

## FireSite Virtual Domain Manager

Welcome to virtual Web site hosting with FireSite! The FireSite Virtual Domain Manager is the most advanced single-IP-address virtual domain system ever created. We believe you be pleased with the power it gives you, but also with the quick and straightforward setup.

The FireSite Virtual Domain Manager is simple to use: you provide the Internet host names that you will be serving, and designate which folder on your server contains the files for each virtual Web site. From there, everything can be automatic, or you can modify a few optional settings if you wish. When properly configured, FireSite can serve an unlimited number of virtual Web sites to every user on the Internet.

The FireSite Virtual Domain Manager is available as a stand-alone product. It can also be combined with FireSite Web Site Accelerator, Speed Booster Edition to create multiple high-performance virtual Web sites on a single server. The Standard and Multimedia Editions of FireSite Web Site Accelerator have the Virtual Domain Manager built-in.

### Overview

The FireSite virtual domain system is the most advanced single-IP-address virtual domain system ever created on any platform.

#### Power

- Serve an unlimited number of virtual domains from a single IP address
- Does not create “messy” URLs in any modern browser
- Compatible with absolutely all “Web crawlers” and Internet search engines
- Provides flawless performance when accessed by any modern Web browser such as Netscape Navigator 2.0 or later, or Microsoft Internet Explorer 2.1 or later
- Even when accessed by “old” browsers such as Mosaic, over 95% of all requests are usually served instantly; others require one extra click
- Automatically makes all other Web server plug-ins and CGIs “virtual-domain-savvy”, even custom CGI scripts

#### Ease of use

- Fully-graphical interface is extremely straightforward to use
- Can be administered locally or over the Internet (password protected)
- Easy to upgrade from older virtual domain systems



## Getting Ready

In order to set up virtual Web sites on your server, you should be fairly familiar with:

**Web (HTTP) Background Concepts.** This includes the interaction of Web browsers and Web servers in general. You should have your Web server already set up and running on the Internet with one domain name before you begin to configure multiple ‘virtual’ domains.

**DNS Concepts and Setup.** You will need to understand “DNS records”, “DNS servers”, “IP addresses”, the “domain name registration” process. You should understand who runs the domain name servers for your registered domains. It may be someone in your organization, or it may be another party, such as your ISP. You will have to create new DNS entries for each virtual host name that you wish to serve from your server.

**Web Server Setup and Configuration.** You should be familiar with the operation of your Web server software (WebSTAR or compatible). In particular, you should understand how to configure and administer CGIs and plug-ins; FireSite Virtual Domain Manager is a Web server plug-in.

## Three “Rules” to remember

There are three important rules that you must keep in mind when using FireSite VDM:

**1) All host names are created equal.** Try to avoid thinking of your server as having one “main” host name and then some other “virtual” names. It may help to think of your existing “main” Web site as simply one of the virtual Web sites you’ll be hosting for a client -- in this case, the client is you.

Your server’s IP address may have a reverse-DNS record that points to one particular name, but that name is not particularly important.

**2) If possible, keep your “main” Web site in a subfolder.** You probably already have some Web files in your server’s root folder, including your “default home page”. To keep the virtual domain setup more consistent, move these “root-level” Web files down into a subfolder (called “~main”, for example). This setup will make it easier to remember item #1 (above), and will make your Web site more readily accessible from old browsers.

**3) Although rarely used, the multisite index page (MSIP) is fairly important.** FireSite does an outstanding job of “sorting out” requests for each virtual Web site’s pages. Typically, around 99.95% of requests can be handled instantly and automatically by FireSite. The other 0.05% of the time, when the request comes from an ‘old’ Web browser or from a Web-crawler’s first visit, FireSite cannot determine what page is being requested. In this case, FireSite serves the “multisite index page.” Once an old browser or Web crawler has accessed this page, they can access all your Web sites without further interruption, so it’s important to make sure this page lists all the virtual Web sites hosted on your server. Setting up the MSIP is covered in detail later on.

## **Installing the FireSite plug-in**

If you have not already done so, install the FireSite plug-in as described here.

The FireSite products are add-ons for Internet Web servers; you should have your Web server up and running on the Internet before installing any FireSite products.

FireSite software is distributed as a single file which contains several “locked” FireSite products; when you enter your FireSite Authorization Key as described below, it will unlock and enable the correct FireSite software on your server. If you enter Authorization Keys for more than one FireSite product, all of the authorized products will be activated at once and will work in concert together. Once you have set up the FireSite plug-in on your Web server, almost all administration will be accessed through a Web browser, as shown above.

### **First: Protect the FireSite Administration Pages with a REALM**

All FireSite configuration is accessed through “virtual” Web pages on your Web server. In order to make sure that only authorized users can access your Administration pages, you must set up a protected Realm and choose a username/password for yourself:

1. Make sure WebSTAR is running
2. Next, Launch WebSTAR Admin and select your WebSTAR server.
3. Under the “Configure” menu, choose “Realms...”
4. Enter a Realm Name of “FIRESITE-ADMIN”
5. Enter a Match String of “admin.fire”
6. Click “Add” and then click “Update”
7. Under the “Configure” menu, choose “Add Password...”
8. Select “FIRESITE-ADMIN” from the popup list of Realms.
9. Choose a username and password for the FireSite Administrator (you), and click “Add”
10. Quit WebSTAR Admin.

### **Next: Install the FireSite plug-in**

1. Quit WebSTAR.
2. In the Finder, select WebSTAR, and choose “Get Info” from the File menu. Increase WebSTAR’s Preferred size by 4000 K. Close the Get Info window. You may be able to reduce the memory partition later, but it is best to make sure you have plenty of memory available for FireSite while you are setting it up.
3. Locate the file called “FireSite” that came with your distribution package (you may have to ‘unstuff’ it). Drag the “FireSite” file into the “Plug-ins” folder located in the WebSTAR folder.
4. Relaunch WebSTAR. The startup may take an extra minute while FireSite sets up its database.

You should see a message in the WebSTAR status window that reads “Welcome to FireSite from ClearWay Technologies!” You may also see a message warning you that no “authorization keys” are installed. This is OK; you will be entering the authorization key shortly.

The rest of the configuration is performed using a Web browser to access virtual “Administration” pages on your Web server. Start up a Web browser such as Netscape Navigator or Microsoft Internet Explorer (the browser must be able to display tables.)

Enter the address of your Primary server into the Web browser, followed by “admin.fire”:

`http://www.your-server.com/admin.fire`

You will be prompted for your FIRESITE-ADMIN username and password. Enter both and click OK. You should be presented with the FireSite License Page. Review the terms of the license agreement, and if you find them acceptable, click Accept. If the license terms are not acceptable, return the software and materials immediately to ClearWay Technologies, Inc.

Once you click “Accept”, you will see the FireSite Authorization Key page.

### **Now enter your first Authorization Key**

You will find your FireSite Authorization Key on a colored paper certificate in your package. If you have Keys for more than one FireSite product, select the one you wish to install first; others can be entered later. (If wish to use both a “Speed Booster” Key and a “Virtual Domain Manager” key, you should install the Speed Booster Key first.)

Enter the selected Authorization Key carefully into the space provided, and click “Enter”. If you entered the Key correctly, FireSite is now up and running on your server.

You should now see the “Welcome To FireSite” central administration page.

## Configuring Your Web Server

If you have not already done so, install the FireSite server plug-in as described in the “Installing FireSite” document.

### Step 1: DNS Setup

Create entries in your DNS server for each host name that you wish to serve. All of the host names should resolve to the single IP address of your Web server computer. If your ISP provides you the primary DNS service for your domains, ask them to create the appropriate DNS records.

We recommend using Address (A) records for each domain name, rather than alias (CNAME) records.

For example if your Web server’s IP address is 123.45.67.89, and you are going to run Web sites for FirstCo, SecondCo, and ThirdCo, then the DNS records for “www.firstco.com”, “www.secondco.com”, and “www.thirdco.com” should all resolve to 123.45.67.89.

**Note:** Besides having DNS records in your DNS server, each new top-level domain (such as “firstco.com”) must be registered with the proper Internet registration authorities, or people browsing the Web will not be able to locate your Web server. Additional registration information can be found at <http://rs.internic.net/>

**Note:** You do not need a new top-level domain for each new virtual Web site. For example, if you already are registered for the “grafyx.com” domain, you can freely add new domain names without having to register them, as long as they all end with “grafyx.com”. You could create “www.grafyx.com”, “portfolio.grafyx.com”, and even “firstco.grafyx.com” or “secondco.grafyx.com” simply by adding the correct DNS records. This can be a quick and easy way to get started with virtual domains without the delay or expense of registering a new top-level domain. Ultimately, most Web hosting clients (FirstCo), will want their own domain names, but you can get started with “firstco.grafyx.com” if you wish.

### Step 2: Clean up your server root folder

Because “all host names are created equal”, all Web sites on your server should also be “equal.” Since you probably already have some Web pages, HTML and graphics in your server’s root (WebSTAR) folder, you should create a new folder (called “~main”) and move your existing “top level” pages and graphics into this folder. There is no need to modify your HTML code.

### Step 3: Create a folder for each additional Web site

Create a new folder for each Web site that you will be serving. By convention, we recommend that you name the new “site folders” names that start with a tilde (“~”). For example, you might create “~firstco” and “~secondco” folders to hold two new Web sites.

If you have been using another virtual domain system, and you already have folders set up for your various Web sites, there is no need to rename them. However, FireSite will be able to handle Web crawlers and old browsers best if your sites’ folder names are distinct from any other folder names on your server. For example, it’s better to have a Web site folder named “firstco-pages” than “firstco”, because it’s less likely that there will be another subfolder somewhere called “firstco-pages.” In most of our examples, we begin Web site folder names with tildes (~).

### Step 4: Create a “multisite index page”

The multisite index page (MSIP) will be served only in those (relatively rare) cases when it is

impossible for FireSite to determine which Web page a browser is requesting. It will only be served to 'old' browsers and Web crawlers, and even then it will only be served when the requested URL cannot be resolved to one specific file on your server. The multisite index page is never served to 'modern browsers' such as any recent versions Netscape Navigator, or Microsoft Internet Explorer. While the AOL 3.0 browser is 'modern', the previous 2.7 version will sometimes be served the multisite index page.

The multisite index page should be in your Web server's root folder, and not inside any of your virtual site folders, since it's not a part of any one site.

Create a new HTML file called "multisite-index.html". This HTML file should (a) greet the user and (b) offer a list of hyperlinks to the various Web sites on your server. Here is a simple example of a multisite index page:

```
<title> Welcome! </title>

Welcome to Grafyx Web Services!
Please select a Web site to visit:

<p><a href="http://www.firstco.com/">FirstCo Ltd. - fine goods</a>
<p><a href="http://www.secondco.com/">SecondCo Inc. - superior services</a>

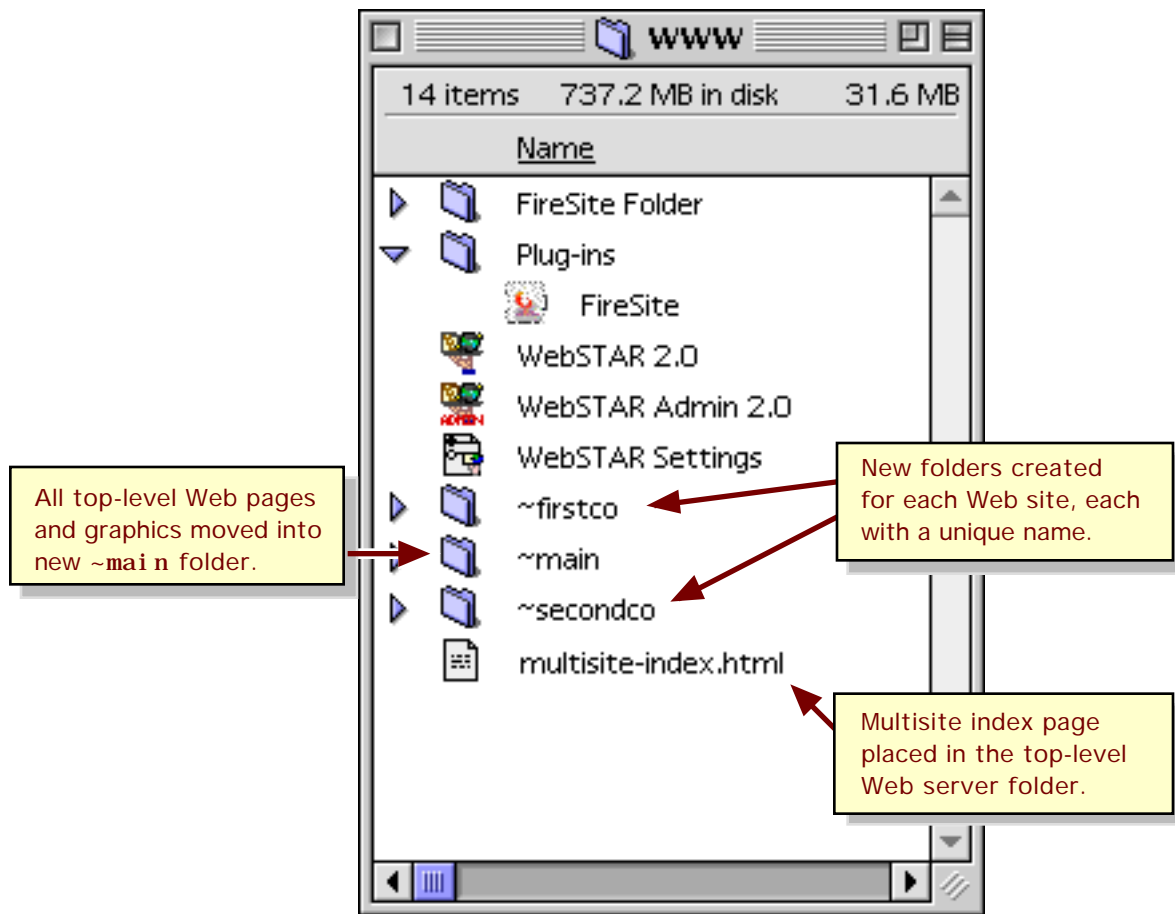
<p>To learn more about our own Web services and technologies,
<a href="http://www.grafyx.com/">visit the Grafyx Web site!</a>
```

Notice a few things about this page: it lists the virtual sites each by their own host name-- it does not provide the "folder" information as to where each site is located. FireSite will dynamically adjust these URLs as needed. You should always write URLs as if each site were on a separate Web server.

Also notice what this page doesn't do: it doesn't say "error", and it doesn't accuse the visitor of "using an outdated browser." Keep the tone positive and friendly, and remember that the person using the browser **wants** to visit your site.

## Checkpoint

Your Web server root folder should look roughly like this:

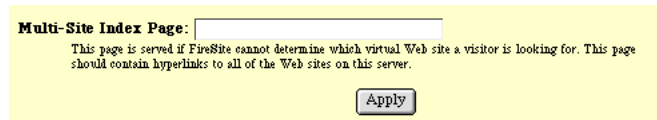


Note that there are no longer any "Web pages" in the server root folder except for the multisite-index.html. All the files that were at this "top level" have been moved into the new "~main" folder.

## Configuring the Multisite Index Page (MSIP)

Once you have created the “multisite index page” (MSIP) as described in the previous section, you need to tell FireSite where it is and what you have called it.

Access `http://www.yourserver.com/admin.fire` from your browser and click on “FireSite Setup.” You will see the FireSite General Settings page. This page has four sections (possibly more depending on the various FireSite products that you are using together.) One early section will be “Multisite Index Page”:



A screenshot of a web form titled "Multi-Site Index Page:". It features a text input field, a paragraph of explanatory text, and an "Apply" button.

**Multi-Site Index Page:**

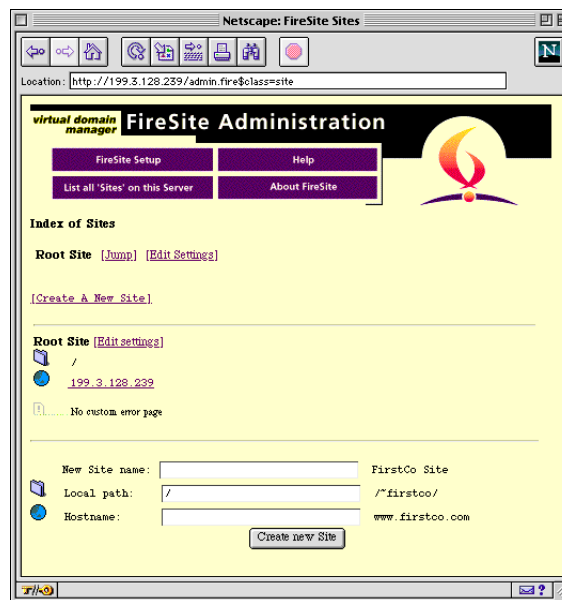
This page is served if FireSite cannot determine which virtual Web site a visitor is looking for. This page should contain hyperlinks to all of the Web sites on this server.

Enter the name of your multisite index page (“multisite-index.html”) and click “Apply.”

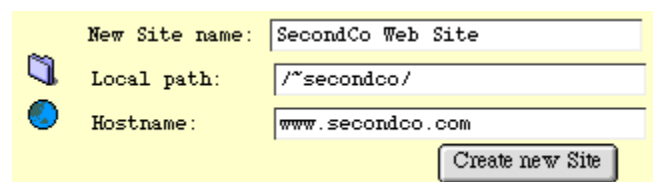
## Configuring FireSite to Serve a New Virtual Domain

To configure FireSite for a new virtual domain, access `http://www.yourserver.com/admin.fire` from your Web browser.

Click on “List all ‘Sites’ on this server”. You will see the “Site Index”:



To configure a new virtual Web site, you will need to enter three pieces of information: a “name” that you want to call the site (“SecondCo Web Site”), the Internet host name (“www.secondco.com”) that will be the “preferred” domain name for this site, and the folder that the Web site’s files will be stored in (“~secondco”).



A screenshot of the "Create new Site" form. It contains three input fields: "New Site name:" with the value "SecondCo Web Site", "Local path:" with the value "/~secondco/", and "Hostname:" with the value "www.secondco.com". A "Create new Site" button is at the bottom right.

New Site name:

Local path:

Hostname:

Enter the information into the fields provided on the administration page, and click “Create new Site.”

The virtual Web site is now defined and active! Put a few pages in the new Web site’s folder and try accessing them through your Web browser. You should be able to access the pages using “clean” URLs such as “<http://www.secondco.com/default.html>” that do not include the “/~secondco” folder name.

You can define as many Web sites as you wish.

In the next section, you’ll learn how to configure some of the optional settings for a virtual site.

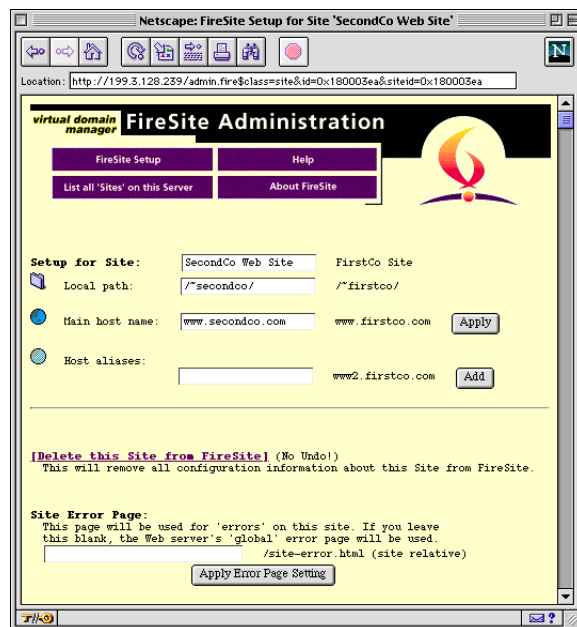
## Custom Settings for a FireSite Virtual Site

FireSite allows you to define a number of “custom” settings for each virtual site. Each virtual site can have a custom “Error” page, as well as multiple Internet host names.

To configure the custom settings for a virtual Web site, click on “List all ‘Sites’ on this server”, and locate the virtual site you wish to edit. Click on the “[Edit settings]” link next to the site.



You will see the settings page for the virtual site you have selected:



## Multiple Host Names for one virtual site

You can assign multiple host names to each Web site. For example, you might want both “[www.secondco.com](http://www.secondco.com)” and the shorter “[secondco.com](http://secondco.com)” to both refer to the same Web site.

To define a “host name alias” within FireSite:

1. Enter the host name alias (“secondco.com”) into the field on the administration page.
2. Click “Add”

This feature is also useful if you are upgrading to FireSite from Open Door Networks’ HomeDoor system. To do this, update all of your virtual sites’ DNS records (www.secondco.com **and** www2.secondco.com) to point directly at your server’s single IP address. Configure FireSite with the simple “www” name as the main host name for each virtual site, and add the “www2” name as a host alias for each site. This way, all old bookmarks will continue to work as usual, even if they contain “www2...”

### Custom Error page for a virtual site

Ordinarily, if a browser requests a Web page that does not exist, WebSTAR responds with a generic error page that is used for the whole server. FireSite lets you design different “error” pages for each virtual site.

To designate a custom Error page for this Web site:

1. Create a custom error page of HTML. Name it something easily recognizable, like “SecondcoError.html”.
2. Place the error page file inside the virtual site folder that it is designed for.
3. In the FireSite administration page, enter the name of the file, proceeded by a slash (“/SecondcoError.html”) into the space provided at the bottom of the page.
4. Click “Apply Error Page Setting.”

Once this is configured, any requests for nonexistent files will be answered with the site-specific Error page instead of the generic WebSTAR page.

**Note:** If you are using a plug-in to handle your HTML files (such as Maxum’s NetCloak or StarNine’s WebSTAR SSI), you should read the “Error/Action Handling” portion of the next section. It tells you how to configure FireSite’s custom Error pages to work with NetCloak or SSI.

### Resettable Statistics

FireSite keeps full traffic statistics for all virtual sites. FireSite maintains a set of cumulative “lifetime” statistics, and a second, resettable set of statistics that can be used to collect billing information or to analyze traffic patterns across your virtual sites.

Site Statistics						
	Since last reset Tue, 17 Jun 1997 01:11:40			Cumulative		
	Physical	Virtual	Total	Physical	Virtual	Total
<b>Hits</b>	52,299	206,286	258,585	115,453	356,503	471,956
<b>Bytes</b>	217,736,592	1,228,105,679	1,445,842,271	428,279,250	2,600,450,692	3,028,729,942
<a href="#">[Reset Now]</a>						

Like a car’s “trip odometer,” you can reset FireSite’s resettable statistics at any time. Each site’s statistics can be reset independently, or you can reset the statistics for all sites simultaneously for all sites from the “List All Sites” administration page. If you are not using FireSite Acceleration, you will see only “Physical” hits and bytes listed.

## FireSite General Settings for Virtual Domains

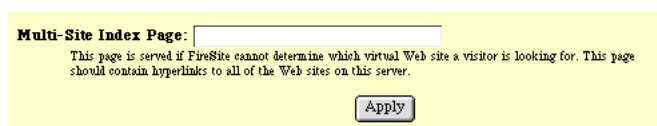
There are a few settings which affect your entire FireSite server. These include how FireSite handles “missing files” handled by an action, the way that FireSite makes all other plug-ins and CGIs virtual-domain savvy, and the “Multisite Index Page” that is sometimes served to old browsers.

To configure FireSite’s General Settings, access <http://www.yourserver.com/admin.fire> from your browser and click on “FireSite Setup.” You will see the FireSite General Settings page.

### The Multisite Index Page

The multisite index page (MSIP) will be served only in those (relatively rare) cases when it is impossible for FireSite to determine which Web page a browser is requesting. It will only be served to 'old' browsers and Web crawlers, and only when the requested URL cannot be resolved to one specific file on your server. The multisite index page is never served to 'modern browsers' such as any recent versions Netscape Navigator, or Microsoft Internet Explorer. While the AOL 3.0 browser is 'modern', the previous 2.7 version will sometimes be served the multisite index page.

The multisite index page should be in your Web server’s root folder, and not inside any of your virtual site folders, since it’s not a part of any one site.



**Multi-Site Index Page:**

This page is served if FireSite cannot determine which virtual Web site a visitor is looking for. This page should contain hyperlinks to all of the Web sites on this server.

Enter the name of your multisite index page (“multisite-index.html”) and click “Apply.”

You should not leave this field blank; that would prevent FireSite from serving your virtual sites to Web crawling search robots.

### Error/Action handling

Most URLs (including HTML, GIF, and JPEG URLs) correspond directly to files on your server’s hard disk drive. Here are some examples of file-based URLs:

<a href="http://www.yourserver.com/default.html">http://www.yourserver.com/default.html</a>	Plain HTML file
<a href="http://www.yourserver.com/logo.gif">http://www.yourserver.com/logo.gif</a>	GIF file
<a href="http://www.yourserver.com/product-db.qry">http://www.yourserver.com/product-db.qry</a>	Tango “Query file”
<a href="http://www.yourserver.com/dynamic.html">http://www.yourserver.com/dynamic.html</a>	HTML processed by NetCloak

However, some URLs have no ‘file’ associated with them. Here are a few example of URLs that do not come from disk files:

<a href="http://www.yourserver.com/RELOAD.ncl">http://www.yourserver.com/RELOAD.ncl</a>	NetCloak reload command
<a href="http://www.yourserver.com/homepage.count">http://www.yourserver.com/homepage.count</a>	CountWWWebula
<a href="http://www.yourserver.com/.clear">http://www.yourserver.com/.clear</a>	ClearWay Simple Cache Manager

Notice that all these “non-file-based” URLs come from plug-ins or CGIs.

Ordinarily, when FireSite sees a request for a URL that should be processed by a plug-in, such as a Tango query URL like “/product-db.qry”, it simply transfers control to the plug-in (Tango) regardless of whether an actual file exists or not, because some ‘plug-in’ URLs don’t need a disk file.

This can result in inconsistent error handling, especially if all your HTML pages are processed

by a plug-in such as NetCloak or WebSTAR SSI. However, FireSite can be configured to handle these situations correctly.

Locate the “Error/Action” panel:

**Action Error Processing**

For a given suffix/action pair, specify whether or not FireSite should serve an error page or let the action handle the request when FireSite can not find a file to satisfy the request.

Action Name	Suffix	If Requested 'File' Is Missing...
CLOAK_PI	.HTML	<input type="button" value="Serve Error page"/>
CLOAK_PI	.NCLK	<input type="button" value="Run Action anyway"/>
COUNTWWEBULA	.COUNT	<input type="button" value="Run Action anyway"/>
SSI	.SSI	<input type="button" value="Serve Error page"/>
TANGO	.QRY	<input type="button" value="Serve Error page"/>

For each Action/Suffix combination, you can instruct FireSite what to do if a requested URL does not have a disk file. In this example, the file-based actions such as NetCloak for HTML files, Tango (.qry files), and WebSTAR SSI (.ssi files), FireSite has been configured to serve the Error page if the file is not found. For the non-file-based actions (CountWWWebula and NetCloak NCLK commands like “RELOAD.NCLK”), FireSite has been configured to pass the requests directly along to the plug-in.

Note that you can have separate settings for separate Action/Suffix combinations, as shown with NetCloak HTML (Error if file missing) and NetCloak NCLK (Run NetCloak even if file missing).

## URL Cleaning

Whenever FireSite receives a request for a file served from a ‘virtual domain’, it checks to see if the requested URL contains redundant information. For example, the URL might contain the path to the virtual domain’s folder, as well as the virtual domain’s host name. FireSite can automatically “clean” these redundant URLs, and does so by default if the browser can handle the “cleaner” URL. FireSite redirects the browser to the “cleaned” URL automatically.

However, if you are using an IP-mapping router (such as Vicom’s SoftRouter), or a multiple-IP address network driver (such as OpenDoor Networks’ HomeDoor, or a version of OpenTransport that supports multiple IP addresses), you may wish to disable this “automatic cleaning”, because it may create redirection loops.

**URL Cleaning**

FireSite can automatically clean URLs that contain redundant information. For example, `http://www.firstco.com/~firstco/` turns into `http://www.firstco.com/`. You should keep this enabled unless you use an IP-mapping router or driver.

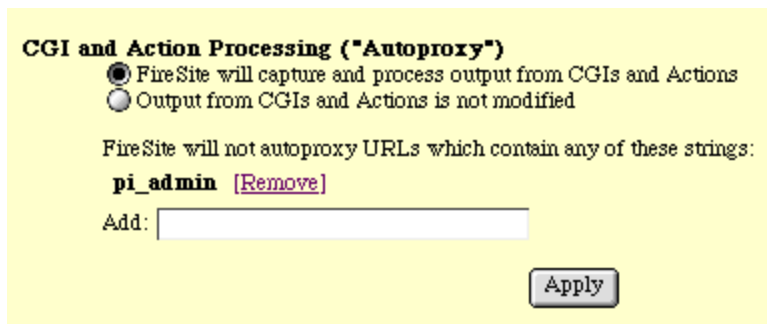
☒ Clean redundant URLs  
☐ Leave redundant URLs unchanged

## CGI and Action Processing (“Autoproxy”)

FireSite re-writes all HTML requested from your Web server as it is being transmitted to the browser. This “on the fly” rewriting allows FireSite to adjust all the HREF and SRC URLs on your Web pages to make sure they reflect your virtual-domain setup correctly.

FireSite uses an advanced technique called “Autoproxy” to capture the HTML generated from dynamic CGIs and plug-ins. By using the Autoproxy system, FireSite can make any other plug-

in “virtual-domain savvy” by adjusting the URLs that it generates. However, you may wish to exclude certain plug-ins, CGIs, or directories. By specifying a an “exclusion string”, you can configure FireSite to not use the Autoproxy system for certain URLs:



The screenshot shows a configuration window titled "CGI and Action Processing ('Autoproxy')". It contains two radio buttons: the first is selected and labeled "FireSite will capture and process output from CGIs and Actions", and the second is labeled "Output from CGIs and Actions is not modified". Below these is a text label "FireSite will not autoproxy URLs which contain any of these strings:". Underneath, the text "pi\_admin" is displayed next to a "[Remove]" link. Below that is a text input field labeled "Add:". At the bottom right is an "Apply" button.

To exclude a certain CGI or plug-in from the Autoproxy system, type the corresponding “suffix” into the space provided and click “Apply.”

**Note:** There are two kinds of “autoproxy” that FireSite will use. With older plug-ins and with all CGIs, FireSite uses a TCP/IP based autoproxy system to capture output from other server add-ons. This will sometimes cause you to see “extra” hits in your WebSTAR log that appear to come from the server itself. This is a normal consequence of using these older CGIs or plug-ins with FireSite virtual domains.

The newest versions of many popular plug-ins support “PIXO” a much faster, more efficient system for communication between plug-ins. FireSite fully supports PIXO communication with PIXO “Source” plug-ins such as NetCloak 2.5 (NetCloak 2.5d6 and later support PIXO.) All other things being equal, you should choose plug-ins that support PIXO communication; they will make your server run faster and smoother. When communicating with a PIXO-enabled plug-in, no “extra hits” are recorded.