

## Communication and Coordination in Small Teams

Today, more and more software is developed “in the small”—by small companies (less than 200 employees) with small teams (less than 20 members). When I worked at a large corporation, many of my co-workers expressed a desire to work in a small team, where communication is direct, feedback on personal initiative is immediate, and dependencies are understood. Many opportunities are associated with working in a small team:

- ❖ **Simpler communication, coordination and decision-making**
- ❖ **Strong ownership of individual work products**
- ❖ **Rewards for talented engineers through breadth of exposure to new technologies**
- ❖ **Fewer dependencies that are better understood**
- ❖ **Increased mutual accountability**

Practicing software development in the small has its challenges. Taking a closer look at the benefits reveals that what is viewed as positive from the individual's perspective may turn into a negative as seen by the company. Strong ownership can turn into a strong dependency on a single engineer if cross-training is not practiced. Formal documentation may be neglected when direct communication is more convenient.

Members of small teams and large organizations need to communicate the content of their work, the events of initiating or completing them, and the decisions they make, so that others who need to know can proceed with their work.

Independent of the group's size, it is important for the producer and consumer of the information to reach each other in a timely manner.

### INSIDE THIS ISSUE...

- 1 Communication and Coordination in Small Teams
- 3 Setting Up Microsoft Project Central

For example, Susan developing the client application needs to know the interface specification for the requests that Bob implements in the server. Doug, who is responsible for preparing the test cases, needs to know when the use cases are scheduled for completion.

In addition, face-to-face communication is becoming less common. Large organizations are familiar with the challenges of communicating and coordinating work across divisions and geographic boundaries. Telecommuting is creating a similar situation for small teams.

### Communication

During the software development life cycle, a wealth of information gets produced that needs to be communicated. Some information is more permanent, such as requirements specifications, design specifications and test cases. Other information is short-lived and relevant only during a particular development cycle, such as status meeting minutes and build reports.

The specific information that needs to be communicated depends on the goals of the community. For example, a program management team responsible for coordinating all products needs to store and access the product feature lists scheduled for the next release and their dependencies. A development team at a small company needs a central, accessible place for people to share and communicate informal design documents or drafts of product documentation.

Whether the information is permanent or short-lived, it needs to be easily accessible to any number of individuals who are not necessarily known when the documents are prepared. Permanent engineering documents are frequently maintained in version control systems. Often, a common place to store informal or short-lived information does not exist.

Finally, other kinds of information enable community members to communicate and coordinate their work, such as a company directory, an automated defect tracking tool, or an automated project management tool. For large organizations, it is necessary to provide easy, online access to the company telephone directory. Telecommuting makes online access to information and coordination tools relevant to small teams.

## Coordination

Coordination involves agreeing on when something is needed versus when something is available. People in the trade may think of this as how and when something gets “thrown over the fence.”

Coordination involves:

- ❖ **Agreeing on the steps for transferring work products, i.e., a workflow**
- ❖ **Agreeing on the acceptability of the work product**
- ❖ **Preparing for the hand-off**

For example, in preparation for handing off the software for QA testing, the engineering team may provide a technology transfer session sharing their tricks of the trade for setting up and troubleshooting the software. During the testing phase, an automated defect tracking tool defines the workflow for coordinating the testing and engineering work involved in fixing defects.

Communication is the most common means for coordination. Communication entails:

- ❖ **Notifying others regarding the completion of a work product**
- ❖ **Notifying others regarding decisions that alter the content and completion date of work products**
- ❖ **Giving others insight into the progress of work they depend on**

Task completion is captured in the project schedule, and the status of deliverables is communicated in project meetings. Increasingly, automated project management tools provide groupware features for communicating task assignments, schedule updates and status reports.

Small teams and large organizations need to document and share decisions that impact the product scope and project schedule. Also, the status shared in project meetings and documented in the minutes is of interest to a larger community whose members are not necessarily present at the meetings.

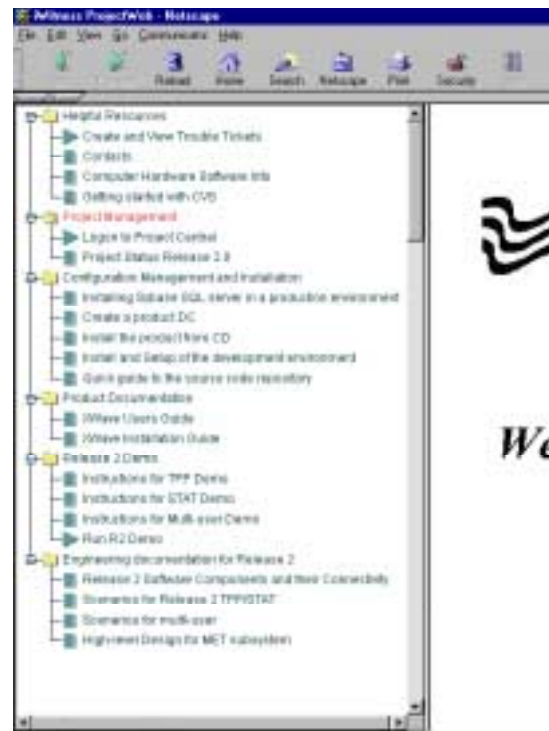
## Using a Project Web Site for Communication and Coordination

Studies observing software development teams have identified opportunities for automated tools to assist software teams with communicating and coordinating their work (“Coordination in Software Development,” Robert E. Kraut and Lynn A. Streeter, *Communications of the ACM*, March 1995/Vol. 38, No. 3; “A Field Study of the Software Design Process for Large Systems, Bill Curtis, Herb Krasner, and Neil Iscoe,” *Communications of the ACM*, November 1988, Vol 31 No 11).

Increasingly, members of software development teams work in different locations. This situation intensifies the need for automated tools supporting effective communication and coordination. Technology, such as HTML, Web servers and the Internet, has created the opportunity to make electronically stored documents and automated tools accessible to members of small and large teams—no matter where they are located.

A **project Web site**, which provides access to information needed throughout the software development cycle, helps small and large teams with their communication and coordination.

The following is a sample table of contents for a project Web site:



The RequisitePro product team at Rational Software introduced me to a project Web site with information similar to that shown above. I very quickly experienced its productivity benefits in a large organization. A project Web site offers benefits to small teams by providing access to contact information, a help desk application, a defect tracking tool, high level design diagrams, product demos, and more.

As communication and coordination are their primary responsibilities, project managers are typically the owners and maintainers of the project Web site. The site allows managers to reach the project community easily and in a timely fashion, no matter where they are located. As a project manager, I find that I frequently review information stored on the project Web site to prepare work and remind myself of past decisions.

Although the project Web site is a tool for connecting team members in different locations, it also assists project managers with “managing by walking around.” The site allows managers to check in with team members to point out new and relevant information, to inquire about their information needs, and to solicit work products to be posted on the site.

Ideally, the site is not a passive tool. It notifies people regarding work product changes and their completion. This can be achieved by providing subscriptions to e-mail lists so team members are notified about the completion of an event, such as a particular software build. The Web site also allows subscribers to find out when document modifications have occurred. In addition, the site may provide access to electronic discussion forums on topics of interest to the development community. These forums allow people to tap into the wealth of knowledge spread throughout the organization and the software development community at large.

In summary, a project Web site is an ideal place to:

- ❖ **Store short-lived or informal documents resulting from collaboration across functions or within a function (feature dependency matrices, informal engineering documents, etc.)**
- ❖ **Provide access to tools for defect tracking, project management and more**
- ❖ **Provide access to permanent engineering documents if they are not stored in a version control system**
- ❖ **Make resources, such as the help desk, accessible**

## Setting Up Microsoft Project Central

In order to use Microsoft Project Central, the following software needs to be installed:

1. Project2000
2. Project Central
3. Microsoft Internet Information Server (IIS)
4. SQL Server or MSDE

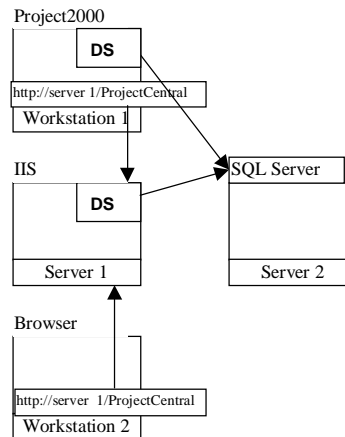
P2E manages projects and we provide training.  
 Whatever you need! Learn more about P2E.  
 Call 303-499-7301,  
 visit <http://members.aol.com/UllaMerz>,  
 or E-mail: [UllaMerz@aol.com](mailto:UllaMerz@aol.com).

## Getting Started

Before you get started, make sure to print the document **srvsetup.htm**, which you find in the <drive letter for CD drive>:\pjcntrl\help\1033 directory on the distribution CD. The document describes how to setup Project Central.

### **It is a must read!!**

The following is a diagram that describes where to install the different components for using Microsoft Project Central and how the components are connected:



Also, here is a description of the different components that need to be installed and the sequence to install them:

### Installing IIS

No special setup or configuration is needed.

### Installing SQL Server

Install SQL Server. Then follow the instructions for creating a default database and tables for SQL Server or MSDE as described in **srvsetup.htm**. Choose “sa” as the account name and specify the necessary password. The database name for storing Project Central data is “Project\_CentralDb.”

Now, use the Enterprise Manager for SQL Server and create another database for storing the data that is commonly stored in an .mpp file. You may want to call this database “Projects.”

Next, define two new logins, one for the “Project\_CentralDb” database and one for the “Projects” database. For example, you may create a login called “pjcentral” and then add the login as a user to the “Project\_CentralDb” database. Assign this user the role of db\_owner and make the “Project\_CentralDb” the default database for the login. Create a login called “mspj” and add it as a user to the “Projects” database. Again, give it the role of db\_owner and define “Projects” as the default database.

## Installing Project Central

Run the custom installation. During install:

1. Specify the name for the web directory in IIS (e.g., ProjectCentral)
2. Specify the database server name
3. Specify the database (e.g., Project\_CentralDb)
4. Specify the login (e.g., pjcentral)
5. Specify the password

If you experience access problems to project data stored in SQL Server, request the CreateSite.exe tool to reset the security settings on the virtual directories in IIS that contain the ProjectCentral files.

## Installing Project2000

Install Project2000. If you install Project2000 and Project Central on the same NT server, always install Project Central first.

## Defining Connectivity between Project2000 and Project Central

Specify a DSN on "workstation 1," which allows Project2000 to save data that is commonly stored in an .mpp file in the "Projects" database in SQL Server. For example, choose the name "Projects" to identify the DSN. The DSN has to be a system DSN.

If Project2000 is installed on NT 4.0, then select Settings, Control Panel and double click on ODBC. Specify a System DSN with SQL Server as the driver. Enter the login and password and set the default database to "Projects." Test the connectivity before you exit.

Now create a project file and store the data using Save As and select the ODBC... button. Specify the previously defined DSN, select entire project and choose a name for the project schedule.

To establish connectivity between the project schedule data and Project Central, select Options from the Tools menu and click on the Workgroup tab. Select Web for the default Web Server messages and enter the URL for the Web server settings, such as <http://<server 1>/ProjectCentral>. Select the check box of Updating Project Information to Microsoft Project Central on every save.

## Defining Connectivity between Project Central and SQL Server

Specify a DSN on "server 1," which allows Project Central to access the project schedule data stored in the Projects database in SQL Server. The name for the DSN has to be identical with the name of the DSN specified on "workstation 1." See instructions above for defining a DSN.

When you update the Project Central Server either by selecting Save or the Workgroup, Update Project to Web Server function in the Tools menu, the DSN that was used for saving the project schedule in SQL Server was transferred to Project Central. Log into Project Central as the Administrator and select Manage Views under the Admin tab. There, select Data Sources for Views, which displays a list of data source names that have been used for storing project data. Select the DSN and choose the modify function. Enter the login and password (for example, mspj, which was defined as the db\_owner of the Projects database). Select Save Changes.

## Additional Help

If you need additional help, go to the discussion group/forum [Microsoft.public.project2000.projectcentral](http://Microsoft.public.project2000.projectcentral). Most likely you will find an answer to your question or problem when you browse through the existing entries.

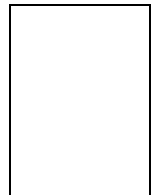
---

Ulla Merz

**P2E**

5501 Aztec Court  
Boulder, CO 80303

ADDRESS CORRECTION REQUESTED



**INSIDE...  
TIPS & INFORMATION FOR  
SOFTWARE PROJECT MANAGERS!**