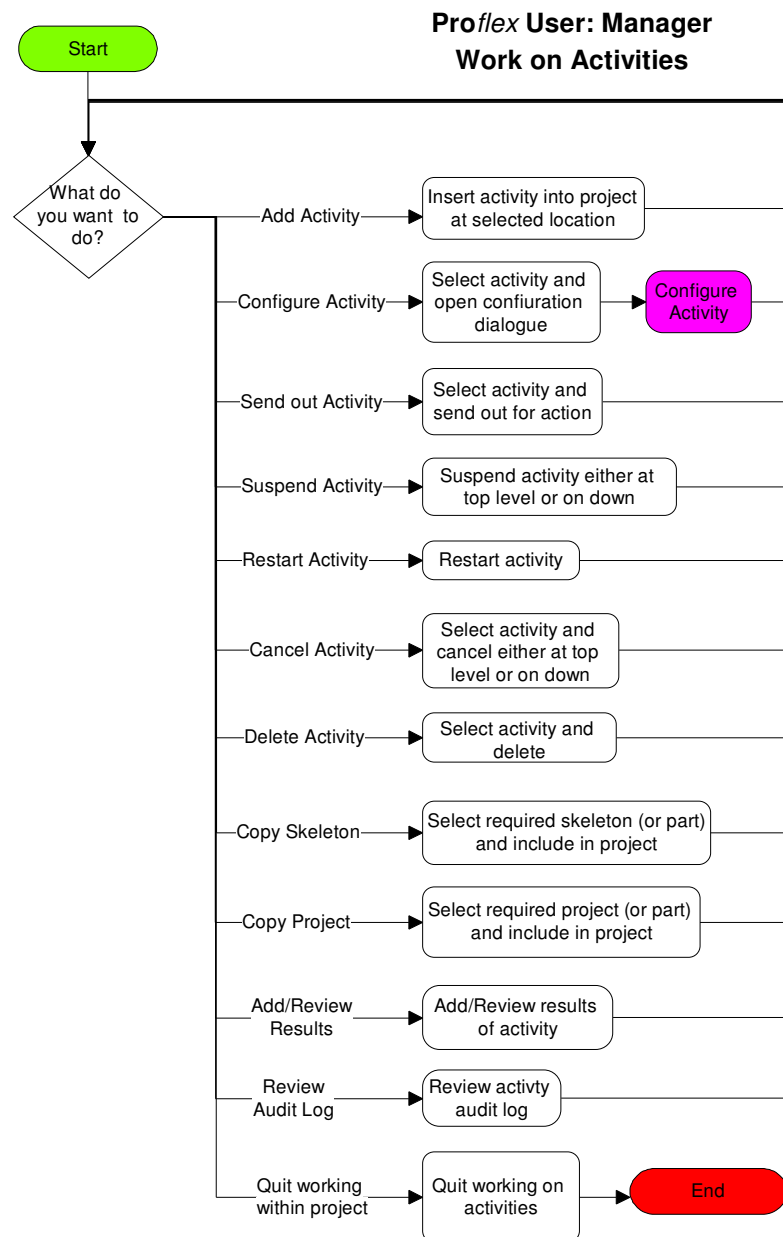
An activity may be cancelled in which case the activity will not be re-startable. Any partial results will be saved with the activity.

Deleting an under design activity will remove it from the project. Deleting any other sort of activity would only show the activity as being *cancelled*.

A project can include skeletons of activities, which are stored sets of activities that are expected to be used in more than one project. Using skeletons prevents managers from re-inventing the wheel each time and perhaps forgetting certain essential steps. Parts of an existing project can also be selected and copied into a project. Copying in a skeleton or part of a project is a useful way of creating a number of activities which can then be tailored to the specific project. Once a skeleton has been included the activities are treated the same as new activities.

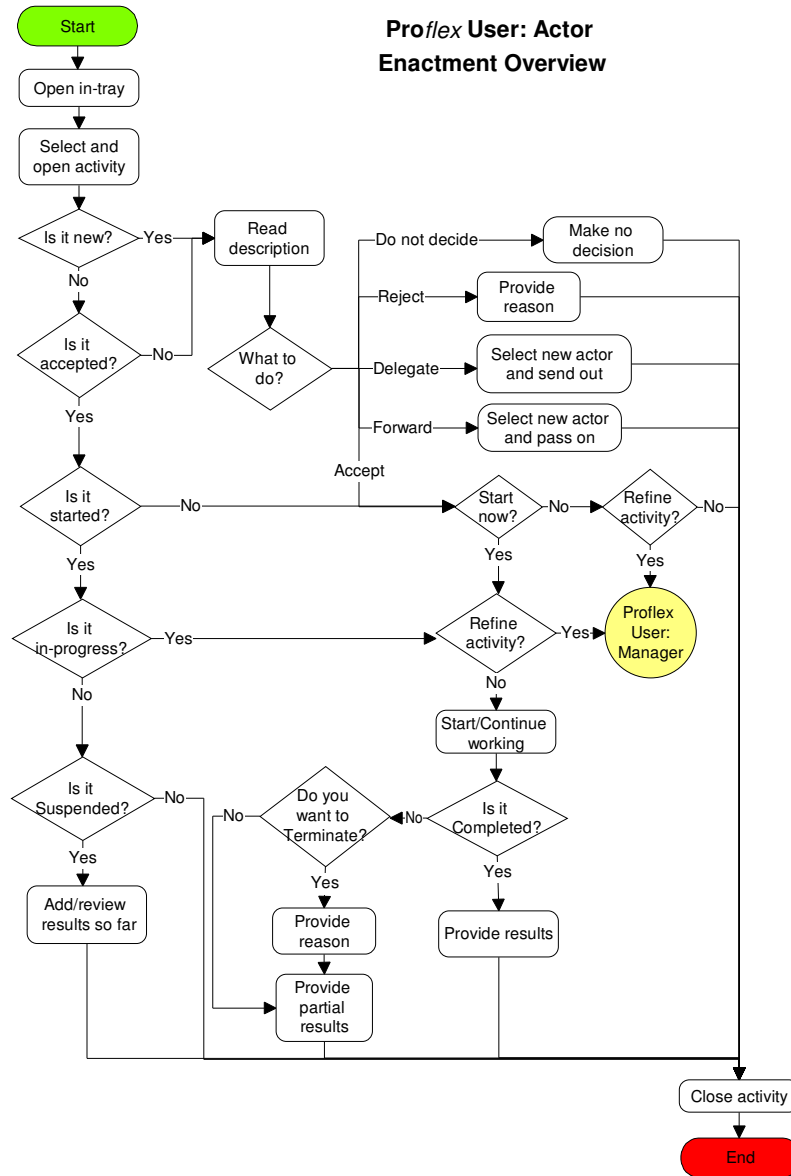
Once an activity has been sent out the manager becomes interested in the results from that activity. Even before an activity has been completed an actor may have attached partial results to the activity. The manager is able to review these results from the activity as seen in the **Proflex** Manager. Once an activity is completed it is only the manager who can add additional results.

Each activity has its own log, capturing information about the changes and events that happen to the activity. For example, creation date, date sent out and to whom, date accepted, date started, date completed. A manager needs access to this log.



## Proflex User: Actor - Enactment Overview

The actor role of the **Proflex** User is concerned with the operations required within the Inbox. Here the activities from **Proflex** arrive in the user's Outlook inbox. Activities that are new will be marked in some way so that the user can immediately see any new work that has arrived. Activities can be opened so that the description of the work can be read, following this a decision is taken as to whether to accept, start, reject, delegate or forward the activity. The decision can of course be deferred until a later date in which case the activity is closed and left in the in-tray.



If the activity is being rejected a reason should be added to the activity before signalling reject.

If the activity is being accepted then another decision is required as to whether to start work on the activity straightaway. If the activity is being accepted but not started immediately then signalling accept allows the project manager to see that the activity has at least been accepted. The activity remains in the in-tray but is no longer marked as being a new item.

If the activity is being accepted and started at once, or if a previously accepted activity is now being started, start is signalled. Once again the project manager will see the new state of the activity in the **Proflex** Manager. There are now two options, either the activity is begun in earnest or the activity can be further refined into sub-activities using the **Proflex** Manager. If refinement is required then the actor now takes on the role of manager for this activity and the **Proflex** Manager is opened with the relevant activity loaded.

Once the activity is completed the results must be provided and attached to the activity and complete is signalled. At any time whilst working on the activity partial results may be added to the activity. Whether the project manager is able to see these partial results is dependent upon the permissions or access rights set.

If an actor has accepted and started an activity but then finds he can not complete it he must terminate it, giving a reason and any partial results.

Access will not normally be available to activities that have been completed, rejected or terminated. For completed activities it may be possible to review the results and the comments added when the activity was signalled as completed. Any additional results which had been forgotten would need to be sent to the manager for addition to the activity.

An activity may only be suspended by the activity manager. Notification will arrive that work must stop on the activity, however any partial results should be attached to the activity and hence they may be reviewed at a later stage.

Delegation is an option when receiving an activity. A different actor is selected and the activity is sent out to that actor. Delegation only moves the doing of the activity to the new actor not the responsibility for getting it done. That remains with the original actor and so the activity remains in their in-tray. Once the delegated actor has completed the activity, the original actor should check the results before completing their own activity.

If an actor receives an activity and wishes to pass on that activity to another actor to actually complete the task then the forward action allows for this. The activity is removed from the original actor's in-tray when the activity is forwarded. The new actor is responsible for completing the activity.

Some of the more frequent or usual actions may be available from the in-tray without the need to open the activity.

## **Proflex User: Viewer**

Viewer can be any of the users of the **Proflex** system. The Manager will want to monitor his projects and the outcome of the actors' activities. The actors will need overviews of their own activities and possibly their place in the overall project. Line Managers do not need to play an active role within **Proflex** Projects in the sense of enacting activities themselves but as they have been appointed to be responsible for a certain number of employees who in their turn may have been appointed as actors by the manager of a project, they will wish to monitor these employees, their output and work load in particular. Hence Line Managers can take on the role of viewer.

There are four issues a state of the art view or report should be covering:

- **selection** : what do we want to see ?
- **sorting** : how do we want the data to be sorted ?
- **lay out** : how do we want the data to be presented : tree, list or graph view ?
- **print destination** : do we want the result to appear on the screen and/or on paper ?

Along with the phased implementation of **Proflex**, new requirements for monitoring and viewing were encountered. What follows is a summary of initial requirements.

## **Proflex User: Viewer - Manager's viewpoint**

Managers require an overview of the projects they manage. The first **selection** a manager will have to make is whether he wants to see all projects he is managing as a whole or all activities within these projects. In both cases, the manager will be able to choose one or more selection criteria. When choosing projects as a whole, he will be able to select projects of a certain status, of a certain category, with a certain keyword, where a certain actor is working on it, with a certain Project Manager. When choosing viewing activities within his projects, the manager will be able to choose from the same selection criteria as for projects as a whole, with the following differences: selecting activity manager rather than project manager; selecting all activities waiting for a predecessor; specifying how many levels down to show; selection by name. Of course it will be possible to select all projects or all activities in one simple operation.

Once the manager has selected what he would like to see, he has the option to decide how he wants the information to be sorted. **Sorting** actually means grouping data with the same properties and presenting them to the end user in a predefined order. The manager will be able to sort on many of the fields of an activity/project, e.g. state, keyword, category, actor. It should also be possible to have projects or activities listed by costs or by average cycle-time. These options will more likely be taken

when viewing projects that have been completed or terminated in order to learn from experiences from the past and to give answers to questions like :“where were the bottlenecks of this project ?”, “Why were the overall costs of this project that high ?”

After sorting comes **Lay out** and presentation decisions.

The viewing performed by the manager may result in further work in the Manager role

### **Proflex User: Viewer - Actor’s viewpoint**

The big difference between monitoring by a manager and by an actor is the fact that the actor is only interested in seeing his activities. This may be activities from all projects he is working on or only the activities within a certain project. This is the first selection that needs to be made.

The actor will then be presented with the same **selection** criteria as the manager has, except for the fact that he will not be able to specify a certain actor as he is only allowed to see his own activities.

Also the same **sorting** facilities will be offered to the actor as to the manager. Only sorting by actor will not be possible for the reasons stated above. Sorting by cost or by average cost time is also not foreseen, as actors are mostly only interested in getting the work done on time. Therefore, there will be the possibility to sort by estimated cycle time; a time calculated by **Proflex** on the basis of the start date and estimated end date. The latter filled in by the actor to give the manager an idea of how long the enactment of a certain activity is likely to take.

The List view is the most appropriate **lay out** to present the data to the actor as he is focusing on his own activities. On the other hand, it is not clear yet whether the actor might not be helped with a tree view or graph view as well when data is required from within one specific project. Default display on the screen is also applicable here as well as printing out on paper will be foreseen.

The viewing performed by the actor may result in “Enactment” in the Actor role or more work in the Manager role.

### **Proflex User: Viewer - Line Manager’s viewpoint**

Line Managers are people responsible for the work of other people. They therefore have the same needs of monitoring as a project manager, even if they are not actively involved in the projects. They should be given the same facilities as managers, be it somehow in a restricted way.

The big difference between viewing as a manager and as a line manager is the fact that the line manager is only interested in seeing the activities in which his staff are involved. This may be activities from all projects his staff are involved in or only the activities within a certain project.

For the line manager the distinction between all or one projects is the first **selection** he has to make. The same selection criteria as for the manager will be available, except for the fact that he will not be able to define how many levels down the activities should be shown. This has no sense here as he is interested in all activities no matter at what level they occur.

**Sorting** facilities are similar to those of the actor, except that it will be possible for the line manager to sort by actor. Ordering the activities by estimated cycle time is interesting for the Line Manager to have an idea of the (coming) workload of his staff.

As in the case of the actor, only the List view makes sense as **Lay out**. It is still under discussion whether the Line Manager should also be offered the Tree and Graph views and this only in the case of activities within one specific project.

Default display on the screen and paper print out will be available as before.

The Line Manager will not interfere in projects himself but will only cause changes through the actor and/or manager. Of course a Line Manager may also be a project manager and/or an actor.

## 7 ARCHITECTURE OVERVIEW

A project is a set of activities related to some business goal(s). An activity is a task to be carried out to help achieve this goal. Activities can be hierarchical, i.e. they can contain sub-activities.

The **Proflex** System consists of the following 3 major components:

- the **Proflex** Manager (or PM),
- the **Proflex** Inray (or inbox),
- the **Proflex** Store.

### 7.1 PROJECT MANAGER

The **Proflex Manager** supports the design, enactment and monitoring of a project.

There are 3 different views available to organise and configure the project — or the activities that define your project:

- the **Tree View** where the project is structured into a hierarchy of sub-activities/sub-projects
- the **List View**, where the most important details of the activities, e.g. the actor, the activity, title, due date, etc. are tabulated.
- the **Graph View**, where the relationships between activities, e.g. activity sequence/dependencies, are shown as a network.

Any or all of these 3 representations can be used to design and manage a **Proflex** project. They support adding/deleting activities, naming activities, linking activities together, etc. The different Views are designed to support different interaction styles appropriate to different modes of working, project types, project development phases, etc. In addition, Views can be customized to show only the information relevant to the manager's current task, whether trying to get an overview of a project or focusing in on the details of a small set of activities.

All the details of an activity can be accessed using the **Manager Form**, accessible from any view. This holds the information needed by the actor e.g. detailed instructions, due date, attached documents etc. as well as information used by the manager, e.g. actor name, current status, previous/next activity, etc.

The design and configuration information entered in the PM will be collected in the **Proflex Store**. The Store also holds status information for all the activities currently in progress.

### 7.2 INTRAY

The **Proflex Inray(s)** present to the actor their personal list of activities. The list view shows some main details, typically the project title, the activity title and due date. Again, what is displayed in the list can be customized. Further details of a selected activity are accessed via the **Actor Form**. This dialogue corresponds to the Manager Form of the PM and shows the details that are necessary to perform the required action. It also allows the actor to indicate when they have finished the activity.

### 7.3 STORE

The **Proflex Store** acts as a central repository for projects and activities, ensuring all users have access to up-to-date information - providing they have the necessary permission. The Store is proactive, i.e. it monitors projects, passing activities to the actor's Inray, flagging overdue activities to the manager, etc.

The following diagram shows the main **Proflex** components.

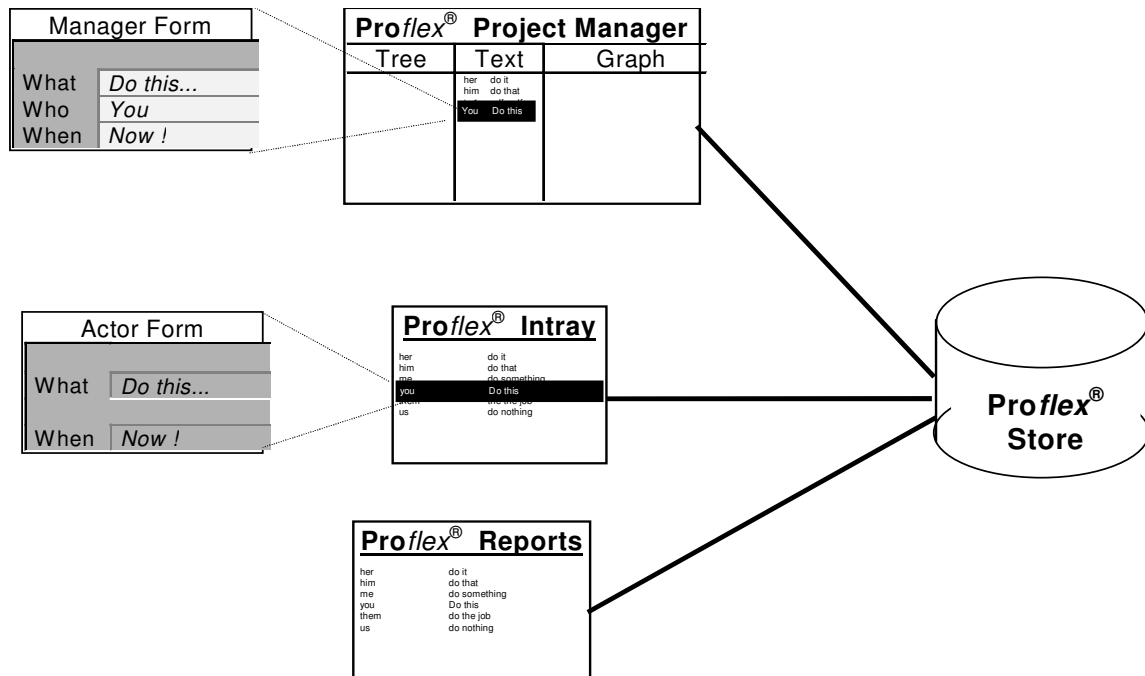


Figure 2: Proflex Overview

## 8 MODULE DESIGN

---

The system is composed of three main software modules: The **Proflex** Manager and associated forms for the manager, the forms for the actor and the Store, managed by the Background Agent.

### 8.1 THE PROFLEX MANAGER

The PM is a tool to allow managers to design, enact and monitor a single project. Projects may be loaded from those held in the **Proflex** Store, or a new (empty) project can be created.

There are three different views available to organise and configure the project — or the activities that define your project:

- the **Tree View** where the project is structured into a hierarchy of sub-activities/sub-projects,
- the **List View**, where the most important details of the activities, e.g. the actor, title, due date, etc. are tabulated,
- the **Graph View**, where the relationships between activities, e.g. activity sequence/dependencies, are shown as a network.

Any or all of these three representations can be used to design and manage a **Proflex** project. They support adding/deleting activities, naming activities, linking activities together, etc. The different Views are designed to support different interaction styles appropriate to different modes of working, project types, project development phases, etc. In addition, Views can be customised to show only the information relevant to the manager's current task, whether trying to get an overview of a project or focusing in on the details of a small set of activities.

All the details of an activity can be accessed using the Manager Form, accessible from any view. This holds the information needed by the actor e.g. detailed instructions, due date, attached documents etc. as well as information used by the manager, e.g. actor name, current status, previous/next activity, etc.

The design and configuration information entered in the PM is collected in the **Proflex** Store. The Store also holds status information for all the activities currently in progress and the library of template projects, also known as skeletons.

#### 8.1.1 ACTIVITIES

An activity is any task to be carried out that the manager would associate with the project. The main attributes of an activity are:

- what has to be done,
- who will do it,
- by when should it be completed,
- what information is required to carry out the task and
- what information is generated.

New activities can be inserted at any point (other than preceding a completed activity) in any view, and their main attributes set. Most information visible in any of the views can be modified directly. In addition, all aspects of an activity (including linking to other activities) can be viewed and modified through a **Manager Form** (see the following section on the client for more information). This is the main way to include detailed instructions for the actor, attach documents to the project or activity, set status flags, etc.

Activities, of course, need not be simple tasks. In many cases, an activity can itself be broken down into simpler activities. To accommodate this, the Tree View and Graph View allow an activity to be "expanded" and new activities placed inside it. Collapsing the activity hides its contents. This enables an overview of the complete project to be seen, without cluttering the screen with unnecessary detail.

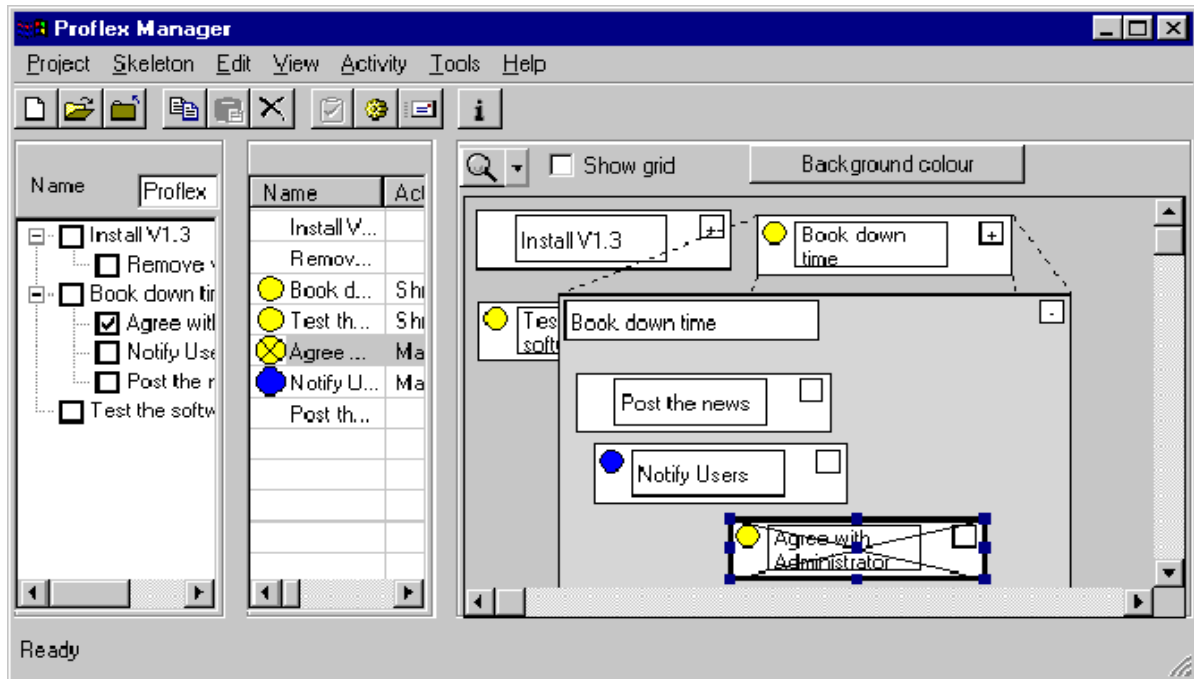


Figure 3: The *Proflex* Manager

A second sort of relationship between activities is execution sequencing, where the order in which the activities are carried out is important. Linking activities together captures this. Of course, any activity can be the source or destination of more than one link. Links, shown by lines between activities on the Graph View, can connect to the ends of the activity box, implying no overlap between the activities, or from the top or bottom of the activity box.

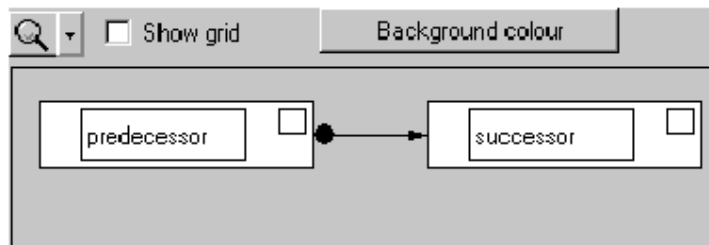


Figure 4: Linking two activities using the PM

### 8.1.2 PROJECT DATA

Most projects, when initiated, will have some information associated with them, and will accumulate more as their activities are completed. Most activities will require information (over and above the instructions to the actor) and will change that information or generate new information.

In *Proflex*, information falls into two categories, attachments (typically letters, reports, forms, etc.) and specific data items (such as a component cost, a delivery date, etc.). An attachment is a named reference to information held elsewhere. If it is to an on-line document, the document is copied into the *Proflex* area (if not already there). Otherwise the textual reference provided by the actor is used. Each activity has a list of attachments and a list of data items where all the activity data can be entered and viewed. It is envisaged that a project will also have its own data as well as that of the individual activities.



Each Manager Form has an area for input data (information to be passed to the actor) and return data (information modified or generated by the actor). Copying a project data item or attachment to the input area will cause the information to be made directly to the actor. Similarly, when the actor completes the return data area of the Actor Form, that information will be copied back to the project data area (i.e. the values in the project data area are updated). Of course, even after copying to the project data area, it will still be accessible via the (now read-only) Manager Form for the completed activity.

### **8.1.3 SKELETONS**

While many of the projects **Proflex** has to handle will be unique, frequently they will have some similarities to previous projects or parts of a project. The similarities will be exploitable by reusing old projects or project parts, via the copy and paste mechanisms (not yet implemented). The pasted activities can then be modified as necessary. This can be taken a stage further, by deleting the instance specific details from an old project (or part thereof) and saving the basic structure as a skeleton or template. Once a collection of these skeletons has been built up, project design can become much quicker and less prone to error. The organisation also can build up a collection of useful project designs.

### **8.1.4 ENACTMENT**

Identifying and specifying an activity is only the first step in achieving the project goals. Once sufficiently configured (the minimum is an actor, the instructions and a due date) the activity is ready to be enacted. First the manager has to specify the actor who will carry out the task. Then the activity is enacted, i.e. "sent out" to that actor. The state of the activity is updated, from "under design" to "sent out", and this is reflected by colouring the activity in the PM views. Of course, the manager may not need to explicitly "send out" an activity - it might have been automatically "sent out" by **Proflex** when the preceding activity in a linked chain was completed. Normally the PM is directly connected to the **Proflex** Store, so the activity details in the Store are immediately updated and the actor gets the activity in their Inray (see below).

The PM tracks the status of the activity. It reflects this by colouring the activity differently in the PM views to reflect the state changes made by the actor, e.g. "in progress", "done". Thus the manager is always aware of the latest state of the activity. For audit purposes, once an activity has been enacted, the manager can no longer delete it. They can only cancel it (or more properly, ask the actor to stop). Furthermore, if the manager makes a change to an enacted activity, the actor is immediately. Once the actor has completed the activity the manager is no longer allowed to modify the activity details, only append extra notes.

As a project progresses, an audit trail can be recorded. The information captured depends on the needs of the particular organization, some requiring a detailed log so that the process could be re-enacted, others only wanting a synopsis of the activity or project outcome. At present any audit information has to be manually included by the manager. Audit information will be kept with the project information in the Store and would be archived with the project.

### **8.1.5 MONITORING**

Since the PM always displays the current state of activities, the manager can use it to monitor the current status of running activities or look at the audit trail of finished activities. Limited multi-project comparisons can be achieved by having several PMs open simultaneously.

## 8.2 THE ACTOR'S INBOX

The Actor's Inbox is a sub-folder of the *Proflex Intrays* folder with the 'display name' of the actor. Microsoft Outlook is used to view the Inbox. The view of the Inbox may be different to that shown here as it can be customised per user or by organisation. This customisation is part of Outlook. The basic functions described should still be available but things may look different.

The Inbox shown in Figure 5 displays a list of activities from a variety of projects. The information displayed about each activity includes the subject, the sender, the state of the activity, the date the activity was received and whether it is under revision, suspended or cancelled. In addition there are the standard Outlook features of showing importance and whether there are any attachments.

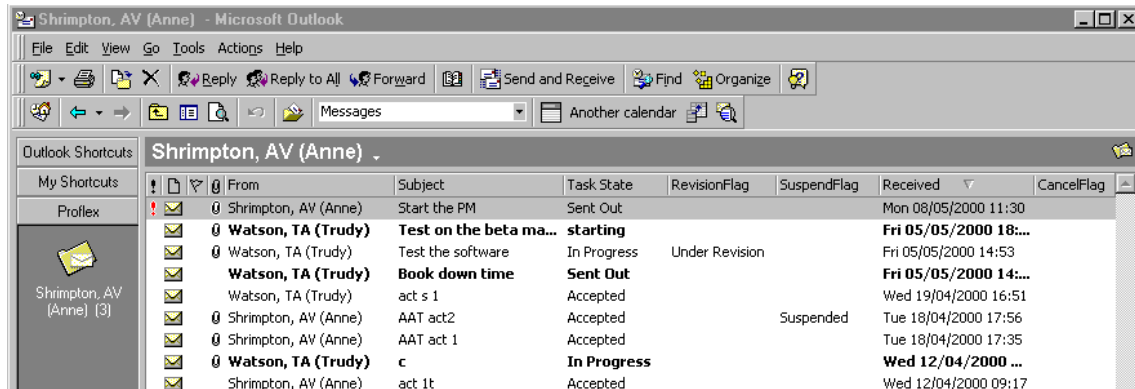


Figure 5: Inbox with new activity

Any activity can be opened by double-clicking on it with the left mouse button. This will display the Activity Form with the first, "Actor", tab sheet on view (see Figure 6).

**Start the PM**

File Edit View Insert Tools Actions Help Form

Accept Start Complete Reject Save & Close Close

Actor Attachments Results Audit Log All Fields

State: [ Sent Out ] Refine Activity

Subject: Start the PM

From: Shrimpton, AV (Anne)

Deadline: Fri 05/05/2000 08:30 Importance: High

Instructions:

Start the Proflex manager (PM) and become accustomed to its many features. There is a Reference Manual and a Users' Guide available (attached).

Expected Result: A certain level of competence, ready to give demos.

Figure 6: The Actor form - Actor tab sheet

This form is similar to that used in the PM but the Manager tab sheet is not displayed. The sender's name replaces the Actor name on the Actor tab sheet. Note that the Task State is shown as *Sent Out* as the Activity has been delivered to the Actor's Inbox. You can read the instructions and expected results for the activity, as provided by the Manager.

Under the toolbar on the form there are six action buttons, Accept, Start, Complete and Reject plus Save & Close and Save. These are the actions that are allowed for this activity whilst it is in its present state. An activity would normally either be accepted or rejected. If it is being rejected then a reason must be given for the manager and the activity would then be removed from the Inbox. Once an activity has been accepted it would normally proceed through being started and then completed with results being added during this process. The expected completion date, the reason for rejection or other comments to the manager and results are provided on the *Results* tab sheet of the Actor form shown in Figure 7. There is also an area for private comments the actor may wish to make a note of during the execution of the activity.

The screenshot shows a software window titled "Start the PM". It has a standard menu bar with "File", "Edit", "View", "Insert", "Format", "Tools", "Actions", "Help", and "Form". Below the menu is a toolbar with icons for printing, saving, and navigation. A row of buttons includes "Accept", "Start", "Complete", "Reject", "Save & Close", and "Close". Below these are tabs for "Actor", "Attachments", "Results", "Audit Log", and "All Fields", with "Results" currently selected. The form contains several text input areas: "Expected Completion Date" with the value "Mon 08/05/2000 08:30", "Comments to Manager" with the text "Sorry for the delay, unforeseen withdrawal of service.", "Results" with the text "Everything worked well, ready to give demos", and "Comments: (Private)" with the text "Had some problems with installation, worked through Users' Guide, tried my own examples, created suitable examples for demos.". A note at the bottom states "Attachments can be added on the Attachments tab sheet".

**Figure 7: The Actor form - Results tab sheet**

Once an activity has been accepted or started it can no longer be rejected. Instead if the activity can not be completed the option offered is to terminate it. This implies as much work as possible has been done but the task has not been completed as expected. Interim results should be added and a comment made to the Manager as to why the termination is being done.

Any changes in state of the activity made by the Actor using the action buttons will be reflected in the Manager's view of the project. This allows an up-to-date view of a project to be maintained.

If an activity is suspended, cancelled or being revised then only appropriate actions will be available to the Actor.

### 8.3 THE BACKGROUND AGENT

The Background Agent performs the necessary work within the **Proflex** Store in order to support the actions requested by both the Manager and the Actor.

The **Proflex** Store consists of:

- the Project storage located on the MS Exchange Server, implemented by a special Exchange folder usually called *Proflex Activities*.
- the Actor Inboxes containing the Activities forwarded to a certain Actor. This is implemented by a set of folders, one for each Actor. They are located as sub-folders of an Exchange folder usually called *Proflex Intrays*.
- the Skeleton store located on the MS Exchange Server, implemented by a special Exchange folder, usually called *Proflex Skeletons*.

When a manager "sends out" an activity it is the Background Agent that makes the new entry in the Actor's Inbox. When an Actor has added results and completed an activity it is the Background Agent that ensures those results are copied back to the original activity in the project store and that the completed activity is removed from the Actor's Inbox.

The original Background Agent methodically steps through each project, checking each activity, to see whether anything needs doing and then steps through each Inbox, checking each activity to see whether anything needs doing. This is obviously time-consuming but this functionality was foreseen, for example if deadline alerts are to be notified, or if expected completion dates being later than deadlines are to be notified. So using this polling algorithm was a sensible place to start. A timer is provided with the Background Agent which allows this trawl through the activities to be undertaken regularly.

Changing activities within the **Proflex** Store generates "events". These events can be captured and reported to an Event Background Agent. This allows the Event Background Agent to check only those activities that have been modified and to perform the background processing much faster. It uses the main processing functions of the original Background Agent but instead of a timer or a user initiating a long trawl through activities, a specific "changed" activity is identified for checking. Subsequent processing only takes place if necessary. Activities are "sent out", accepted, completed, etc. much quicker with this Event Background Agent.

## 9 EXAMPLE APPLICATION RUN

Let us assume a manager needs to review his project and find out whether there are any problems. After starting the **Proflex** Manager, the required project is opened.

The status bar at the bottom of the **Proflex** Manager window displays messages indicating what is happening until the Project is displayed.

Once the project has completed the loading, the Project name is highlighted and no activities are selected. The little plus signs in the tree view indicate those activities that have sub-activities. The sub-activities can be seen by clicking on the plus signs. The list view displays all the activities. The graphic view displays the top level activities and any expanded boxes that were opened when the project was last closed. Different symbols can be seen by the names of the activities, indicating different states. No symbol implies "Under Design", a yellow circle implies "Sent Out", a blue circle implies "In Progress", a green circle implies "Completed" and a red circle implies "Rejected".

The **Proflex** Manager with the loaded project is shown in Figure 8.

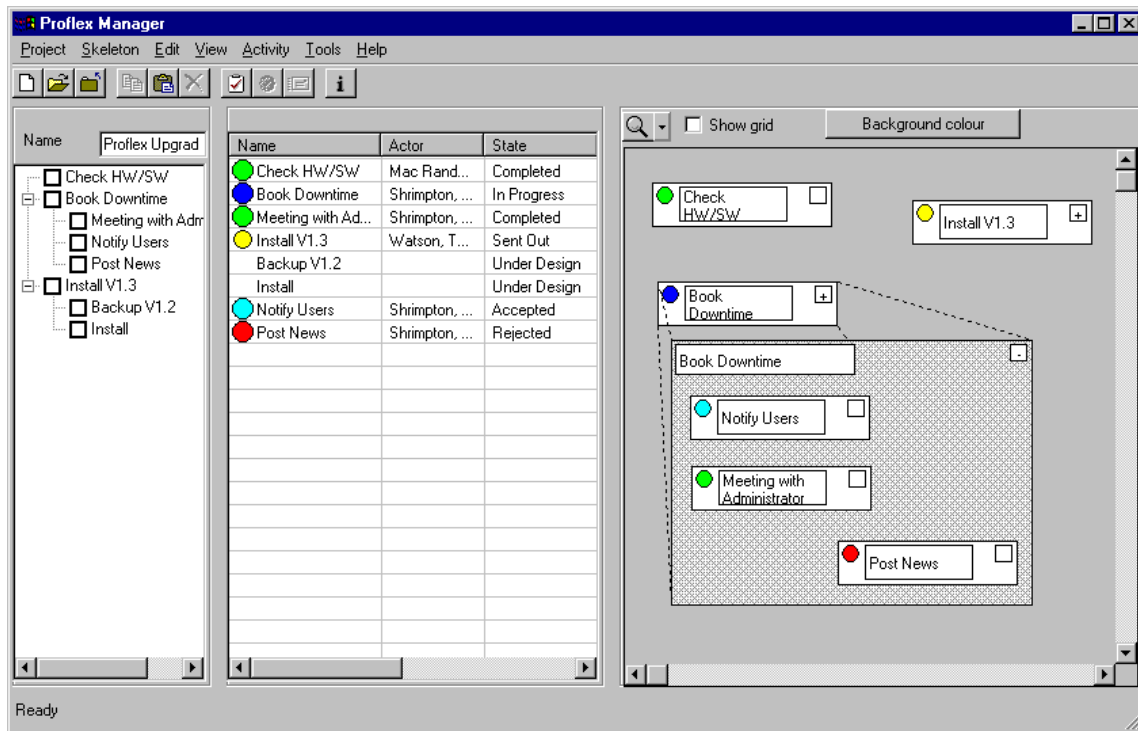


Figure 8: A newly loaded project

The manager can see at once that an activity has been rejected. The red circle highlights this activity. To review the reason either check the Actor Comment field in the list view or use the Activity Form by clicking on the Configure button and checking the Results tab sheet.

It appears that the activity has been sent to the wrong Actor. The Manager needs to revise the activity.

Clicking on the rejected activity in the list view to select it and then selecting the 'Revise' item from the Activity menu causes the Activity Form to appear. This version of the form has the same appearance as the standard Activity Form when configuring an activity, except the only action button is the "Save, Close & Re-send" one. After changing the actor to a more suitable one the "Save, Close & Re-send" button is clicked.

The Activity Form (in Revision 2 mode which is used for rejected and terminated activities) is shown in Figure 9 below.

Figure 9: The Activity Form in use to revise an activity

The activity is then sent out to the new actor and the PM, when updated, will show this state. Figure 10 shows the activity now in the 'Sent Out' state in the PM.

Backup V1.2		Under Design
Install		Under Design
Notify Users	Shrimpton, A...	Accepted
Post News	Watson, TA ...	Sent Out

Figure 10: The updated state of an activity after revision

## 10 EVALUATION

---

A business software tool has two main aspects: (1) it is designed to support a certain part of the everyday business or administration and (2) it has a set of features that provide the intended support and encourage people to use it on a daily basis. Furthermore, its use may bring to light some indirect spin-offs that are equally important to evaluate as they may become important marketing assets.

The **Proflex** system is designed to help the management and enactment of flexible, customer driven, responsive processes (**projects**) via a focus on:

- easy (re-)design of projects and their set of **activities**,
- simple allocation of activities to the appropriate **actors** by use of a groupware system,
- support for controlling and monitoring projects by the **manager**,
- facilities for negotiating, executing and reporting on activities by the actor,
- **intrays** for accessing and manipulating activity lists,
- integrating and interfacing existing products and
- providing a common information pool that enables automatic update of all project and activity related data that is visible to both manager and actor given the appropriate access rights.

The evaluation of the ability of **Proflex** to support these kinds of processes mainly considered the system's core functionality that make it a support tool without going into too much detail on the individual features. It is based on general questions and the related answers that were asked of the Field Testers within the User Organisations.

The findings of the Field Testers revealed that they indeed think of **Proflex** as a possible everyday tool that would help them to co-ordinate and monitor their projects. Consequently it would enable them to take the necessary actions in order to reach the project goals set.

Support for Total Quality Management is seen as one of the most important indirect spin-offs of the use of **Proflex**. To measure whether the Field Testers joined the **Proflex** developers in this, they were asked to look at the four main aspects of TQM: (1) customer directed working, (2) Process Thinking, (3) involvement by everyone and (4) taking cost saving measures.

The majority of the Field Testers really believe that **Proflex** can contribute to TQM, especially by the improvement of the process thinking through the display of the project in three different views (Tree, List and Graphical View). Providing the actors with the possibility to refine their own activities and thus become managers of their own work, is seen as a good feature to get the people to gain a greater feeling for involvement in their work and the related projects.

The common information pool, which offers up to date status of the project at any time for both manager and actor, and the overall better project management that can be attained with **Proflex** will help to save time and costs for the organisation.

Another spin off that was concluded is the ability of **Proflex** to contribute to Knowledge Management through the use of Skeletons. These Skeletons are considered to be a good means to re-use the work already done in former flexible projects and to learn from experience as they allow the user to store projects or part of projects for later re-use.

Evaluation of the **Proflex** tool and its functionality was carried out in two domains: system transparency and system functionality.

Firstly, the system transparency was evaluated by looking at the clarity of the concept behind the **Proflex** components (**Proflex** manager, MS Outlook Inbox and MS Exchange Server as common information pool), the system's ease of use and ease of learning.

The concept behind the tools seemed not to be that evident for the end users whose answers revealed that the purpose of the Microsoft software was clearer to them than **Proflex**'s own programs. The fact that all End Users are used to working with these Microsoft components was the reason behind this.

The majority found **Proflex** to be a "nice to use" tool which did not take too much time to get acquainted with. Very little additional support was therefore needed. Ease of use and ease of learning is highly dependant on the IT or non-IT background of the End User. Users with more knowledge than their colleagues or an aptitude for new software were more likely to speak in favour of **Proflex** in this respect.

Secondly, in order to evaluate the system functionality of the current **Proflex** version (1.3), the delivered features were compared against the process maps of the User Requirements as they had been laid down in the early stages of the project. Not all foreseen core functionality has been implemented in the system yet. Throughout the project, the developers, in close co-operation with the User Organisations, frequently changed the priority list.

It can be concluded, however, that the functionality provided in **Proflex** enables the user to perform the core actions necessary to be able to evaluate the system as a system for the support of project management. It allows the user to create and structure one's project, to send out activities to actors, to enter (partial) results into the system and to further use these results as a manager. It even provides the user with some advanced tasks like skeletons, refinement by the actor, manipulations of sent out activities by the manager (suspend/restart, cancel, revise), re-parenting, etc..

Missing features involve blocks of required functionality that for reasons of further investigation were not yet implemented in the system. Nevertheless, the developers acknowledge that some of these outstanding issues do contribute to the selling potential of their newly created software.

Other missing functionality includes additional features that were required by the Field Testers during the exhaustive testing at the User Organisations. However, laying down the User Requirements is a flexible and thus evolutionary process of its own. Considering the diversity in the backgrounds of the future and constantly growing group of users, a different use of the system may be expected, in its turn generating more and other requirements. Consequently it will be advisable to make a distinction between "must haves", "should haves" and "desirable" requirements and always bear in mind that the main intention of the **Proflex** Developers has always been to create a transparent, easy to use tool with no functional overkill.

**The User Organisations final conclusion is that the Proflex Developers have reached, with Proflex V1.3, a highly advanced prototype that has the potential to become a successful and exploitable business tool that is suitable for the support of project management for flexible, customer driven, responsive processes.**

**With its set of functionality, a finalised version of Proflex offers an easy way to create, manage and monitor projects for the manager and a clear execution for the actor. Its common information pool provides all project members, from top level managers to day-to-day clerks, with the latest information.**

It is the User Organisation's opinion that this product, being itself a product of a successful European co-operation, has the ability to become a successful tool to support the kind of processes it has been designed for and we really hope it will.



## 11 CONTRIBUTION TO THE STATE OF THE ART

The user requirements exercise carried out at the start of the project confirmed the underlying requirement for a tool like **Proflex**. The initial market survey carried out at that time showed a range of different tools that each satisfied some of the requirements, but nothing that covered all the requirements. The existing tools fell into categories as shown in Figure 11.

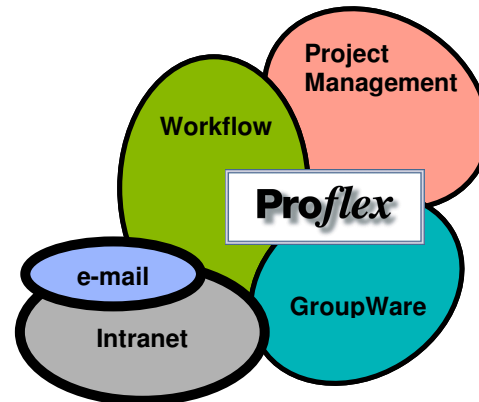


Figure 11: **Proflex** positioning in the marketplace

Looking at recent developments in all three areas, there has been a clear trend to extend each category of application towards one or other of the complementary areas. For example, MS Project 2000 (a Project Management tool) has been given the ability to export tasks to Outlook (a Groupware tool). However, the links are still primitive, and in any case serve to make the tool set much more complicated – already a major criticism of existing products. The conclusion drawn from the recent market studies is that the niche targeted by **Proflex** still exists and is unlikely to be properly addressed by the major vendors in the near future.

The **Proflex** system has a number of novel or unusual features intended to make it easier for busy managers to design, enact and monitor their processes. These have been described in more detail earlier, but are summarised here:

- The ProjectManager simultaneously displays the hierarchical structure of the process, details of the individual activities and a graphical view of the relationships between the activities. The manager can use any view to input, edit and view the activities.
- The manager can delegate activities to colleagues, see how they break the task into parts and/or delegate work to others, monitor progress and be instantly aware of any problems, missed deadlines, etc.
- Any actor (subject to access permissions) can see the context of the activities they have been given, and have access to the full power of **Proflex** to handle that activity i.e. to define sub-activities, delegate to others, monitor progress, etc.
- At any stage in a project (normally at the end), the specific data for that project can be stripped out and the remaining structure saved as a template or skeleton for later reuse as the basis for similar projects or as a specialist component of a larger project.

One of the most important aspects of **Proflex** is its simplicity. Throughout the entire system, the emphasis has been on ease of learning and ease of use – the intended users are busy managers who are not IT experts.

## 12 BENEFIT TO THE PARTNERS

The main benefit arising from the project is clearly the **Proflex** system itself. All partners believe that the finalized version of **Proflex**, offering an easy transparent tool for project management, will help them to increase their competitiveness in the market by increasing the efficiency of their current practice and supporting further Business Process Re-engineering (BPR), Total Quality Management (TQM) and Knowledge Management (KM).

In addition to the delivered system, all partners feel that they have benefited significantly from participation in the project. The user partners and technical partners came to the project with very different backgrounds and experiences, and so benefited from the collaboration in different ways.

### 12.1 USER PARTNERS

The extensive training and frequent presentations of **Proflex** and its background within the companies has not only contributed to the dissemination of new software but also helped to wake up both the staff within the Administrative & Commercial Departments and within other sheltered workshops to the necessity of good project management.

There has been significant cross-fertilization between partners throughout the project. In depth exposure to other organizations with different sizes and types of administrative and management styles gained during the requirements capture and evaluation phases has been enriching for all concerned. It was interesting to come into contact with a non-industrial environment in RAL and see their more academic approach to business – quite different to that of the user partners.

From a technical point of view, the user partners have gained a better understanding of how to:

- (better) draw up a user requirements document,
- evaluate the new tool, its design and functionality,
- introduce a new tool and its background to an unaware audience, adapting the presentation to the different kinds of positions/background of the people who form part of that audience.
- train people and change explanations for people with an IT or non-IT background,
- evaluate the acceptance/resistance from people and to reposition ourselves accordingly,
- evaluate the delivery of the tool through comparison with the user requirements.

Being involved in the development stage of a new software system also improved understanding of software, particularly the implementation complexity required to achieve some of the requirements and the factors beyond the developers control that lead to initial requirements being unfulfilled or, more commonly, fulfilled indirectly.

### 12.2 TECHNICAL PARTNER

Similarly to the User partners, one of the most valuable intangible benefit of the collaboration is the insight RAL gained into how commercial organization in the manufacturing supply chain operate. The knowledge of commercial administration/management gained during the Requirements Capture, the roll-out of the Field Trial and the Evaluation phase has been invaluable.

Useful experience was gained when the project coordinator and technical partner had to pull out of the consortium. Taking over the project management and re-planning the technical workplan was challenging, but with the full support and cooperation of the other partners the project was successfully realigned – a model of how a cohesive Consortium can handle change. One positive consequence of the whole affair was the need for RAL to take on the role of lead exploitation partner, leading to a much better understanding of the way the commercial software market operates and to useful contacts in this area.

Technically RAL has gained useful knowledge of

- working with the latest software development tools;
- interfacing to Microsoft Office and Back Office software;

- introducing software systems to non-IT users
- leading the exploitation planning for a new software product,
- requirements and environments in the manufacturing supply chain.

In summary, all the partners feel that the project has been a success, and they have gained significant collateral knowledge as a result of participation.

## 13 PROJECT MANAGEMENT ISSUES

---

This is not the place to report details of the management of the project. However, there are two issues that should be of general interest.

Firstly, it is reassuring to know, that participants in a project devoted to process management were sufficiently aware of this topic, that it was completed on schedule and within budget. A trite observation perhaps, but many projects do not show the application of their own R&D subject to themselves.

Secondly, as a multi-national, multi-partner project we ourselves had to dynamically reorganise our processes in May 1999 when the company acting as our prime contractor and project leader went into liquidation. This followed a management buy-out a year earlier and was not entirely surprising. However, the consequences for the project were that we lost our management partner, needed to formally re-structure the project, and contracts with our funding body the CEC, and we also lost our major exploitation partner, leaving a technology developer, and three user organisations in the consortium. We were able to resolve the management issues, but the loss of any experienced software exploitation and marketing expertise was a significant blow, which may be evident from other parts of this final report. We have produced exploitation and marketing plans for the **Proflex** results, and have approached various organisations with the potential to exploit these results as commercial products, with the expectation that it will be exploited after the completion of the project.

However, the loss of a committed, experienced exploitation partner is very significant. There were many marketing and exploitation consultants willing to take on the exploitation planning as a paid activity. But they obviously lacked any commitment, indeed the longer they extended the exploitation planning activity, the more they would be paid, so they lacked any motivation to progress quickly to productisation, marketing and sales.

Equally, CLRC, the technical development partner in the project was willing to lead the exploitation activities, and competent in terms of planning, but without a network of personal contacts it was hard to make any actual progress towards sales.

These comments are made here to provide a context in which the next section of the report should be read, but also to underline the commitment of the project participants to commercial exploitation of the project results, even though we admit that such activity is not our main pastime.

## 14 RESULTS AVAILABILITY AND THEIR EXPLOITATION

### 14.1 PROJECT RESULTS

Several kinds of project results can be exploited:

The user organisations intensify their already good co-operation and their pool of services and products, which they market together on the European market. The knowledge gained during the project will be disseminated to other sheltered workshops and SME's.

One of the objectives of the technical suppliers is to gain reference sites by introducing the developed **Proflex** solution into companies. Along with that the **Proflex** products and services will have been tested and evaluated, The major results that are subject to dissemination and exploitation are:

- An operational pilot of a system supporting flexible, customer driven, responsive processes,
- a proven methodology on how to design flexible, customer driven, responsive processes
- promotion materials to educate the markets,
- the knowledge on how to introduce a system to support flexible, customer driven, responsive processes into organisations,
- marketing materials to access potential customers.

The project has a dissemination and exploitation plan as a planning and control instrument for dissemination and exploitation activities. First the plan gives an overview of the organisations, outlining the objectives and infrastructure that can be used for dissemination and exploitation. Afterwards the products and services that are subject to exploitation are described. The position of **Proflex** on the market is updated and its potential customer markets are laid down. Finally the market related strategies and concrete action plans are developed. The dissemination and exploitation plan is completed by a business plan for the future marketing of the developed services and products.

### 14.2 EXPLOITATION METHODOLOGY

The marketing methodology adopted in the exploitation was the SOSTAM method that follows the following stages:

- Situation Analysis
- Objectives
- Strategy
- Tactics
- Action Plan
- Monitoring

The Situation Analysis covered:

- Structure of the market
- Trend Analysis
- Economic Factors
- Competition
- Technology
- Social and Cultural Issues
- Political Environment

Most of which has been described elsewhere in this final report, showing a clear market niche which the technology addresses.

### 14.3 COMPETITION

This niche can also be characterised in terms of competitor products currently in the market:

Product Category	Groupware Platform: Exchange (Microsoft)	
Messaging/ Calendaring/ Scheduling	Outlook (Microsoft)	
Project Management	MS Project (Microsoft)	
Work Management	Workfolder for Exchange (EastmanSoftware)	<b>Proflex</b> ⚡
(Ad HOC) Workflow	KeyFlow (KeyFile)	
(Production) Workflow	Eastman Workflow (Eastman Kodak)	

### 14.4 EXPLOITATION WINDOW

The timing of exploitation is limited by:

- 6-12 months with no competitors
- 2 years before MS extends Exchange/MSPProject to do ad hoc workflow
- 4 years before **Proflex** is killed by MS or Web or other new technology.

### 14.5 PRICING

The closest competitors in the market for workflow tools price their products at around \$1500 for the server pack and then \$100 per seat to use them. Since the **Proflex** solution uses the MS-Exchange server, we would intend to sell initially into a market of organisations who already have this installed.

### 14.6 PRODUCTS

The products to be exploited are:

- 1) The **Proflex** software.
- 2) Maintenance and support of the software.
- 3) Consultancy to use the software.
- 4) Training courses & materials for the software.

### 14.7 CONSORTIUM EXPLOITATION OBJECTIVES

The main objective of the consortium is to generate a revenue stream from these products that will:

- a) support and maintain **Proflex**,
- b) enable further development of **Proflex**, and
- c) support development of other downstream products to supplement and replace **Proflex**.

Given that the project partners are either technical developers, or users, the route chosen to achieve this is to identify an exploitation partner. The Exploitation Partner could be found to take over the productising, marketing and sales roles, in exchange for exploitation rights to the product. This would generate a royalty stream (10%) that could be used to achieve objective b and c above – objective a and possibly b would be covered by the exploitation partner.

The advantages of this approach are that it keeps the level of further investment by the consortium small, and minimizes the size of the development team required. The Consortium's sub-objectives then become:

- a) Support a development team (1-2 people X 2 years)
- b) Build a relationship with partner to:
  1. establish contact with customers to identify unfulfilled needs (new projects) and
  2. maintain an exploitation channel for use with other products.
- c) Generate profits and repay investment (i.e. invest in other projects).

## 14.8 EXPLOITATION PARTNER

To meet the consortium objectives, and implement this strategy, the exploitation partner can be characterised as:

A medium sized software sales company:

with more than one product generating revenue to cover the investment required in launching **Proflex**, but not enough products that **Proflex** will be ignored.

Appropriate customer base covering the whole of Europe, or with arrangements with sales outlets across Europe to take advantage of the multinational, multi-language interface to **Proflex**.

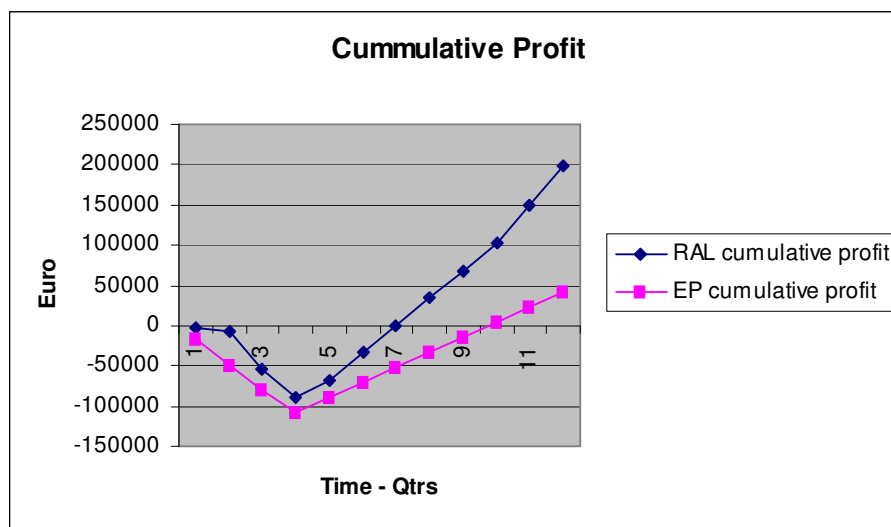
The customer base should include the support and use of Microsoft BackOffice, in particular MS-Exchange since the **Proflex** tool is dependent on this server, and it would be easier to sell to those already using it, than to have to sell it into companies too.

Complementary products such as QA documentation, contact management, accounting etc. which already sell to managers so that they have salespersons with established appropriate contacts in potential customer companies.

The necessary IT skills to assess the further developments and required investment in **Proflex**, and act as a partner in that development.

## 14.9 BUSINESS PLAN

For such a partner, and the **Proflex** consortium the business plan suggests the following returns over quarters following the start of post-project technology implementation:



After the period shown in the above graph, the return to the consortium flattens off, while that to the exploitation partner continues to grow with sales. This graph does show the maximum investment required of an exploitation partner for productisation, marketing and sales.

The consortium has approached several potential exploitation partners who fit this description, and are approaching others. At the time of writing this report, none has yet agreed to exploit the results.



## **Appendix 1**

~

### ***Proflex* Glossary**



## Appendix 1 – Proflex Glossary

<b>Proflex Term</b>	<b>Definition</b>
Activity	An activity is an aggregate or single item of work, containing a description of what is to be done and by whom, what results are required etc. An aggregate activity is usually split into sub-activities.
Actor	An actor is an internal or external person or group of persons like a department or even possibly a machine (e.g. fax machine) that carries out activities from a project.
Administrator	The administrator is a person responsible for maintenance of the stored projects, activities and supporting information which is required for the <b>Proflex</b> system to run smoothly. They also register users for <b>Proflex</b> .
Attachment	Documents related to a project or an activity are “attached” to the activity or project. Attachments may be electronic files (MS Word-files, e-mail files, spreadsheets, etc.) or paper-files.
Cancelled	A project or activity can be cancelled. This action stops any further work being done on the project or activity. A cancelled project/activity is not expected to be <u>restarted</u> .
Category	A category is used to classify projects or activities and is used for managing, finding and <u>monitoring</u> projects and activities. It is a term describing the kind of project or activity. Examples include “Sales”, “Motor Goods”, “White Goods”, “Marketing”, “Esprit” etc.
Completion	An <u>activity</u> is completed by entering the results of the “real” work into <b>Proflex</b> and signalling that work on the activity has been done to the point the <u>actor</u> considers complete..
Delegation	Delegation is the action of passing on the work to another but keeping the responsibility for that work being done.
Design/ re-design	Design and re-design are the processes that allow the continual evolution of a project, adding, changing and removing activities. A project can be designed from scratch or by using <u>skeletons</u> .
Enactment	Enactment is the collective term for the different phases within execution of designed activities: <u>sending out</u> , <u>negotiation</u> , (re-) <u>design</u> , in progress, <u>completed</u> and <u>monitoring</u> .
Execution	Execution of an activity means carrying out a work item (“real” work) within or outside of <b>Proflex</b> .
Forwarding	Forwarding is the action of passing on the work to be done and the responsibility for it to another.
Graphical View	The name given to the display of the project within the <b>Proflex</b> Manager that shows activities in their hierarchy as boxes of variable size and location according to the user. Links between activities are also shown. Typically this view is the right-most view.
Keyword	A keyword is used to classify projects and activities and is used for managing, finding and <u>monitoring</u> projects and activities. It is a term that relates to the data within a project, examples include customer name, part number, part type etc.
Line Manager	A line manager is an <u>actor's</u> immediate manager and typically relates to the organisation's structure rather than the projects undertaken.
List View	The name given to the tabular display showing all the activities within a

	project within the <b>Proflex</b> Manager. Typically it is the central view.
Manager	A manager is a person defining a <u>project</u> or an <u>activity</u> and performing actions upon the project or activity, e.g. (re-)name, send-out, suspend, terminate. The manager <u>role</u> is one of the roles the <b>Proflex</b> user may take on.
Monitoring	Checking the status of work by using standard/user defined reports by setting filters on project information and visualising them in various ways.
Negotiation	Negotiation within <b>Proflex</b> is about the <u>actor</u> accepting or not accepting the activity assigned to him. The actor negotiates on the communicated activity with the manager and may either accept it, <u>delegate/forward</u> it to another actor or reject it. Negotiation may also involve the line manager.
Organisation's fields	An organisation's field on a project or activity is one where the meaning and the usage are defined by the organisation. They can be used in the same way as <u>keywords</u> for managing, finding and <u>monitoring</u> .
Parent activity	An activity that encapsulates other activities, i.e. has internal structure and hence contains <u>sub-activities</u> .
Predecessor activity	Activity preceding another or several other activities.
<b>Proflex</b> User	Anyone who uses the <b>Proflex</b> system is a <b>Proflex</b> user. There are three main <u>roles</u> that the user can take; <u>manager</u> , <u>actor</u> and <u>viewer</u> .
Project	Projects are a collection of loosely or hierarchical structured activities guiding a set of business processes that need to be executed to fulfil the goal of the project.
Project lists	Each project may have lists of people attached to it. Each list is given a name. These lists can represent different management styles. Examples include a Steering Group and a Reference Group.
Project team	A team consists of a group of managers and the <u>actors</u> . Each project team would typically have a project team leader who has overall responsibility for the project.
Realisation	A <u>UML</u> term used to describe the models that express how to achieve a <u>use case</u> . There can be many realisations for any one <u>use case</u> .
Restart	A project or activity that has been <u>suspended</u> may be restarted. This releases the project or activity so that on it work may continue.
Role	A part played by the user. Each role has a defined set of characteristics. There are three roles used within <b>Proflex</b> ; <u>manager</u> , <u>actor</u> and <u>viewer</u> .
Sending out	The sending out of an activity means communicating to an actor by various channels that work (described in the activity) has to be done.
Sequence diagram	A <u>UML</u> diagram that shows a sequence of actions over time in order to achieve all or one path through a <u>use case</u> .
Skeleton	A skeleton is a project where all project specific (not generic) information is removed (i.e. no transactional data), in essence a set of activities and their structure.
Sub-activity	A sub-activity is one part of a larger (aggregate) activity and so has a <u>parent activity</u> . There is no difference in designing sub-activities except that they are on a level lower than their corresponding <u>parent activity</u> . All attributes and the graphical presentation refer as well to a sub-activity.
Successor activity	Activity following another or several other activities.
Suspend	A project or an activity can be suspended. This action stops any further work being done on the project or activity. A suspended project/activity

	may be <u>restarted</u> or <u>terminated</u> .
Terminate	An activity may be terminated by the actor, after acceptance and before completion. It is used to signify work has been finished on the activity but the activity as not been completed as expected.
Tree View	The name given to the display showing a project in its hierarchical structure with the <b>Proflex</b> Manager. Typically it is the left-most view.
UML	The Unified Modelling Language is a notation used to express designs in a variety of ways, e.g. <u>sequence diagrams</u> , <u>realisations</u> .
Use Case	A typical interaction between a user and the system. A use case captures some user-visible function usually achieving a discrete goal for the user.
Viewer	The <b>Proflex</b> user acting in the viewer role makes use of the <u>monitoring</u> facilities provided within <b>Proflex</b> to review the status of work, its planning and its progress.
Work item	'Real' work carried out by an actor. The work item is usually not part of <b>Proflex</b> . However the results of work items are part of <b>Proflex</b> and they become <u>attachments</u> .



## **Appendix 2**

~

### **Internal Publications & references**





## Appendix 2 - Internal Publications & references

This appendix lists the papers produced by project partners for themselves or shared with other partners which are not official deliverables of the project.

Title	Author(s)	Reference	Date
Initial Thoughts on User Requirements	CLRC	Proflex-RAL-1	02/98
Note on Microsoft Exchange 5.5 Briefing	CLRC	Proflex-RAL-2	02/'98
System Development Process	CLRC	Proflex-RAL-3	02/98
How to store Activities in MS Exchange	CLS		05/98
Communication between the <b>Proflex</b> Project Manager (PPM) and the <b>Proflex</b> Store	CLRC	Proflex-RAL-8	07/98
Implementation of Short and Long Thin Slices	CLS		07/98
Evaluation of development support tools for <b>Proflex</b>	CLRC	Proflex-RAL-11	10/98
Installing the PPM	CLRC	Proflex-RAL-13	10/98
Outstanding User Requirements	CLRC	Proflex-RAL-15	11/98
Process Maps	CLRC	Proflex-RAL-16	11/98
Use Cases and Scenarios	CLRC	Proflex-RAL-18	01/99
Activity States and the Actor	CLRC	Proflex-RAL-19	01/99
Activity States and Actions	CLRC	Proflex-RAL-20	02/99
Key Actions	CLRC	Proflex-RAL-22	03/99
Graphic View within the PM	CLRC	Proflex-RAL-24	03/99
Evaluation of toolkits for Graphic View	CLRC & CLS	Proflex-RAL-25	04/99
Graphic View within the PM (revised)	CLRC	Proflex-RAL-27	04/99
Skeletons for <b>Proflex</b>	CLRC	Proflex-RAL-28	05/99
Linking Activities	CLRC	Proflex-RAL-31	10/99
Background Agent State Changes	CLRC	Proflex-RAL-32	11/99
Reuse in <b>Proflex</b>	CLRC	Proflex-RAL-35	03/00



## **Appendix 3**

~

### **External Project Publications**



### **Appendix 3 - External Project Publications**

Damian Mac Randal, Business Processes ; **ATLAS Magazine** - CLRC Open Days Special Issue June/July 1998.

Jacques Waelkens, "Beschutte werkplaatsen gaan veranderende administratieve processen beheren met Proflex"; **WAAK, VLAB NIEUWSFLASH**, Spring 1999. "Sheltered Workshops are going to manage changing administrative processes with Proflex" (English version).

Nachtsheim, Reinhard; Lippert, Detlef , Proflex - Professional Management of flexible, customer driven, responsive processes. In Proceedings of the European Multimedia, Microprocessor Systems and Electronic Commerce Conference (EMMSEC), June 21-23 1999, Stockholm, Sweden.

Klaus-Dieter Kreplin, Damian Mac Randal, Herbert Schmauch Real Supply Chains and Virtual Enterprises in Proceedings of the International Conference on Enterprise Information Systems (ICEIS2000) (July 2000) .



## **Appendix 4**

~

## **Project Consortium**





## Appendix 4- Proflex Consortium

### Project Manager

Dr Michael Wilson  
CLRC Rutherford Appleton Laboratory  
Chilton  
UK - OX11 0QX  
Didcot  
UK

Tel: +44-1235-446619  
Fax: +44-1235-445831  
e-mail: [m.d.wilson@rl.ac.uk](mailto:m.d.wilson@rl.ac.uk)

### Project Technical Manager

Mr. Damian Mac Randal  
CLRC Rutherford Appleton Laboratory  
Chilton  
UK - OX11 0QX  
Didcot  
UK

Tel: +44-1235-445403  
Fax: +44-1235-445831  
e-mail: [d.mac.randal@rl.ac.uk](mailto:d.mac.randal@rl.ac.uk)

### User Partners

Mr. Jacques Waelkens  
WAAK Beschutte Werkplaats vzw  
Heirweg 125  
B-8520  
Kuurne  
Belgium

Fax: +32-56-350287  
Tel: +32 75 87 37 48(variable)  
+32-56-3634 10  
e-mail: [info@waak.be](mailto:info@waak.be)

Mr. Gerhard Lang  
Lebenshilfe Werkstatt GmbH  
Scharnitzstrasse 11  
D-81312  
Munich  
Germany

Tel: +49-89-71008-16  
+49-172-9449018  
Fax: +49-89-71008-18  
e-mail: [lebenshilfe-werkstatt@t-online.de](mailto:lebenshilfe-werkstatt@t-online.de)

Mr. Kent Halldén  
Alcatel IKO Kabel AB  
S-51481  
Grimsås  
Sweden

Tel: +46-325-80285  
+46-7052-80285  
Fax: +46-325-80320  
e-mail: [kent.hallden@iko.alcatel.se](mailto:kent.hallden@iko.alcatel.se)