Baseline Tool Development and Configuration
Work Package Reference 2.3
Document C

Scope

This document describes the implementation and development of the protocols, techniques and tools to support the outsourcing scenarios of the PILOT projects. The identification and definition of the functionality of these tools is given in Document B.
A clear pre-requisite of the CARIBCAD programme is to utilise existing state of the art IT environments and to configure these to the project needs but not to develop IT solutions. The following describes the configurations made by work package 2 to meet the needs of the pilot projects and the testing process. It is presented in three sections selection and configuration of the groupware environment, selection and configuration of the workflow environment and the special support the CAD in a remote working environment.

Groupware Tools

Microsoft Exchange Server 5.5 (MSEXCH) was chosen as the basic platform upon which to build the specialist CAD remote working support required by the Pilots. The reasons behind this choice are outlined in Document B.
It was clear that there were two alternatives to configuring this environment, an HTML/Java based approach and a Microsoft Outlook/ Visual Basic approach.
The former was chosen for the following reasons

- Open standards language definition
- Utilises a standard browser rather than specialist software
- Server side development testing and deployment

Experience in the early stages of the project setting up the computing environments for the research team had shown that installing specialist or even normal software on client machines was fraught with problems and error prone furthermore this difficulty was independent of Northern/Southern hemisphere issues. Indeed Universities partners often experienced more difficulties due to the complexity of their existing environments. It was therefore decided to opt for solutions that could be developed, tested and deployed on a server and accessed universally through a WEB browser by the client. This too, proved to be more complex than initially perceived and is discussed later.
The first phase of the configuration was to model the individuals, their organisations and roles within the standard framework provided by MSEXCH. Each user was modelled as part of the GroupWare environment as well as an external participant in the wider Internet. This was to enable CARIBCAD to manage members of the Pilot teams both as individuals working within discrete companies accessible via the Internet and as members of a collaborative team brought together to complete a specific project. In essence we aimed to create and Intranet on top of the Internet, as opposed to a virtual private network.

Having set up the basic GroupWare support an online tutorial was put together to train each user in collaborative group working. This can be seen at http://archhive.ncl.ac.uk/caribcad/grouptut.htm.
This initial training session began with simple email exchange working through conversation thread tracking, shared files, diaries and appointment management. Several problems were identified with incompatibilities of the web browser employed. The majority of these problems related to different levels of compliance to HTML standards employed by the two key Web Browser tools, Netscape and Internet Explorer. Netscape was found to be the most rigorous in its implementation of a standard and was used to carry out quality checks, Internet Explorer was found to be the most effective in use and often implemented the more recent versions of standards, it was generally used by the team members. A key lesson learnt was that "standard" does not necessarily mean "the same", standards can be interpreted and implemented differently by software authors. Furthermore, standards such as HTML are emergent and evolving, careful consideration needs to be given to the version and the specific subset of that version that is to be used within a project where repeatable, consistent and vendor independent behaviour is required in a configured system.
Following this initial Groupware configuration, deployment and testing the workflow components were added and configured.

Workflow Tools

As part of the Baseline Implementation and Definition, Action Workflow Manager had been adopted as the process-modelling tool. This embodied the Language Action Approach defined by Medina-Mora et al 1992[i], and workflow models were constructed using this paradigm for Pilot 1 and 2 Figure 1. However, at the June workshop in Delft in 1998, a review of this approach concluded that the enactment service required for its implementation was over complicated and beyond the resources of the project. A further review of workflow enactment engines was carried out and it was clear that the technology had developed considerable in the year between the two reviews. The trend identified in the initial review of workflow engines building upon the more developed Groupware engines had gained momentum.

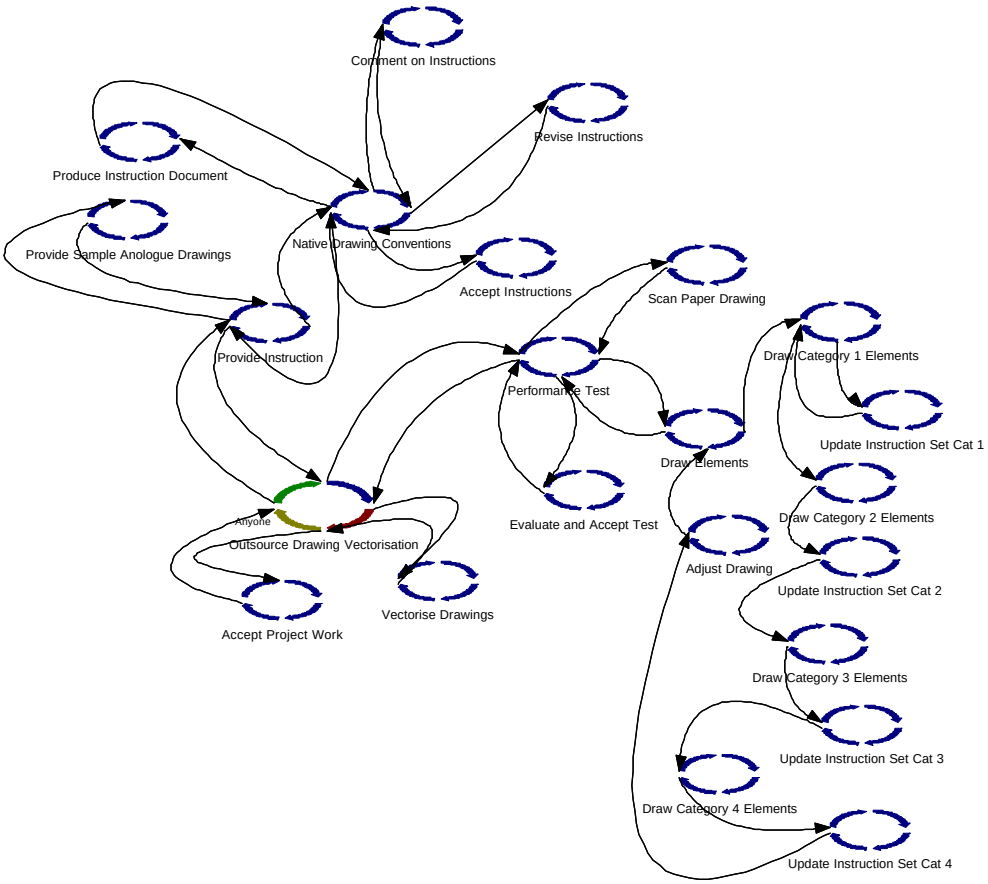

Figure 1 Typical Language Action Workflow Model

A major factor in reducing the complication of the workflow environment was the process definition language. It was evident that the workflow models that were to be developed by WP2 had to be transparent and understandable by all actors, not only during development but also during execution. There was an obvious division of the available workflow systems into those, which were communication centric, and those which were data centric. It was decided to investigate one engine from each aspect that met our needs to reduce complication. Keyflow by Keyfile inc. was selected as the primary workflow engine and Figure 2 illustrates the configuration that was defined for the initial workflow enactment environment.
The following sections outline the configuration and deployment of these components.

Definition of the Data Objects

Both pilots have the need to handle and exchange CAD drawings. At the June 1998 Delft workshop it has become clear that scanned drawings may not need any special consideration.
Figure 3 illustrates the top level view of the data model proposed to support the CAD drawings needs of the Pilots. The words in circles are objects the lines between the circles relationships between those objects and the letters "1" and "M" the cardinality of those relationships (1 or Many).

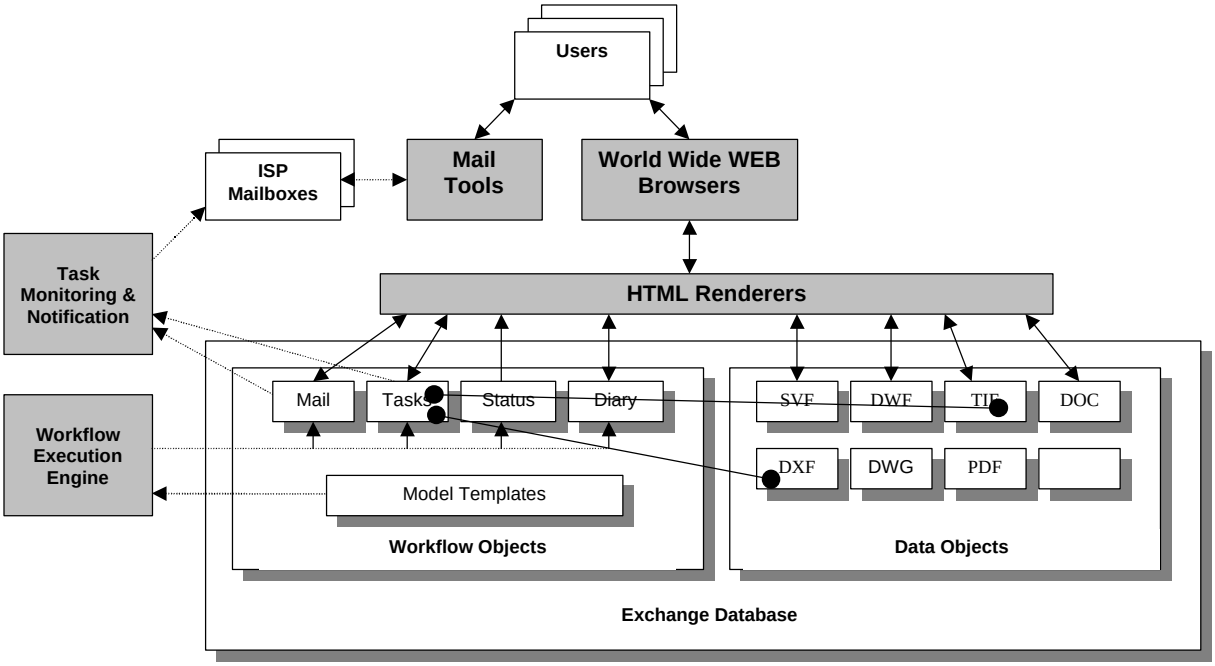

Figure 2 Overall System Architecture

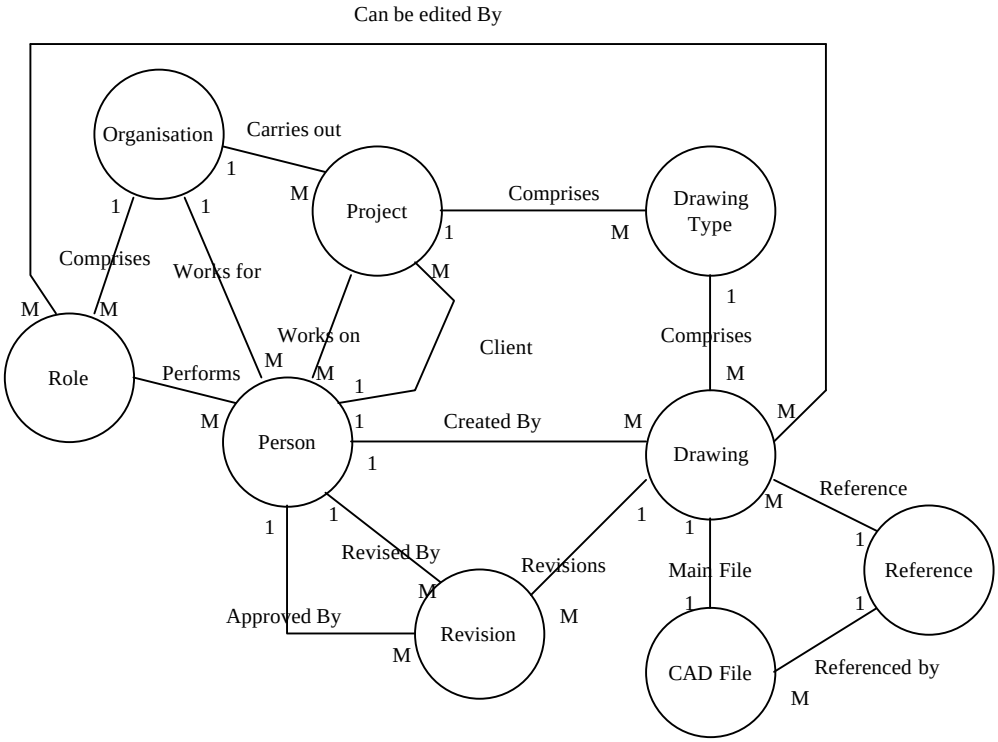

Figure 3 Data Model Overview

This is only the top-level view of the key CARIBCAD data components and their relationships. In addition we need to consider the attributes of these objects and the relationship to Internet Items or objects.
The data model is implemented on top of the Collaborative Data Objects (CDO) Model for Internet Items, the following shows the relationship between these and the CARIBCAD objects.

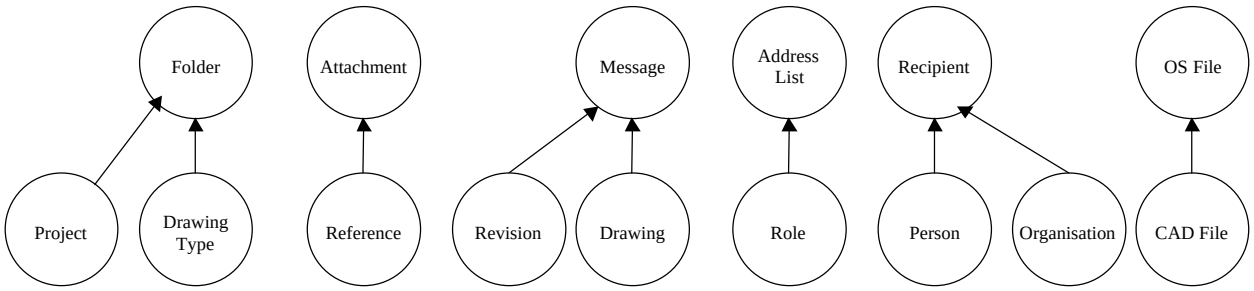

Figure 4 Relationship to Collaborative Data Objects

The arrow terminated line indicates inheritance or "Special Kind of". All the CDO items have the facility to support programmatically added fields of a wide range of types. These are used to add attribute data, such as "Drawing Title", "Scan Resolution" etc.

The following is an outline of the attributes/fields that have been implemented for each object over and above those that are inherited from the CDO layer. Where the "Folder." Or "Message." notation is used it indicates that an inherited CDO attribute is employed.

Project
Name [Folder.Name]
Drawing Type
Name [Folder.Name]
Drawing
Created [Date]
Reference Number [String]
Title [Message.Subject]
Public Release Version [String]
Latest Revision [Number]
Scale [String]
CAD File Type [String]
Comments [Message.Subject]
Revision
Description [Message Body]
Number [Number]
Created Date [Date]
Task
Planned Start [Date]

Actual Start [Date]
Planned End [Date]
Actual End [Date]
Completion Status [Enumeration]
Percentage Complete [Number]
Assigned To [String]
Assigned By [String]

Reference

Inserted By [String]
Inserted On [Date]

The CAD objects have been implemented employing the Microsoft Collaborative Data Objects within Exchange Server environment. The result is a special kind of email message that can be sent through the Internet and received as a CAD object not merely as a mail message.
The WEB based implementation was found to be neither as flexible nor extensible as the alternative Outlook based approach outlined earlier. Neither did it result in the simplification envisaged for end users. The differences in implementation and conformance levels to HTML of the various browsers available created unnecessary complications for the project. A major area of difficulty was encountered with supposedly simple aspects, such as file up and downloads using HTTP protocols. These were found to be unreliable over low bandwidth and low quality Internet connections. This is a general problem with WEB based file transfer and tools were identified that could resolve the file download issues i.e. GetRight ™ [ii].

Definition of the Workflow Objects

The workflow object model was built upon the Keyflow Data Object environment that in turn was built upon the Microsoft Collaborative Data Objects environment. The most significant workflow objects are Flow, Task and Documents. In order to meet the Pilot requirements the most configuration was required in the Task and Documents objects.

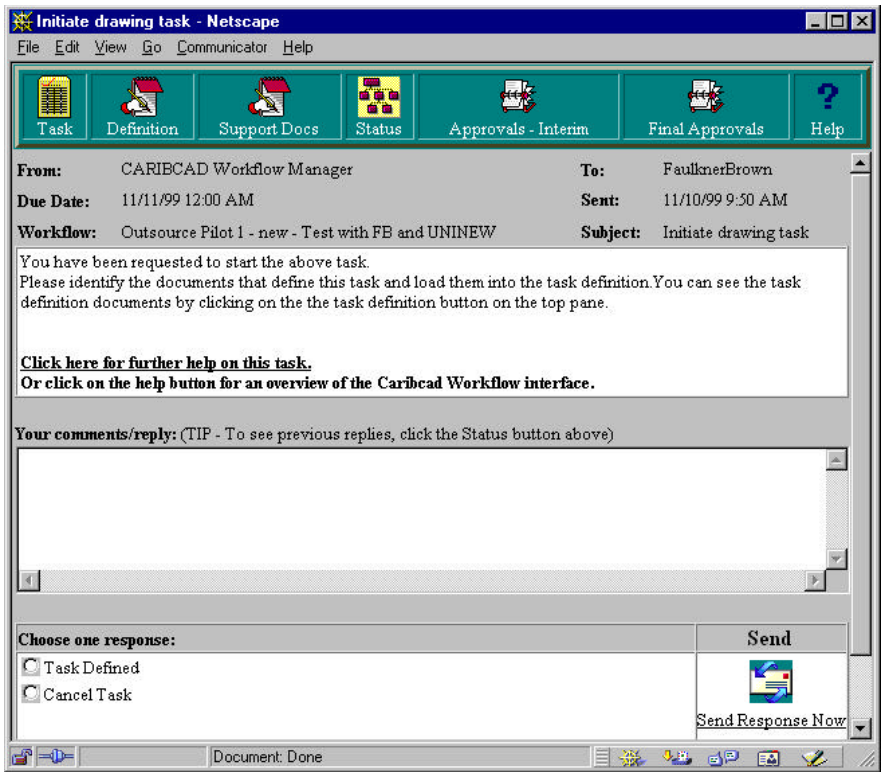

Figure 5 Typical Configured Workflow Task

Figure 5 illustrates how a task was configured to suite the needs of the both Pilots. The major components added to the generic task were

- Definition
- Supporting Documents
- Interim Approvals
- Final Approvals

Definition, each task was expanded to contain a set of documents that defined the specifc work to be executed by the entire workflow. It was conceived that this definition would expand as the problem became resolved and better understood by actors adding further documents. Supporting Documents, were necessary to describe the protocols and procedures that needed to be understood and consulted during execution of the flow. Interim and Final Approvals were where the outputs of the workflow were placed by actors for submission and acceptance. Appendix

Based on this general strategy and the two informal Pilot models a general CAD or document transaction process model was proposed upon which the specific workflow models for the Pilots could be built. The lifecycle of a CAD file has been mapped throughout its usage within the project and the processes that surround this have been abstracted into four models, they are

1 Create Drawing/Document
2 Edit Drawing/Document
3 Review Drawing/Document

4 Issue Drawing/Document

Informal models were then made for each of these areas. They are

1 Create Drawing/Document

a) Specify Project
b) Identify Drawing/Document Type
c) Identify Reference Drawings/Documents
d) Prepare Drawing/Document Meta Data
e) Nominate Editors
f) Nominate Readers
g) Nominate Reviewers
h) Nominate Owner
i) Invoke the create operation (NB no CAD file or document file exists)
j) Receive confirmation of creation.
h) Notify owner of its existence.

2 Edit Drawing/Document

a) Define the editing work to be executed
b) Identify and locate the relevant drawing/document
c) Allocate work and agree/negotiate timescale
d) Accept/Understand work to be done
e) Check out the Drawing/Document
f) Raise queries about work to be done as they occur
g) Monitor progress/status of editing operation
h) Notify that editing is complete
i) Review editing work, comment redline etc. (Invoke "Review Drawing/Document" session)
j) Approve drawing/document
k) Issue Drawing (Invoke Issue CAD file model)

3) Review Drawing/Document

a) Receive instruction drawing is ready for review
b) Obtain project and drawing requirements (what is needed, brief etc.)
c) Redline and comment on drawing (This and the next three are not necessarily sequential)
d) Raise queries with Editors
e) Issue drawing (make it public) or
f) Invoke a new "Edit Drawing" session

4) Issue CAD Drawing/Document

a) Identify recipient
b) Prepare transmittal advice
c) Transmit package to public area
d) Notify recipients of its availability
e) Recipients acknowledge receipt

f) Recipients invoke "Review Drawing/Document" session
g) If review requires changes sender invokes another "Edit Drawing/Document" session
h) If recipient approves drawing the "Issue for Construction"

It should be noted that these are general process models for Pilot 1 and Pilot 2, they are extended to meet the specific needs of the formal models defined for each Pilot. An examples of how these fitted into Pilot 1 model is,

Item 1.1 Prepare Initial Instruction Document
Item 1.2 Comment on Instruction Document
Item 1.3 Revise Instruction Document

Have counterparts of
"Create Document", "Edit Document", "Review Document", "Issue Document"

In the more CAD specific areas,

Item 3.3.1 Draw elements until correct and
Item 3.3.1 Check Correctness

Have a direct counterpart as they decompose into a "Create Drawing" followed by a series of "Edit Drawing" operations followed by a "Review Drawing" and finally an "Issue Drawing"
The difference between the WP2 model and the Pilot models is that the WP2 model has no specific project application and needs to be configured for a specific project related set of actions.
This general framework was implemented in a series of formal workflow models that are defined in documents D and E.

CAD Specific Configuration
***TUD input to be inserted

---

[i] Medina-Mora, R; Winograd, T.; Flores, R; Flores, CF.; (1992). The Action Workflow Approach to Workflow Management Technology. In Turner, J.; Kraut, R.; (ed.). *Proceedings of the 4th Conference on Computer Supported Cooperative Work.* ACM, 31 October-4 November, Toronto, Canada:281-288
[ii] http://www.getright.com