Chapter 9: Web Site and Server Maintenance

As this book goes to press, the Web has just passed a milestone, with over five million Web servers online today. This chapter helps you make choices as you launch another Web site into the mix. When you embark upon a Web-based community information (CI) project, the most basic question you face is where you will host your content. Before assuming that you will run your own general-purpose Web server on your own premises, you will want to consider alternatives:

- Running on someone else s server, either by payment for services or as part of an existing relationship (e.g. your library cooperative).
- Hiring someone else to house and administer your own server on their premises.

One survey finds that *nine out of ten* Web sites are hosted externally from the site s premises. So let s first consider reasons why you might want to find space on an external server instead of running your own:

- You and your staff are freed from having to develop expertise in server administration; you can concentrate on content and on authoring technology, and leave the server technology to others.
- You need not worry about backing up your server data.
- You don t have to worry about when to install the latest version of Web server software or the newest server operating system.
- You don t have to carry a beeper that goes off Sunday at midnight.
- You do not have to have a direct connection to the Internet.

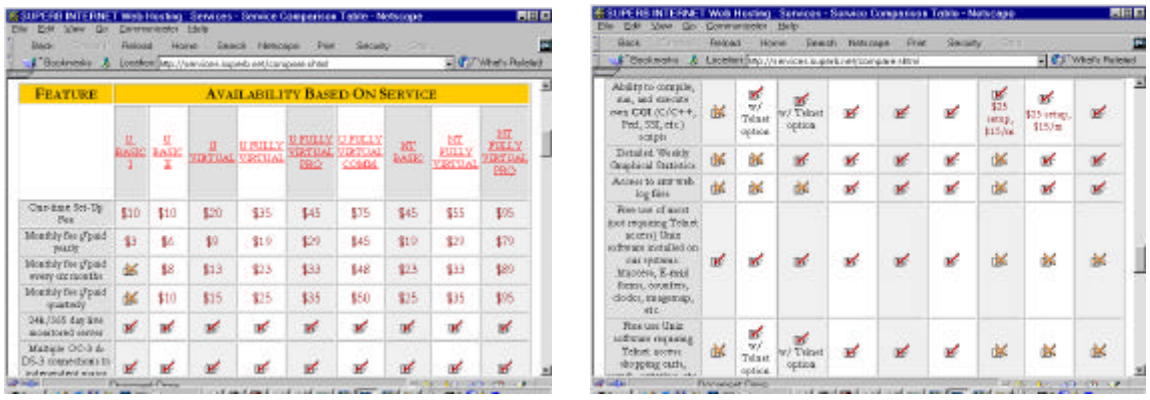
The option of hiring someone else to run your server on their premises is a good middle ground choice in many cases. Here, you own the server and all software and content on the server, but you simply choose to house it at another location.

The virtual host concept is another option that offers some of the benefits of owning your own server but typically at a much lower cost. Under a virtual host scenario, you share a single physical server with one or more other content publishers. However, you are assigned your own domain name, and your Web content appears at the root of that domain name, so that your users cannot tell whether you have your own server or not.

Because a virtual host divides a multi-thousand-dollar server box up among a number of users from a handful of virtual sites to a dozen or more the service provider can afford to charge each virtual site much less than you d have to pay if you ran your own server. You also avoid the capital expenditure of buying the server up-front, exchanging that cost for rental of the shared server, typically on a monthly or annual basis.

With virtual hosting, you typically have some access to the server to run your own CGI scripts, but the service provider may demand the right to inspect all scripts before they are installed. You may or may not be able to choose the server extensions (such as FrontPage

extensions) or middleware tools you prefer. Many virtual host services provide an upgrade path from basic HTML to FrontPage to CGI scripts to full database support. Fees also vary based on the amount of traffic your site generates. You will want to evaluate the upgrade path and potential future cost tiers before you make a deal with your vendor. Here is an example of the services comparison chart offered by one host content service, Superb.net:



Host content services that offer full detailed charts such as this one make it easy to compare various service levels and to evaluate upgrade options. Many smaller ISPs may offer similar services, but may not have sufficient virtual host business to cause them to post such a detailed service comparison chart. If a local ISP or other organization offers an ad-hoc quote for virtual host service, be sure you get the quote in writing.

The global nature of the Web makes it possible for you to choose a content hosting vendor potentially anywhere on the global Internet. In practice, you will want to consider geographic proximity of the service provider, as well as the speed and level of congestion of Internet links between your main body of users and the remote provider. There is no single rule of thumb here: it is possible for a host content service 2000 miles away to provide better service than a local ISP; it is also possible for the remote service to offer dreadfully slow page download times due to distance and congestion. Performance can vary dramatically with time of day. You may want to test performance on more than one service before you go into production.

On the other hand, if you choose to run your own server on your own premises, or if you choose a local ISP to host your content, and, if the server has good connectivity to the fabric of the Internet in your area, you can expect to deliver good service locally. Since most of your users will for the most part by definition be members of your local community, you can be confident you are serving them well.

Whether you want to have an external service provider run your own server, or whether you want to pursue the virtual host option, you have a number of choices as to who might host your content:

- Your library cooperative
- A library service organization such as The Library Network (TLN)
- A local Internet service provider

- A local unit of government or school system with its own Web presence
- A statewide service provider (for instance, in Michigan, Merit is such a provider)
- A telephone or cable company that has entered the Internet and Web content hosting businesses (e.g. Ameritech, AT&T, TCI, Media One, Time-Warner).
- A national host content service (e.g. Mindspring, Intermedia, Superb.net, IMC Online).

If your server is housed off-premises, you will need to ensure that your own staff has adequate access to the server for posting new and updated content. In many cases, dial-up access may be adequate for this purpose. However, if you re hosting graphical or multimedia content, or databases, you probably want a faster connection than dial-up perhaps ISDN or a cable modem is for you.

Server "Co-Location"

Many ISPs offer server co-location which simply means that your server is housed on their premises, located alongside servers belonging to other clients.

Under this scenario, either you can hire the ISP to administer the server, or you can take on some or all of the server administration responsibilities yourself.

Here are reasons why you might choose server co-location over running a server on your premises:

- The ISP has faster or more reliable connectivity to the greater Internet.
- The ISP has a better physical environment for the server better environmental conditions, better security, etc.
- The ISP can perform regular tasks, such as backup, restarts after failure, or security monitoring, on a 24 hour basis.

Co-location differs from the virtual host option. Under co-location, typically you buy and own the server hardware. Under the virtual host option, typically you rent a share of the server resources under your own domain name.

External Service Provider Caveats

You need to do some up-front research and negotiation before you decide to put your content on an external server. Here are some caveats:

- Have a clear agreement that states that you own your own intellectual property: the content that you put on the Web site, the domain name you register, etc.
- Make sure you have access to server log information, including all raw data, and that you can access this data as frequently as you want. Many commercial host content services charge extra for access to the logs. You may be willing to pay the fees, but be sure they are within your budget.
- Be sure you understand all fees in general. For instance, many service providers charge more if you exceed certain monthly bandwidth limitations. These costs can surprise you,

especially if you begin by serving small HTML documents, then at a later date you decide to offer streaming multimedia. Also you can expect to pay relatively high rates for rented server disk space; in many case one year s rental would more than buy an equivalent amount of disk. (You are paying not only for the capital cost of disk, but for connectivity, backup, server hardware maintenance, the ISP s profit, etc.)

- Ensure that you will be allowed to run your own CGI scripts or middleware products you need.
- Be sure there is a clear understanding of the exit strategy for the day when you decide to move your content to another provider, or in-house. For instance, the service provider will give you free access to all content, scripts, configuration information, logs, etc. within one week of your request.

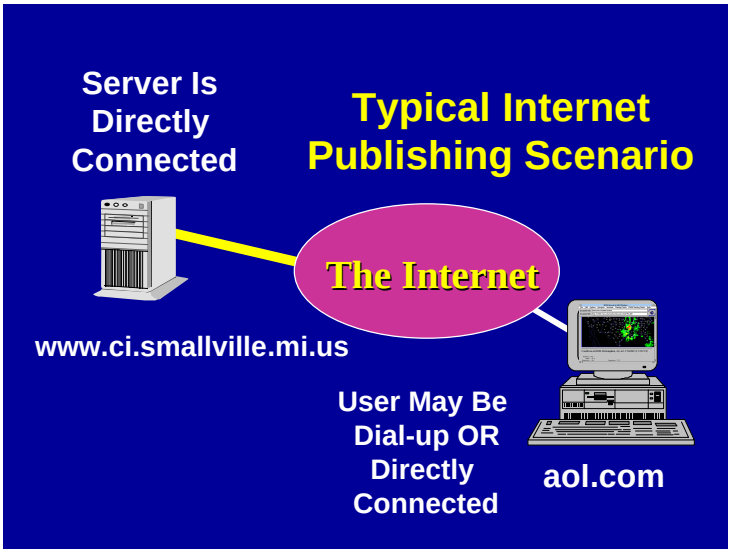
Choosing to Run Your Own Server on Your Premises

Here are reasons why you might want to run your own server:

- You want to do sophisticated animations or implement front end Web connections with existing databases, and your service provider can t offer services as efficiently or effectively as you could on-premises.
- You have a very large amount of data so much that it is impractical to move the data to the off-premises server.
- You already have an Internet infrastructure in place for instance, a high-speed link to the greater Internet and you want to take advantage of that existing infrastructure.
- You feel it is inevitable that you will learn server technology, and now is as good a time as any.

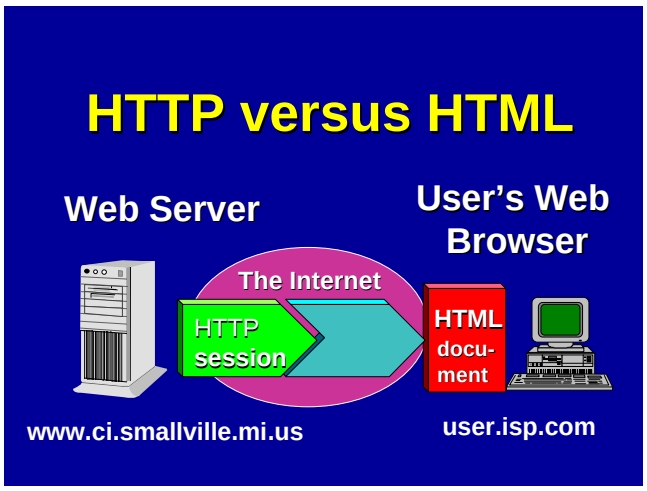
Basic Web Publishing Infrastructure

No matter where your server resides, you will need some basic components in your Web publishing infrastructure. This diagram depicts some of the core elements:



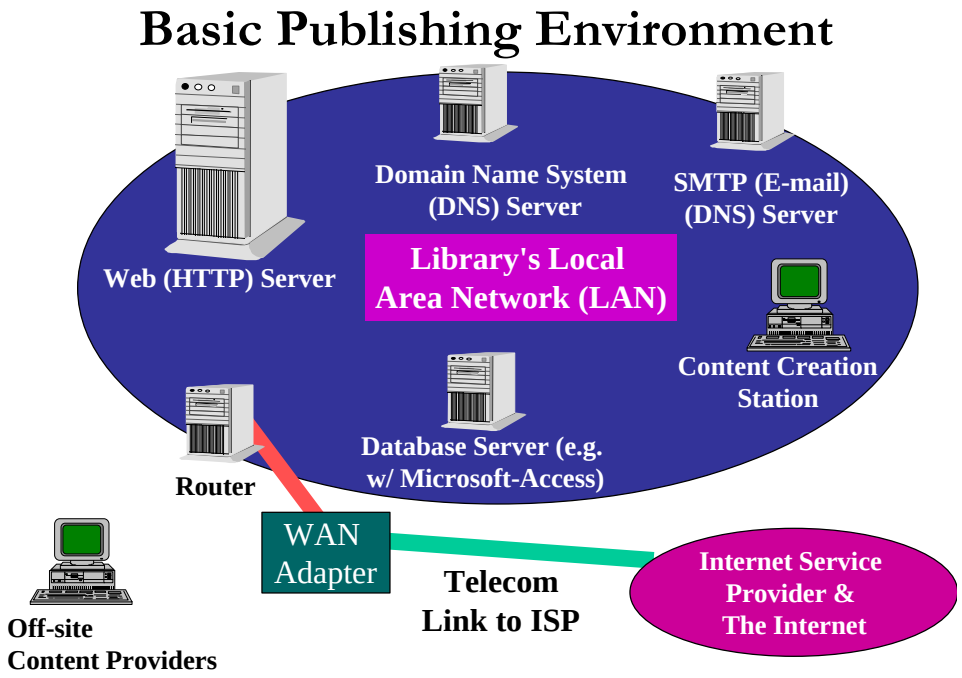
Your server is named `www.ci.smallville.mi.us`. (We ll discuss server names and Internet domain names in more detail later in this chapter.) Your server has a direct, permanent connection to an Internet Service Provider or ISP. Your users connect to any service provider of their own choice (such as America Online in this example) and across the global Internet to your server.

In essence, your server waits until it hears a request for a Web page, then delivers it on demand. That request comes across a TCP session established by your user s Web browser to your Web server:



Thus, your server needs to know how to speak the Hypertext Transfer Protocol (HTTP) in order to understand and process requests to transfer Web documents to the user. Generally speaking, your server does not need to understand anything about HTML in order to deliver content to your users. Your content providers and your authoring tools take care of producing good HTML documents; the server just sees them as files to be sent on demand. (In the case of dynamic documents and live database content, the scenario is somewhat more complicated; the server and associated software will produce HTML content on the fly. The HTTP server itself still doesn t concern itself with that content, but associated tools need to generate correct HTML.)

In order to achieve the publishing mission, you will typically need more components than a simple HTTP server. This diagram depicts some of the pieces you may need:



Here we assume that the Web server is a standalone box dedicated to the function of serving Web pages. It will thus run a process or service that is capable of listening for HTTP requests and replying to them.

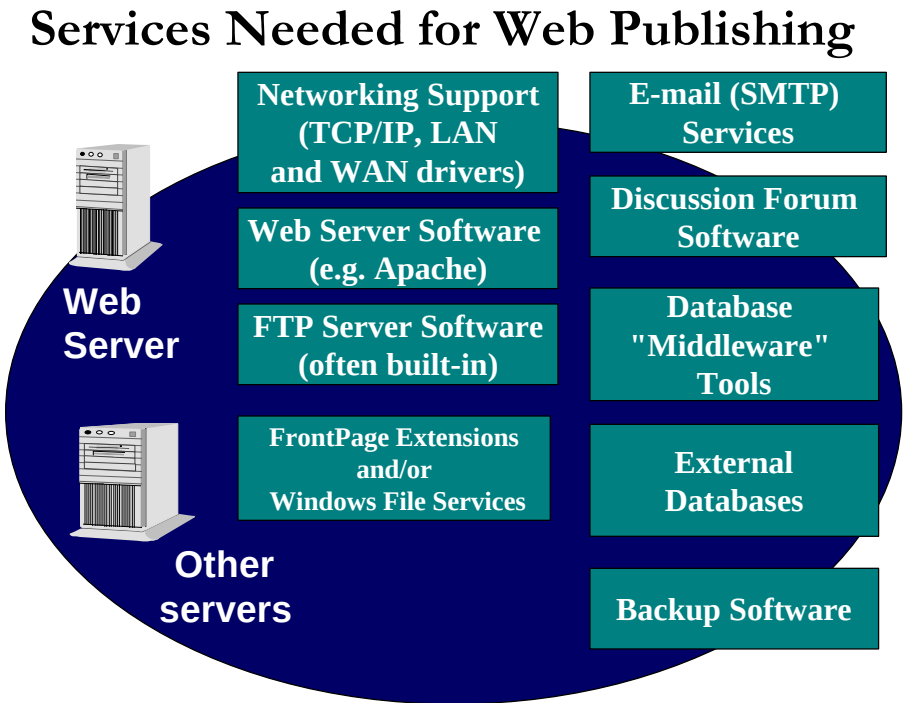
Note that this diagram depicts a box called a router. A router is a device that s capable of deciding which traffic needs to leave the local network, and be routed to other networks on the Internet. The router is connected to some sort of Wide Area Network (WAN) adapter, which could vary from a simple ISDN modem to a digital interface for a modern high-speed link.

If you have an existing Internet infrastructure, you already have a router on your premises. Some server appliances (discussed later in this chapter) come with their own router functionality built-in; the more typical case is to have a router running in its own box. The leading vendors of routers are Cisco Systems, 3Com, and Bay Networks.

One optional device is not shown in this diagram, and that is the firewall. A firewall is a device that filters network activity and rejects traffic that isn t appropriate for your local network. Firewalls also help monitor traffic and detect attacks on your various servers. Firewall functionality can be built into a router, or you might choose a separate router. While many commercial enterprises run firewalls as a matter of course, many institutions such as libraries and universities historically have chosen not to. Over time as security needs increase, firewalls may become popular in virtually all sites. Today, whether your site incorporates a firewall depends on how you balance concerns about safeguarding information, maintaining open access, and budgetary constraints.

You have a choice of whether to dedicate your Web server to the sole task of serving Web content, as opposed to running multiple ancillary services on the same box. These ancillary

services may reside on other servers on your premises, or even on servers hosted off-premise by someone else. If your site is small you may opt to run a variety of ancillary services on a single physical server box. For instance:



Many people find it helpful to spread services out among multiple boxes in order to isolate functions; they argue this improves performance of each function, and minimizes disruptions when servers fail. For instance, in this example, if your external database server goes down, your static HTML content (such as the main home page for your site) could still be available to users.

The choice of how many server computers to install depends on many factors: performance, budget, and the philosophy of the server administrator. In one extreme case, you would run a single Web service and a mechanism for transferring content to the server and nothing else on your Web server. In the other extreme case, you might run *all* of the services shown here on a single piece of server hardware. Most sites will choose between these extremes.

Whether you use a single server or many, note that some of these related services are essential to every Web publishing project. Others may not be necessary for your publishing project. Let s consider each of these services:

- **Networking support:** Your server must be able to communicate with users on your local network as well as the greater Internet. Modern operating systems now come with built-in support for the Internet standard communications protocol known as TCP/IP, whether a box is a desktop, a laptop, or a server system. Your server will have thus include built-in support for TCP/IP, which in our examples is used both by your

content providers on the local network as well as for your Internet users. At initial setup of the server, you ll need to configure it to talk TCP/IP in your environment by entering certain configuration options and perhaps by installing drivers for your local network and for your connection to the greater Internet. (Think of your local network as your LAN and your Internet connection as your WAN or Wide Area Network..)

- **Web server software:** Your Web server software is a separate tool that runs as its own service or process on the server. A variety of packages are available; we discuss options later in this chapter.
- **FTP server software:** Most Web sites support FTP (for File Transfer Protocol) as a least-common-denominator mechanism for content providers to place new content (HTML files, images in GIF or JPEG format, etc.) on the server. Many sites use FTP as the only mechanism for posting content. FTP can be cumbersome when done manually via a line-mode interface; with graphical interfaces, it can be reasonably intuitive.
- **FrontPage extensions:** Microsoft s popular FrontPage authoring tool requires specialized support in the server so that authorized content providers can publish their content. Microsoft s Internet Information Server (IIS) supports FrontPage natively. Other server products may do so as well; others may require installation of extensions to support FrontPage.
- **Windows file services:** Many shops run a local area network with NT file sharing enabled, so that authorized users can mount disks on other computers, including the Web server. This allows content providers to use drag and drop operations in Windows to move files or folders to the content station for editing, or to the server for publishing. (Note that Windows-style file sharing can be enabled even for non-Windows Web servers thanks to tools such as Samba. Note also that in a pure Macintosh shop, with Mac servers and desktop computers, you would use Mac-style drag-and-drop file importing and exporting.)
- **E-mail services:** It s almost impossible to conceive of running an active Web publishing project without a mechanism for team members and users of the site to communicate. Your e-mail service need not reside on the same server as the one that hosts your Web content; indeed, many system administrators prefer not to run e-mail on Web servers for security reasons.
- ♦ **Database middleware tools:** if you want to offer live content by connecting to a database, you ll need a database middleware product such as Cold Fusion. (For the Toolkit, we use Microsoft Active Server Pages, or ASP, as our middleware.) In order to connect to a live database, you typically would install middleware tools on the Web server itself. If you are not connecting to a live database, you do not need middleware software.
- ♦ **Database server:** If you offer a live database, you need database software such as MS-Access, SQL Server, Oracle, Informix, or FileMaker Pro to host the actual database content. It is quite common to run a database on an external box for performance and/or security reasons. In order to communicate with your middleware and in turn your Web server, you will want to run a database that supports the ODBC (Open Database Connectivity) standard. Note that many database products now are shipped with built-in Web middleware.

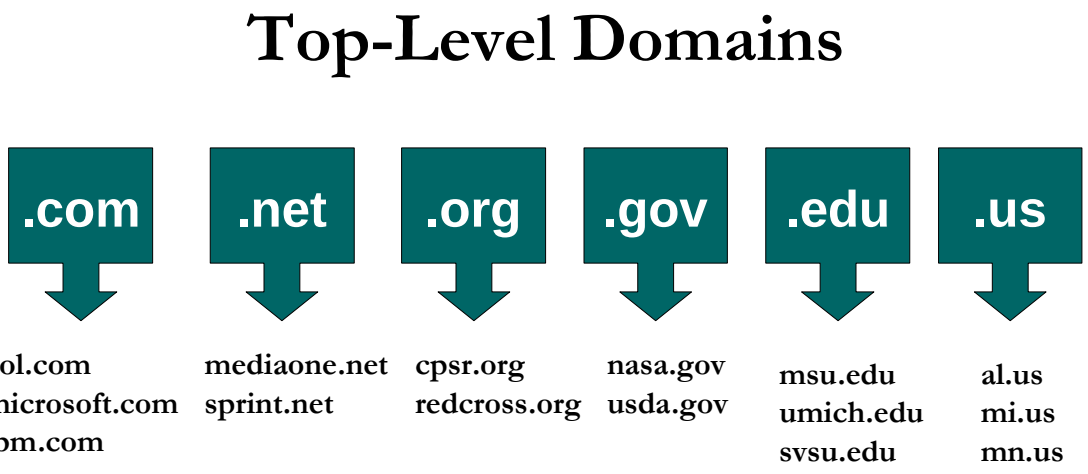
- **Backup software:** You will want to periodically back up your entire server software environment, and you will want to back up your Web content frequently as well. Backup software allows you to automate this task. You may want to run backup software on the Web server and on each of the servers with production services, or you may want to install a single, centralized network backup service that is capable of accessing all production disk volumes and backing them up as a batch.

Keep in mind that your choices as to what services to run, as well as your how many servers to use, are not carved in stone. You can always add new server hardware, and re-allocate production services among server boxes, as time goes on.

The Domain Name System (DNS) and Your Domain

The Domain Name System, or DNS, is a remarkable distributed database that provides a mechanism for naming hosts on the Internet. It s called the *Domain* Name System because it divides the global Internet into a number of administrative domains, each of which can be sub-divided according to the desires of its administrators.

At the top of the hierarchy are, appropriately enough, a series of top-level domains many of which will be familiar to you:



Each of these top-level domains has its own meaning which dates back to the beginning of the Internet:

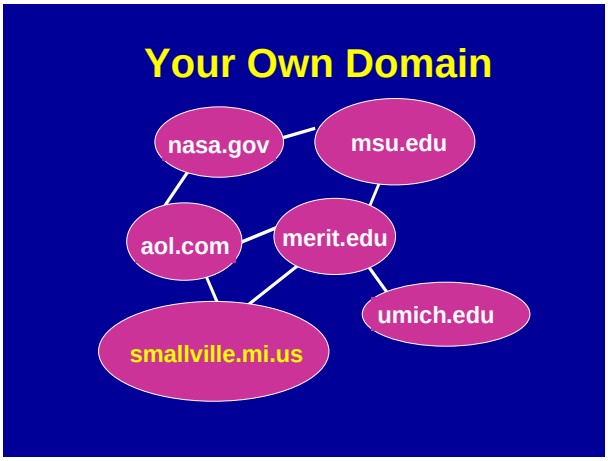
.com Commercial entity, typically a corporation or small business.
.net Traditionally, an Internet service provider. Today, anyone who wishes to register a .net domain is able to do so, regardless of whether they represent a computer network of any sort.
.org Non-profit organization.
.gov Governmental entity.
.edu Institution of higher education.

.us Any entity associated with a geographic or political subdivision of the United States. In particular, a `.us` subdomain exists for every state in the nation; thus `mi.us` is the subdomain for Michigan.

The state hierarchy within the `.us` domain is well-suited to domains assigned to municipalities, counties, and school districts.

This naming system is oriented towards the United States due to the historical fact that the Internet was born in the United States. Two-character country codes analogous to .us exist for every country on Earth. For example, `.uk` is the United Kingdom; `.c`a is Canada; `.ch` is Switzerland. (Note, however, that many `.org`, `.net`, and `.com` registrations apply to non-U.S. organizations.)

You may already have a domain assigned to your organization, which you may decide to use to house your new community information site. In this case you do not need to register a new domain. Instead, you need merely ask your domain administrator to assign you a new host name if necessary. You may want to read further about domains and host names to understand this process better.



In many cases, a new CI project *will* want to establish its own new domain. How you proceed depends on the domain you wish to use. For instance, you may want to register your CI site in the `.org`, `.com`, or `.net` domains. In this case, you will need to contact a registrar for these top-level domains, either directly or through the services of your ISP.

Note that domain names are not case-sensitive. For marketing purposes, you may publish your domain with capital letters; smallville.org and Smallville.org refer to the same site.

As this book goes to press, the handling of these top-level domains is undergoing a major change, as the process moves from a monopoly to a new shared scheme. From 1993 until 1999, all registration in these domains was handled by the registration InterNIC, a service of Network Solutions, Inc. As of this writing, Network Solutions remains a service provider in this arena, but a number of new companies will be added to the mix, each offering registration on a competitive basis.

If you wish to register a new domain in `.org`, `.net`, or `.com`, your options are:

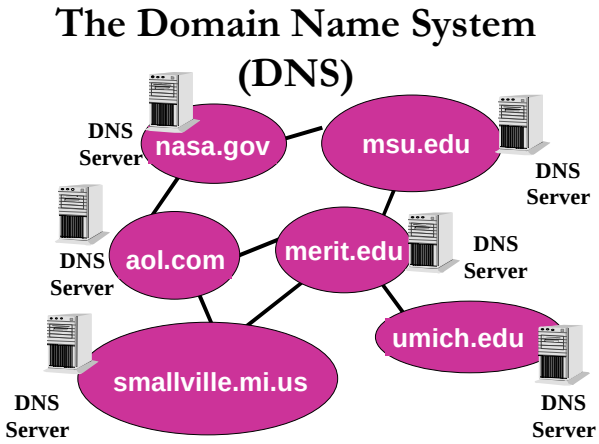
- Register using the services of Network Solutions. Visit `www.networksolutions.com`.
- Use one of the new competitive registration services. For a list, visit `www.icann.org.`
- Work with your ISP to register the domain. You can use the services of your existing ISP, or you can use the services of one of a large number of host content services to register a domain on your behalf. Many ISPs offer free domain parking which allows you to register a new domain without any special fee for the creation of the domain. (In all cases, you must pay a fee to the actual registrar, whether it is Network Solutions or one of the new registrars. The current rate as of this writing is $70 per two years per domain.)

If you want a domain outside of `.org`, .`net`, and .`com`, such as a domain in the `mi.us` hierarchy, you must pursue this with the administrator of the domain in question. The handling of the .us domain is documented here:

`http://www.isi.edu/in-notes/usdnr`

A convention that's commonly applied calls for `ci` to be in the domain name of a community information site. For instance, the domain `ci.east-lansing.mi.us` might be used for a CI project for East Lansing, Michigan.

The DNS works on a distributed basis. The existence of your domain (for instance, smallville.org) is recorded in a root server. Every domain must run its own DNS server — in fact, a primary server and a backup are required. In turn, each domain is able to add hosts within its domain. Thus, if you wish to have a server called `www.smallville.org`, it's up to your domain administrator to add an entry into your domain server. The DNS allows any user anywhere on the global Internet to discover `www.smallville.org` through a simple process: first the user's computer discovers where the DNS server for `smallville.org` resides; then, the user's computer interrogates that server to find the specific address of `www.smallville.org`.

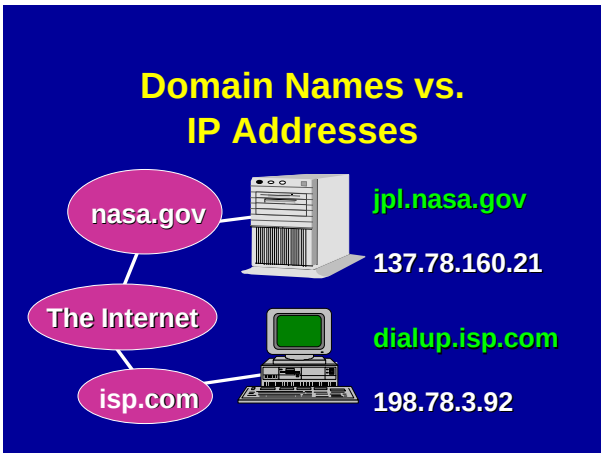


For obvious reasons, it is important that your DNS server remain operational as much as possible. If it s down, your users won t be able to locate your server. (In fact, you re expected to have a primary and a backup DNS server in order to assure reliable participation in the Domain Name System.) Often your ISP will be willing to provide DNS services on your behalf as part of the cost of providing you with a dedicated Internet connection.

Domain Names and IP Addresses

Every host computer server or otherwise on the Internet is assigned an IP address. IP addresses are four-part, all-numeric handles that are easily used by networks. For instance, the Community Information site for East Lansing, Michigan has a host name of `www.ci.east-lansing.mi.us`; it has a corresponding IP address of `205.153.190.140`.

In general, each computer on the Internet has its own host name and its own IP address. However, it s possible for a single computer to have more than one host name. For instance, you may want to register both `ci.smallville.mi.us` and `www.ci.smallville.mi.us` as host names for your Web server. This allows users who guess your host name to find the computer either way. In this case, your server administrator simply creates an alias in your DNS tables so that both host names correspond to the same IP address.



In this example, the Web server for the Jet Propulsion Laboratory, `jpl.nasa.gov`, has been assigned the IP address 137.78.160.21. Normally the only people who have to worry about IP addresses are the administrators of servers or networks. If an organization handles its own IP address administration, it will be assigned a pool of IP addresses to use when new computers are installed. If IP administration is handled by an organization s ISP, new addresses will be assigned by the ISP s domain administrator, and given to the administrator of each new server as it is installed.

It s also possible for a single physical computer to have more than one IP address associated with it. This may be done to support virtual hosts, or it may be done if the server has more than one LAN (Ethernet) adapter installed for performance reasons. In the most common case, you will only have one IP address per server (or other computer) you install.

Generally speaking, you *never* want to publish the IP address of your Web server or any other server. Let the DNS map the friendly host names to IP addresses for you and your users. Users should see friendly, domain-style host names, not IP addresses, in the URLs your publish and in the Location box on their Web browsers.

Obtaining a New Domain

In many cases, you will want to use an existing domain, such as the one already assigned to your public library, governmental unit, or civic organization. In other cases, you may want to obtain a new domain just for your new community networking project. The choice is entirely up to you.

You may need to work with more than one domain administrator if you wish to assign multiple names to a server. For instance, suppose the Ann Arbor Public Library houses a server for information about that community. Within the library s existing domain of `ann-arbor.lib.mi.us`, the server might be assigned an address of `server3.ann-arbor.lib.mi.us`.

For a name appropriate for a community information site, the library would want a name more meaningful to the public. The library might choose `AnnArborInfo.org` (which would mean contacting Network Solutions or one of the new registrars of the .org domain).

Or, the library might choose `www.ci.ann-arbor.mi.us` which would mean contacting the administrator of the `ann-arbor.mi.us` domain. We find out who serves as the administrator of the `ann-arbor.mi.us` domain by consulting the list at `www.isi.edu/in-notes/us-domain-delegated.txt` which tells us the right place to send our e-mail request is: `us-domain@i-theta.com`.

What if you wish to register a new domain, and you find that it is already in use? For instance, suppose you want to register `smallville.org`, and you learn that someone in another town named Smallville in another state has already registered that domain? Once a domain name has been registered by another party, it can be difficult to wrest it away, unless someone has an existing trademark that has been infringed by the registrant. Therefore, it s a good idea to register a domain name as soon as you concoct it, even if you don t plan to deploy a Web site within that domain immediately. You can always deploy a site under the choice domain at a later date.

Within your domain, how you name your servers is pretty much a matter of your own taste and desires. However, functional names are a popular and reasonable approach. For instance, if you run e-mail services on a separate server, a server name of `mail.smallville.org` would make sense. Your database server might be `database.smallville.org`. Different conventions may be used for servers that are used only by your staff, such as internal servers or desktop computers.

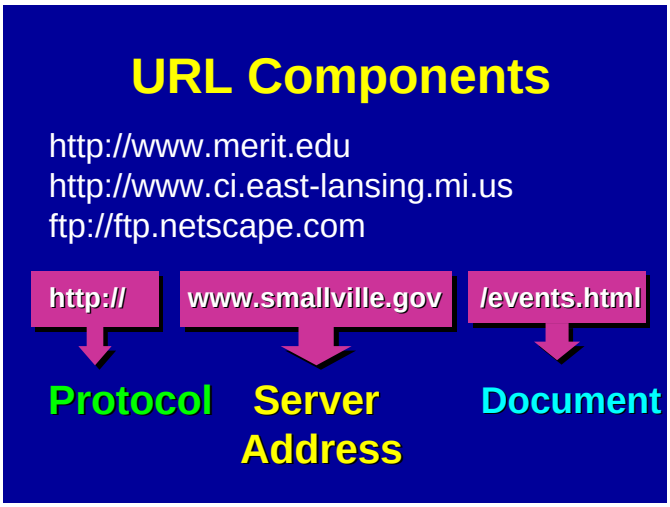
The Concept of TCP Ports

Now that we ve seen how host names are assigned, let s consider another important concept that applies to the Uniform Resource Locator your users will see.

The TCP/IP protocol suite carries with it the concept of a TCP port. When your user s browser connects to your Web server to fetch your CI home page, it connects to a particular host name that of your server and as part of the connection process it requests a particular TCP port number. By default, for the Web the relevant port number is 80. When you install Web server software, you can configure it to operate on any port number you choose. The vast majority of Web servers operate on the default port of 80. Other Internet services have their own assigned default port numbers; for instance, the remote login protocol Telnet is assigned port 23. By assigning different port numbers to different kinds of services, a single physical server box can handle multiple kinds of Internet services without conflict.

For your main Web presence, it s a good idea to stick with the convention of using port 80. It is possible to run more than one Web server process or service on a single server, in which case you may wish to choose other port numbers, such as 8000, for secondary servers.

With this discussion, we can see how your server name choice, TCP port choice, and file name choices combine to determine the URL of a document on your server:



- Every URL for a Web page begins with `http://` though many browsers allow users to omit that prefix.
- Next comes the address of the server. This is a host name within a domain. In the diagram s example, the server s address is `www.smallville.org`.
- Optionally, the host name may be followed by a colon and a TCP port number, for example `www.smallville.org:80`. You will only specify a TCP port other than 80 if you have installed a Web server that listens on the port in question. (If you publish a URL with a port number for which there is no corresponding Web server, your users browsers will display errors.)
- Next comes the path to the specific file on the server. As you ll recall from Chapter 4, this path name begins at the start of the server s hierarchy, as specified in a server configuration file, not the absolute root of the server hard disk.

Thus, we might offer all of these URLs:

```
http://www.smallville.org/
http://www.smallville.org/events.html
http://www.smallville.org:8000/
http://www.smallville.org:8000/events.html
```

The first two examples assume a server on TCP port 80. The second two explicitly refer to a server on port 8000.

If you never deviate from the default of port 80, you will never need to publish URLs with a port number in them, and your users will never encounter such URLs when traversing your site.

Running Your Own Server – "Ready-to-Run" Options

At the beginning of this chapter, we considered the basic question of whether to run your own server or to hire someone else to run a server for you. Assuming you do wish to run your own server, you still have choices to make: Your choices include:

- Server appliances or thin servers
- Turnkey or ready-to-run servers

So-called servers appliances or thin servers are an alternative to installing a general-purpose server. With a server appliance, you purchase a server that s ready to run with a minimal amount of setup. Typically, you need only tell the server its new IP address and host name, along with a minimal amount of additional information. Then you plug the server box into your local network, hook up a link to your ISP, and you are ready to serve pages. All that s left is preparing HTML documents and placing them on the server.

Configuration of the server appliance is usually performed via inputs on a keypad on the server box itself, and/or use of a Web browser. There is no editing of configuration files nor is there any downloading of software or compiling of utility programs. Setting up a server appliance, in theory is no more difficult than installing a Web browser or other simple end-user application program.

Server appliances are a relatively new concept. The goal of server appliances is to create a server box that has had all of the administration and installation work done for you, with setup work kept to the absolute minimum. Server appliances tend to be based on processors that are somewhat less powerful than the most current general-purpose servers, but vendors tune the operating system and Web server software for the best performance possible. In any event, serving static HTML pages does not require a huge amount of processing horsepower.

Server appliances usually are designed to have a level of robustness not always found in general purpose servers. In the event of a power outage, for instance, some server appliances are able to quiesce all activity, shut down the server, and restart automatically when power is restored.

Some ISPs are working with vendors of server appliances to offer simple package purchase arrangements. You order the server with the assistance of the ISP, which has prepared special documentation and procedures to allow the simplest installation possible. In some cases, such servers can be installed in as little as 15 minutes.

If you are new to network and server administration, but you do wish to run your own server, think carefully about a server appliance as your first solution. Even if you expect to graduate to a general-purpose server at a later date, your server appliance can free you to concentrate on content as you begin your project. If you eventually acquire a general-purpose server, you can always reuse the server appliance for other projects after your new, fancy general-purpose box is in production.

Server appliances may be based on proprietary hardware and software, or they may use commodity hardware and freely-available software such as the Linux operating system and the Apache Web server. In theory, you will not even be aware of what is running under the covers; those details are hidden from you.

Prices for server appliances range from under $1000 to $3000 or so.

Vendors of server appliances include:

- Cobalt Networks, `www.cobaltnet.com`
- Encanto Networks, `www.encanto.com`
- Technauts, `www.technauts.com`
- Whistle, `www.whistle.com`

General-purpose servers are increasingly being offered with ready-to-run configurations. Vendors will pre-install your operating system of choice. Such turnkey servers may offer many of the advantages of server appliances. For instance, Compaq markets the Prosignia Neoserver, an entry-level server for sites that have a network but not a network department. Most vendors of proprietary Unix systems also offer ready-to-run Web server packages. Over time, vendors will increasingly deliver systems ready to run and to serve Web pages out of the box.

Paradoxically, some people argue that the first thing you should do with a brand new server is to format the hard disk and reinstall the operating system from starter CD-ROMs. The argument here is that if you rely on the vendor s pre-installed software, you won t understand how to do an installation from scratch and when your disk fails, you ll be unable to recover. This could lead to an outage lasting days instead of hours. If you use a pre-installed server, at least be sure you do complete backups.

Server Platform Choices

Assuming you decide to purchase a general-purpose server, you have a number of options from which to choose. Most generally, these include: what hardware, what operating system, and what Web server software.

Among hardware choices, you can pick among:

- Wintel based PCs or PC servers. These machines run on Intel or compatible processors. Vendors include Dell, Gateway, Compaq, IBM, HP, and others.
- Proprietary server hardware. Typically offered by traditional vendors such as HP, Sun, IBM, Compaq/DEC, etc., these servers generally use proprietary processors.
- The Macintosh, running on the PowerPC processor.

Your server operating system choice may be implied by the hardware choice. If for instance you choose a Sun server, the system will come with Sun s version of Unix, Solaris. In other cases, your hardware choice may not dictate your operating system. For instance, you might

pick a Compaq Prosignia server, which would be able to run either Windows NT or an Intel-or-compatible-processor-friendly Unix clone such as Linux.

The core server operating system choices, then, are:

- Windows NT. Windows NT is Microsoft s operating system for server applications. Windows NT comes in two flavors, Workstation and Server. Microsoft does not intend Windows NT Workstation for production services, but rather for low-volume testing, development, and desktop use.
- A proprietary version of Unix.
- A Unix-like system for Intel servers such as Linux or FreeBSD.
- The Macintosh operating system.

Some sites may be tempted to run Windows 98 as a server platform, as it comes pre-installed with Microsoft s Personal Web server. In general this is not advisable. Windows 98 is not intended as a multiuser, multitasking server environment, and the Personal Web Server is limited in its functionality. This tool could make a Windows 98 desktop PC a good testbed, on which content providers pre-publish their new documents for review.

In the Macintosh realm, you have a choice between Apple s traditional operating system, at level 8.5 as of this writing, and Apple s new OS-X operating system, which is intended as a server environment. OS-X blends some Unix-like server capabilities into the core Apple operating system; Apple claims this yields an easy-to-administer but robust server environment.

Finally, you have a choice of Web server software packages. Although there are dozens of Web server packages, only a handful are commonly used:

- Apache, a freely-available server software package, is used on about one-half of the production servers in use today. According to a survey by Netcraft.com, Apache represents about 54% of Web servers online.
- Microsoft s Internet Information Server, or IIS, is popular as a server for the NT platform. Netcraft estimates that the various flavors of IIS represent about 24% of the server market, with share increasing.
- Netscape s various servers (Fasttrack, Enterprise, Commerce) represent just under 7% of the server market..
- Web Site Pro, a tool from publisher O Reilly & Associates, has just under two percent of the market.
- The most popular Mac server software, WebStar, has about 1 ½ % of the overall Web server market.

If you choose a server environment that s popular, you are more likely to find training materials, magazine articles, tools, and online discussions to help support your developing site. Less-popular server environments may be appealing because of special features such as ease of administration.

Connecting to the Internet: Wide Area Network (WAN) Choices

If you host your own server, it will need to be connected to your ISP via a permanent, direct connection. There are a number of choices for such a WAN connection:

- **ISDN**, or Integrated Services Digital Network, is a mature technology available from the telephone company. There are several variations on ISDN but the most common variety moves data at 128 kilobits per second.
- **T1** lines are a form of leased telephone line moving data at 1.544 megabits per second. A T1 line is a basic, reasonably high-speed mode of connecting across town or across a long distance.
- **Cable modems** are offered by some cable television systems. They move data at relatively high speeds up to 10 megabits per second, which is much faster than ISDN. (Because you share the local loop between your premises and a neighborhood facility, you cannot assume all of the advertised raw bandwidth for yourself.) A relatively new technology, cable modems do not always offer the level of reliability of traditional telephone links, but service may become sufficiently solid as the technology matures.
- **DSL**, or Digital Subscriber Loop, is a relatively new technology for digital data transfer over traditional telephone lines. DSL can move data at about two megabits per second. DSL is only available in some localities and only subject to constraints having to do with the local telephone plant. DSL is expected to be a very popular connectivity option for small businesses and other small organizations. (Sometimes DSL is referred to as ADSL.)
- **ATM** and **Frame Relay** are higher-speed data transfer methodologies often associated with fiber optic rings found in larger metropolitan areas. Speeds in the multi-megabit range are typical. Only a few community information networks are likely to demand these speeds in the near term.

The cost of connecting from your premises to your ISP varies with speed and with distance. You can spend anywhere from under $50 per month to thousands of dollars per month. As a practical matter, most CI networking projects are unlikely to generate a large number of concurrent users. Also, most projects are not likely to deliver large quantities of data. An exception would a site that relies heavily on multimedia content (such as an audio voice archive) or on delivering large graphical images to many users (such as a geographical information system site). Therefore, many if not most CI sites can survive quite handily on a link such as a T1 line.

Many public libraries hosting a new CI site will have their own existing communications link. Such sites need merely evaluate whether their existing link has sufficient spare capacity to handle the new load induced by the CI site.

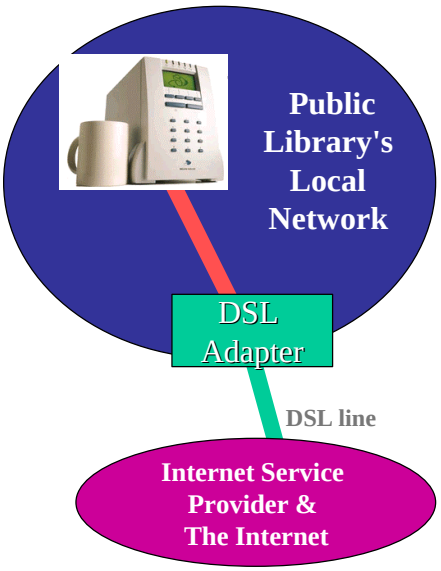
Server Scenarios

Let s consider some possible server scenarios. Note as you consider these examples that the choices of WAN links are shown as examples only; any one of these servers could handle relatively high-speed or low-speed links.

Here are some scenarios of some typical server environments.

A "Thin Server" Scenario

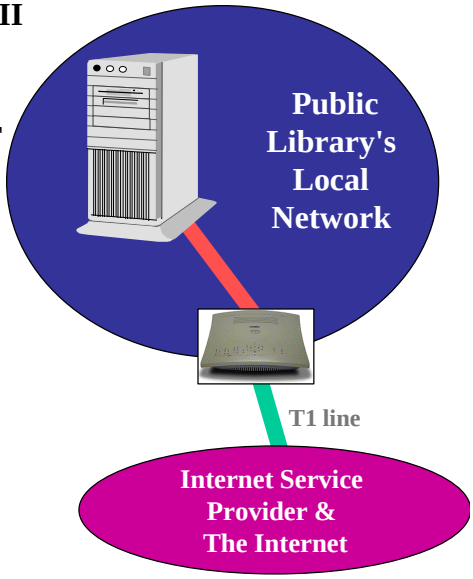
- **Hardware: Whistle InterJet**
- **Operating System: BSD Unix (built-in)**
- **Web Server: Apache (built-in)**
- **Static HTML documents – uploaded via FTP**
- **DNS (Domain Name System): built-in**
- **SMTP (e-mail) Server: built-in**
- **Connectivity: Built-in Router**
- **DSL ("Digital Subscriber Loop") link via external DSL adapter**



Here, the site has chosen the Interjet server appliance from a vendor called Whistle. The Interjet provides all of the server functions, both Web and ancillary, the site needs. The Interjet even includes routing functionality. This scenario would be very appealing to a library with little technical staff whose ISP is willing to support a server appliance.

A Windows NT Scenario

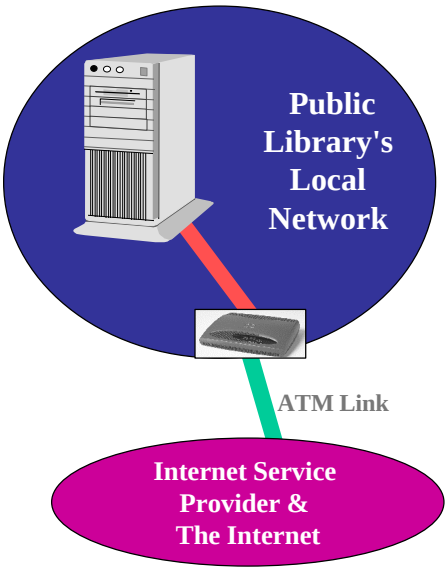
- **Hardware: Wintel PC, Pentium II Processor @ 500 MHz, 128M Memory**
- **Operating System: Windows NT**
- **Web Server: Microsoft IIS and Microsoft Site Server**
- **Static HTML documents – uploaded via FrontPage**
- **DNS (Domain Name System): provided by server at ISP**
- **SMTP (e-mail) Server: on separate server at library**
- **Connectivity: Cisco 4500 Router with T1 adapter**



Here we are running Windows NT Server as our operating system, with Microsoft s Internet Information Server, or IIS, as our Web server software. We are also running Microsoft s SiteServer, a collection of Web publishing tools that assist in managing large, complex sites. We rely on FrontPage for posting content to the server. This is a fairly typical Microsoft-oriented configuration. It would be easy to extend this server to run other Microsoft tools, such as Active Server Pages and a live database. Our DNS services are provided off-premises by our ISP. Our e-mail services are handled by a separate server at our site, which might or might not be a Microsoft-based server.

A Proprietary Unix Server Scenario

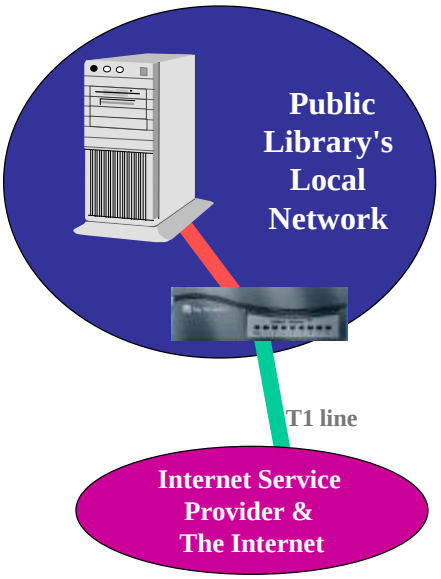
- **Hardware: Unix server (from HP, Sun, IBM, Compaq, etc.)**
- **Operating System: Vendor's Unix**
- **Web Server: Netscape FastTrack Server**
- **Static HTML documents – uploaded via FTP or HTTP**
- **DNS (Domain Name System): separate server at library**
- **SMTP (e-mail) Server: on separate server at library**
- **Connectivity: Leased T1 Phone Line, Cisco 4500 Router with ATM adapter**



Here we ve chosen a server from a traditional vendor of Unix hardware. We re running Netscape s FastTrack server. Our content providers publish to the server either via FTP or via Netscape s One Button Publish extensions. We are relying on separate server boxes at the library to provide DNS and e-mail services.

A Macintosh Scenario

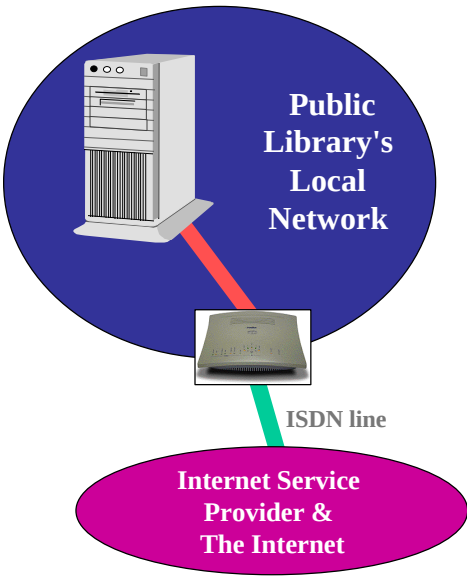
- **Hardware: MacintoshG3**
- **Operating System: Mac OS 8.5 with Appleshare IP**
- **Web Server: Webstar (by StarNine)**
- **FileMaker Pro database front ended by Lasso middleware**
- **DNS (Domain Name System): provided by server at ISP**
- **SMTP (e-mail) Server: on separate server at library**
- **Connectivity: Bay Networks Instant Internet Adapter**



Here we've chosen a Macintosh as our server. We're running Apple's conventional operating system at version 8.5 along with Appleshare IP extensions. Our Web server software is WebStar, which we've coupled with middleware from Lasso to offer live content managed by the popular FileMaker Pro database software, with the database software and Web server both on the same physical server.

A Linux / Apache Scenario

- **Hardware: Wintel PC, Pentium II Processor @ 500 MHz, 128M Memory**
- **Operating System: Linux**
- **Web Server: Apache**
- **Static HTML documents – uploaded via FTP**
- **DNS (Domain Name System): provided by server at ISP**
- **SMTP (e-mail) Server: on separate server at library**
- **Connectivity: 10 megabit bidirectional cable modem**



This scenario exploits the most popular Web server package, Apache, and the Linux operating system, which is enjoying increasing popularity as an alternative to proprietary Unix systems (and Windows NT). This scenario is appealing because it relies on an operating system and on server software that are free.

Although in this example we show DNS and e-mail being provided by separate servers, Linux comes with built-in DNS and e-mail capabilities, so it would be just as easy to run those services on the same box as our Web content. Here we might choose not to do so for several possible reasons:

- DNS and e-mail were already running on separate servers on site.
- We don t want any performance problems with our Web server.
- We don t want an outage in DNS or e-mail to force us to take down the Web server for service.
- We want different staff to work on the Web server than the DNS and e-mail, and we don t want to give superuser or root permission to other services for our Webmaster.

Interfacing to Databases

Besides the basic Web server software, you may need to install additional software to round out your server s capabilities. For instance, if you are going to connect to a live database, you will need to install the database software, whether it be Oracle, Microsoft Access, Microsoft SQL Server, or some other database. Your scenario may call for the server software and database software to reside on a single piece of server hardware, or you may choose to run your database software on a separate box.

Whether your database resides on the same box or a separate one, you will need to be able to interface your Web server software with your database. Fortunately, a standard known as ODBC provides a standard way to interface with most popular databases. Database middleware tools such as Cold Fusion provide the glue to connect your Web server to any ODBC-compliant database. If you choose to run a Windows NT server with Active Server Pages under Microsoft s IIS, you will be able to interface directly with an Access or SQL database. The Toolkit demonstration software assumes this model.

A Search Engine for Your Site

If your site grows to include a large number of pages, no matter how good a job you do of making the site navigable, navigation can be made easier with a local search engine. You may want to consider acquiring such a tool for your site. Here are your options:

- Install a free tool, such as the ICE indexer and search engine. Written by Christian Neuss and made available to anyone at no charge, ICE is written in the language Perl and is suitable for use on servers with up to several thousand documents. For a tutorial on installing and using ICE, see:

www.webreference.com/dev/dndcgi/start.html

- Some Web server software packages may come bundled with their own search engine. For instance, Microsoft s SiteServer includes a search engine capable of indexing content on your own server as well as crawling other servers you wish to index.
- You could acquire a commercial search engine. For instance, Infoseek markets its search engine under the Ultraseek brand name, and Altavista markets its engine as well. Fees for using these engines vary with the size of the site. Both vendors products are available for Unix or Windows NT servers. Both vendors offer evaluation copies of their products for free download and use during a limited evaluation period. Altavista also offers free use of their product to index sites with up to 3000 documents. Note that these products can be rather expensive; be sure to pursue any non-profit, governmental, or library discounts. See www.infoseek.com and/or www.altavista.com.
- You could use Excite for Web Servers, an older version of the search engine used on the Excite service and available for a variety of versions of Unix servers. This tool is available for free and offers much of the functionality users of the global Excite service are used to. See: www.excite.com/navigate/download.html.
- In lieu of installing a search engine on your site, you can leverage the index of a global search engine. For instance, Altavista allows a user to limit a search to a particular domain. For instance, to look for information on events at www.smallville.org, you would submit a search of the form events host:www.smallville.org. The search would return all pages in the index that have the word events from the Smallville Web site. In order to leverage the central index, you could install on your server a CGI script that appends the appropriate host: information to each query, after which the request is funneled to www.altavista.com for processing. Or you could accomplish this on the client side by using a bookmarklet in JavaScript. For details on how to implement this approach, see:
  www.webreference.com/js/column35/search.html

An excellent general resource on picking a search engine for your site is:

www.searchtools.com

There you will find information on the currently-available free and commercial alternatives, as well as case studies showing how some sites evaluated alternatives and chose their search engines.

Server Administration Roles

Many Web publishing shops make a basic distinction between the Webmaster and the System Administrator. The Webmaster has primary responsibility for organizing and placing content on the server, for coordinating content provider tasks, for connecting the server to external databases, and for analyzing server logs. The System Administrator is responsible for the administration of the hardware, the server operating system, the maintenance of authorized users, system backup, network management, and security.

Keep in mind that these are vague conventions, not hard rules. In some sites, the Webmaster and the System Administrator are the same person. In others, a team of a dozen

people may be responsible for different aspects of site management. In fact, a site may not use these labels to describe staff members. In any event, the project manager will have to decide who on the team has what responsibilities, and who is granted what level of system administrator privileges on the various servers deployed. (See Chapter 2 for a further discussion of the many roles in a Web publishing project.)

Choosing Your Server Hardware

We ve seen the wide variety of server hardware, operating system, and server software options. Now suppose you ve decided to buy an Intel-based server. How powerful a system should you buy?

Many sites get along just fine by buying a commodity PC intended for desktop use (with an Intel processor or with a competitive processor such as an AMD K6) and running Windows NT or Linux on the box. Vendors will try to sell you boxes marketed specifically to run as servers. What are the differences between server boxes and desktop ones?

- Server PCs tend to come with greater expansion capabilities: more PCI slots and more room for memory.
- Server PCs tend to come with SCSI disk drives built-in. Today s desktop PCs come with Ultra ATA IDE drives, whose performance is impressive, but which lags behind modern SCSI devices.
- Server PCs may offer RAID (Redundant Array of Inexpensive Disks) with important advantages in reliability and performance (see below).
- Server PCs tend to provide special disk drive bays that make it easy to swap defective drives with the system turned off (cold swap) or even with the system still running (hot swap).
- Server PCs tend to have processors (such as the Intel Xeon) and internal memory caching tuned for transaction processing.
- Server PCs tend to offer error correcting memory, which most of today s commodity PCs do not. This can improve reliability.
- Server PCs offer the option of multiple CPUs, which can be exploited by Windows NT or Unix for very high performance requirements.
- Server PCs tend to offer high-performance networking interfaces.
- Server PCs may offer with built-in high capacity tape drives and backup software.

Let s consider one server option, RAID, in a little more detail. RAID allows you to make efficient use of multiple relatively inexpensive disk drives to achieve better performance, reliability, or both. RAID spreads data across multiple physical disk volumes to achieve these goals. The industry has defined these levels of RAID:

- **RAID 0** is also known as data striping. This level improves performance by spreading data across two or more drives. Because data can be retrieved in parallel from multiple physical disks, performance can improve dramatically.
- **RAID 1** is also known as mirroring. Here you run pairs of identical drives. All data written to disk is written as a mirror image on both drives. Here you are gaining total

redundancy for your data at the expense of doubling the amount of disk you must buy for a given amount of content. If a mirrored drive dies, your server can continue running until you are able to install a replacement drive; then, the new drive is automatically brought back in sync with the remaining one. Mirroring offers some performance improvements when both drives are operational.

- **RAID 5** provides data striping with parity. This provides the performance benefit of striping along with redundant information spread across the disk volumes, so that your system can continue running if a single drive fails. RAID 5 isn t as simple as mirroring but you buy reliability at a far lower cost.

The alternative to RAID is to use a single disk volume or a set of volumes in the conventional way each disk holds its own data entirely, and no data within a single file is spread across drives. This is perfectly adequate for many Web sites, and probably adequate for the majority of CI projects.

Other considerations to look at when buying a server include the speed of the processor and the amount of memory. In recent years memory prices have declined so dramatically that it makes little sense to buy a server with less than 128 megabytes (or even 256 megabytes or more) of system memory. Buying more memory can mean a dramatic improvement in performance, especially if you have many concurrent users or are doing a great deal of live content work.

The case for buying the most powerful processor or multiple processors is not so clear. Vendors charge a premium for the very fastest CPUs. Most CI sites have no need to buy the very latest CPU at the very highest clock speed. Systems that support multiple CPUs cost more, and each additional CPU adds to the price. In addition, depending on your software environment, you may not be able to take full advantage of multiple CPUs. Think twice before you spend hundreds or thousands of dollars on additional CPU horsepower for your server.

Increasingly, vendor Web sites make it easy to comparison shop for systems both within the server and desktop categories. For instance, the Dell site makes it particularly easy to choose options from a Web form, seeing how much various configurations would cost.

A very powerful commodity PC that would make a perfectly adequate server for many if not most CI projects can be had for under $2500 including tape backup. A server-class machine can easily cost $5000 to $10,000 or more. Similar ranges of prices can be found among proprietary Unix server systems or Macintosh systems intend to be used as servers.

There are two schools of thought when it comes to buying server hardware:

- Buy the cheapest system that will handle your load for the next year, and plan on replacing it.
- Buy a system that will handle your anticipated load for three to five years.

Which school you follow will depend on your current budget and your budget cycles. Some sites with grant funds or other one time money like to buy extra capacity while capital funds are available. However, since computing power for the same amount of money doubles every 18 months, it s very expensive to buy capacity very far into the future.

Installing Your Server Operating System

If you buy a system that comes with the operating system installed, and if you choose not to do a re-install for learning purposes, you are ready to configure and run your server when you take it out of the box.

If you buy a system that doesn t have your desired operating system installed, you will need to install it from scratch. Typically you will work from a series of installation CD-ROMs. You may have to begin the process with a bootable floppy if the system arrives totally bare.

Because the Toolkit includes demonstration applications that run under Windows NT, we include complete instructions on installing Windows NT Server 4.0 software. This chapter is extremely comprehensive, including annotated screen shots of every step along the process of installation. However, because not all sites will install this operating system, we omit that material from the printed book. You will find this material on the Toolkit CD and Web site under *Software*.

System and Content Backup; Archiving

You will need to establish a regular program of backup for your system. Backups protect you from disk failures as well as human catastrophes such as accidental deletion of data. Here are some general guidelines:

- Most shops back up disk drives to tape. Tapes with sufficient capacity to match today s disk drives can be expensive; see discussion below.
- A common approach calls for daily backups of all changed data files, and weekly backup of all data on your system. You create pools of tapes for your daily and weekly dumps. You might have one tape series for every day of the week (or one each for Monday through Friday) and a separate series for your weekly dumps. The more

depth to your tape pools, the more confidence you have that you ve got all your important files are backed up.

- Some sites handle content backup separately from operating system and software backup; some even put software and data on separate disk drives. Because software tends to change at a different pace than content, this can provide important efficiencies. For instance if you make software changes infrequently, you might do a daily change dump of your content, and only back up your software weekly. Note that it s important to capture important configuration files, which tend to reside in the same folders as software; the only sure-fire way to do this is to back up everything.
- It s very important to periodically store your most critical files off-site. You can do this by taking one of your full dump tape set to an off-site storage location. A safety deposit box at a bank is a good choice. Increasingly network backup is becoming an option; vendors provide ways to archive your most critical files at their site across the Internet.
- If you employ RAID 1 or RAID 5, you may feel confident that your system is adequately backed up. Unfortunately, there have been cases in which RAID systems have failed in such a way that you are not protected. For instance, in some cases, a mirrored drive may fail, and you may not notice the failure. Eventually its mirrored partner fails, and now you have no data and no backup. Bottom line: it s a good idea to back up all data periodically even when you have RAID protection.

The industry offers data backup drives based on formats created for other purposes. For instance, DAT, or Digital Audio Tape, is a popular format for data backup; it is probably the most commonly-used format for server backup applications. The major formats for server backup are:

- DAT. These drives can hold from 8 gigabytes (8GB) of data up to 24G at a cost from under $1000 for the drive to over $3000, depending on transfer speed and data capacity. Data is backed up onto a 4mm cassette that looks like a DAT tape. Media costs are from $5 per tape to $25 per tape depending on capacity.
- Exabyte, or 8mm. These tapes resemble those used in camcorders. Drives cost $1000 or more. Tapes hold from 2.5 to 7 GB and cost from $5 to $12 each.
- DLT. These half-inch cartridges hold from 10GB to 75GB of data. Drives cost from $2000 to $5000 or more. Tapes cost about $50 each. These are very high-quality, high-performance backup devices.

Note that vendors may quote capacities of 12/24 or similar numbers. The first number is uncompressed; the second number assumes two-to-one compression. If your content is already in a compressed format such as JPEG, or if your backup software does data compression, you won t achieve an additional two-to-one compression on tape.

In any event, if you buy a drive with sufficient capacity, you may be able to back up your *entire* Web site onto a single tape. This saves the manual effort of loading contiunation volumes during the backup process and can be a great convenience.

In order to perform backups, you will need backup software. Your NT, Unix, or Mac server will come with built-in backup software, but you may find such software to be limited. Commercial backup software tools allow you to schedule backups, manage multiple tape

pools, and back up user desktop computers along with your server. In the Windows NT environment, ARCserve from Computer Associates is a popular tool. Seagate Software markets a competing tool, Backup Exec. These tools cost less than $500.

Archiving of content is a concept related to backup. When we speak of archiving, we generally think in terms of taking a snapshot of part or all of our content, with that snapshot kept indefinitely. Archiving can be done to the same tape media you use for backup. Alternatively, you may want to consider CD-R, CD-RW, or the new DVD-RAM as archive formats.

- CD-R allows you to store about 650 megabytes of data on a single CD, which can be read by any PC with a CD-ROM drive. A CD-R probably could not hold an entire CI site including software, but in many cases it could hold most or all of a site s content. Individual CD-R discs can be found for under $2 each.
- CD-RW offers the advantages of CD-R at a higher media cost about $12 per disc as of this writing. Unlike CD-R, CD-RW allows a single disc to be written on multiple times. The extra media expense of CD-RW would not be justified for archiving content; by definition, you want to write on an archive disc only once.
- DVD-RAM offers several times the capacity of CD-R up to 5.2 gigabytes. Thus DVD-RAM could in many cases back up on a single disc an entire CI site, or all of the content of a multimedia-rich site. As of this writing each DVD-RAM blank disc is expensive $50 or so but these prices will fall.

These formats can be useful for exchange of data as well. For instance, an off-site content provider may wish to deliver large amounts of content in CD-R form. Your webmaster can retain the CD-R disc or return it to the content provider; either way, another backup copy can be retained.

Other popular archive and exchange formats include Iomega s ZIP and JAZ drives, and Imation s Superdisk drives.

Log Analysis

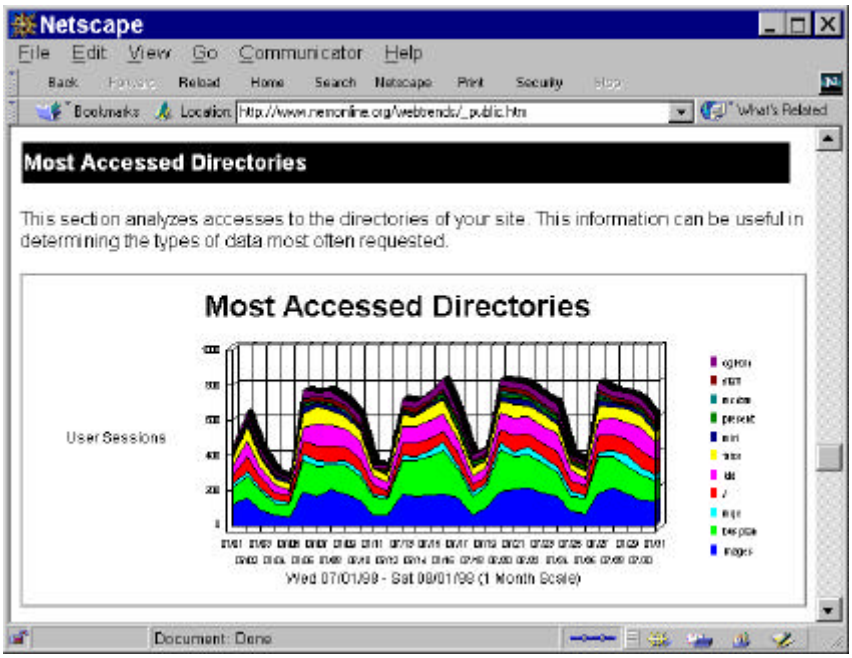
Most people who run Web sites want to know their hit counts how often the site in general, and certain pages in particular, are visited. Your Web server generates log files that hold information about every HTTP transaction: every time a file is fetched from the server, a log entry documents the date and time as well as the host name of the user s computer. Log analysis tools convert your logs from almost incomprehensible raw data into useful summaries of activity.

Log analysis tools allow you to summarize and analyze your traffic patterns across time, presenting tabular and graphical reports that greatly assist you in tuning your content and your site. Examples of the kinds of reports you can get include:

- A list of the most popular pages on your site.
- A list of the least popular pages on your site. If a page you think should be popular isn t, you can tune your site s layout and link structure to improve its visibility so more users will find it. Some tools offer path analysis, showing what hyperlinks users follow through your site.
- A report showing how users arrive at your site. Your logs contain referrer information that says what Web site a user visited *before* they followed a link to your site. You can tell which directories, search engines, personal or other kinds of sites are the most popular starting points from which your users find your content.
- A graph showing activity across days, weeks, or months.
- A list of the most popular domains from which your users visit your site.

These tools vary from free, public domain tools to commercial packages costing from $100 or so to $10,000 tools. Extremely expensive log analysis tools feature real-time data analysis and sophisticated multi-server analysis functions that are required by only the busiest commercial sites on the planet. Most community information sites could choose to spend no more than $250 for an adequate solution.

Here is a sample graph from a popular tool, Webtrends.



Here is a list of some of the most popular log analysis tools:

Tool	Vendor	Vendor Web Site	Notes
WebTrends	WebTrends	http://www.webtrends.com	A very popular log analysis tool that presents reports in tabular and graphical form.
Site Server	Microsoft	http://www.microsoft.com	Site Server is a family of server support and administration tools, including log analysis, site mapping, and a search engine.
Funnel Web	Active Concepts	www.activeconcepts.com	A high-end tool for Windows, Mac, and Unix servers.
NetIntellect	WebManage	http://www.webmanage.com	Another log analysis tool.

You will want to establish a policy as to what log information is kept for how long. Because logs can be used to trace activity by IP address, it is possible to discern in some cases which users access which pages on your CI site. This can be a violation of privacy, especially if the log files are published on the site or otherwise become public. Libraries will be especially sensitive to this issue, as there are ethical considerations and in some jurisdictions, legal ramifications of releasing individual patron access information.