

UMEA: User-Monitoring Environment for Activities

Victor Kaptelinin

Department of Informatics

Umeå University

S-901 87 Umeå, Sweden

+46-90-786 5927

vklinin@informatik.umu.se

ABSTRACT

A common problem with virtual work environments is a lack of support for setting up and managing project contexts. The paper presents a system that addresses this problem by providing low-overhead support for personal project management. The system monitors the user and creates an interaction history containing MS Office 2000 events mapped to individual projects. By selecting a project the user enters a project context, that is, gets a convenient access to project-related resources. When a new resource is used within a project, the resource is automatically added to the project context.

KEYWORDS: project contexts, interaction histories, desktop metaphor

INTRODUCTION

Carrying out a higher-level task (or “project”) typically involves setting up and maintaining a *project context*, that is, arranging necessary resources to make them readily available when working on the project. Computer users often spend considerable time and effort finding and opening documents, web pages, email messages, contact details, etc., to be able to carry out their work.

Conventional virtual work environments provide little support for managing project contexts, especially when projects span several applications and require various types of information objects. This limitation had been recognized in the area of Human-Computer Interaction (HCI). A number of systems, such as ROOMS [1] or the Task Gallery [3], were developed in the last decades to help users create dedicated project spaces. However, these systems have not become widely used. It appears one of the main problems with these systems is overhead. If the user has to set up and maintain project contexts manually, the costs of using a system are apparent and immediate, while the benefits are uncertain and remote.

This paper presents a system named UMEA (User-Monitoring Environment for Activities), which addresses

the above problem by utilizing interaction histories (see also [2]). The rationale behind the system is making project contexts a by-product of the actual work on projects.

SYSTEM ARCHITECTURE

The UMEA system is implemented as an application running under Microsoft Windows (see Fig. 1). The application was developed in Microsoft Visual C++ 6.0 using DISCo interface library (www.disco.ru).

In the foreground mode (1) the system allows the user to view the list of projects (2), select or set up a project, open a resource -- a document, a folder, a web page, or a contact, from a project-specific list of resources (3), and use a number of personal information management (PIM) tools (4).

In the background mode (5) the system receives Microsoft Office 2000 events, such as opening a web page, printing a file, or sending an email. The events are received through a COM add-in DLL (6). When the system receives an event, the event is linked to the currently active project and saved in a Microsoft Access database (7). If the event includes a new resource, that is, a resource that has not been used within the currently active project, the resource is added to an appropriate list of project-specific resources (3).

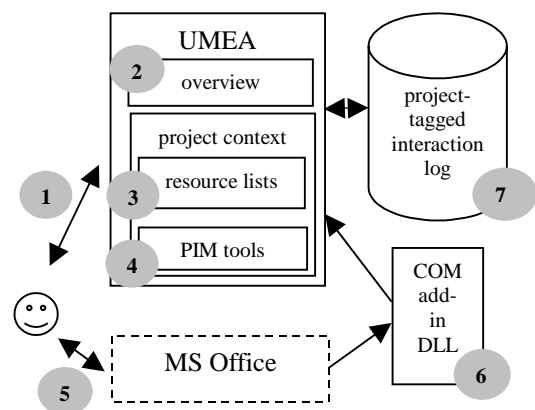


Figure 1. The general architecture of the system

USER INTERFACE

The user can switch between three different views. The *minimized overview* window (Fig. 2) displays two vertical panels: (a) the project panel showing project icons and the "maximize" button, and (b) the resource panel, showing pop-up menu buttons, corresponding to four types of project-specific resources: documents, folders, URLs, and contacts (email addresses). By clicking on a button on the resource panel and then selecting an object on a pop-up menu the user can open a resource related to the project selected on the project panel.

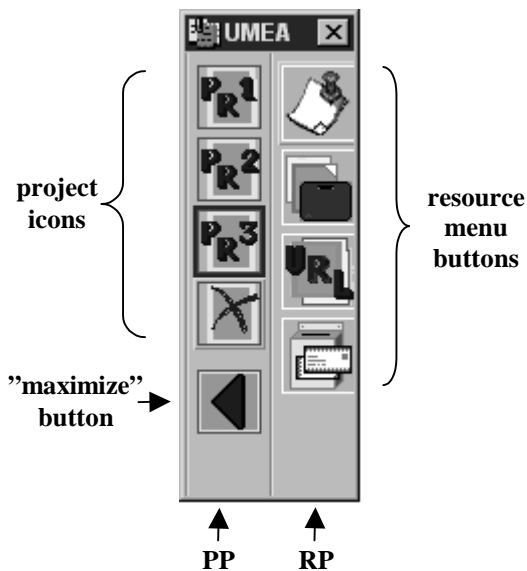


Figure 2. A minimized overview window (PP-project panel, RP-resource panel)

The *maximized overview* window (Fig. 3) extends the minimized overview window by including: (a) a complete list of projects, (b) a PIM/ history area, which displays a PIM tool or the interaction history of the active project, and (c) a control panel.

Project windows, which look similar to the maximized overview window, allow the user to define attributes of a project, such as its name, associated color, icon, and deadline. Besides, a project can be decomposed into lower-level tasks.

CONCLUSIONS

To sum up, the UMEA system allows the user to set up a number of projects and select one of them as active. Events in the interaction history created by the system are associated with active projects. Therefore, resources actually used when working on a certain project, can be identified, organized into categories, made readily available to the user, and automatically updated.

First experiences with UMEA show that the minimal initial effort necessary to start enjoying the benefits of the

system can be as low as setting up a new project (one mouse click) and giving it a meaningful name. With time, the benefits are increasing. By selecting a project the user gets an easy access to project-related resources and, at the same time, makes it possible for the system to create even more elaborated project contexts.

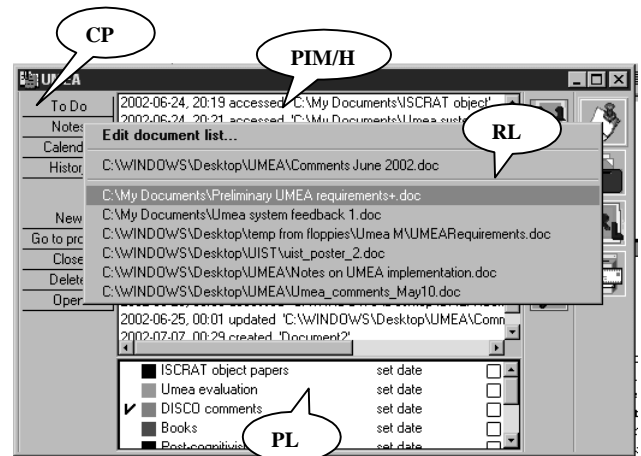


Figure 3. A maximized overview window with an opened list of documents (PL – projects list, PIM/H – PIM/ history area, CP – control panel, RL – resource list opened by clicking on the Document button)

Additional advantages of UMEA appear to be an integration of PIM tools and other types of information management within a single environment, and a lightweight, non-restrictive type of support it provides to the user.

ACKNOWLEDGEMENTS

The study was supported by the Swedish Agency for Innovation Systems (VINNOVA).

REFERENCES

1. Henderson, A., Card, S. K. (1986) Rooms: The use of virtual workspaces to reduce space contention in a window-based graphical user interface. *ACM Transactions on Graphics*, 5 (3).
2. Kaptelinin, V. (1997) *Supporting Higher-Level User Actions with Representations of Project Contexts and Interaction Histories*. Working Paper. Department of Informatics, Umeå University.
3. Robertson, G., van Dantzich, M., Robbins, D., Czerwinski, M., Hinckly, K., Risdan, K., Thiel, D., Gorokhovsky, V. (2000) The Task Gallery: A 3D Window Manager. In: *Proceedings CHI 2000 Conference on Human Factors in Computing Systems*.