

Chapter 3

Planning a Dynamic Site

This will inevitably happen to you.

You are sitting at your desk, and the phone rings. When you answer it, the individual, from the Oakbridge Recreation Center, will start a conversation that goes something like this:



“Do you build web sites?”

Your answer is, “Of course,” and then you talk about the work you have done for other clients.

“Here’s what we need,” says the voice on the phone. “We need to rethink our site and are looking at adding a couple of features that cement our relationship with the community. They would include an interactive tour of the facility based on interest. We would also like to streamline our current facility booking procedures and enable people to book sports facilities or meeting rooms online. Finally, we do a lot of community work and want to create something on the site that enables people to meet online...or something like that. Do you do that sort of thing?”

This is typically the start of the process, and a positive answer will result in a client meeting. The project’s success or failure is dependent upon what you do next.

In the good old days of the Wild, Wild Web, up to about 1998, the answer was a resounding, “Yes.” In those days, the developer met with the client, got a rough idea of what was needed, and gave the client a rough idea of what he or she was going to do. The client, not having a clue what the developer’s proposal entailed, inevitably bought in because it sure sounded “cool.” At that point, the developer bought the necessary software and spent the next month learning how to do what he or she had just promised the client.

With the rise of dynamic sites and the introduction of the work group and some pretty sophisticated software, the technical and creative demands placed upon web developers have increased. This means the day of the “One-Guy-Does-It-All” shop is a thing of the past. In today’s production environment, speed and accuracy are critical. This means projects can only be completed by a team of specialists working with a carefully crafted production and project plan.

The old business adage, “Plan your work and work your plan,” is more appropriate than ever. More often than not, that means the production cycle doesn’t start with pixels lighting up on a screen. It starts with a sheet of paper.

Want vs. Need

When talking to the client, a key skill to have is the ability to distinguish between a “want” and a “need.” All too often, clients and developers get caught up in the technology and the software when they start discussing the project. A huge number of Flash MX 2004

sites out there could have been done more easily in Dreamweaver MX 2004. Why Flash? The client, discovering how cool it is to have a Flash site, decides they want one.

The problem here is technology is constantly changing and can't be fixed in place. A great example is the MX Studio. In April of 2002, we were all using individual Macromedia products. Six months later, they were completely reinvented as the MX Studio of integrated products that also included a developer edition of ColdFusion MX. Director developers planning a number of projects around the Multi-user server discovered, with the release of Director MX in late 2002, that the Flash Communication server replaced the Multi-user server. A valid question, therefore, is, "How long will the online meeting facility be available through the site, and will it be able to adapt to changing technologies?"

In the case of our community center, they want an online meeting room. This room will be used by members of the clubs or other associations based in the community center to get together online to discuss plans, events, and so on. It should have a shelf-life of at least two to three years, which means the enabling technologies—ColdFusion and Flash—will most likely change, but the upgrades and changes are easy to effect. Then again, there is an even more fundamental question that should be asked.

The question is, do they need such a facility? Ask that question of both yourselves and the client, and the production process takes on an entirely new dimension.

Defining Exactly What Is Needed

The last thing a web developer needs is the client changing his or her mind mid-project. This situation usually results from the clients' discovery that the project is moving in a direction they never expected. Even worse is the client discovering they have a technologically-advanced site their market either can't or doesn't use.

In the case of the Oakbridge Community Center asking for an online meeting room, it is interesting, but do they need it?

In this situation, it is incumbent upon the developer to break out a pad of paper, meet with the client, and start doing some very careful fact-finding. The questions that could be asked are:

- Have your clients expressed a need for this?
- What sort of market do you serve?

- Do you have any idea about what kind of connection they have to the Internet? Broadband or dial-up?
- Are they technologically sophisticated enough to use video cameras connected to their computers?
- Does your ISP have the necessary software to accommodate dynamic data transfer?

The key here is discovering whether a full-bore interactive area with streaming video, audio, and chat is needed, or whether the client would be better served with an instant messaging feature.

Never forget, your client is rarely as technologically adept as you are. Thus, your role is twofold: web developer and educator. You will be building the site but also translating the technology to the client and presenting it in terms he or she understands. If the client needs instant messaging, recommend it. Although instant messaging isn't as "cool" as a Chat Room built in Flash, you will have saved your client a serious amount of money that would have been spent on unnecessary technology.

Developing the Project Scope

When you understand what is needed, there is an understandable urge to start up the computer and get to work. This would be a huge mistake. The only software you will need at this point is word processing software. It is used to produce a document that lays out the parameters of the project, including items such as a budget, deadlines, a creative brief, and a technical specification. This is called the project scope. It should be regarded as a deliverable for the client to sign and date.

By producing this document and having the client sign and date it, you and the client reach an agreement regarding exactly what is to be produced, by when, and at what cost. In this manner, you avoid scope creep.

Scope creep is composed of those changes that, at the time, appear insignificant, but when added up can actually result in charges that can increase the original budget by a factor of twenty percent or more. For example, the decision is made to use JavaScript rollover buttons for the navigation. Midway through the process, the client decides they should be done in Flash. In the total scheme of things, the three hours to make the change seem insignificant. If you have a number of them, though, they can have a huge, negative impact upon the final bill, potentially culminating with the involvement of lawyers and lawsuits.

To avoid this potential messiness, create the aforementioned written document and include the fact that changes, additions, or deletions through such things as cancellation penalty clauses in contracts will incur extra charges. Some agencies go as far as setting a limit for these charges, at which time they will actually stop work and renegotiate a new agreement with the client.

Identifying the Technologies

There are a number of technologies that can be utilized in the building of a dynamic site. This book's topic suggests you already own Studio MX 2004. This means you have a serious number of very powerful tools available to you. From Dreamweaver MX 2004 to ColdFusion MX, your production needs are met. Still, the decision regarding which technology to use to meet the needs of the client requires some serious consideration and trade-offs.

The way we chose MySQL as the database for this book is a good example of the considerations and trade-offs you will have to make.

When we first considered producing this book, we assumed we would be using Microsoft Access as our database of choice. We had used it in our previous book—*Building Web Sites with Macromedia Studio MX*—and we just assumed we would stay with it. The reasons were valid:

- Microsoft Access is easy to use.
- It is widely available because it is bundled in the Microsoft Office Suite.
- ColdFusion can quickly “plug into” it.

The times have changed between our first book and this one. When we started looking at the market—our beloved readers—we came to the conclusion that Microsoft Access was not a suitable choice. For example:

- Microsoft Access is only available on the PC. Even then, not everyone owns it. The likelihood of someone purchasing it just for the sake of following the exercises within this book is rather slim.
- ColdFusion MX is widely available on many platforms. Mac OS X, as well as various J2EE platforms, can work with ColdFusion MX.
- ISPs accommodate MySQL. Our site's host, NI Solutions in Toronto, offers MySQL in addition to the package of technologies available to us and, by association, to you. However, Access was not mentioned as part of their packages offered alongside ColdFusion.

When you ask those questions, the number of technology options tends to narrow to one choice. That choice is MySQL. Here's why:

- It is freely available under the GNU General Public Licensing. This simply means that it is free for regular users as long they don't need to incorporate it in a commercial application. Go to <http://www.mysql.com/downloads/index.html> and download and install MySQL Public release 4.0 for your operating system. Mac OS X users can also download a complete installer from <http://www.aaronfaby.com/>, which also contains a System Preferences panel to control the MySQL daemon.
- Various GUI (Graphic User Interface) MySQL management tools are available. There are a number of them out there, but for the purposes of this book, we have decided to go with YourSQL by Magister Ludi (<http://www.mludi.net/YourSQL/>) for Mac OS X. PC users can use <http://www.mysql.com/downloads/mysqlcc.html>. If all else fails, you can use the widely popular phpMyAdmin, available at <http://phpmyadmin.sourceforge.net> available on almost all platforms. Just remember that you would need to download and install PHP for this.
- Connecting to MySQL through ColdFusion MX is done using a JDBC connection. ColdFusion 5 supports ODBC and thus connection strings. Macromedia dropped ODBC in the jump to Cold Fusion 6. That means the only way to connect ColdFusion MX to a database is through a datasource name.

Dreamweaver MX 2004 Server Behaviors write SQL queries that are understood by all SQL-compliant databases. Thus, anything we do with MySQL here can most likely be used on Microsoft Access.

The Creative Brief

When one introduces complexity into the web design process, one also creates a management problem. The problem is retaining the consistency of design throughout the project. If the site is composed of images, text, artwork, video, sound, and other media, it is absolutely critical they all have the same "look and feel." This way, the site visitor is not left wondering whether he or she has left the web site. A good example would be Nike.com. The site is massive. Yet visit any page in the site, and you still feel you are on the Nike site.

The other issue you will have to contend with is search engines. You have absolutely no guarantee the visitor will start at the index or home page. Again, a consistent design ensures the visitor is not disoriented.

The solution is a solid creative brief. As we pointed out earlier, the web design process begins on a sheet of paper. The software used is usually word processing software, and one of the key documents produced will be a creative brief.

What Is a Creative Brief?

The creative team essentially receives its direction from the creative brief. The document is a written summary of the goals and objectives of the project, with very close attention being paid to the visitor. It also serves to give the team an idea of the creative direction for the project. It does this by laying out the demographics of the audience, their expectations, and the communication strategy behind the site.

This is also a valuable document for the client. It is not uncommon for clients to have unreasonable expectations regarding the design of a site. A carefully-crafted creative brief will serve to focus the client's attention on the creative goals of the site. It can also be used as the vehicle that enables the creative group and the client to agree to a common set of objectives.

The document can be as short as one page, or, if the project is complex, it can be many pages long. Regardless of size, each member of the team should receive a copy of the brief. It should also be given to the client to be signed and dated.

How The Brief Is Developed

The brief is usually the result of a number of meetings held between the developer and the client.

The document can take any form, simply because no two web shops approach the process in the same way. Still, there are some broad topics that can be covered, such as:

- **Project Objectives:** List two or three key objectives for the project.
- **Communication Objectives:** Two or three key objectives that can be measured should be listed.
- **Audience:** Who is using the site? Try to be as specific as possible.
- **Tone and Manner:** What is the site's "look and feel?"
- **Brand Idea:** What impression should visitors get from the site? A brand idea can be developed by laying out the idea, why you think it works, and the rationale.

Having wrestled with those broad topics, you can produce a brief that is acceptable to both the client and the creative team. The creative brief we will be working with throughout the book is shown in Figure 3.1.

Creative Brief

Job Title: Oakbridge Community Center

Project Objectives:

- ❖ To provide a communication opportunity between the Center and the Oakbridge community.
- ❖ To position the Oakbridge Community Center brand favorably amongst the community.
- ❖ To educate the community regarding the depth of the facilities offered by the Oakbridge Community Center.

Communication Objectives:

- ❖ To drive a measurable response throughout the Oakbridge community to increase facility usage.
- ❖ To drive a measurable awareness of the OCC brand explicitly associated with facility usage.
- ❖ To educate the general public regarding the variety and breadth of facilities and programs offered by the OCC.

Target Audience:

- ❖ Male/Female (assume a 60/40 split).
- ❖ Age: 20 to 49.
- ❖ Internet-aware professionals and community members.
- ❖ Current and former users of the facility.

Tone and manner:

- ❖ Contemporary
- ❖ Approachable
- ❖ Engaging

Brand Idea

What's the idea?

To position the OCC as being more than just a sports complex.

How does it work?

- ❖ The visuals portray breadth of the Center.
- ❖ The copy educates the visitor about the nature and scope of the OCC.
- ❖ All visuals positioned in proximity to programs or facilities.
- ❖ The design guides the visitor visually on each page of the site and directs the eye on the page.

Why is it right?

- ❖ Visitors will be exposed to the variety and depth of the OCC.
- ❖ For visitors new to the site, the copy and visuals will position the OCC as a community resource and not just a sports complex.
- ❖ The descriptive copy reinforces the community resource philosophy.
- ❖ The design conveys the key brand personality and enforces visual approachability and ease of use.

Figure 3.1 The Oakbridge Community Center creative brief will drive the design of the site.

The Technical Brief

Dynamic web sites use a lot of technology. As such, it makes sense to produce a document that gives the technical team its “marching orders.” From the client’s perspective, this document tends to demystify the process.

Let’s take a look at the key features the Oakbridge Community Center site will employ. First is an interactive tour based upon user interest. This tells us we will need to create a facility that enables those interested in indoor sports, outdoor sports, and community use features—meeting rooms and conference rooms—to quickly review those features. We’ll also need to create an application that enables the visitors to book the various sports and meeting facilities.

A new wrinkle introduced by the client is the creation of an online meeting facility. This is being included as a part of the center's efforts to broaden its reach into the Oakbridge community and to enable a variety of community groups to either communicate their message to the Oakbridge community or to hold online meetings.

Having done that, the process can be reduced even further, and we can describe and rationalize the functionality of each feature.

Rationales for the Features

The tour will be the part of the site that will receive the most visitor interaction. Visitors will need to be able to view the various programs and facilities, based upon their specific interest. After the visitor selects a category, he or she should be presented with an overview of that particular facility. If more specific information is required, the visitor can reach it with a single click.

The booking facility will perform a number of tasks. It will need to keep a running list of the facilities the visitor wishes to book. It must also include functionality to enable the visitor to change his or her mind and add or delete items. It should also include the capability to refuse a booking based on over-capacity or, if the facility is unavailable, provide a few alternative dates or times that can be booked. It should also be able to issue an email confirmation regarding the booking.

The online meeting facility should be flexible enough to accommodate all manner of users, from novices to the technically adept. Thus, it has to be flexible enough to offer services ranging from a text input chat facility to streaming video and audio. It should be able to accommodate groups ranging in size from three to three hundred. It will also contain a "help agent" that will "create" the meeting facility based upon user input.

Cool vs. Practical

The rationale, if you are on the web team, just oozes "cool." Focus on the "cool" factor and the project is doomed to failure. Never forget the web is a communications medium. This means information is being transmitted. Focusing on the technology—"cool"—and not the process—"practical"—will overwhelm the message. For dynamic sites to work, the technology has to be transparent.

The tour could be done in any number of applications, but we decided to feed the content into Dreamweaver templates using content located in or referenced by the database. This way, the information is not fixed in place and can be quickly amended to accommodate change. For example, outdoor tennis courts and baseball diamonds can easily be

removed during the winter months, and programs like “spinning” or aerobics can be added without major modifications to the site.

The booking application will be created in both Flash MX 2004 and Director MX. The reason here is to demonstrate to you how both applications can obtain and display data using ColdFusion and Flash Remoting.

The online meeting facility will be a Flash application that shows you how to add some “cool”—streaming video and audio—to the site. It will also be built to accommodate those visitors not as interested in “cool” and will offer a text chat option using the Flash Communications Server.

Timelines/Deadlines

More and more web development companies are starting to understand the importance of project management and the use of a project manager. This individual, in many respects, could be the difference between the success or failure of a project and, to be blunt, the web development company. In fact, one of the authors has gotten used to hearing the principals of many large and small web development firms tell him the smartest thing they ever did was to actually hire a “project manager.”

This individual knows the key to the successful completion of the project is the schedule, and that means he or she must get the team and the client to “sign on” to the agreed deadlines. If the schedule is properly developed, both groups will see a continuum from the start of the project to upload to the client’s server. This continuum involves timelines and deadlines and helps everyone involved to differentiate between forests and trees.

Timelines, the forest, are usually contained in a key event schedule that lays out the broad parameters of the project and the scheduled completion date for the elements of the project. Depending on the complexity of the project, this document presents the major milestones of the project’s progression through the production process as a series of weekly or monthly deadlines. This document should be included in the proposal presented to the client.

A detailed schedule—the trees—serves to move the team toward the completion of the project. By breaking the project into a daily schedule, all the major team members have a daily “to-do” list. This should be a fairly detailed document, but don’t make it rigid. Spawned from the key event schedule, this document should accommodate the fact that scope creep will occur and that tasks will inevitably be added or subtracted and deadlines will expand or contract as the project moves toward completion.



The Tardy Client

There are any number of solutions available to the project manager who has to deal with a team member who consistently misses deadlines. What about the client that misses them on a regular basis?

The solutions here range from blood oaths and weapons on the table to account resignation. To avoid this situation, make the client aware right at the start of the process that time is money...the client's money. This tends to get their undivided attention.

Clearly explain each of the project's goals and the financial implications of the client missing his or her goals for content delivery and approvals. In certain instances, it also doesn't hurt to request that the client appoint a back-up individual with the authority to approve, sign, and date documents, should the client be unavailable.

The Communication Process

It makes no difference whether a site is dynamic or static when you reach the core concept behind its existence on the web. The web is nothing more than a high-speed communications vehicle. The page being viewed will communicate with the visitor, or, in the case of dynamic sites, the visitor can communicate with the page.

If you understand that you are dealing with a communications medium, then you can also grasp the importance of clear lines of communication between all the players, from client to team member, in the development process. This communication can be anything from interpersonal communication to the trading of email. Regardless of how you choose to communicate with each other, the key is simply to communicate. The last thing a client or project manager needs to hear is, "Well, no one told me."

Client/Team Communication

A key communication vehicle for all parties is a staging area where the client can view the progression of the project. This usually takes the form of a site created in Dreamweaver.

If you are considering a client site, set up areas for the team and the client, and password-protect both areas. This way, you can control, on the client side, whom the client talks to. If properly designed, a client site imposes tight controls on the communications and avoids conflicting messages to the players.

If you do build a client site, consider adding these features:

- **Security:** Most work done involves a legally binding Non-Disclosure Agreement (NDA). To avoid infringing on the project's release date or other confidential

information, password protection—to avoid leaks and to protect the confidentiality of the client’s work—is the first line of defense.

- **Threaded Communications:** By having all the communications in one place, you have a dated record of all communications. This is especially useful in situations where a team member, for example, claims he or she never saw the authorization to proceed.
- **Contact Information:** Ensure everyone on the team has current contact information for all team members. In the case of contract members, this will include phone numbers and a valid email address. The client should provide this information on his side of the fence to the project manager.
- **Critical Path:** Make the critical path or other scheduling device available to both the client and the team.

We aren’t going to explain how to build a client site because these sites inevitably reflect the complexity of the project and the design standards of the developer. If you are looking for a couple of excellent overviews of how create a client site, Kelly Goto presents in her New Riders book, *Web Redesign: Workflow that Works*, a rather simple example, and Todd Purgasson shows in his New Riders book, *Flash deConstruction*, how his company, Juxt Interactive, uses a fairly complex site as a communications vehicle.

We are going to show you how to construct a relatively simple chat room in Flash MX 2004. This could be used for real-time communication between the client and the team or as a communications vehicle between the various members of the project team.

Creating a Chat Room in Flash MX 2004

We have been quite consistent to this point in the book. Being firm believers in “let the software do the work,” we are constantly looking for ways that make the production process more efficient.

In the case of inter-team communication, the easy way is to simply rely on email. The problem with that is there is simply no way of adding immediacy to the communication process. The only way of doing this is to pick up the phone, yell across a divider, or jump in your car and have a face-to-face meeting. The use of the Chat component in Flash adds a distinct measure of “cool,” but, most important of all, it provides a sense of immediacy and, unlike instant messaging, focuses solely around the task at hand.

This final section of the chapter, therefore, injects immediacy into the production process, and it revolves around a new feature of Flash MX Professional 2004 and an “old” feature of Flash MX. The new feature is the Project panel in Flash MX Professional 2004. The old one is the use of the Chat component in Flash MX.

The Flash MX Professional 2004 Project Panel

Until this version of Flash, the capability to share a file around a work group involved servers, “sneaker Net,” or email. This version of Flash further cements the integration of the Studio and web development workflow.

In the previous chapter, we discussed a work group aspect of Dreamweaver MX 2004—Check In/Check Out. For some very odd reason, Flash has always been regarded as a single-user product. That may have been true when it was first introduced, but the dynamic and coding features added to the application in subsequent revisions have served to make it far more complex than its VectorWorks roots. When the MX Studio was introduced, we found it odd that the new Flash workflow integration capabilities—Check In/Check Out for example—were simply not there. This has all changed. Sort of.

There are now two versions of Flash on the market. The version packaged with the Studio is Flash MX 2004. Flash MX Professional 2004 is aimed squarely at the high-end application developer market, such as those building e-commerce solutions in Flash. The feature sets between the two versions are not radically different, apart from the addition of some very powerful components, video tools, streaming media components, and the Project panel in the Pro Edition.

We include the Project panel because it is such an important addition to the application, and it is beyond us why it is not included in Flash MX 2004.

Defining a Site in Flash MX Professional 2004

Flash now includes version control capabilities on par with Dreamweaver. This includes Local/Network and FTP access protocols, as well as a plugin interface to allow for integration with version control systems like SourceSafe, WebDAV, and RDS. To get started, you need to define a site. Here’s how:

1. Open the Project panel. If the panel isn’t in your panel grouping, select Window, Project. The panel is then activated by selecting either Create a New Project or Open an Existing Project. When the panel activates, you will see two buttons. The first one, which looks like a box with a Flash icon leaning against it, is the Project button. The one beside it with the arrows is the Version Control button.

2. Click the Version Control button to open the Version Control pop-down menu. Select Edit Sites to open the Edit Sites dialog box.

**Tip**

The Edit Site dialog box may already contain a number of sites. Flash MX Professional 2004 now shares the Site Definitions created in Dreamweaver MX 2004.

3. Click the New button to open the Site Definitions dialog box, as shown in Figure 3.2. Name the site and set the root folder. Add your email address and the name that will be used to indicate you have checked out a file. Determine your connection from the pop-down list. Depending on your selection, the Connection area will expand and present you with a series of input boxes to determine the appropriate connection information.
4. When you are finished, click OK. This will return you to the Edit Sites dialog box. Your new site will now be included on the list. Click OK.

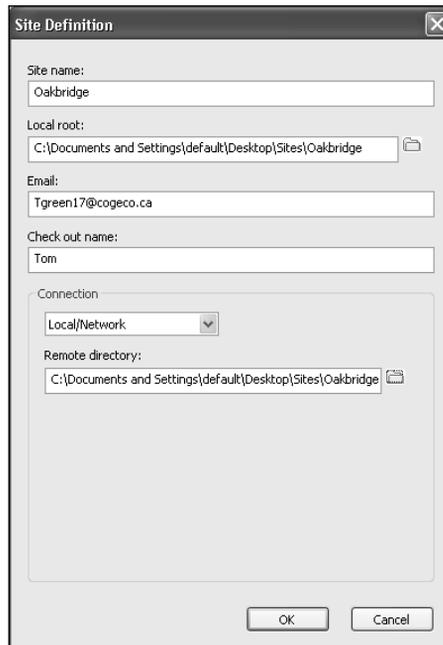


Figure 3.2 Flash shares Site Definitions with Dreamweaver, which permits easy Check In and Check Out of files among members of the team.

Defining and Working with a Project in Flash MX 2004

For the purpose of simplicity, a project is simply all the Flash files used in that particular project. Here's how to define a new project:

1. Click the Project Button on the Project panel. Select New Project, as shown in Figure 3.3, from the pop-down menu. The New Project window will open, and you will be prompted to name the project and select a location for it to be stored. Note the new file format for a Flash project, .flp.

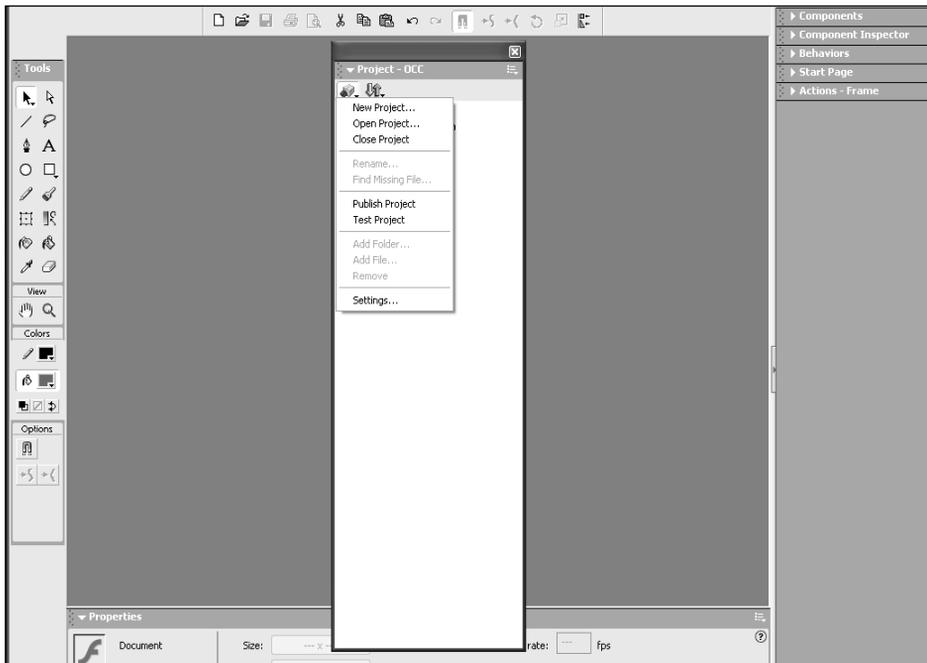


Figure 3.3 A new project is created by selecting New Project in the pop-down menu.

2. When you click OK in the New Project dialog box, the project file is created in the location chosen, and it also appears in the Project panel. Both the .flp file and the project in the panel will have the new Project Icon mentioned earlier.
3. With the project defined, you can now add files and folders containing files to the project. Select the project in the panel and either right-click (PC) or Control-click (Mac) to open the Context pop-down menu. Select Add File to open the Add Files to Project dialog box. Navigate to the file, select it, and click

the Open button. The dialog box will close, and the file will appear under the Project File in the Project panel, as shown in Figure 3.4.

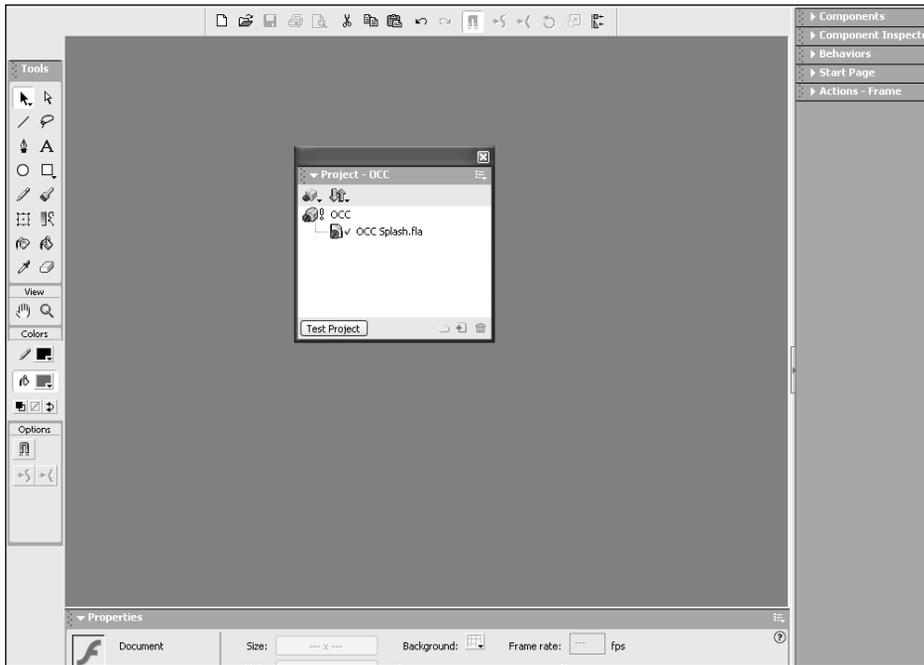


Figure 3.4 Files can be added to Flash Projects.

4. Checking files in and out follows the same procedures as those used in Dreamweaver. You select the file in the Project panel, click the Version Control button to open the Context menu, and select Check Out or Check In, as shown in Figure 3.5.

Two people can't check out the file at the same time. In case you try to check out a file already in use, you will be asked if you want to override the other person's check out. However, two people may not have the same file checked out at the same time.

The members of the team can see the status of files at all times. The green check mark indicates the file is checked out. The red check mark means that someone other than you has the file checked out. The lock indicates that the file is "in" version control but is not currently checked out by anyone, and the yellow exclamation point indicates that there is a later version of the file available in version control.

**Note**

One of the version control menus is Update. Update was added to help users of Flash Projects working on teams to be aware of changes made to files by other members of the team. When a file in version control has an older version on the local system, the update status icon will appear to inform the user that there is a newer file available in the version control system.

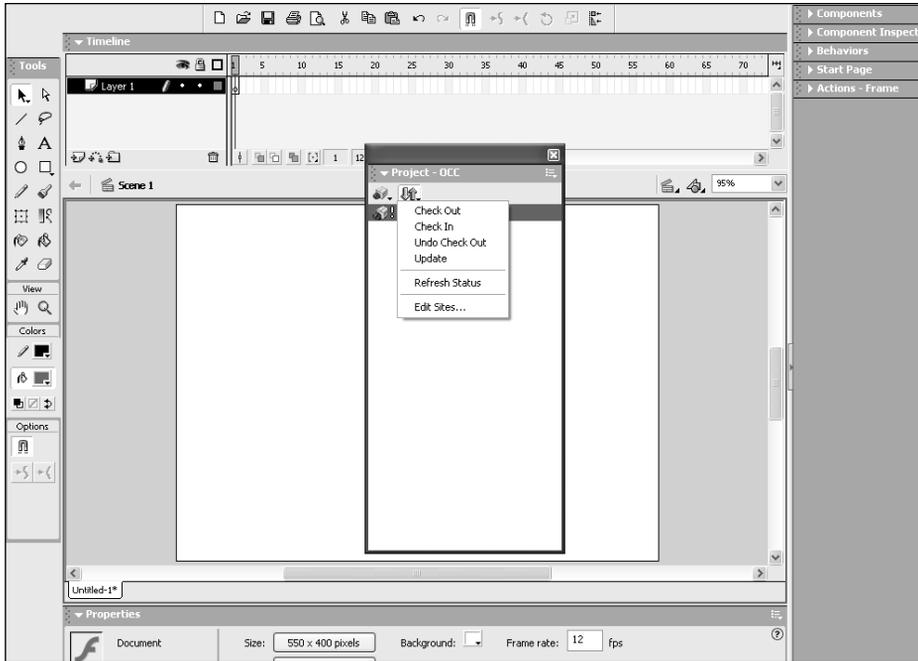


Figure 3.5 The Version Tracking menu is a key file management and versioning control tool in Flash MX 2004.

Setting Up a Chat Room in Flash MX 2004

The decision has been made to give the team an area to talk through their plans, problems, and issues. Though instant messaging and other technologies are readily available, it is decided to build a simple chat room using Flash MX 2004.

If this is your first exposure to using Flash MX 2004 and the Flash Communication Server MX, this decision could be a bit overwhelming. In fact, it is easier than you may think. Essentially, it involves three steps:

1. Write a script that connects the chat room to the Flash Communication Server.
2. Build the facility using simple Drag and Drop Components provided by Flash.
3. Test and tweak.

**Note**

Before proceeding, you should have Flash Communication Server MX installed on your PC. A free version is available to download at <http://www.macromedia.com/software/flashcom/>. You should also have IIS 5 up and running on your PC.

Connecting to the Flash Communication server is not terribly difficult. You simply have to create a folder called an application folder on the server that holds the files. Our example assumes you are using your PC as the testing server. If you are using your ISP's server, you can create a subdirectory on the server in your FlashCommMX area that will be made available by your provider.

Follow these steps to create the application folder:

1. On your PC, double-click My Computer and follow this path: `c:\inetpub\wwwroot\flashcomm\applications`. This takes you to the applications folder for the Flash Communications server on your PC.

2. Create a new folder named **OccChat**, as shown in Figure 3.6.

Having created the folder, you now have to create the script that will make the Flash Chat Room function. This can be done in Dreamweaver.

3. Open Dreamweaver MX 2004. When Dreamweaver opens, select File, New to open the New Document Window. Select Other from the list of choices in the Categories column. Select ActionScript Communications document from this document list and click Create.

4. A script document opens. Click once in line 2 and enter the following code, as shown in Figure 3.7:

```
load("components.asc");
```

5. Select File, Save As. When the Save As dialog box opens, browse to your OccChat folder and name the file **main**. Select ActionScript Communication Files from the Save As Type pop-down list. Click Save. Quit Dreamweaver.

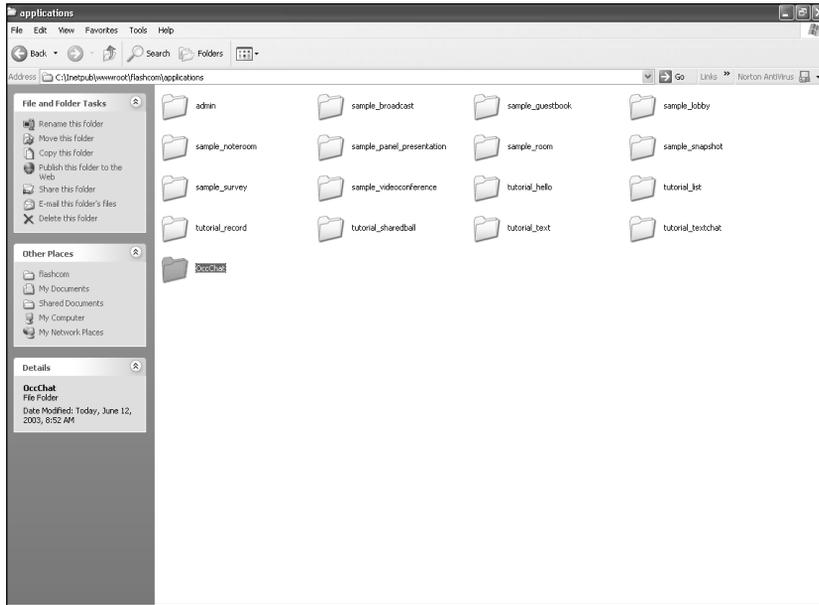


Figure 3.6 The “hook” to the Flash Communication Server is provided by the OccChat folder placed in the Comm Server’s Admin folder.

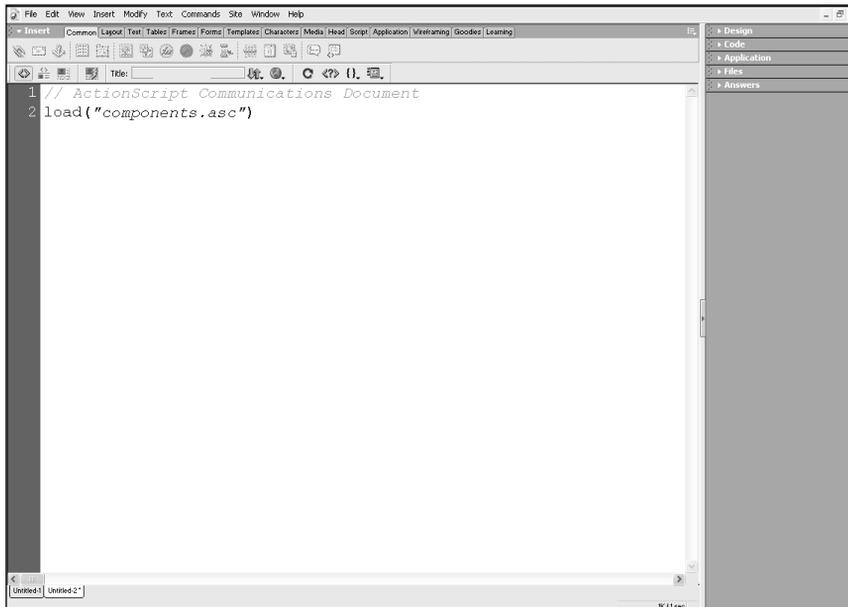


Figure 3.7 The chat room connection script can be created in Dreamweaver.

**Note**

The previous technique was met with the usual “That ain’t the way I would do it” from one of the coauthors. Here’s how he would do it:

1. Open the OccChat folder and right-click inside the folder.
2. Select New, Text Document from the pop-down list and name it **main.asc**.
3. Open a new Notepad document and enter the code.
4. Save the file and quit.

Building the Chat Room in Flash MX 2004

The chat room is to be constructed using the Flash MX 2004 Communication UI Components. The beauty of these items is the fact that they are prebuilt Flash movies requiring, at least initially, no deep understanding of ActionScript. Components are quick and simple to use. Before you start construction, be sure that you have installed the Flash Communication Server MX Components into Flash. Here’s how to construct a simple chat room in Flash MX 2004.

1. Open Flash MX 2004 and create new document. When the document opens, open your Components panel. If it isn’t visible in the panel grouping, select Window, Development Panels, Components, or press Control-F7 (PC) or Command-F7 (Mac).

**Note**

Though this may seem relatively basic, Flash MX 2004 has moved things around from Flash MX. One change is the move of the Components Menu Item to the Window menu.

2. Drag the Chat component to the stage. In the Property Inspector, provide an instance name, such as **ch3_chat**, as shown in Figure 3.8.
3. Drag the UserColor component to the stage. In the Property Inspector, provide an instance name, such as **ch3_color**. This component is really interesting. It is a combo box that color-codes all the users.
4. For fun, we added a ConnectionLight component that will tell us if we have connected to the Flash Comm server. Drag this component to the stage, and, in the property Inspector, give it a name, such as **ch3_led**.
5. Drag a SimpleConnect component to stage. This is an important component with a rather deceiving name. This is the component that will manage all the communications between the Flash Communication Server and the Flash Player.

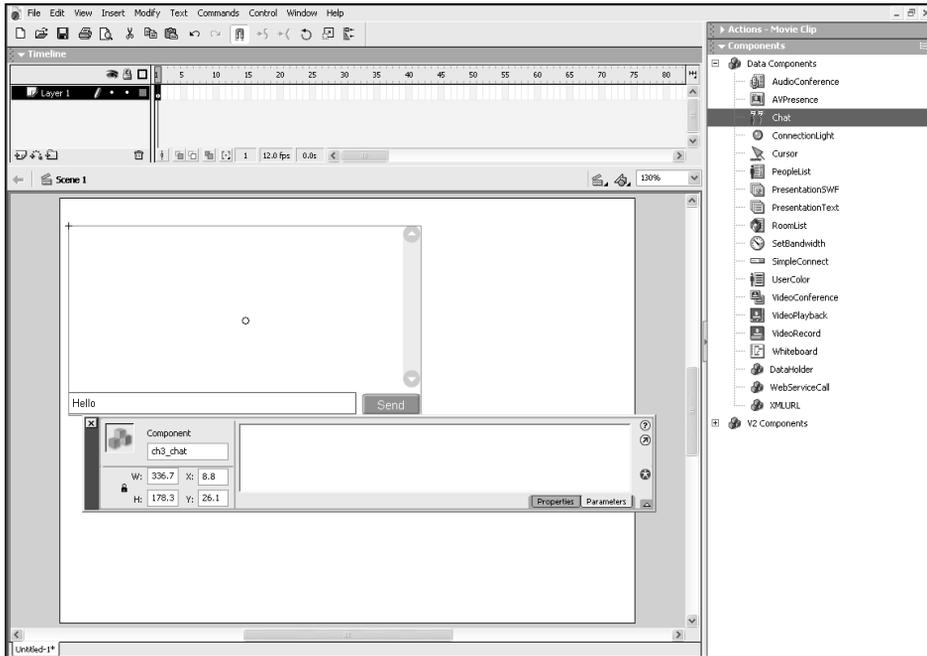


Figure 3.8 The Chat component is dragged onto the stage from the Components panel and given an instance name in the Property Inspector.

- This component won't need an instance name, but you will need to point it to your OccChat directory. In the Property Inspector, click once on the Application Directory and enter the URL for your chat on your Flash Comm Server. For our local server, we entered `rtmp://10.0.1.21/OccChat`, as shown in Figure 3.9. Your values may be quite different from ours. For example, one of our tech editors tested the exercise on his computer using `rtmp://localhost/flascom/samples/OccHat`. If you don't want to go through all the hunting and pecking for rtmp addresses, do what one of our tech editors suggests: use `rtmp://OccChat`.



Note

RTMP stands for Real Time Messaging Protocol. It is a proprietary standard developed by Macromedia to transfer data and messages between the Flash Player and the Communications server. It has no effect on the way the HTTP protocol is handled by the server.

7. Having told the SimpleConnect component where the directory is located, we also have to tell it what other components are being used. In the Property Inspector, click the [] beside the Communications Component to open the Values dialog box. This is where you will connect your SimpleConnect component to our other components that are on the stage.
8. Click the (+) sign in the Values dialog box to add a value. Enter **ch3_chat**.
9. Click the (+) sign. Enter **ch3_color**.
10. Click the (+) sign to add a value. Enter **ch3_led**. Click OK to close the dialog box. Your chat room should resemble Figure 3.9.
11. Save movie and test!

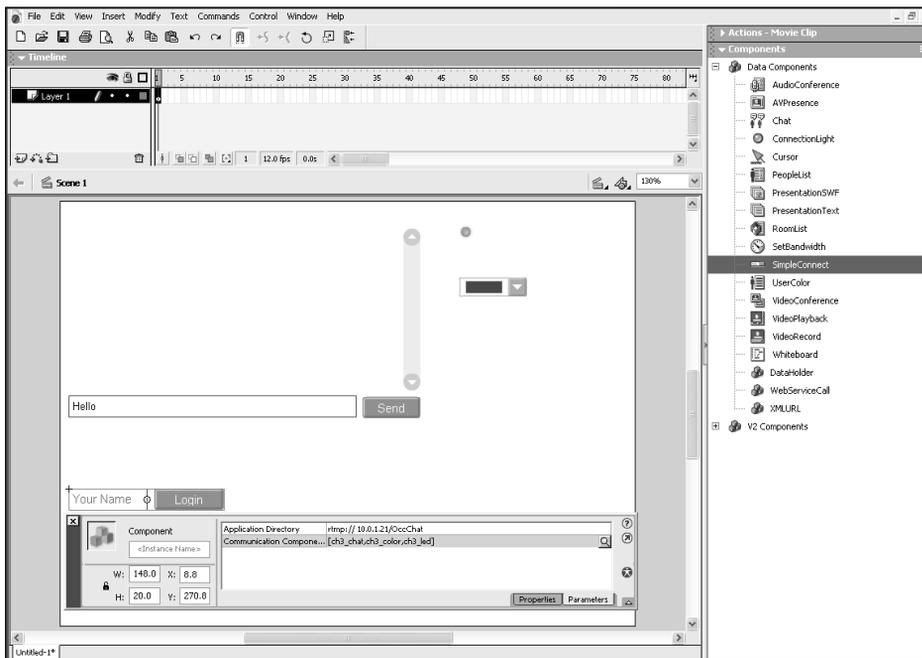


Figure 3.9 The SimpleConnect component is configured and the chat room is “good to go.”

Summary

Planning a dynamic site is hard work.

The first thing you need is an ability to help the client distinguish between a “want” and a “need.” This will require some rather in-depth discussions between you and the client, but the time spent here will avoid serious problems at a later date when the client discovers what he wants isn’t what you are creating.

A key document to prepare is an outline of the project scope. This document lays out the parameters of the project, including such items as budget, deadlines, a creative brief, and a technical specification. This document can be regarded as a “deliverable,” and the client should sign off on it.

Using a variety of technologies can make for some interesting decisions and discussions that will have a profound impact upon the finished product. This is why we reviewed how we made the decision to use MySQL as the database for this book. We also explained the purpose of a creative brief, how it is developed, and supplied a sample brief for the Oakbridge Community Center.

The technical brief is another document that should be prepared. It is the document that gives the technical team its marching orders and the client a succinct overview of the technologies being used in the project.

Client/team communication is important, and we also reviewed the reasons behind creating a client site, how to work with timelines and deadlines, and the importance of communication between the members of the team.

This last point is facilitated through the use of a chat room that is constructed using a number of Flash MX 2004 Components.

With the broad issues of the project under control, the next chapter, “Planning the Data for a Dynamic Site,” shows you how to plan and organize the raw material for the Oakbridge Community Center site.

