Zope Administrator's Guide

Document revision 1.0

Modified September 29, 1999

For Zope version 2.0.1

Table of Contents

Introduction

This is the Zope Administrators Guide for Windows as such this guide's focus is for running Zope on Windows 95, Windows 98 and Windows NT. Please refer to the Zope Administrators Guide for Unix, if you are looking for instructions for running Zope on a Unix derived operating system.

This guide covers almost everything that anyone would want to know about the different ways Zope can be installed and ran. This guide is for developers who wish to immediately begin developing web applications, system administrators that have been charged with integrating Zope into their organizations current infostructure, users who wish to run Zope with their current ISP, or ISPs that wish to offer Zope as a service to their customers.

The details of constructing a web site using Zope is beyond the scope of this document. See the "Zope Content Manager's Guide" for information on this subject. This document focuses on how Zope can be installed to run with various web servers like Apache, Microsoft's Internet Information Server, Netscape's Enterprise Servers, or ZServer which is included with the Zope distribution. Although we will try to make as few assumptions of your technical background as possible; a working knowledge of how to install and run CGI scripts using your respective web server would be beneficial. You might consider reading your web server documentation first before pursuing the rest of this documentation.

NOTE: Digital Creations has made a commitment to the Zope community to deliver a set of documentation that will shed some light on the "inner mysteries of Zope" and remove some of the hurdles that the first time Zope user must currently face. Developing a set of quality documentation takes time. Often it takes longer to properly document a large system than it does to develop it! In the interest of supplying the community with the much sought after documentation this guide has been released in a state that is far from complete. It is hoped that the community can benefit from the information contained in this documentation immediately rather than make them wait for the lengthy documentation process to complete. Rest assured that the incompleteness of this documentation will be remedied shortly.

Getting Started

What is Zope

"**Zope**" is an acronym for "**Z Object Publishing Environment**." In Zope, web pages are built by adding "objects" and modifying object "properties." Several built-in objects like *Folders, Documents*, and *Images* are included in the software. The system is organized into *Object Hierarchies*, much like a conventional file system. A simple hierarchy of objects might include a *folder* containing *documents*, *images* called in those documents, and even sub-folders to further organize or delegate content. This is all done through the web browser rather than accessing the file system directly.[1]

Getting Zope

Zope is distributed in two formats pre-compiled for your respective operating system or in source form.You will need one of these distributions whether you plan on running Zope standalone with ZServer or in conjunction with a third party web server. The pre-compiled versions are simpler to install and are more convenient because there is no need to download additional components and an installation program is included that will prompt you for the required information necessary to run the application. On the other hand the source distribution are smaller and takes less time to download because you do not need to download because additional programs or libraries if you already have them installed on your computer. Detailed instruction for compiling Zope from the source distribution are available in the Maintenance and Administration chapter.

Download the latest stable version of **Zope**. At the time of this writing the current precompiled version of Zope for Microsoft Windows is available at **http://www.zope.org/Products/Zope/2.0.1/Zope2.0.1-win32-x86.exe** but you might want to check **http://www.zope.org/Products/Zope** for the latest version.

Note: For some reason most browsers for Windows will change the name of the file from *Zope-x.x.x-win32-x86.exe* to *Zope-x_x_x-win32-x86.exe* which won't make a difference when installing the program.

Installing Zope

The Zope distributions for the Windows platform are all self-extracting compressed archives. To begin the installation process we need to run the file, so double-click it or run it from a DOS window. You'll notice that the archive has both the version number of Zope and the platform which you are running.

You should see a standard Windows installation screen, with a window welcoming you to the Zope installation. The window will tell you which version of Zope you are installing. Click **Next** to begin the installation.

The window will now display for you the terms of the Zope Public License. You should read through this and make sure you understand the terms and conditions of the license. Once you have finished, click **Accept** to accept the terms and conditions, and continue the installation.

1. Actually Zope offers a few other methods for creating and manipulating content including WebDAV support. If you are running Zope in combination with ZServer, you can use FTP.

Next, you will be prompted for a name for the site. This name will be used as the name of the directory under which this Zope installation will be placed. Because this name will be used as a directory name it cannot have any spaces or punctuation characters. It will also appear in the Add/Remove Programs list in the Windows Control Panel so you can easily uninstall each one separately. Click **Next** when you have filled in the site name.

You will be asked to choose a location to install Zope. Whatever directory you choose will be the *Zope installation directory.* This is where all of Zope's files will be placed. This screen also shows how much disk space you have and how much will be left after you have installed Zope. Click **Next** once you have selected a directory.

Finally, you will be prompted for a username and password to be used for the super manager account. This is the account that will give you complete access to the entire site. Pick a username and password and click **Next**.

This is all the information Zope needs to begin installing. Click **Next** to start the installation procedure. You should see the program begin copying files. If you are installing Zope on Windows NT the installation program will prompt you at this point wether you would like to have Zope started automatically as an NT Service or you would like to start Zope manually. When the installation is completed, click **Finish** to exit the installer.

Running Zope for the First Time

Assuming that you chose the default options while installing Zope you should now have a copy of Zope installed in ***c:\program files\website***. Zope comes with it's own web server named ZServer. ZServer is a very fast very easy to use web server. The installation program has already configured ZServer to run on your machine the only thing that you may need to do is start it. If you are running Windows NT and chose to ran Zope as an NT Service when you installed Zope, you can skip the next five steps and continue reading from there. If you did not run Zope as a Windows NT service or you are running Windows 95 or Windows 98 you will need to manually start the Zope process. Follow these steps below.

1. Start windows explorer by clicking on **Start->Programs->Windows Explorer**.
2. Double click on the **c** drive in **Explorers** left hand pane.
3. Double click on the "Program Files" folder in the right hand pane.
4. Double click on the folder named "WebSite" in the right hand pane.
5. Start Zope by double clicking on *start.bat*

Of course if you picked a different directory and drive when you installed Zope you will need to adjust the above directions accordingly.

Once Zope is started you can point your favorite web browser at **http://*localhost:8080***[1]. If everything worked you should see the welcome page which gives you the option to visit the management screens. If you select this link you will be prompted for a username and password. Enter the username and password that you selected when you installed Zope, you remember them don't you? If not, consult the section further on in this document about changing the superuser password.

If your intention was to get Zope installed and running so that you can begin developing web-sites then feel free to stop here. In the next chapter we are going to examine the background technologies that a Zope administrator needs to be aware of to get the most benefit from Zope.

1. The address **http://localhost** has a special meaning to web browsers. It means "this machine." If your machine has a different name you can just as easily access Zope by type pointing a web browser at **http://yourmachinename.com:8080**

Background

In this section, we are going to cover some of the basics of running web applications. The information contained here should give you a good idea of the different ways Zope can be ran.

It is assumed that the you know how to properly configure your computer to use TCP/IP Networking and that your machine is functioning on your network. If it is not please make sure that the networking is properly setup on your machine at this time.

What is CGI?

CGI for the uninitiated is an acronym that stands for the "Common Gateway Interface". CGI is the method that web servers use to communicate with programs that web developers write to respond to user input from the web. If you have ever filled out a form on the Internet to comment on a article or purchase something online chances are that, your input was processed by a CGI script in the background. The diagram below depicts a generic web request that is processed by a CGI request.

1. Someone requests a document from the web server [1]
2. The server determines that requested document is actually a program[2] and runs the program passing the user input, if any.
3. The CGI server process the input and does something special with it, perhaps stores in a database or uses the input to preform a query on a search engine. The script returns the results formatted as a proper HTML document back to the server.
4. The server then returns the results back to the web browser.

Figure 1. CGI Overview

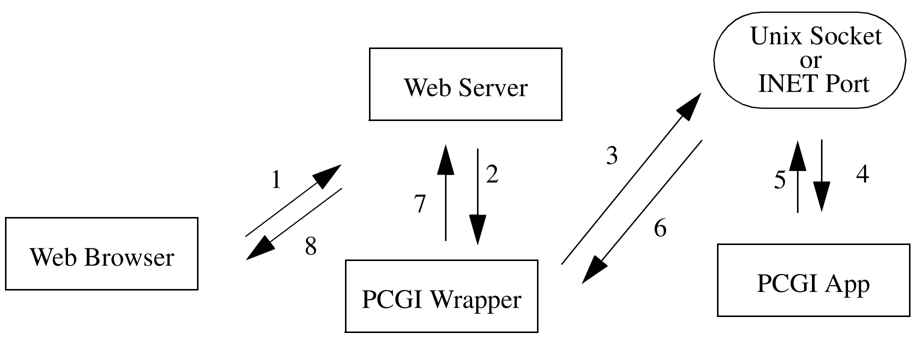
1. Someone typed *http://www.somewhere.com/somedocument* into their web browser.
2. How this is determined is based upon your web server, refer to your web server's documentation for in depth detail on configuring your server to run CGI scripts.

This method works for the majority of tasks that modern websites need to do, such as collect input from a form and e-mail it to somebody. One disadvantage from this method though is that the web server must start the CGI script every time to process the request. This wastes valuable CPU cycles loading the program into memory not mention the CPU cycles that are lost if the CGI program has to work with any configuration files or perform some other form of data processing. Zope is slightly more complicated than the typical CGI script and includes it's own database for storing thousand if not millions of objects. You can imagine how slow Zope would respond if it had to be started a new every time the web server receives a new request.

Zope is a "long running process" so named because it runs as a background service (or daemon in Unix parlance) that does not end after the web request. In this way, Zope can answer the next request that is received by the web server and avoid the start up overhead that CGI programs incur. If you are not using ZServer, the web server that comes with Zope, then you will need to configure your web server to either use a "helper" application called PCGI. PCGI is a light weight program that your web server launches on each request to talk to Zope. PCGI is usually running in the background or you can use a special plug-in written speciffically for your web server, if available.

What is PCGI?

PCGI is an acronym that stands for "Persistent CGI". It is a method developed originally by Digital Creations to circumvent some of the process launch overhead that was mentioned in the above CGI section. The way it works is that a smaller application known as the "pcgi_wrapper" forwards a web request from the web server to an external application that is listening to a specified INET port or a named socket if running on UNIX. The diagram below illustrated the process.



As you can see the process is similar to the way that Web Servers work with CGI scripts with some additional components.

1. Someone requests a document from the web server.
2. The server launches the PCGI Wrapper program passing it the request.
3. The PCGI Wrapper program sends the request to an INET port or a Unix Socket. (Note: the PCGI Wrapper is capable of launching the PCGI application, if it is not currently running.)
4. The PCGI Application receives the request from the socket or port.
5. When the PCGI Application is done processing the request, it sends the results back through the socket.
6. The PCGI Wrapper reads the results from the socket.

7. Then the PCGI Wrapper passes the results back to web server and quits.
8. The server then returns the results back to the web browser.

If you are a quick one, you might be asking yourself whether there is a performance hit taken from starting a new instance of the PCGI Wrapper program on every request. The answer to this question is yes. There is a small hit taken launching the PCGI Wrapper but it is small enough to not be noticed on any but the most trafficked of sites.

If you are using the Apache server, you might want to check out mod_pcgi which is currently being developed by somebody. This plug-in allows the Apache web server to communicate with the INET or Unix Socket directly and avoids the need to launch an intermediate process to pass the request to the PCGI Application. This module is still in early stages of development but many have reported success using this module. You can find out more information about mod_pcgi at http://somewhere.com.

If you plan on running Zope using PCGI you will need to use it in conjunction with ZServer or another third party web server.

What is FastCGI?

FastCGI is a solution to the CGI process launch problem developed by OpenMarket that is almost identical to Digital Creations PCGI. FastCGI has a few advantages over PCGI. FastCGI is not limited to Python development libraries exist for C, C++, Perl and other languages. In addition, plug-ins for multiple web servers already exits so that web servers can communicate directly with the "long running process" without having to launch any intermediate programs. In addition, a program called **cgi-fcgi** exists that works just like the pcgi-wrapper mentioned above for servers that do not have a native FastCGI plug-in.

Unfortunately at the time of this writing FastCGI support for Zope is not available to the public. Although, this situation should be remedied rather shortly. In fact this document will cover everything you need to know get your respective web server to run with FastCGI in anticipation of FastCGI for Zope's future release.

The only other difference between PCGI and FastCGI integration with Zope is unlike PCGI, FastCGI support will only work if you use ZServer. You will see in the next section that you that not only can you run ZServer by itself that you can also run it in conjunction with another third party server. There are a number of reasons why you would want to do this.

Running Zope

ZServer

ZServer is by far the easiest and fastest way to start using Zope. ZServer is the integration of Zope with the high-performance, multi-protocol Medusa server toolkit.

The advantages of ZServer are:

- You do not need to run Zope behind a third party web server.
- You can access Zope through other protocols such as WebDAV and FTP.
- ZServer can handle multiple web requests at the same time because it is multi-threaded.

ZServer Command Line Switches

The table below contains a list and description of all the available options that can be passed to ZServer when it is first started. In the next section we will show you how to modify the startup script to use these switches.

ZServer Command Line Switches	
-h	Displays this list of commands
-z *path*	The location of the Zope installation. The default is the location of this script.
-t *n*	Use *n* number of threads if using ZODB3. The default is 4.
-D	Run in Zope debug mode. This causes the Zope process not to detach from the controlling terminal, and is equivalent to supplying the environment variable setting Z_DEBUG_MODE=1
-a *ipaddress*	The IP address to listen on. If this is an empty string (e.g. -a ’’), then all addresses on the machine are used. The default is .
-d ipaddress	IP address of your DNS server. If this is an empty string (e.g. -d ''), then IP addresses will not be logged. If you have DNS service on your local machine then you can set this to 127.0.0.1. The default is: .
-u username or uid number	The username to run ZServer as. You may want to run ZServer as 'nobody' or some other user with limited resources. The only works under Unix, and if ZServer is started by root. The default is: nobody

ZServer Command Line Switches	
-P number	Set the web, ftp and monitor port numbers simultaneously as offsets from the number. The web port number will be number+80. The FTP port number will be number+21. The monitor port number will be number+99.
-w port	The Web server (HTTP) port. This defaults to 8080. The standard port for HTTP services is 80. If this is an empty string (e.g. -w ''), then HTTP is disabled.
-f port	The FTP port. If this is an empty string (e.g. -f ''), then FTP is disabled. The standard port for FTP services is 21. The default is 8021.
-p path	Path to the PCGI resource file. The default value is Zope.cgi, relative to the Zope location. If this is an empty string (-p '') or the file does not exist, then PCGI is disabled.
-F path_or_port	Either a port number (for inet sockets) or a path name (for unix domain sockets) for the FastCGI Server. If the flag and value are not specified then the FastCGI Server is disabled.
-m port	The secure monitor server port. If this is an empty string (-m ''), then the monitor server is disabled. The monitor server allows interactive Python style access to a running ZServer. To access the server see medusa/monitor_client.py or medusa/monitor_client_win32.py. The monitor server password is the same as the Zope super manager password set in the 'access' file. The default is 8099.
-2	Use ZODB 2 (aka BoboPOS) rather than ZODB 3
-l path	Path to the ZServer log file. If this is a relative path then the log file will be written to the 'var' directory. The default is Z2.log.
-r	Run ZServer is read-only mode. ZServer won't write anything to disk. No log files, no pid files, nothing. This means that you can't do a lot of stuff like use PCGI, and zdaemon. ZServer will log hits to STDOUT and zLOG will log to STDERR.

Closer look at starting ZServer

When you install Zope the installation program unpacks the distribution into **C:\Program Files\WebSite** that is assuming that you have not changed the default name or path. There exists in the directory mentioned above a file by the name of **z2.py**. This is a python program that contains the logic necessary to launch ZServer. Don't worry about it's contents, all though it might have some interest for you if you are a developer. The important thing to know about this file is that by running the python interpreter and passing it as an argument along with various command line switches is how you control many of ZServer's settings.

To start ZServer manually using Python

1. Go to the command prompt

- (Windows 95/98) Click **Start->Programs->MS-DOS Prompt**
- (Windows NT) Click **Start->Programs->Command Prompt**

2. Type **`cd "C:\Program Files\WebSite"`** `including the double quotes.`
3. The prompt should now look like this **C:\Program Files\WebSite>** type **bin\python z2.py *-xx* ...** where -xx is one of the options from the chart below.

To start ZServer using the startup batch file

A file by the name of **start.bat** is in **C:\Program Files\WebSite** the contents are listed below. You can edit this file and place additional command line switched after the "-D" from the list above.

```
"C:\Program Files\WebSite\bin\python.exe" "C:\Program Files\WebSite\z2.py" -D %1 %2 %3 %4
%5 %6 %7 %8 %9
```

Note: The above text should all be on one line, not split as it is. Another thing to be aware of is that windows needs the double quotes to surround the names of the files because of the spaces in the file name.

To start ZServer using NT's service manager

If you installed Zope onto NT Server, you were then prompted whether you would like to run Zope as an NT service. If you selected yes, then you can use the NT Service Manager to start and stop ZServer. To use the NT Service Manager:

1. Click on **Start->Settings->Control Panel**
2. Double click on the Services icon in the Control Panel
3. The NT Service Manager window will be displayed on the screen. Scroll down through the list of services and select the one labeled Zope.
4. Push the start or stop button

Running Zope with third party web servers

Before there was ZServer the only way to run Zope was in tandem with a third party web server[1]. This was done using PCGI as described earlier. With the inclusion of ZServer the need to setup Zope with a third party web server is dwindling every day. However there still exist a few scenarios that require the use of a third party server.

- You already have a third party web server running that you can not take down and you can not run ZServer on a different port. Or maybe the existing content is shared and/or modified over a network file system of some sort, or you are behind a firewall that only allows port 80 through.
- You would like to protect all of your Zope content with SSL.
- You would like to use your existing web server to do advanced proxying so that you can do such things as advanced load balancing or use your server to map virtual hosts to different folder in the Zope hierarchy.
- You would like to run Zope on your ISP's machine.
- You are an ISP who would like to run an instance of Zope for each of your customers.

Whatever your reasons the next few sections will help you to get Zope running with the three most common web servers. We cover some of the above mention topic in the advanced sections as well.

The Number One Problem Encountered when running Zope for the First Time

Installing Zope should be as easy or maybe as hard as it is to install a normal CGI script for your respective web server. One caveat is that Zope normally performs both authentication and authorization of users, however some web servers don't pass authentication information to CGI scripts. If you keep getting rejected when you try to access Zope through the web with the super manager user name and password, there is a good chance that your web server is not passing authentication information to Zope. Here we have instructions on how to get three of the most common web servers to pass authentication information to Zope.

Running Zope with IIS 4.0

IIS is only available with Windows NT Server 4.0. Windows NT Server includes a free copy of Internet Information Server 2.0 which is often referred to as IIS. The following instructions assume that you have installed the Microsoft Options Pack which will upgrade your IIS to version 4.0. The option pack is available for free from **http://www.microsoft.com**.

Note the rest of this section has not been completed yet.

1. This actually a small fib, earlier versions of Zope were distributed with an application called ZopeHTTPServer. Although ZopeHTTPServer was ideal for the developer who wanted to quickly setup a Zope environment, it's limited functionality kept most people from using it in a production environment.

IIS + PCGI

IIS + ZServer + PCGI

IIS + ZServer + FastCGI

Running Zope with Netscape

Netscape + PCGI

Netscape + ZServer + PCGI

Netscape + ZServer + FastCGI

Running Zope with Apache

Apache + PCGI

Apache + ZServer + PCGI

Apache + ZServer + PCGI using mod_pcgi

Apache + ZServer + FastCGI

Maintenance and Administration

Changing the Administrator Name and Password

Because Zope is managed through the web, user names and passwords must be used to ensure that only authorized people can make changes to a Zope installation. User names and passwords are normally defined by creating and modifying user folders within Zope. A special *super manager* user name and password is defined outside of Zope for two reasons:

- A user name and password is needed to enable creation of initial normal managers of your Zope site.
- The super manager provides an all-powerful user that can do anything in the application and whose password cannot be changed through the application user interface.

This user name and password is defined in the *access* file located in the Zope directory. It should be readable only by the user as which your web server runs. The super manager username and password should only be used when defining the normal management users and passwords and when dealing with unusual situations, like lost or hacked manager user names and passwords.

The *access* file should consist of a single line of the form *username:password.* The build scripts automatically create an *access* file for you, using a default username of *superuser* and a randomly generated password. For Windows, you will have an opportunity to name this superuser before installation. It is highly recommended that you change the default super manager name and password by editing your *access* file after the installation.

Note that you may also add an optional third component to the line in the *access* file to restrict super manager access by domain. For example, the line *mario:nintendoRules:*.mydomain.com* in your *access* file will only allow super manager access to your installation from **.mydomain.com* machines. Attempts to access the system from other domains will fail, even if the correct superuser name and password are used.

Upgrading Zope

Upgrading your Zope installation is a pretty easy task. Even still, there are a few precautions that you'll want to take, to ensure that any problems that may occur don't become serious problems.

Always, always, always back up your database before doing any kind of modification to Zope. This way you'll always have something to fall back on if everything doesn't work out the way you planned. The file you should be most interested in is called *Data.fs*, and it lives in the *var* directory of your Zope installation. You'll want to copy this file to a safe place. The really paranoid can make a backup copy of their entire Zope installation, but *Data.fs* contains all your data.

I'll assume that by now you have downloaded a new version of Zope. You'll need to uncompress the archive that you downloaded and run the install script if you're using Unix, or run the executable if you're using Windows. You don't want to overwrite your old installation just yet, so make sure it goes into a new directory!

Now that you have your new Zope installation, you need to copy any Products and External Methods that you may have installed over to your new Zope installation. You don't need to worry about having to reinstall the Products at all, you should just be able to copy them from the old *lib/products/* directory to the new one. Make sure you don't copy any Products that are already included with Zope, as you may overwrite a new one with an older one! Don't worry about installing the External Methods within Zope, just copy the module files over to the *Extensions* directory. When we move the object database over, all your External Methods will be there.

Now, shut down your old site. We need to copy the data file over and it just wouldn't do to have people adding new stuff while we're copying it over, so if you're using PCGI it would be a good idea to shut down your web server also. The data file, as we said earlier, is *Data.fs* and is located in the *var* directory of your Zope installation. Copy it over to the new installation.

If you're using PCGI, move the old Zope.cgi out of the directory it's in, and copy the new one that's in your new Zope installation directory over. Make sure the environment settings match the ones from your old Zope.cgi. If you're using PCGI and you stopped your web server, you should restart it now.

Otherwise, start Zope the way you normally do. Verify that it's all working by visiting the site and the management interface. Make sure that all the Products are installed and functional by going to the Control Panel and selecting Product Management.

If there are any problems, refer to the Troubleshooting section of this document.

Compiling Zope from Source

The first thing we need to do is pick a directory to install Zope to. The directory you pick shouldn't be accessible to your current web server (if any). For this example we're going to use */usr/local/*. By now you should have downloaded a compressed archive of the Zope distribution. For purposes of this example we're going to use *Zope-2.0.0-src.tgz* which we downloaded to the */usr/downloads/* directory.

We're going to change to the directory we want to install Zope under using *cd /usr/local*. Once there, we need to extract Zope from the compressed archive. We'll use the command *tar xfz /usr/downloads/Zope-2.0.0-src.tgz*. Depending on which directory you pick, you may need to become root to do this. When the file uncompresses, it will create a directory called *Zope-2.0.0-src*. This is the *Zope installation directory.* If you want you can rename it, but you should do it now because the build scripts use the Zope installation directory during configuration.

Now that we're done extracting the Zope source from the archive, we have to go to the Zope installation directory using *cd Zope-2.0.0-src*.

If you're planning on running Zope with your existing web server, then you need to build Zope with PCGI support. To do this, you need to run the command *python w_pcgi.py* to build Zope. If you're just planning on using the included ZServer, then use the command *python wo_pcgi.py.*

When the build script is finished, it will tell you that it has created a super manager called superuser with a random password. Make a note of this, as you will need it to manage your Zope installation. You can also edit the *access* file in the Zope installation directory and change the username and password if you wish. We will go into more detail about the access file in the next section.

Extending Zope

Zope was designed from the beginning to be extensible, to allow added functionality to be included. There are two ways to add new functionality to **Zope:** ***Products*** **and** ***External Methods***. *External Methods* are simply functions, written in Python, that have been imported into the Object Database so they can be accessed in Zope. A *Product* is a drop in object that can extend Zope's functionality.

Products

Products are "plug-ins" for Zope. There are a growing number of Products available for download from *http://www.zope.org/* as well as a tutorial for the Product API, if you're interested in creating your own Products. In the Zope Content Manager's Guide we create a simple Product for creating a web site. you can refer to that as well.

Installing New Products

The procedure for installing new Products should be exactly the same on both Unix and Windows, with a few syntactic differences.

First of all, download the Product you want to install. Typically, you will receive a compressed archive with the name and version number of the Product. Let's say for example that we have just downloaded *ZLDAPConnection.tgz* which is the Zope LDAP Connection Product and have placed it, under Unix, in */usr/downloads/* or, under Windows, in *C:\Downloads*.

Most Products, if they have been constructed right, will need to be uncompressed into your Zope installation directory. Under Unix, we would have to change to the Zope directory: *cd /usr/local/Zope-2.0.0*. From there, we would do *tar xvfz /usr/downloads/ZLDAPConnection.tgz* which will extract the Product and place it in the directory *lib/python/Products/ZLDAPConnection/*.

Under Windows, you would have to use a program (WinZip) that understands the TAR and gzip compression schemes. A good place to look for these is *http://www.winfiles.com*. Instruct your program to extract the files, using full path information, to your Zope installation directory.

Check the directory that the Product was installed under for any README or INSTALL files. If any exist, we should read them thoroughly and follow any other installation instructions that they may have.

After this, your Product should be ready to go! We just need to restart our Zope server by going into the Zope Control Panel and clicking the Shutdown button. Start your Zope server back up the way you normally would.

Once we are back in Zope's management interface, we want to verify that we can use the Product. All we have to do in this case, is click the list of items to add and verify that ZLDAPConnection is listed in there.

Product Troubleshooting

If the Product does not show up in the *add list,* we can get more information as to what the problem might be. Under the *Control Panel*, there exists a special folder called *Product Management*. This folder has a list of all the Products currently installed on your system. If a Product is broken, it will say so very clearly next to the name of the product, as well as display an icon of a broken box. Clicking the Product will take you to a page that has a link to the *Traceback*, which will tell you what error the Product is giving.

For more information on troubleshooting and getting help, see the global Troubleshooting section.

External Methods

External Methods are significantly easier to install and troubleshoot than Products, if you're familiar with Python that is. An *External Method* is simply a Python function that resides within a *.py* file.

Let's say, for example, that you have a file called *spam.py* with the following contents:

```
def spamandeggs():
  return "Spam and Eggs!"
```

If you wanted to call this function from within Zope, you would need to set up an *External Method*.

Installing New External Methods

The first thing we need to do is put the file, *spam.py*, in a directory for Zope to be able to read it. Copy the file to the *Extensions* directory in the Zope installation directory. If you don't have an *Extensions* directory, then you should go ahead and create it using *md Extensions* from the Zope installation directory.

Now that we have the file in the right spot, we need to import it into Zope. Point your web browser to the management screen of any Zope folder. From the list of item types to add, select External Method.

We are now at a screen that's asking for information on the External Method we want to create. We need to enter in an id for it in the object database, so let's call it *breakfast*. Let's set the title to *Spam and Eggs!* just so we don't forget what this External Method does. The function name, if you remember from the contents of our file, is *spamandeggs*, so we type it in just like that. The Python module file is the name our file, which is *spam.py.*

Click the *Add* button, and voila! We have an External Method! Since this is External Method just returns information and doesn't take any arguments, we could call it from a DTML Document or Method by using *<dtml-var breakfast>*.

External Methods' Troubleshooting

You should test the function in Python outside of Zope to make sure that it works as expected before trying to import it into Zope. Zope will usually be able to catch syntax errors when importing, and will notify you when one occurs. If there is an error while executing the External Method, you will get a standard error message. Typically, if you look at the source code for the page with the error message it will give you more information.

For more information on troubleshooting and getting help, see the global Troubleshooting section.

Advanced Topics

Running Zope with an ISP

TBD

Virtual Hosting

TBD.

Launching Multiple Instances of Zope

TBD

Troubleshooting

Common Problems

TBD

Directory Permissions

TBD

Can’t login

TBD

Understanding Tracebacks

Every now and then, you’ll have a problem with your installation of Zope. Maybe you installed a Product and it doesn’t work, or maybe there’s some DTML that’s giving you grief. Whatever the reason, errors do happen. Whenever Zope gives you an error, it will spit out an error page with a *Traceback* that is meant to help you identify the problem.

A traceback will look something like this:

```
Traceback (innermost last):
  File /usr/local/Zope-1.10.2-src/lib/python/ZPublisher/Publish.py, line 877, in
publish_module
  File /usr/local/Zope-1.10.2-src/lib/python/ZPublisher/Publish.py, line 590, in
publish
    (Info: /invisible)
  File /usr/local/Zope-1.10.2-src/lib/python/OFS/DTMLMethod.py, line 155, in
__call__
    (Object: invisible)
  File /usr/local/Zope-1.10.2-src/lib/python/OFS/DTMLMethod.py, line 151, in
__call__
    (Object: invisible)
  File /usr/local/Zope-1.10.2-src/lib/python/DocumentTemplate/DT_String.py, line
513, in __call__
    (Object: invisible)
KeyError: (see above)
```

What this means, beginning at the top, is that the function *publish_module* in *Publish.py* called the function *publish* in *Publish.py*, which called the function *__call__* in *DTMLMethod.py*, which called the function *__call__* in *DTMLMethod.py*, which called the function *__call__* in *DT_String.py* where an error of type *KeyError* occurred. All this for the object called *invisible*, whose path is */invisble*.

If you view the source to this document you will see, above the traceback, a comment that looks something like this:

```
<!--
 Error type:
 Error value: output_data
 -->
```

We already know that the error being raised is a KeyError, and now we know that the error is being raised over *output_data*, which means that the variable *output_data* doesn't exist.

For this example I used a DTML Method called *invisible* that looks like this:

```
<!--#var standard_html_header-->
<!--#var output_data-->
<!--#var standard_html_footer-->
```

Since I don't have an object named *output_data*, Zope displays a KeyError when trying to render this page.

Where to Go for Support

If you can't figure out a problem, help is always only an email away. Zope currently has two mailing lists for getting help or discussing Zope-related issues. The *zope@zope.org* list provides help for end-users, while the *zope-dev@zope.org* list provides a place for people programming Zope to ask for help and discuss problems.

Both lists have a searchable archive at *http://www.zope.org/Support/SearchList*. You can also subscribe to either list by going to *http://www.zope.org/Community/*.

Reporting Bugs

On the rare occasion that you might encounter a bug in Zope, Digital Creations has set up a Collector to facilitate the tracking and reporting of bugs. It features a completely searchable interface, so you can find out if the bug you are reporting has been reported yet and if there is a solution available to it. Entries in the Collector are updated as information regarding reported bugs is found. The Collector can be found at *http://www.zope.org/Collector/*.

GLOSSARY

Soon to come:

Zope

PCGI

ZServer

FTP

HTTP

IIS

ZODB

DA

Extensions

ZTables

ZClass

Tracker or Collecter

etc...

Index