Project Management via the Web

Author: Robert Green

Robelle Solutions Technology Inc.

Second Floor, 15399-102A Avenue

Surrey, BC, Canada V3R 3K1

Telephone: 1.264.497.5088

Fax: 1.264.497.3295

Email: bob.green@robelle.com

Introduction

Now that the HP 3000 computer platform is reborn as the **e3000**, it is time to consider innovative new uses for the system. This paper will teach you Internet techniques that can improve the management of your software development projects, and any other projects that your group, department or company may have. I am going to show you how to use your browser, a Web site, and some basic tools like Qedit for Windows to simplify project management.

There is more to project management than just knowing if the project is on schedule. Traditional tools like MS Project do this, and well, but they do little else. The rest of project management is sharing information and results throughout the project team, and that is the primary topic of this paper. The issues addressed are task collaboration, sharing, distribution and updating of digital products, testing, and detailed communication within the team (see "Working Together" by Bennett in the References).

Contents
Why the Web?
A First Project – HP World Prep
Add a Sub-Project
A Programming Project Using the Net
Preliminary Design via the Web
Portable Design and Coding via the Web
Server Coding via the Internet
Automated Testing Over the Internet
Documentation, Installation and Distribution
The Future

About the Author – Bob Green.   I am a software developer, author and speaker, member of the HP Hall of Fame and founder of Robelle (http://www.robelle.com) . I welcome your comments at bob.green@robelle.com

Why the Web?

- Robelle is truly a distributed company.
- How to collaborate over 4 time zones, in 3 different countries?
- Email is a great communicator, but email is not enough.
- The Web is accessible by all and open to all types of documents.

Robelle is a distributed company. For example, our R&D team is sprinkled between Vancouver, Calgary, Montreal, Phoenix, and the Caribbean. And we have partners all around the world. How are we to collaborate on projects? We needed a simple, easily available method to communicate ideas and to collaborate on various projects.

Naturally, our first use of the Internet was email, which immediately improves project communication. Team members can pose questions and answer them without the time synchronicity required for phone calls. But email isn't enough. It eliminates the need to be in the same place and/or time, but it doesn't do much to manage the masses of data and relations that a project can generate. What you need is a repository for the digital products of project planning and status information.
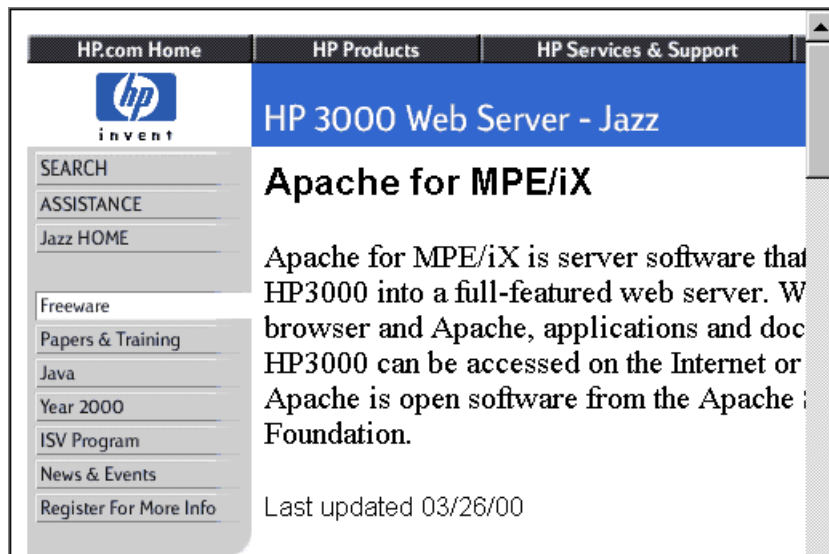
As our projects grew in scope to include more people and especially more people in remote locations, the Web provided a medium that was supported, easy to use, and widely accessible.
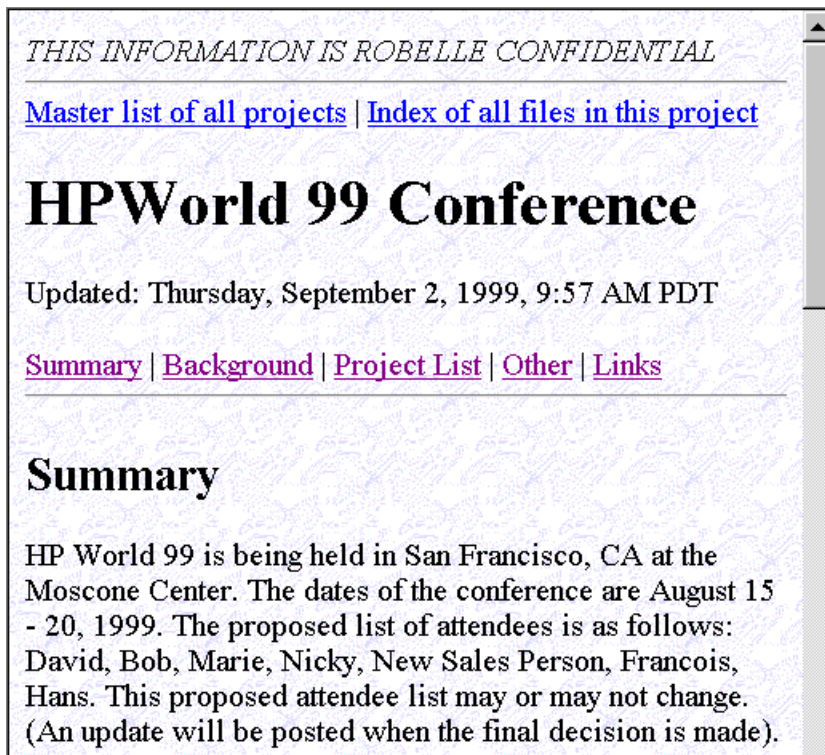In addition, the Web is open, flexible. and adaptable to changing needs, not a closed straightjacket. From a Web page you can link to other Web pages, text files, DOC files, PDF files, CAD drawings, databases, supplier Web sites, customer Web sites, source code, mailing list archives, almost anything. And the industry leader in Web servers, Apache, is available for the e3000 on Hewlett-Packard's Jazz Web server.

A First Project ~ HP World Prep

- HP World Project Web page.
- Create the HTML file with your favorite editor.
- Link to related pages, internal and external.
- Divide project into smaller steps, which also have pages.

Let's start with a simple project that we **all** can understand: preparing for the HP World conference: how to make all of the arrangements to come to HP World, set up the booth, write these tutorials and, more importantly, provide you with T-shirts..

The first step is an internal company Web site for recording the status of projects. Then we create a Web page for a new project: HpWorld.

> *THIS INFORMATION IS ROBELLE CONFIDENTIAL*
>
> Master list of all projects | Index of all files in this project
>
> ## HPWorld 99 Conference
>
> Updated: Thursday, September 2, 1999, 9:57 AM PDT
>
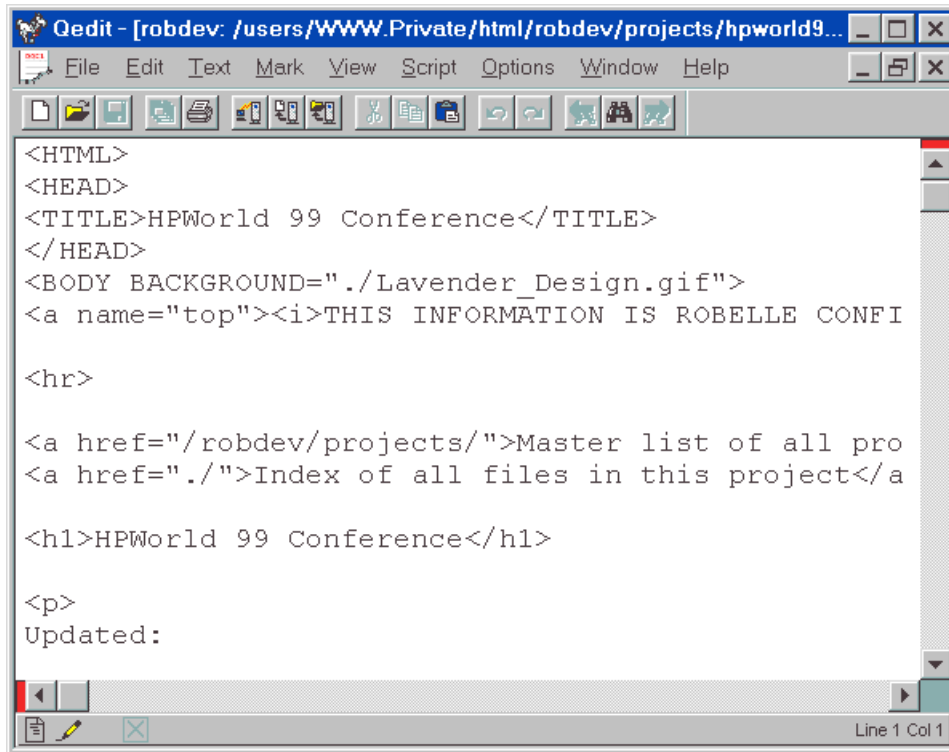> Summary | Background | Project List | Other | Links
>
> ## Summary
>
> HP World 99 is being held in San Francisco, CA at the Moscone Center. The dates of the conference are August 15 - 20, 1999. The proposed list of attendees is as follows: David, Bob, Marie, Nicky, New Sales Person, Francois, Hans. This proposed attendee list may or may not change. (An update will be posted when the final decision is made).

A Web page is just a file with text that describes both the content and how to present it in the browser. The markup language is called HTML and is easy to learn – everyone in our company does their own HTML for status reports, project pages, etc. Notice the "Updated: Thursday…" comment on the page above. This is created automatically by the following Server Side Include (see **References** for links to more information):

```
Updated:
<!--#config timefmt="%A, %B %.1d, %Y, %.1I:%M %p %z" -->
<!--#echo var="LAST_MODIFIED" -->
```
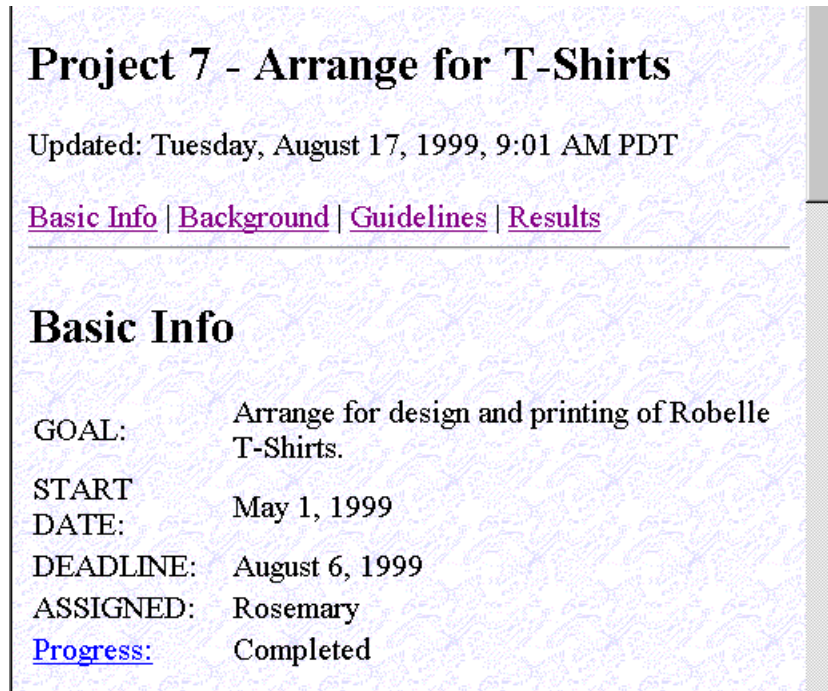
Here is how the HPWorld projects page above looks when you are editing it:

```
Qedit - [robdev: /users/WWW.Private/html/robdev/projects/hpworld9...

 File  Edit  Text  Mark  View  Script  Options  Window  Help

<HTML>
<HEAD>
<TITLE>HPWorld 99 Conference</TITLE>
</HEAD>
<BODY BACKGROUND="./Lavender_Design.gif">
<a name="top"><i>THIS INFORMATION IS ROBELLE CONFI

<hr>

<a href="/robdev/projects/">Master list of all pro
<a href="./">Index of all files in this project</a

<h1>HPWorld 99 Conference</h1>

<p>
Updated:

                                                            Line 1 Col 1
```

Using the Internet as your project management platform allows you to integrate material from external partners into the project. In this case, Interex is our main partner for HP World and they have a Web site also: http://www.interex.org

Putting your project planning documents on the public Internet raises the serious issue of **security**. To make a Web site accessible to remote project members, but without making it freely available to strangers, there are several techniques you can use: a password on the site or sub-site, or a firewall that limits who can do what from beyond the wall.

Returning to our HP World project - there were 10 major steps in getting Robelle to HP World. Let's take a look at the sub-project 7, the T-shirt sub-project.

Project 7 - Arrange for T-Shirts

Updated: Tuesday, August 17, 1999, 9:01 AM PDT

Basic Info | Background | Guidelines | Results

Basic Info

GOAL:	Arrange for design and printing of Robelle T-Shirts.
START DATE:	May 1, 1999
DEADLINE:	August 6, 1999
ASSIGNED:	Rosemary
Progress:	Completed

Each sub-project has generally the same layout. We have a spot for the Basic Information of the Project, some of the background information and the guidelines on how to get the project completed and what the steps are. These Web pages keep everyone informed of the tasks required, the results so far, where we are with each task, and clearly outline who is responsible for what.

All of this is done with simple HTML, which even non-technical persons within the company have mastered. The surprising thing is that the non-technical people were the first to begin using color to indicate the status of a task.

Add a Sub-Project

- Edit the main project page to link to a new project 11.
- Edit Web pages direct on your server.
- Create the new Web page using project 10 as a template.

Let's add a step to the Hp World project to Recap the conference and plan for next year.

10	Wait	Post Show Activity (Marie, Nicky)	Tue, Aug 17, 1999
11	Wait	Conference Recap - Ideas for 2000 (Hans)	Thu, Sep 2, 1999

Other

None.

Links

THIS INFORMATION IS ROBELLE CONFIDENTIAL

Previous Project | Main | Next Project

HPWorld 99 Conference

Project 11 - Conference Recap for 1999 And Ideas/Suggestions for 2000

Updated: Thursday, September 2, 1999, 1:43 PM PDT

Basic Info | Background | Guidelines | Results

```
Qedit - obdev: /users,
File   dit  Text  Mark

<td>Wait
<td><a href="pro
<br>(Marie, Nick
<td><!--#flastmo


<tr>
<td><a href="project-11.html">11</a>
<td>Wait
<td><a href="project-11.html">Conference Recap - I
<br>(Hans)
<td><!--#flastmod virtual="project-11.html" -->

</table>
```

Line 139 Col 34

We will need to create a new Web page for a step 11, the Recap project, connect it with the main HPWorld page. The main page is edited to contain a link to the new project. Links are the key to the Web:

```
<a href="project-11.html">Conference Recap - Ideas for 2000</a>
```

The page we are pointing to is "project-11.html" in the same directory as the main page. But it could easily link to a page at another Web site if we wished. For example, here is a link to the HPWorld 99 conference proceedings on the Interex Web site:

```
<a href=http://www.interex.org/conference/hpworld99/proceedings/index.html>Click</a>
```

A Programming Project Using the Net

- A project with a client, server, and several team members.
- Delete Columns: part of Rectangular selecting and editing.
- Local, Server MPE & HP-UX files.
- Used Web and Qedit to manage project.

Since I am most familiar with software development, let's look at a software project. Customers wanted columnar operations in Qedit for Windows: delete a rectangular selection, insert, copy, paste, plus shift left and right. For example, here is an entry from our customer support Knowledge Base:

http://server.robelle.com/cgi-bin/knowledgebase?kb=14062

KB Ref No 14062

```
Ref:        14062
Company:    St. Paul Pioneer Press
Caller:     Cindy Sirois
Phone:      (235) 712-8375
Serial#:    42A
Subject:    Add & Delete Columns in QW
Line1:      She would like to be able to add or delete columns throughout
Line2:      the file.
Product:    QW 4.6.08
Techie:     TORY KILGOUR
Pri:
Keywords:
Status:     Enhancement Request
Orig Date:  19980717 14:21


In response to Eunice's follow-up email to QW users. Cindy stated that she
thought QW was wonderful, enjoyed using it, makes her job a little bit
easier. However, the one thing she would like to see incorporated is the
ability to delete or add columns throughout the file. She is their Sr.
Programmer.
```

I was able to access this enhancement request in my Web browser because we have written a CGI interface from the Web server to the ImageSQL database that holds the data. Ken Robertson (email: Ken.Robertson@ROBELLE.com) explains how the CGI interface works: "We tried two methods. The first was to put a Visual Basic program on our NT server. The drawback was relatively slow loading times and you had to use Internet Explorer as your browser. The second and current method uses Apache running on the e3000, with a small CGI script. The script actually runs the existing MPE application and

re-directs the output to a flat file, which we then display. This was strictly an R&D project, i.e. how to set up and use Apache. We have thought of better ways to do it in the future, where the Web interface can be outside the firewall and the Knowledge Base database is inside the firewall."

The goal for this customer enhancement was for column operations to work identically for local files as well as server-hosted files. That meant defining the edit operation, coding it in the client, designing a client user interface, porting the edit code to the server, and defining a transaction between the client and server.

In the Robelle tradition, we wanted the Delete Column enhancement to be reliable, but we also wanted to produce it quickly. It was just one of several Qedit enhancements we were managing. For a period of 16 months, we released a new production Qedit version every month, including updated documentation, complete with instant Web download.

Here is how we envisioned the project results in our beginning, naïve design:
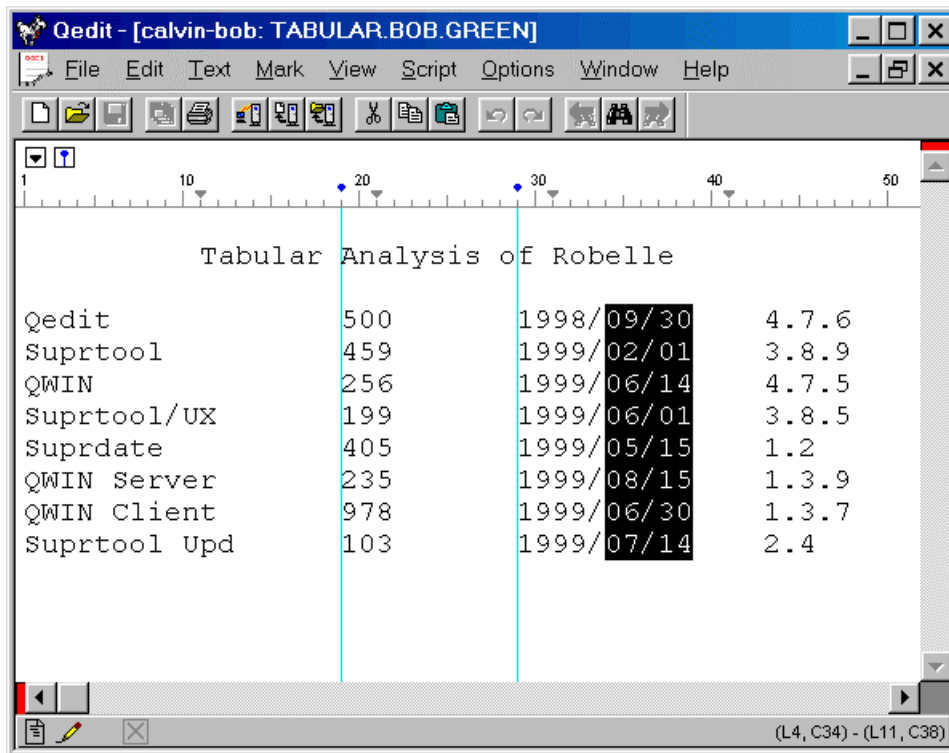
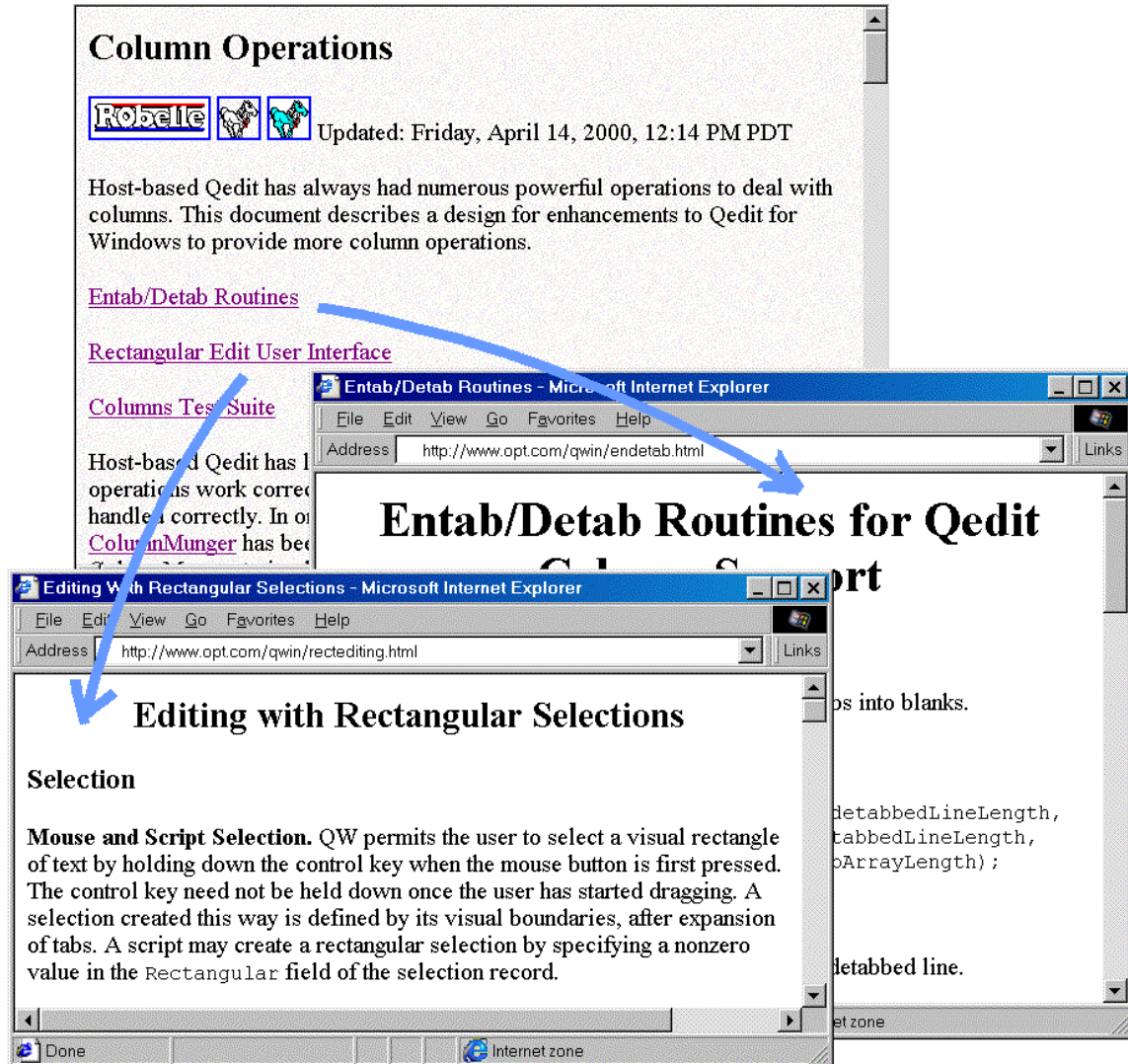We chose to highlight the rectangular selection by holding down the Control key and moving the mouse or the cursor keys. Once the rectangle is highlighted, you can delete it with the Delete key. This paradigm is used in a number of other software products - why re-invent the wheel and make things difficult for your users. Note: The vertical blue lines in this image are Guide Lines you can place at important columns.

Preliminary Design via the Web

- Design is published on Web pages.
- Coping with tabs.
- User interface design: Control-Drag.

As the project ramped up, a number of Web pages appear:

Column Operations

Updated: Friday, April 14, 2000, 12:14 PM PDT

Host-based Qedit has always had numerous powerful operations to deal with columns. This document describes a design for enhancements to Qedit for Windows to provide more column operations.

Entab/Detab Routines

Rectangular Edit User Interface

Columns Test Suite

Host-based Qedit has 1...
operations work corre...
handles correctly. In o...
ColumnMunger has be...

Entab/Detab Routines - Microsoft Internet Explorer

File   Edit   View   Go   Favorites   Help

Address   http://www.opt.com/qwin/endetab.html   Links

Entab/Detab Routines for Qedit Column S... ...ort

Editing With Rectangular Selections - Microsoft Internet Explorer

File   Edit   View   Go   Favorites   Help

Address   http://www.opt.com/qwin/rectediting.html   Links

Editing with Rectangular Selections

Selection

Mouse and Script Selection. QW permits the user to select a visual rectangle of text by holding down the control key when the mouse button is first pressed. The control key need not be held down once the user has started dragging. A selection created this way is defined by its visual boundaries, after expansion of tabs. A script may create a rectangular selection by specifying a nonzero value in the Rectangular field of the selection record.

...bs into blanks.

...detabbedLineLength,
...tabbedLineLength,
...bArrayLength);

...detabbed line.

...et zone

Done   Internet zone

From a quick design, the team implemented a **prototype** rectangular option for the Delete transaction. Notice that the links are to pages on the Web site of a technical partner: OPT. This is one of the great advantages of the Web as a collaboration technique.

As soon as we tried the naïve design on real data, we could see that tabs were quite a problem. If the user deletes a rectangle of text from the screen that has tab characters before, within, and after it, the user expects that things will still line up (at least most of

the time!). This means explicitly distinguishing between physical column numbers and visual column numbers.

We started the design over by thinking about converting tabs to spaces and vice-versa and created code to Entab and Detab lines. See the Entab/Detab Web page in the preceding graphic. Isolating this issue in a single routine, encouraged spirited "discussions" (over the net) about how each boundary case with tabs should be handled.

We then worked on a detailed User Interface Design. We knew we wanted Control-drag to define a rectangle, since that is what the Reflection emulator uses, then the Delete key to delete it. But we hadn't worked out what happened in every situation (copy to clipboard, insert a rectangle from the clipboard, etc.).

Notice that all of the design work is done on Web pages, which reside on a Web server anywhere in the world. The project team member doesn't really need to know where the page is located, unless he must update it. The HTML code used is extremely simple, on purpose, so that any team member can make changes when necessary. One advantage of using Qedit for Windows for this task is that we can cut and paste any of the existing code or interface definitions simply by pasting directly into the HTML source. Or from the Web page into the source code!

Portable Design and Coding via the Web

- Put tab-aware edits in new ColumnMunger() object.
- Written in Visual C++, also valid in C-UX and C-MPE.
- Define a new transaction to use ColumnMunger().

As our thinking on tabs matured, and because we wanted to deal with visual columns in a consistent way throughout the project, we created a single function, ColumnMunger(), to encapsulate our results.

Transactions of Client-Server Qedit – Microsoft Internet Explorer

File   Edit   View   Go   Favorites   Help

Address   http://dev.robelle.com/qedit/doc/transactions.html   Links

DeleteColumn TX (34)

See ColumnMunger documentation and source code.

This tagged TX works in a column range (i.e., logically a rectangle) to either remove text or fill it with spaces. See also Insert Column TX and Columns Design.

Overflow...
transacti...
overflow...

Client T...

- kt:
- kt:
- kt:
- kt:

Network Column Munger – Microsoft Internet Explorer

File   Edit   View   Go   Favorites   Help

Address   http://www.o...t.com/qwint/columnmunge...html   Links

ColumnMunger

ColumnMunger (rhymes with "plunger") is a routine that handles tab-aware column operations for Qedit. It takes the form of a three-operand instruction, where two of the operands are treated is input and the third is treated as a result. There are nine operations currently defined.

The calling sequence is:

int ColumnM...

```
/* ColumnMunger
 *
 * Column manipulation routine and auxiliaries for Qedit
 *
 */

#include "stdafx.h"
#include "typedef.h"
#include <stdlib.h>
#include <ctype.h>
#include <string.h>
#include "colmunge.h"
```

Done

Here you see the Delete Columns transaction from the Client/Server design, linking to the underlying ColumnMunger() design and ColumnMunger() source code.

The programmer passes ColumnMunger() a tab-stop record, a line from the file, and a function code such as SubstituteVisual. ColumnMunger() takes into account the tab stops and any tabs in the line, then passes back the modified line.

The ColumnMunger() source code is written in Visual C++ on the PC, but is carefully coded so that it is also valid for the C/UX and C/IX compilers on the servers. The same code is executed on all platforms, both client and server, ensuring that the tricky boundary cases are handled the same everywhere.

Again, we use the Internet as the collaboration backbone for the project development process.

Server Coding via the Internet

- Geographically dispersed programmers.
- Centralized source code on hosts.
- Edit and compile over the net to multiple hosts.

For this project, we have server source code in Surrey, Canada and programmers in Montreal, the Caribbean, Phoenix and Vancouver.
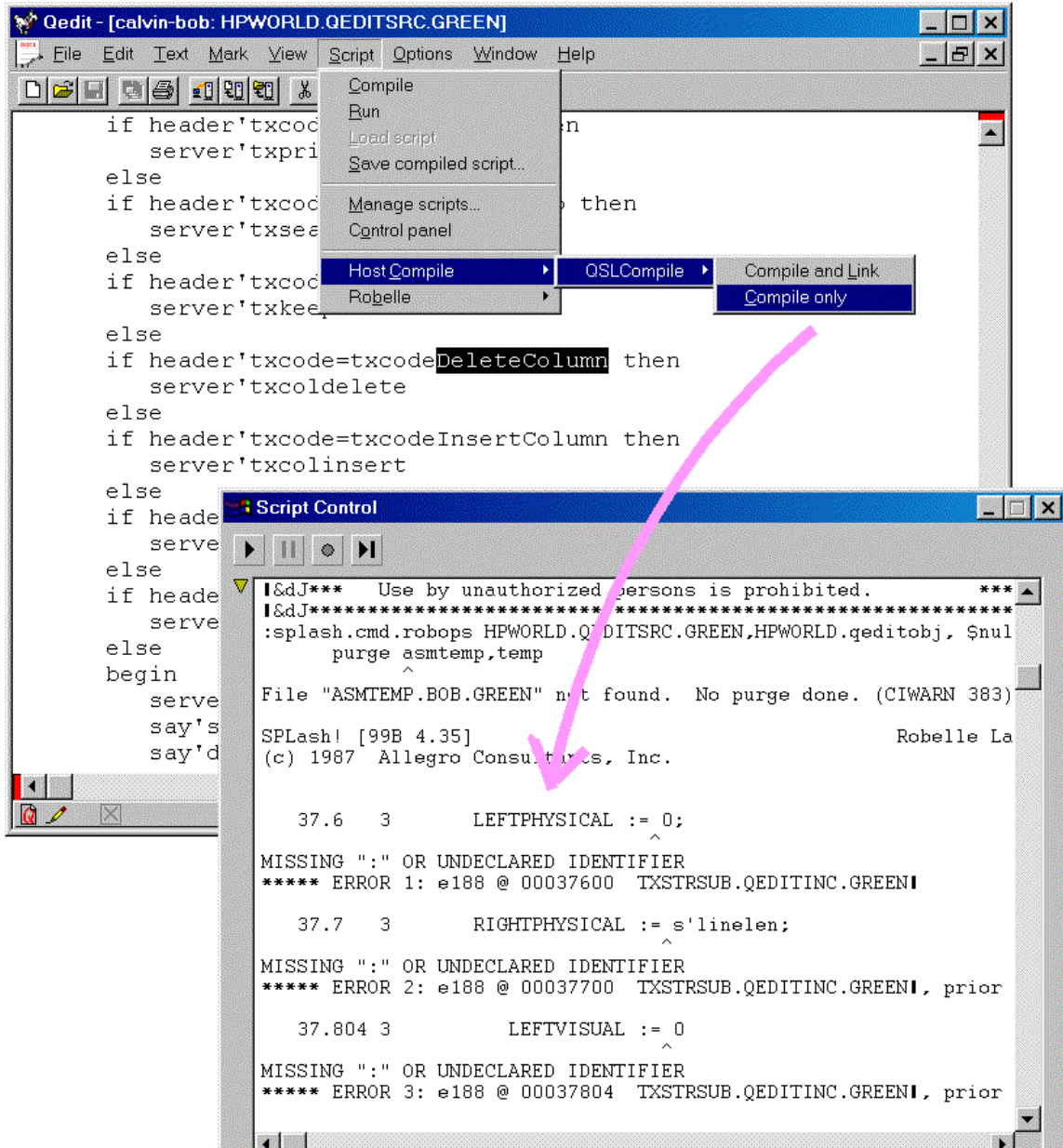
The source code lives on the host servers, not the PCs. In this way there is less chance of conflicting changes by team members. Qedit warns you if you try to save a file that

14

someone else has modified since you started editing. And for Qedit work files, only one editor at a time is allowed.

We have source code and testing facilities on both the proprietary MPE platform and the UNIX platform. The executables are created and installed automatically by batch scripts that execute after every development cycle (often daily or more often).

We did all our server development using Qedit for Windows, which allows editing source on all our host systems, both UNIX and MPE. Alternatively, we could have downloaded the source files to our PC, edited them locally, then uploaded the revisions.

In addition, the latest Qedit for Windows can execute host commands, making it possible to do host compiles from within the PC editor. Once syntax errors are detected and corrected, you link the executable on the host with Qedit for Windows.

Automated Testing Over the Internet

- Use TCP/IP ports to access test versions.
- Store test scripts as Web pages.
- Test suite tries all options and boundary cases.
- Do client/server testing to check timing and network code.

Once the new server is compiled and linked, it is time to test it with the latest client.
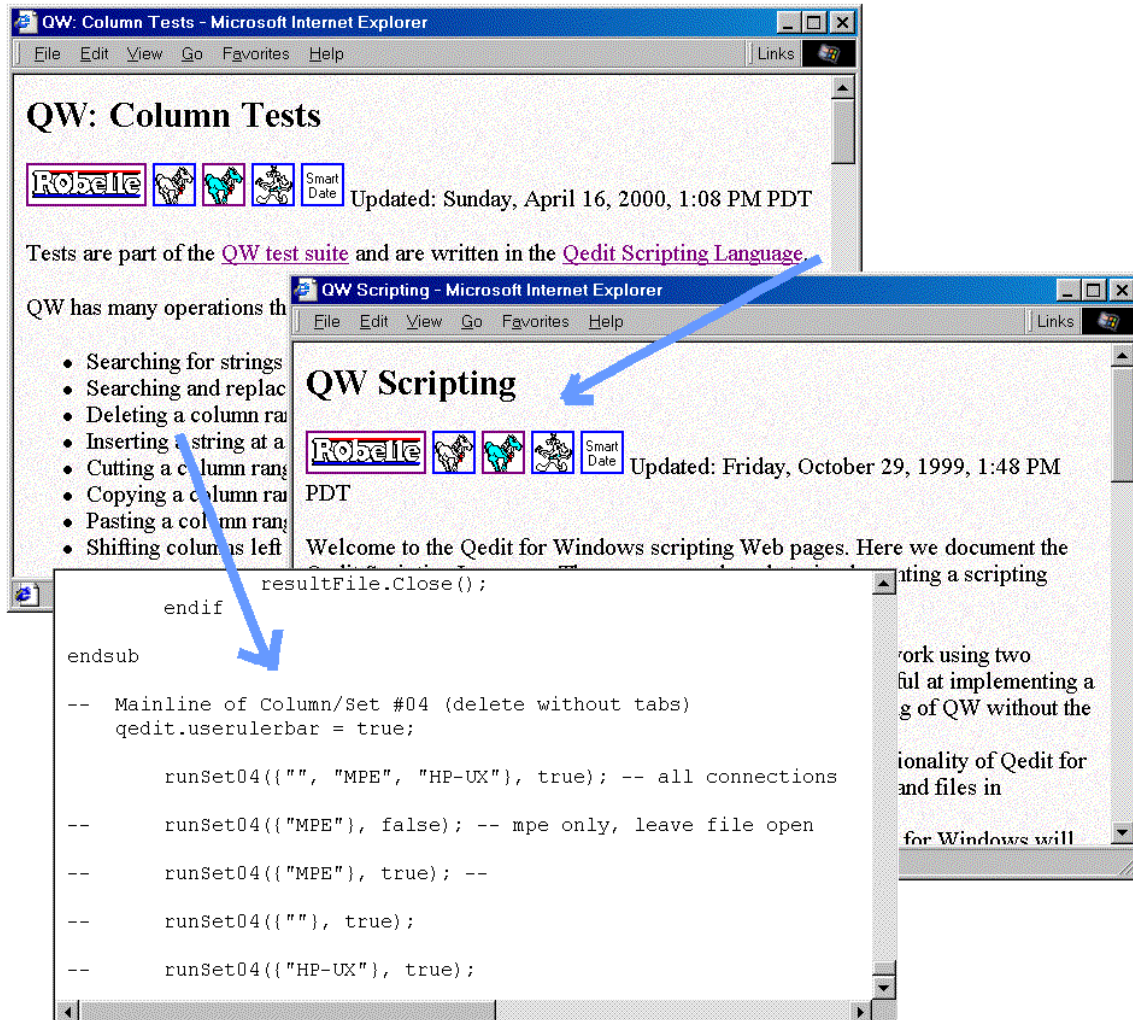
Since we are also using the project software in production, there is a potential problem: how to test the latest release when other people are actively using the previous release. We solved this problem by allowing you to have several Qedit servers running on the same host. On UNIX, you specify which server the PC client should connect to by specifying a TCP/IP **port**. A port is like a mailbox for communication and we can have several Qedit UNIX servers "listening" for connection requests on several ports. For MPE the PC specifies the file name of the server, so you just have several versions with different file names.

16

To obtain a number of benefits, including repeatable, controlled testing of the product, we added a Scripting Language to Qedit. The Scripting language allows for direct calls to the underlying methods that change the text and is very similar to many other programming languages. Tests of new features are coded in the Qedit Scripting Language, and the QSL test scripts are stored on the Web server, with documentation in Web pages.

The tests allow us to simulate manual testing in a highly accurate way, over a real Internet connection with a real PC. The goal is a **test suite**, consisting of scripts that verify all the basic functions of the system, plus as many of the boundary cases as we can think of.

When implementing new features, we often write a test suite script to help define how we expect the feature to work, then modify the script if necessary to record how it actually works when finally accepted. This was the case with the Delete Columns project.

 By using a script, you can perform many more tests per release cycle than if you had to do them by hand. Manual testing is tiresome and boring and it is difficult to test all the old features whenever a new version is released. For example, with a test suite, it is possible to repeat the same tests precisely on Windows NT, Windows 95, Windows 98 and Windows 2000, something that would be extremely difficult to do manually.

A thorough test suite exercises all major options of the program. This is not as difficult as it sounds. Notice the script in the graphic above. It has calls to execute the tests on local files, UNIX files, and MPE files. In all three cases it is executing the same test code. Then it repeats the tests for multiple file types, such as files with trailing spaces retained and files with trailing spaces dropped.

Because the test suite lives on the Web and can be executed over the Internet, it is possible to have developers and testers in any geographic location execute the current test suite at any time.

Documentation, Installation and Distribution

- Documentation is essential too.
- Then the rest of the staff and beta users.
- Automated updates and distribution.
- New version available on the Web every month.

Once the Delete Columns feature is designed, coded and tested, it is ready for internal use by the R&D staff. But before it can spread beyond R&D it must be documented.

Before the Delete Columns enhancement is done, it must be documented. For daily documentation of changes we use Qedit for Windows on our lab diary, but for the end-user documentation we use Doc-to-Help from Wex Tech Systems, Inc. (www.wextech.com). We maintain a single Microsoft Word document from which Doc-to-Help can generate a WinHelp file and/or HTML files. Since the source document is a Word document, the documentation can be printed or written to PDF file using Adobe Acrobat (www.adobe.com). Although the editing is all done on a PC, we use the Web as our primary distribution method, enabling us to have current documentation bundled with the newest release of the program.

Wherever possible, automated scripts install and distribute the digital results. For example, here is a QSL script to install a new version of ColumnMunger() source from the C++ file on a PC to directories where it resides on our UNIX and MPE development machines.

```
<<   Colmungeupdate.qsl:

        Update the server copies of ColumnMunger() from the PC-based copy.

        This script assumes that you are in the CWD of the PC-copy of colmu
>>

sub docopy(localfilename, serverconnection, serverfilename)

        localfile = open(localfilename);

        localfile.select(startline: 1, endline: localfile.linecount);

        localfile.copy();

        serverfile = open(connection: serverconnection, filename: serverfi
```

The updated documentation, executables, PDF files and help files are packaged up and an Installation build is made using a combination of Windows Scripting Host, ftp and InstallShield tools. The script that does this is invoked through a single command and can

automatically deliver the final products over the Internet to our Web Site and Media making machine for final production CD cutting.

The MPE and UNIX servers for Qedit for Windows are also delivered automatically when they are updated, for inclusion with the CD. The installation process is smart enough to detect when we are including server files for both HP-UX and MPE. They can be installed over the Internet or our Intranet, directly onto the servers from the PC installation process to any host server.

Once the new release is installed, distributors and beta-test users can download it using the Web, and Robelle staff can make new CDs for distribution by conventional methods.

We send the new release to the customers that asked for the new feature, as recorded in our Knowledge Base system (remember the original customer request from several sections ago).

When we need additional testers, we use another Internet resource. We post a message to the **Robelle-L mailing list**. A mailing list is an email message distribution system to which users can subscribe. Subscribers receive each message posted by any other subscriber (or just a daily summary of the postings) and can post replies to any message for the rest of the group.

The Future

- Browser-based editing of Web pages.
- Integrated email notification/collaboration.
- eServices: project management Web sites.
- Web links between Web project pages and project data.

The Future of Project Management using the Web holds great promise. There are already many Web tools, such as discussion forums, to assist collaboration. One of the most unique is

Crit, an open source method for annotating other people's web pages.
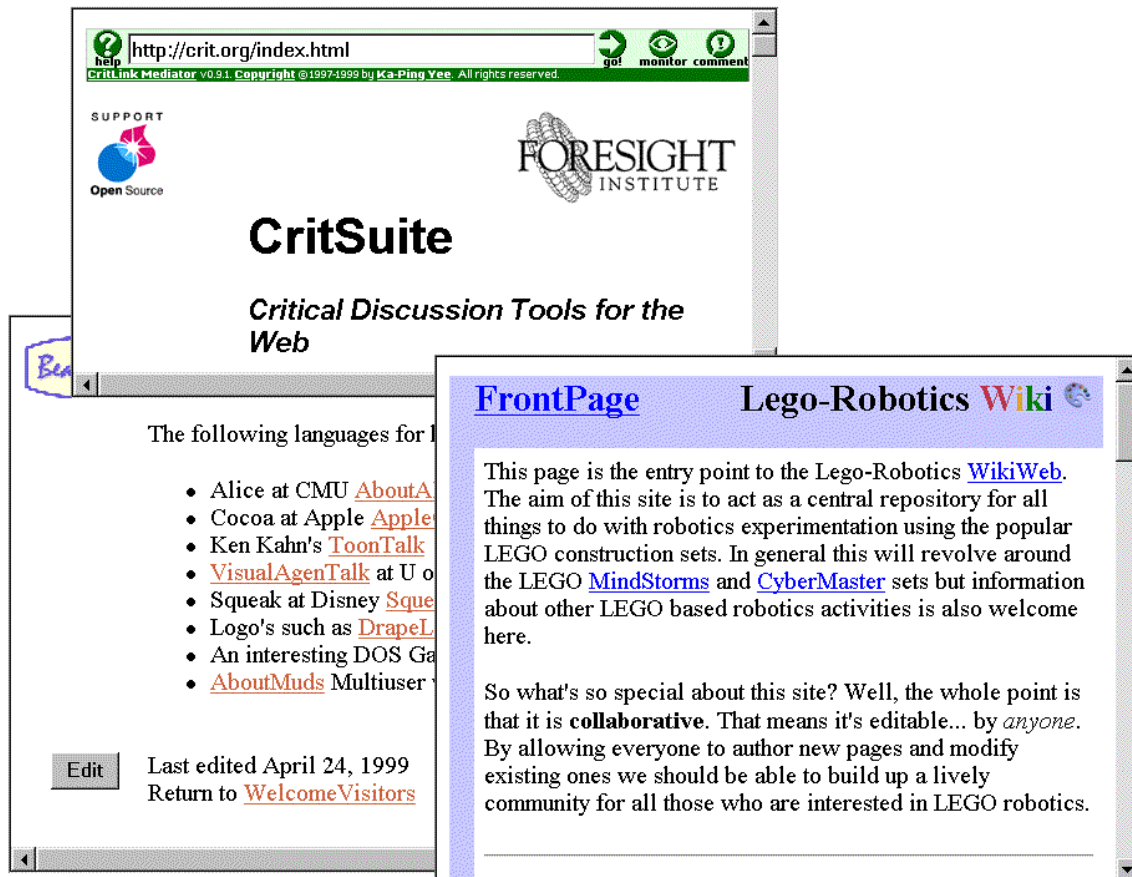
CritSuite, as it is now known, allows you to add notes to an existing Web page. The notes are stored on a Crit server, which merges the notes with the original page when it serves the page up to your browser. Someone can post a draft proposal on their Web site and colleagues can criticize it.

After you annotate a Web page with Crit, "text relevant to the annotation will be surrounded by small colored markers. The colors indicate the nature of the comment: green for support, red for an issue, blue for a comment, or orange for a query. The markers also contain small symbols to help them appear more distinct: a plus sign for support, a minus

sign for an issue, a hash mark for a comment, or a question mark for a query." The Crit software and a working Crit server are available on the Web at http://crit.org

WikiWiki

Ward Cunningham's WikiWiki database was the first collaborative browser-based tool I used. Using only your browser, you create a database of Web pages, each of which has an Edit button at the bottom. When you click on Edit, it pulls the page contents up in a browser textbox for you to edit. There is a very simplified markup language that allows bold, italic, horizontal rule, http links, and inline graphics. You create links to other WikiWiki pages by running the words of the title together, as in SoftwareTestingTools or HpWorldPapers. To create a new Wiki page, you create a link to it by title, then click on the link and an empty textbox appears.

All recently added or changed pages are listed on a Recent Changes page, making it easy to catch up on team progress after a break. You can use email to create new pages or append notes to existing pages – just put the page title in the email subject area. WikiWiki is very simple, very elegant and very flexible. A reasonable starting point for WikiWiki is the original at the Portland Pattern Repository: http://c2.com/cgi/wiki (for more information, see the References).

Another option that is likely to be widely adopted in the near future is the Web-based project management system. In this case, the software to support Project Management and scheduling runs on a shared eServices host. All you need is a browser. And if the systems encourage Web links, you will be able to integrate this date-oriented automatic processing with the content-oriented project Web pages that I described earlier in this paper.

iTeamWork

http://iTeamWork.com is the first of the Web-based project managers that I noticed. It is simple, free, advertiser supported, and makes good use of email. Users are identified by their email address, plus nickname. The first time you assign a task to a new person, they are invited by email to join iTeamWork. Projects and tasks do not currently allow links to external Web pages, but you can request an email summary of your outstanding tasks to be produced on specified days of the week.

eProject

http://eProject.com looks a great deal like the MS Outlook email/calendar system. You can set up teams and projects and assign tasks to people. eProject is more powerful than iTeamWork, but a little harder to use, and it doesn't generate a daily email summary of your work to do (you have to go to the Web site). You can upload documents to your Web-based project for sharing with the team. You can put Web links into project descriptions and they are useable links when displayed. Each project has a home page that is either generated for you by eProject or linked to a Web page on your own site.

WebProject

http://www.wproj.com is the most traditional of the Web-based systems I found, much like MS Project, and is implemented through the use of Java in the browser.

WebProject is a lot more complex than the other two and includes threaded discussion lists, virtual project status meetings you can attend in chat-mode, tracking time actually spent on tasks, integrated document management, cross-platform server, plus export and import to and from all major project management systems. You can run the rent WebProject as an eService on their servers, or run the software on an in-house server.

The future of project management on the Web is unlimited. Consider just two examples – XML for projects and the Mesa/Vista system.

XML Standard for Project Data

Pacific Edge Software is working with eProject, Great Plains and other project software suppliers to define the XML (Extensible Markup Language) schema for project management. This will someday allow intelligent interchange of project information among different systems.

Mesa/Vista Project Manager

Mesa/Vista Project Manager is a high-end product that provides process integration rather than project management. It inter-operates with many of the major project management systems, bringing their capabilities to the Web and enhancing them as well. The advantage of this approach is that it raises the prospect of a general solution that links industry-specific tools such as CAD with project management tools. For more information, check http://www.mesasys.com

Conclusion

With the simple Web page techniques discussed in this paper and the many Web-based tools coming on line, there is no good reason to put off using the Web to help manage your projects. Start today.

Trademarks

Qedit and Suprtool are trademarks of Robelle Solutions Technology Ltd., Windows, MS Outlook, and MS Project are trademarks of Microsoft Corporation, InstallShield is a trademark of InstallShield Software Corporation, and Mesa/Vista is a trademark of Mesa Systems Guild. All other trademarks belong to their respective holders.

References

Project Management:

Sippey, Michael. "Pyra's Killer App". Stating the Obvious, May 1, 2000, Web site: http://www.theobvious.com/archives/050100.html

Gilbert, Alorie. "Online Collaboration Tools Help Simplify Product Design". Information Week, April 24, 2000, Web site: http://www.informationweek.com/783/collaborate.htm

Bennett, Gordon. "Working Together, Apart The Web as Project Infrastructure." IDM Web site: http://idm.internet.com/features/idm0398-pm1.shtml

Bumbo and Coleman. "Case Study: Collaboration for Distributed Project Management." http://www.collaborate.com/publications/case2.html

Apache Web Server: http://www.apache.org/
Apache for e3000: http://jazz.external.hp.com/src/apache/index.html
Apache Server Side Includes: http://support.pair.com/tutorials/ssi.html

HTML: http://beachtech.net/webclass/getstarted.html

WikiWiki database: http://c2.com/cgi/wiki
Wiki Clones and variations, such as: http://wcsb.org/~swain/php/wiki/
Dolphin Wiki system: http://www.object-arts.com/cgi-local/view.pl/Dolphin
Dolphin Wiki at LEGO Robotics: http://www.object-arts.com/cgi-local/view.pl/Lego-Robotics

XML for Project Data: http://www.pacificedge.com/pr24.htm