So far we have been using Pharo’s ‘Seaside One-Click Experience’ to learn Seaside and Smalltalk. Pharo is just one of several dialects of Smalltalk supporting Seaside, each with a unique focus. One dialect of Smalltalk, GemStone/S, has been available from GemStone Systems, Inc. since 1985 and provides a distinctive emphasis on server-side scaling and persistence. That is, while other Smalltalk dialects provide a system in which a single virtual machine interacts with an object space that fits in RAM (generally up to a few hundred megabytes on modern hardware), GemStone provides a system in which thousands of virtual machines on multiple machines can interact—using database semantics—with an object space that can be terabytes in size.

For its first couple of decades, GemStone’s traditional customers have been large enterprises that do internal development of mission-critical applications using Smalltalk. Typically, they use Cincom’s VisualWorks (formerly from ParcPlace) or Instantiation’s VA Smalltalk (formerly VisualAge Smalltalk from IBM) to build applications that run on employee workstations (the client) and connect to a GemStone/S database (the server). By using Smalltalk on the client they are able to rapidly develop complex applications using pure objects and the powerful libraries provided by their dialect’s vendors. By using Smalltalk on the server, they are able to eliminate the complexity and impedance mismatch of object-relational mapping. In addition, selected back-end processing can be transferred from the client to the server reducing network overhead and allowing more powerful servers to be used.

As the Internet became increasingly popular, libraries and frameworks supporting web application were added to various Smalltalk dialects (including VisualWorks and VisualAge). None of these, however, achieved the traction that Seaside realized and maintainers of other dialects have generally abandoned their proprietary approaches and are focused on supporting Seaside. Seaside originated in Squeak (<http://www.squeak.org/>) and is now being maintained primarily in Pharo (<http://www.pharo-project.org/>). At this point there are ports of Seaside to Cincom Smalltalk from Cincom ([http://www.cincomsmalltalk.com/](http://www.cincomsmalltalk.com/userblogs/cincom/blogView)), VA Smalltalk from Instantiations (<http://instantiations.com/VAST/index.html>), Dolphin Smalltalk from Object Arts (<http://www.object-arts.com/>), GNU Smalltalk (<http://smalltalk.gnu.org/>), and, of course, GemStone Smalltalk from VMware (<http://seaside.gemstone.com/>).

While each dialect brings unique strengths (and weaknesses) to Seaside, we will be focusing on GemStone/S in this (and subsequent) chapters. Because of GemStone’s background as a server Smalltalk, it is uniquely positioned to provide scalability (in both objects and number of users) to a Seaside application. The trade-off is that a GemStone installation tends to be larger (thus more complex to use and manage) does not have a native suite of GUI tools (so relies on other Smalltalk dialects for development tools such as browsers), and the Seaside-enabled version only runs on 64-bit \*nix operating systems (including AIX, HPUX, Linux, Macintosh, and Solaris, but not Microsoft’s Windows).

GemStone has a reputation in the Smalltalk community of being very desirable (who would want to use SQL when they could use Smalltalk?) but also very expensive. Fortunately, the cost issue has been addressed in part with a no-cost (‘free as in beer’) license of the product for Linux with the primary limitation of 2 CPUs and a 2 gigabyte shared page cache (see <http://seaside.gemstone.com/docs/GLASS-Announcement.htm> for details). A version is also available for Macintosh.

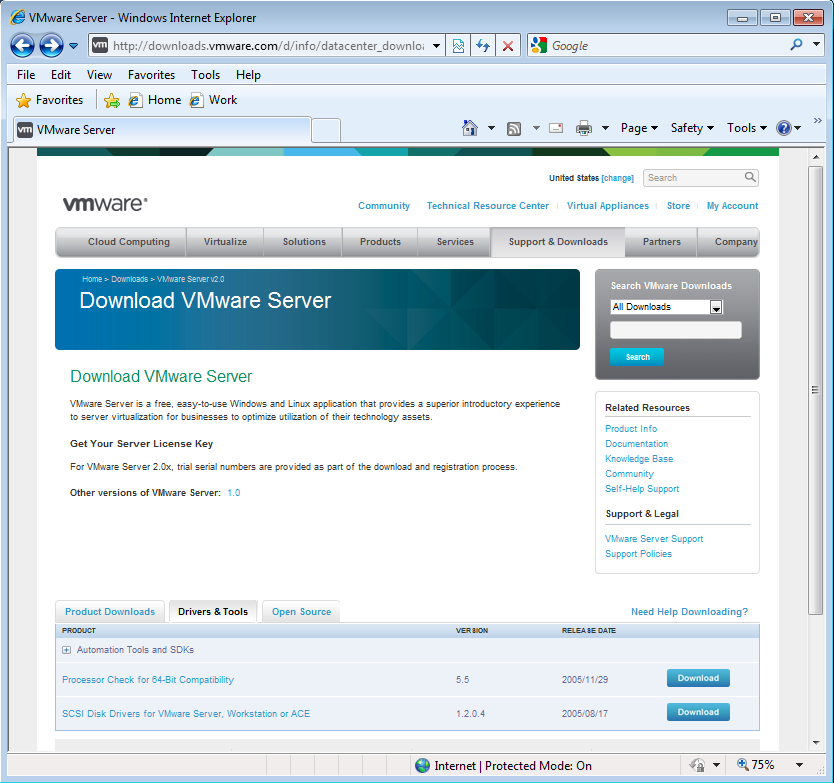
While getting started with GemStone is not quite as easy as using Pharo’s ‘Seaside One-Click Experience,’ it is not nearly as difficult as, say, an Oracle install. There are two options for installing GemStone/S 64 Bit: (1) native, and (2) VMware. In this chapter we focus on the VMware approach.

**VMware Virtual Appliance**. This is the approach most likely to get everything working quickly, but requires 64-bit hardware with virtualization support and an installed copy of VMware Server (free for Linux and Windows) or VMware Fusion (for Macintosh) along with a license key. The Virtual Appliance has a full install of 64-bit Linux, Apache, GemStone/S 64 Bit, and other components (such as FastCGI).

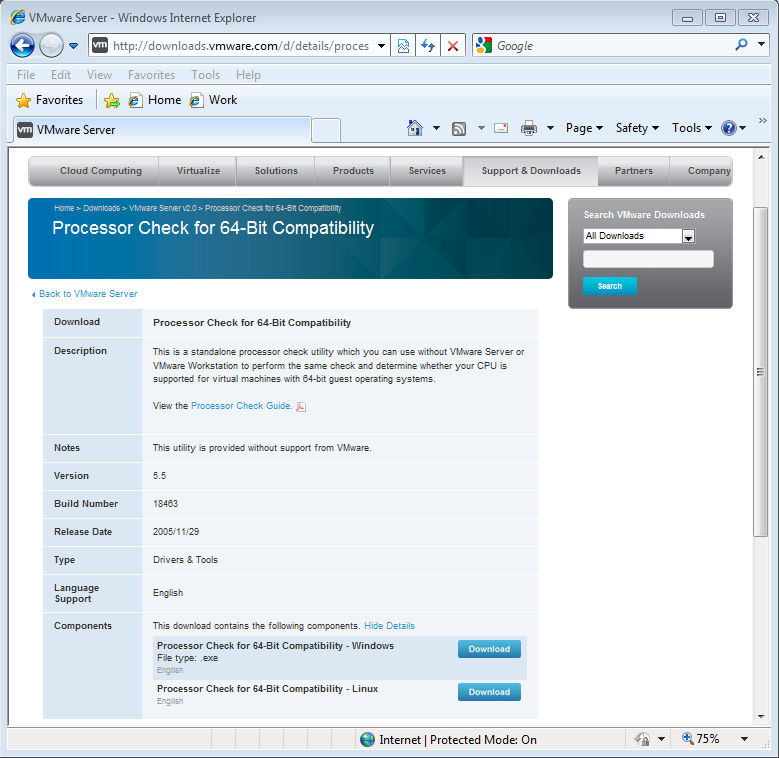
The first step is to see if your hardware supports 64-bit virtualization. If you are running a recent MacBook, then you are fine. If you are running Windows or Linux, then you need to check your machine. Open a web browser on the following link:

<http://www.vmware.com/download/server/drivers_tools.html>

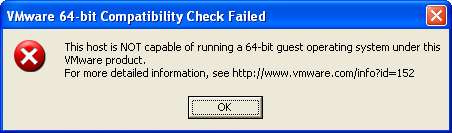
Click the link 'Processor Check for 64-Bit Compatibility' in the Drivers & Tools tab.



This will take you to a download page where you can select the download for either Windows or Linux.

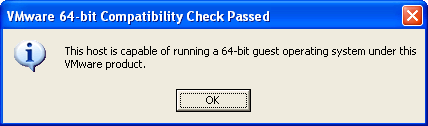


After downloading the guest check application run it to see if your hardware supports virtualization. If the answer is ‘no,’ then you need to pursue another approach (run GemStone/S native if you are on a 64-bit Linux OS or get access to another computer).

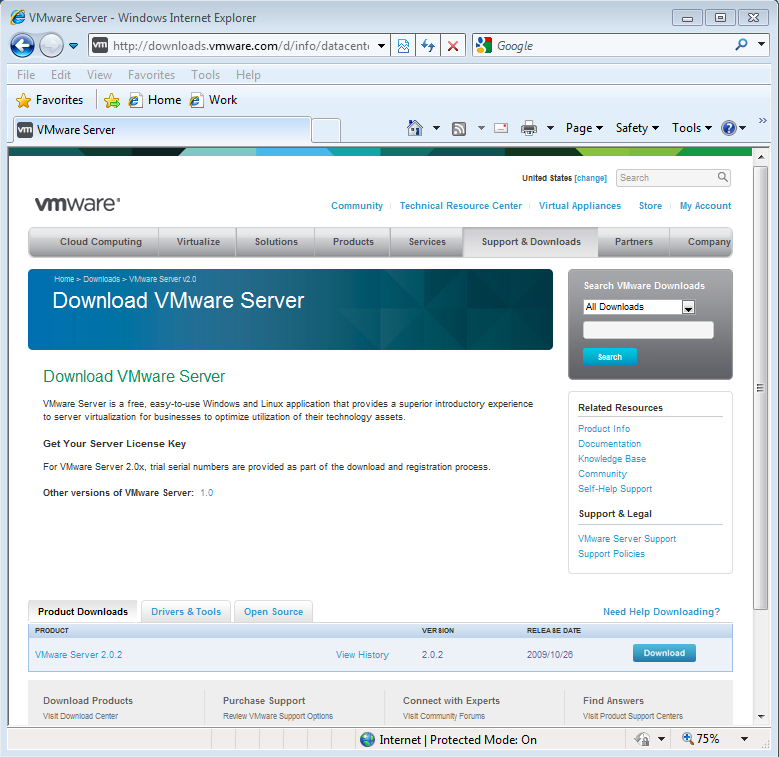


If your processor supports 64-bit virtualization, then you need to see if it is enabled in your computer’s BIOS. Reboot your PC and press <F2> during the boot process to get into the BIOS setup. Look for a setting named something like ‘Enable VT,’ see that it is true/yes, save the settings, and then reboot the machine. Unfortunately, we are unable to give more detailed instructions because the BIOS settings differ depending on your machine. See <http://kb.vmware.com/selfservice/viewContent.do?externalId=1003944&sliceId=1> for some instructions from VMware.

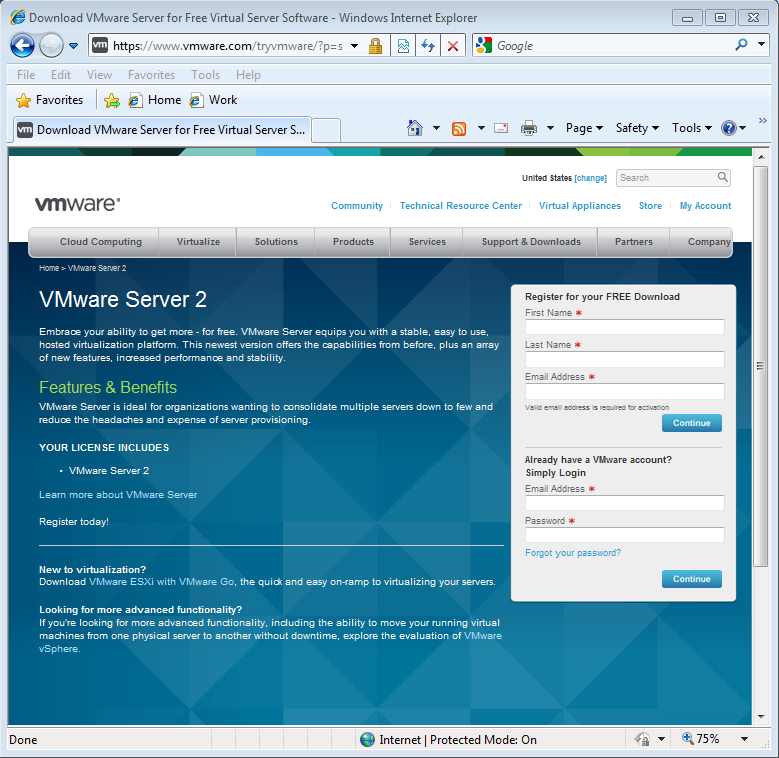
If things work properly, then you should see the following dialog.



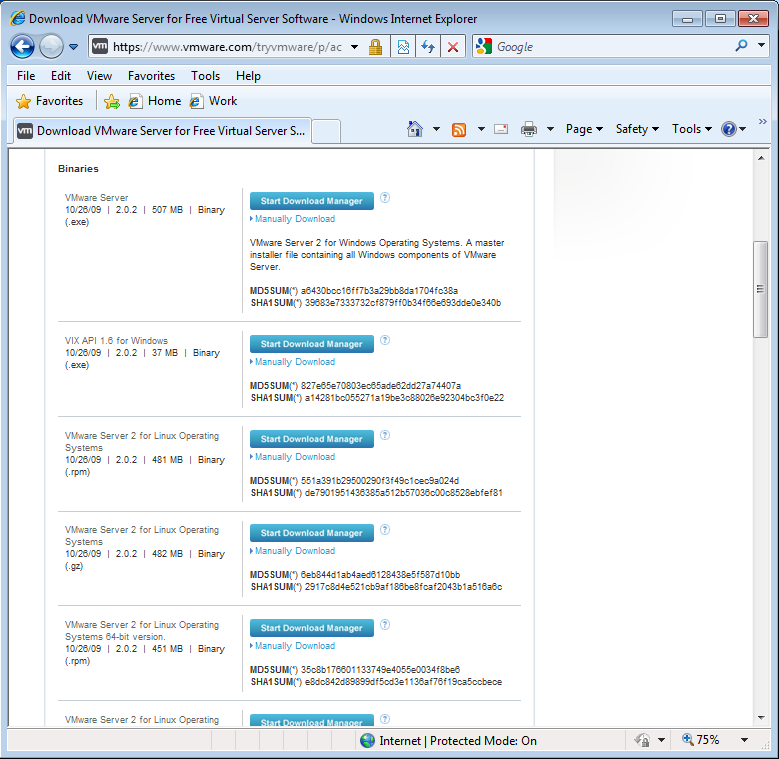
Next, download VMware Server 2.0 from the previous web site by selecting the 'Product Downloads' tab (instead of the 'Drivers U Tools' tab).



During this download process you will be required to register and a registration key will be sent to an email address you provide.



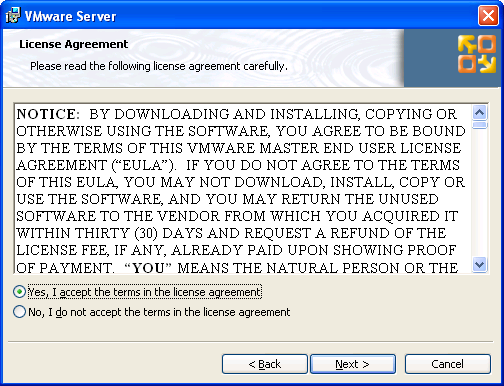
Once you complete the registration process (including a license page) you will be presented with a download page. Download the version of VMware Server 2 appropriate for your operating system. We will demonstrate the process of installing on 32-bit Windows XP.



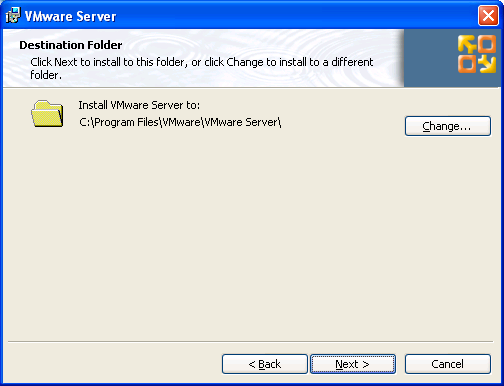
Once you download the installer, launch it.



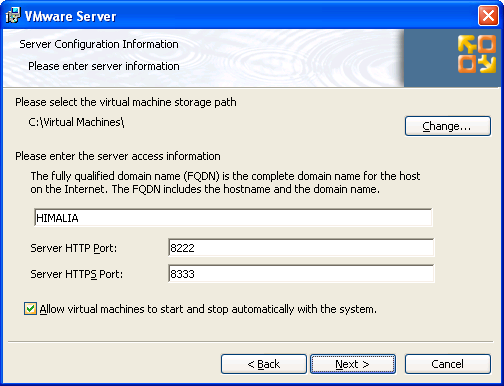
After reading the license (you do that don’t you?), click ‘yes’ (if you agree) and then ‘Next.’



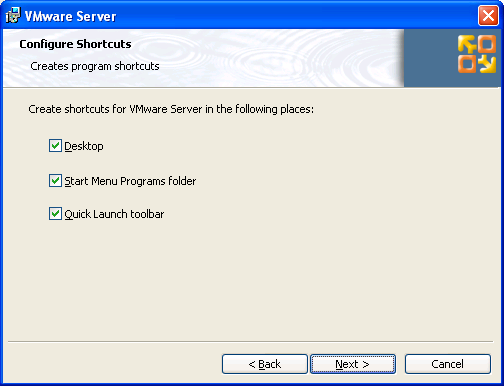
Select a location in which to install the software. By default it will go in Program Files.



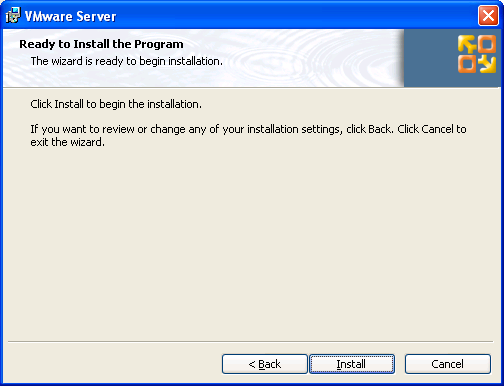
Unless you have a good reason, the server configuration information should be left as is.



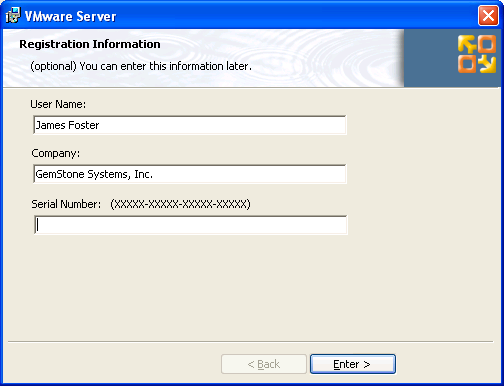
Decide whether to allow shortcuts to be included in various places.



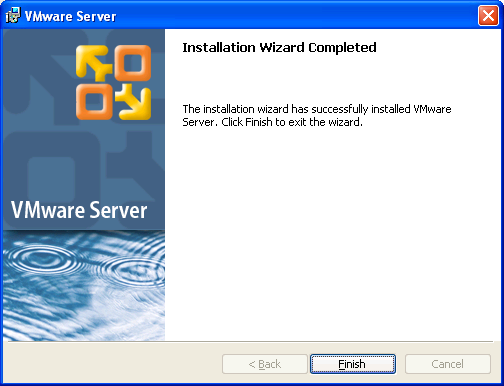
When the installer has all the setup information it needs, it can start.



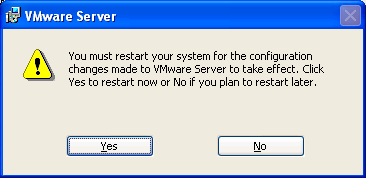
When you registered for the download you gave an email address. Check that mailbox for a key.



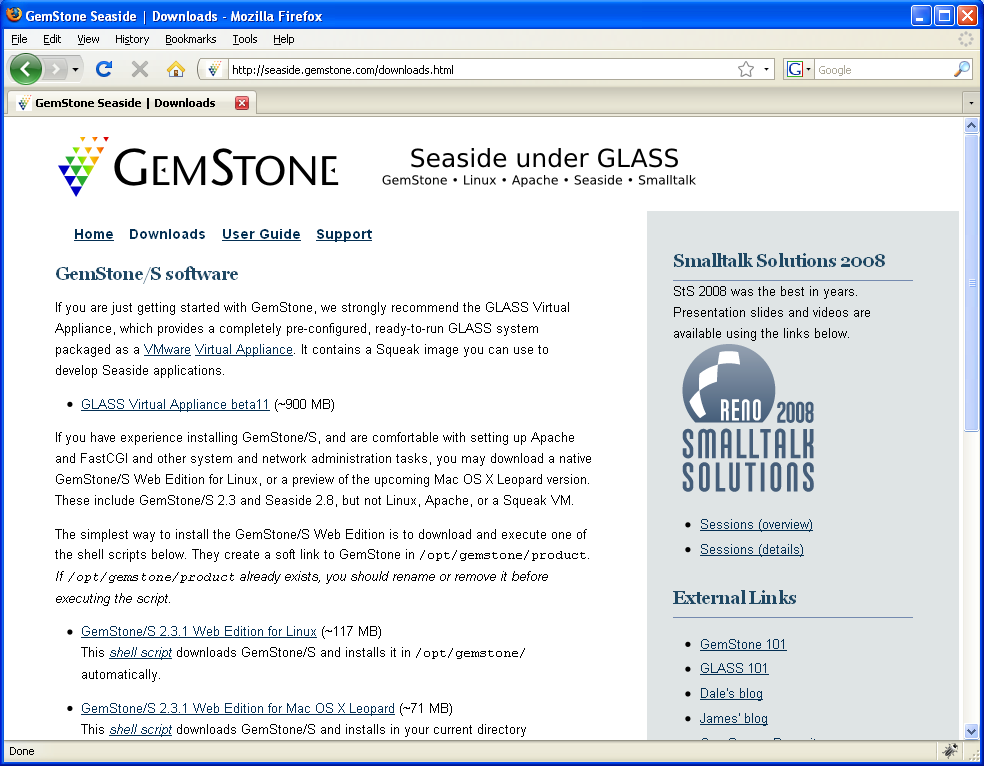
When the installation is done click ‘Finish.’



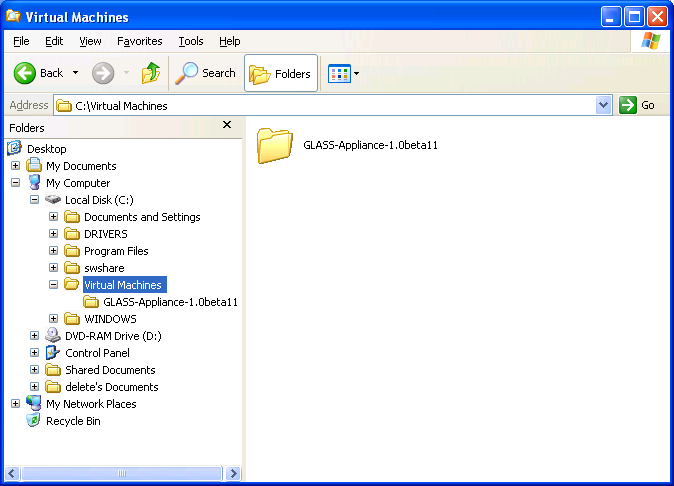
Like many other applications, an installation requires a restart.



Using a web browser, navigate to GemStone’s download page (<http://seaside.gemstone.com/downloads.html>) and download the GLASS Virtual Appliance.



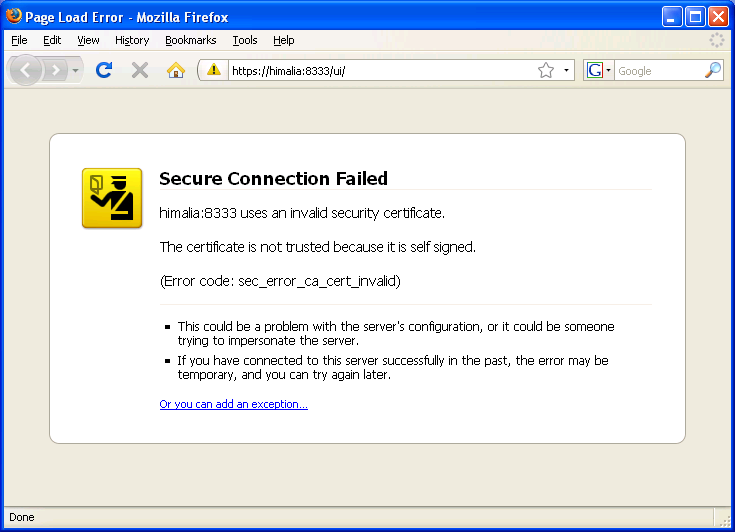
After the file downloads, unzip it and place it in ‘C:\Virtual Machines’ for VMware to find.



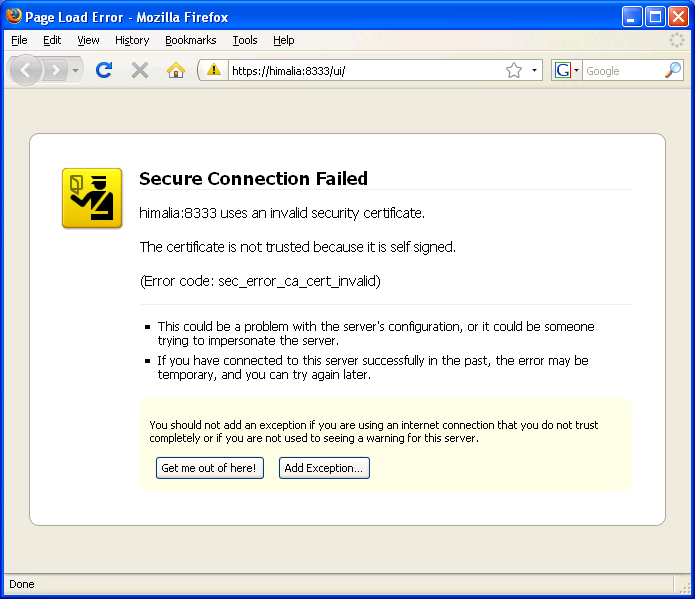
Now launch VMware Server using one of the shortcuts provided by the installer. This will open a web browser and, depending on your security settings, will report a problem.



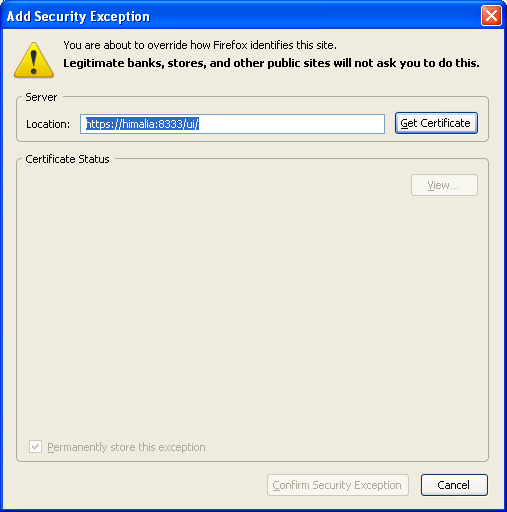
Click OK and the browser will show an error. Click on the ‘Or you can add an exception…’ link.



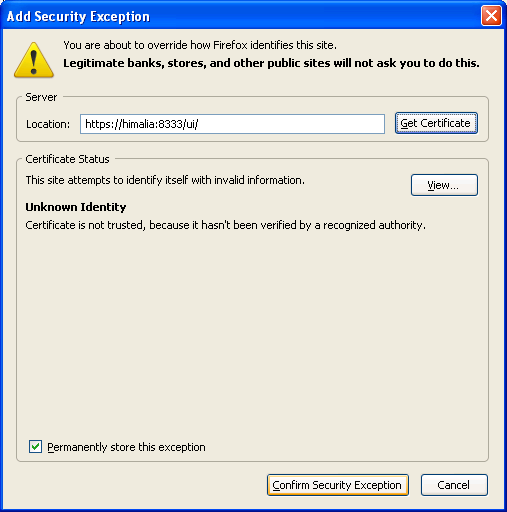
After you click the link you will be shown two buttons. Click the ‘Add Exception…’ button.



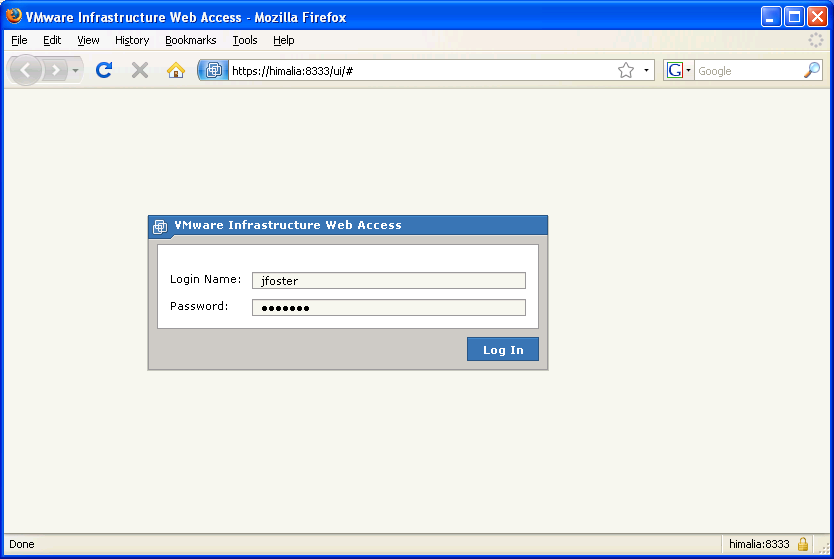
Firefox requires you to get the certificate before you can confirm the exception.



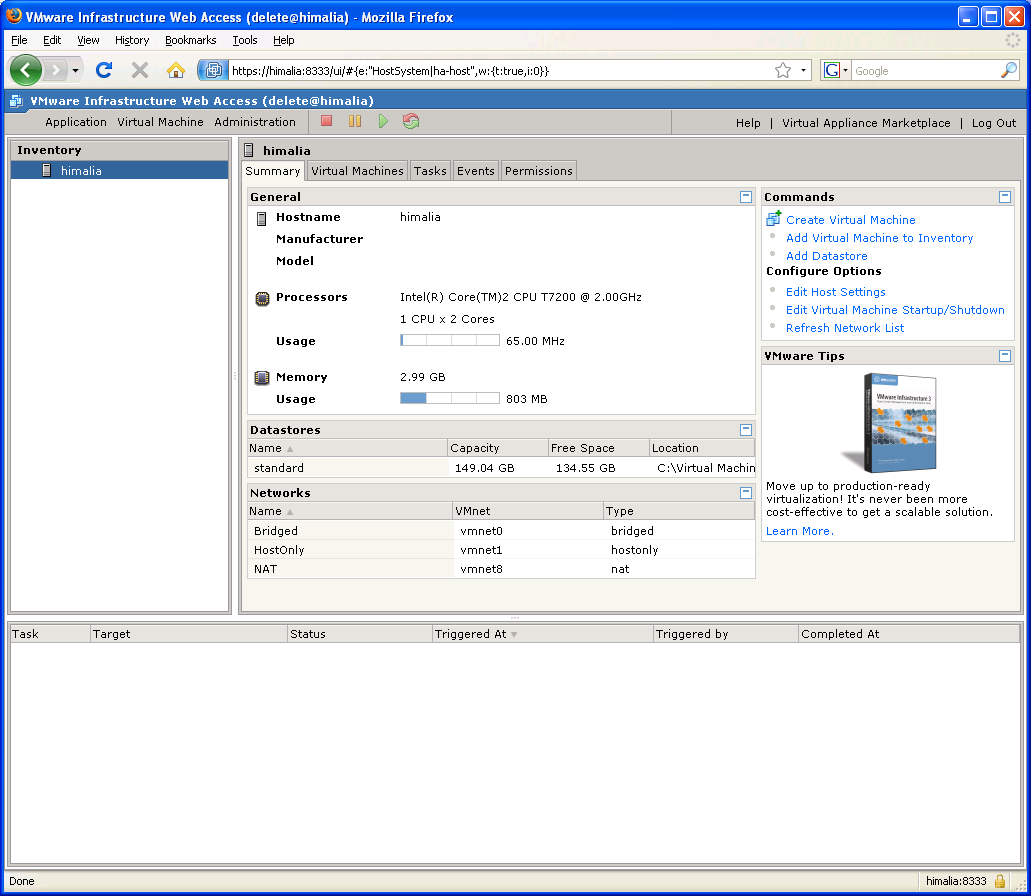
Once you get the certificate, click the ‘Confirm Security Exception’ button.



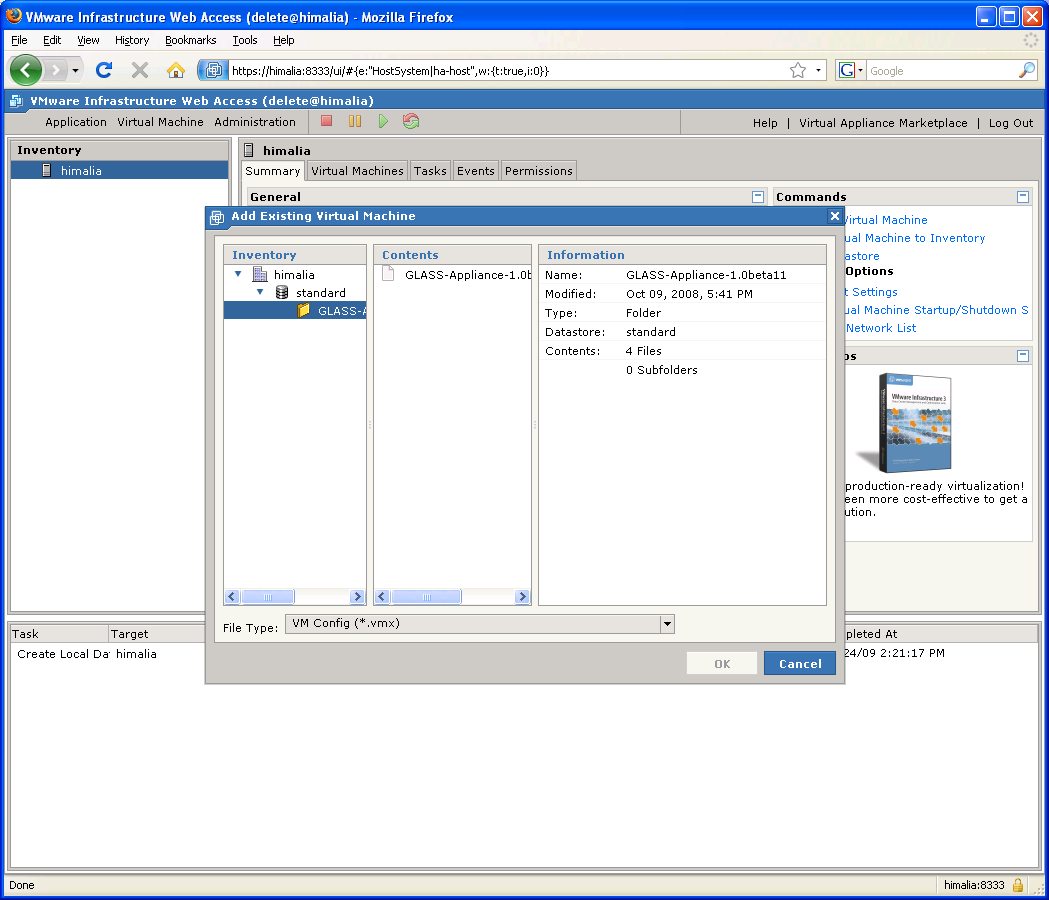
Once you confirm the security exception, you should get a web browser with a login screen asking for you host operating system user and password. Note that VMware requires a password. If your Windows account does not have a password you will need to add one and then give VMware your Windows password. Note that you could access this screen from another computer on the network using the same URL.



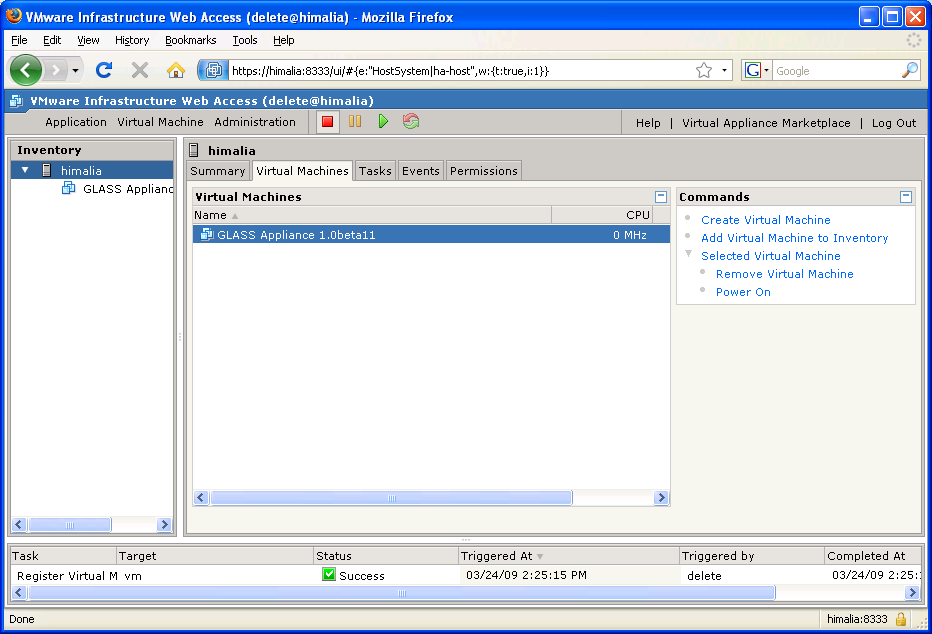
Once you have a successful login, you should see the VMware Infrastructure Web Access. Click on the command link ‘Add Virtual Machine to Inventory.’



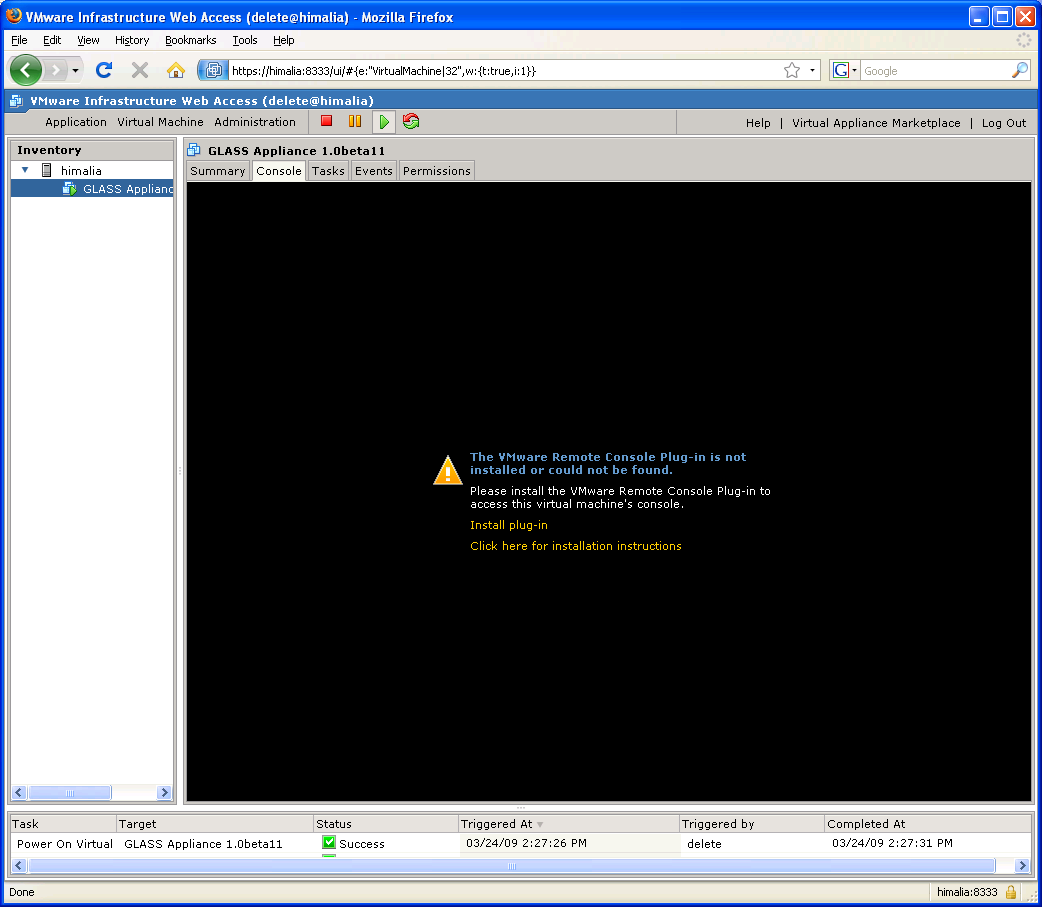
This will present a dialog. In the first column (Inventory), expand the machine, the inventory, and select the GLASS directory. In the second column (Contents) select the GLASS-Appliance. Finally, click the ‘OK’ button.



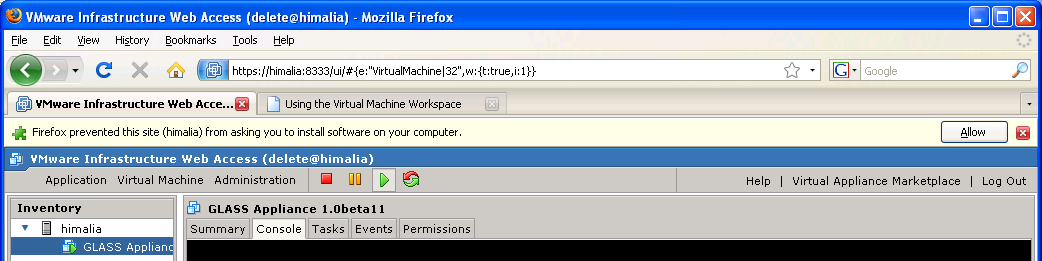
Once you have added the GLASS Appliance it should show up in the list of virtual machines.



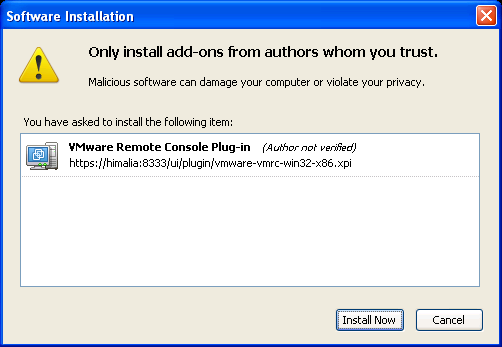
Click on ‘GLASS Appliance’ in the Inventory list (the first column). This will change the area to the right showing a summary and other tabs about the appliance. Click the green start button and then switch to the ‘Console’ tab. This will inform you that the VMware Remote Console Plug-in is not installed. Click on the ‘Install plug-in’ link.



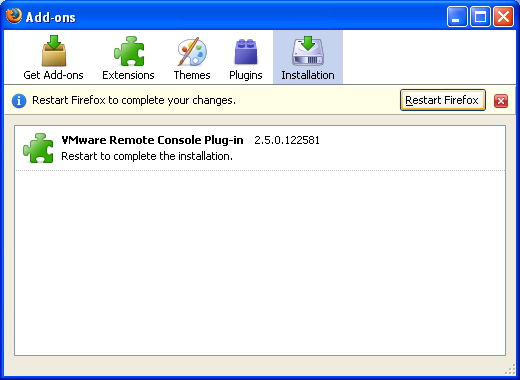
The plug-in installation might prompt you to allow the installation. Click the ‘Allow’ button.



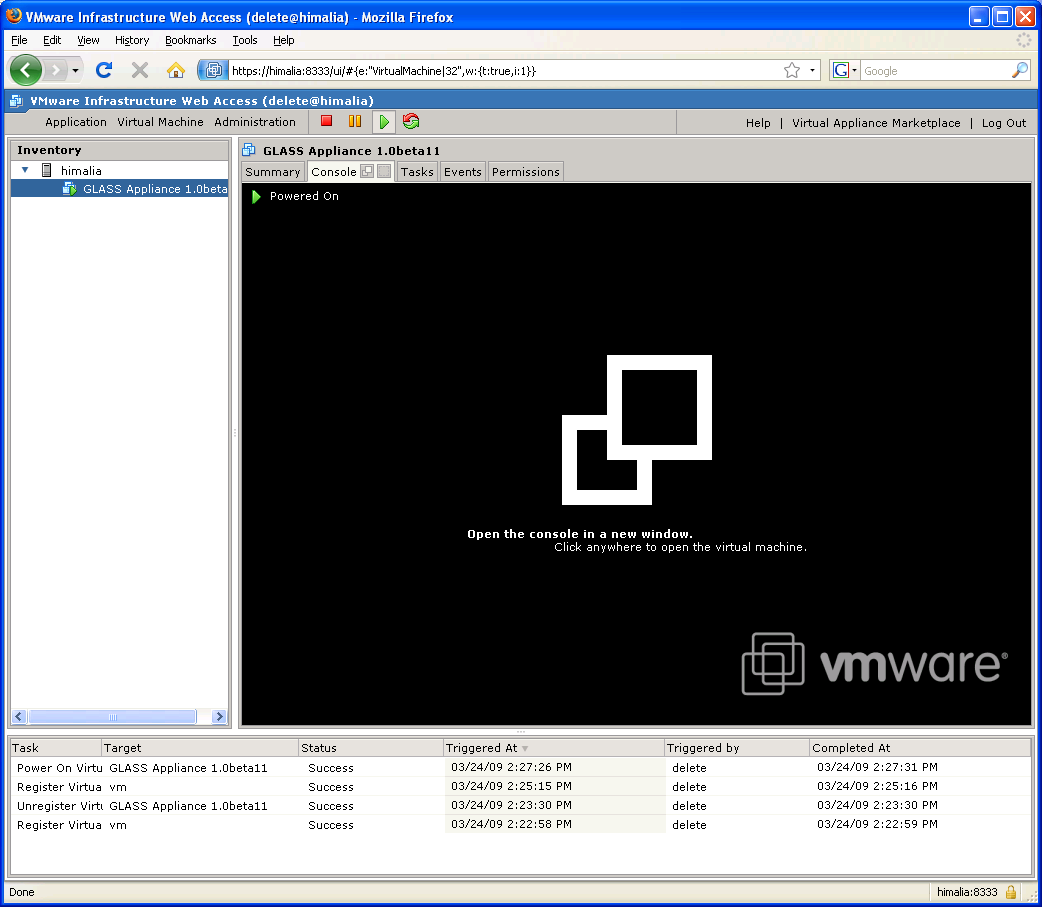
You might be prompted to confirm the installation.



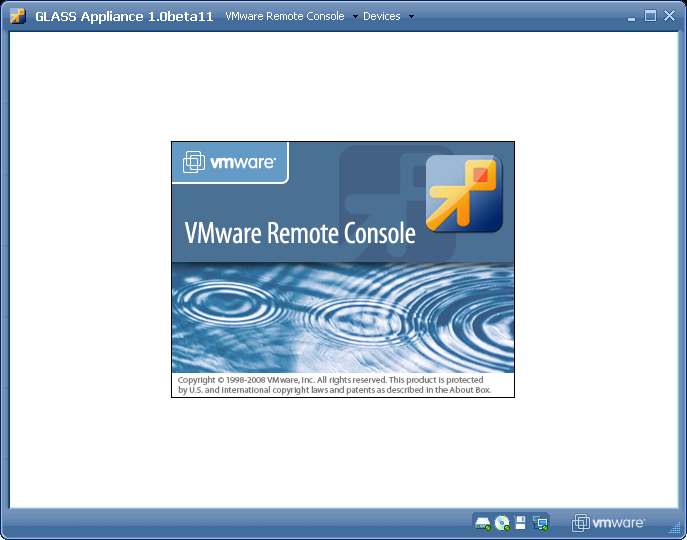
You might need to restart your browser.



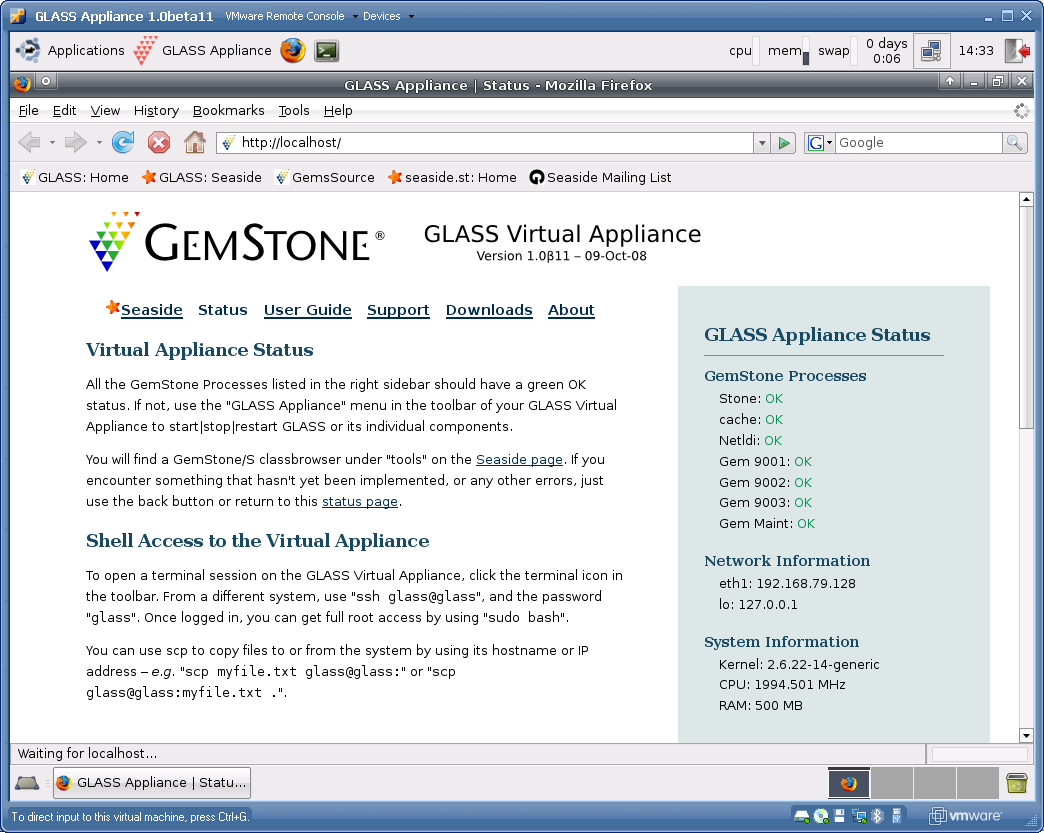
If the browser does not reopen to the same place, restart the VMware Infrastructure Web Access, select the GLASS Appliance in the Inventory list, select the Console tab, and click anywhere in the black area to open the virtual machine.



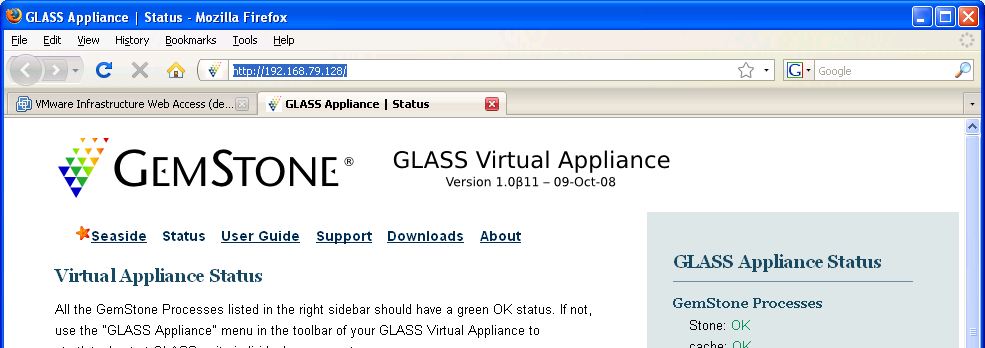
This will open a new window with a splash screen while the console is connecting to the virtual machine.



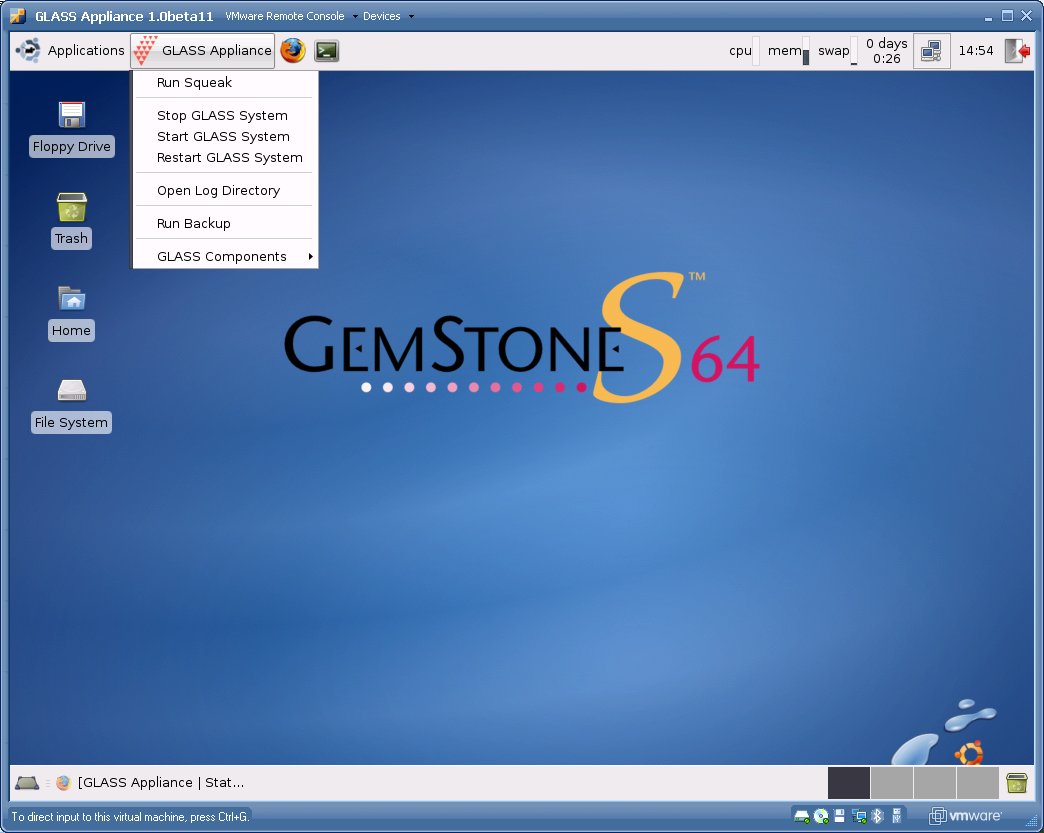
Once the console connects to the virtual machine, you should have a window showing what you would see if the GLASS Virtual Appliance were running on a real machine. In this case we have Linux with a GUI and a Firefox browser launched looking at <http://localhost/>. This is host is not your Windows machine, but a ‘new’ (‘virtual’) machine running in VMware that has its own operating system (Ubuntu Linux), its own web server (Apache), and a GemStone Smalltalk system already configured to start when the operating system starts. This machine has its own IP address (in my case it is 192.168.79.128, but it might be different for you). Note the series of green ‘OK’s for the seven GemStone processes.



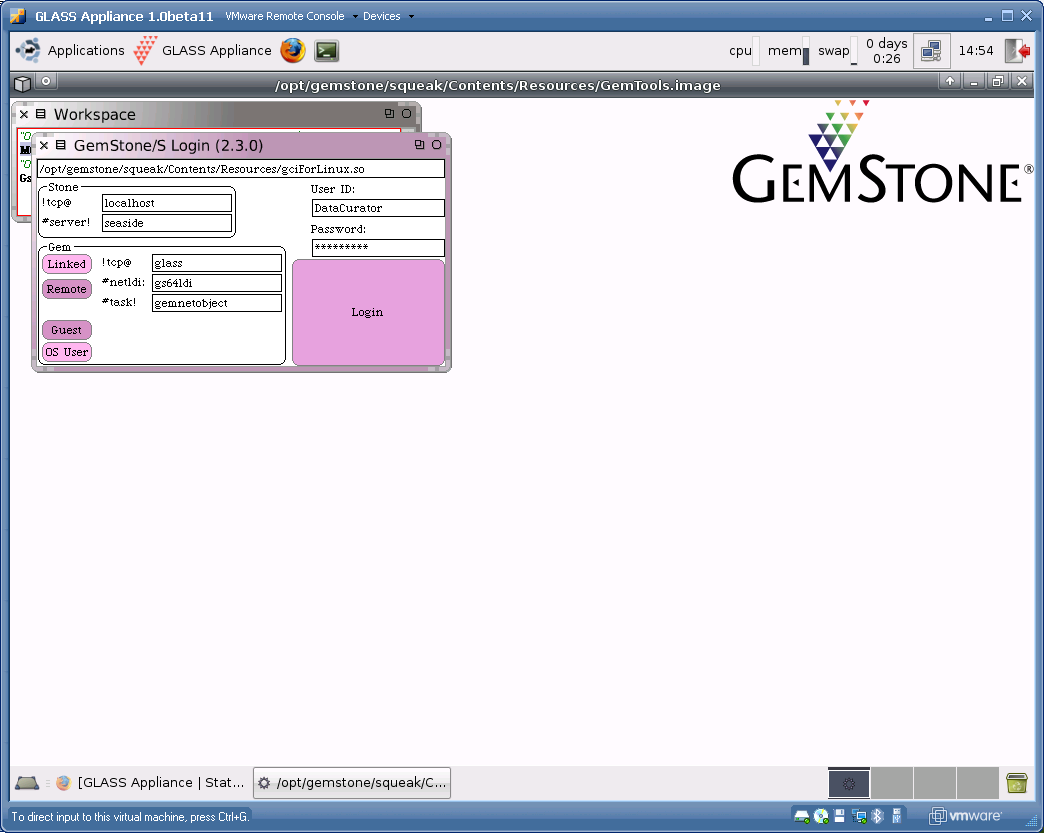
Copy the IP address out of the GLASS Appliance and past it into a web browser outside the virtual machine. Note that the web server in the virtual machine now is serving a page to you outside the virtual machine. You can start to think of the virtual machine as a host running in your company’s server room that you access over the web using TCP/IP.



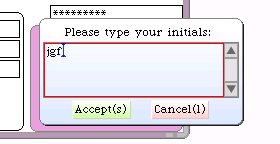
Returning to the virtual machine console, you can minimize Firefox and see the Linux desktop with menus. Select ‘Run Pharo’ from the ‘GLASS Appliance’ menu to start the GemTools application.



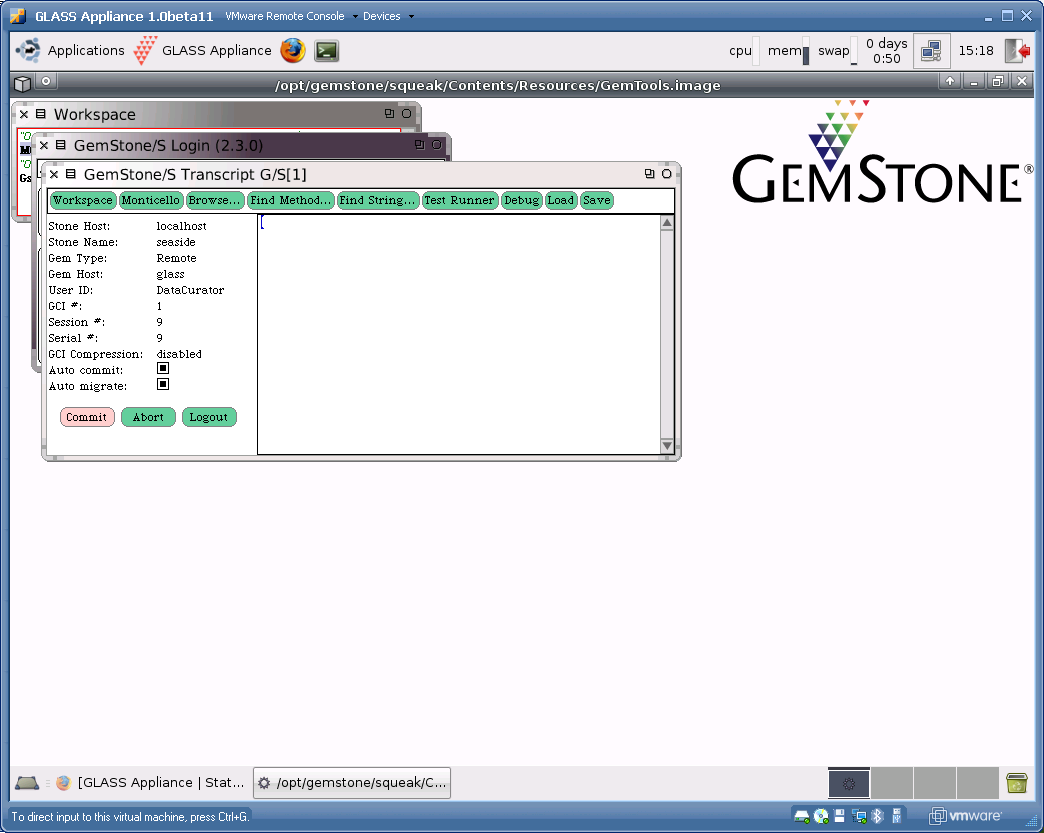
The GemTools application that ships with GLASS-Appliance-1.0beta11 is a Pharo application that fills the screen. Click the ‘Login’ button



When prompted, type your initials (this is used by the Monticello code management tools to identify authors of methods). After typing your initials (by convention, in lowercase), click the ‘Accept(s)’ button.



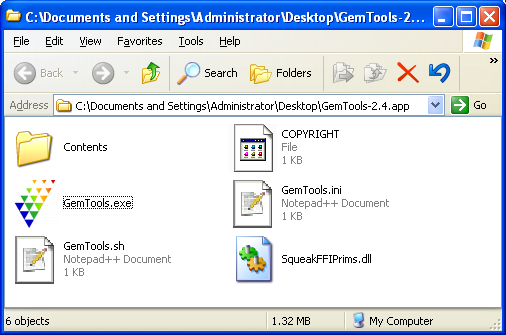
After you login, you should see a ‘Transcript’ window. We will do most of our interaction with the system using GemTools running outside the appliance, so go ahead and click the ‘Logout’ button on the Transcript.



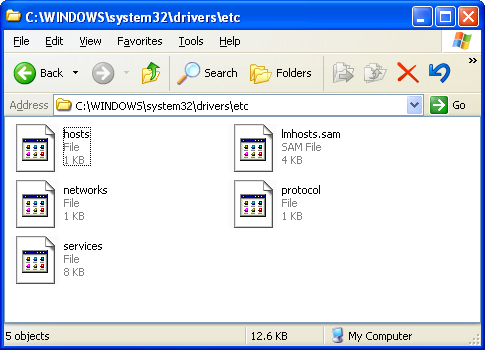
The tools have been substantially updated since the version included with the appliance. To get the latest tools, navigate to the GemStone downloads page where you got the appliance and scroll down to the bottom. Click the ‘GemTools 2.3.0 Preview for all platforms’ link to download a zip file containing the tools package. (This version matches the server executables in the virtual appliance.)



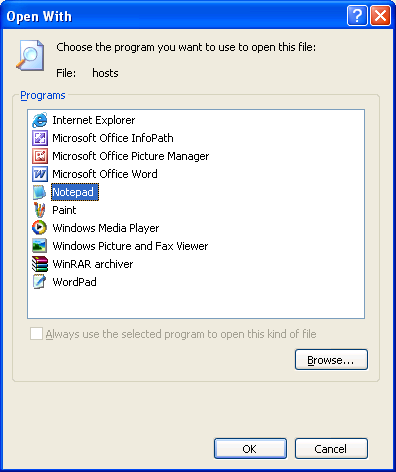
When you unzip this package it will consist of a folder (or directory) containing various pieces. On the Macintosh, the entire folder will appear as an executable package. Double-click the icon to launch the tools. Otherwise, open the folder and launch ‘GemTools.exe’ on Window or ‘GemTools.sh’ on Linux.



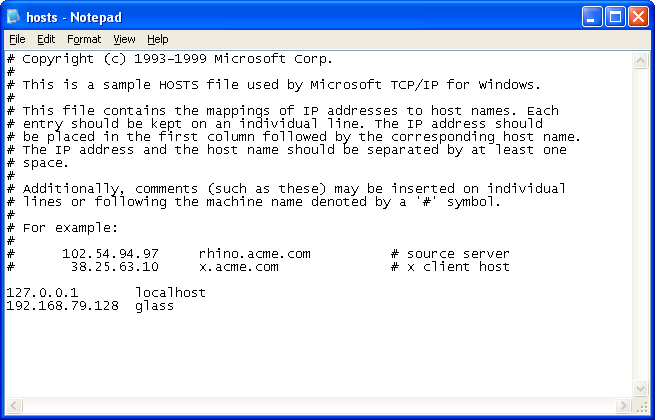
Before we can connect to GemStone on the virtual machine, we need to edit a few files that are used by the local machine to handle networking. First, we need to add the virtual machine to the hosts file so we can refer to it by name. On Linux or Macintosh you should find ‘hosts’ in /etc/ and you can edit it from a terminal using the command ‘sudo vi /etc/hosts’. We will give more detailed instructions to our friends on Windows. On Windows, open a Windows Explorer to C:\WINDOWS\system32\drivers\etc and double-click on the ‘hosts’ icon.



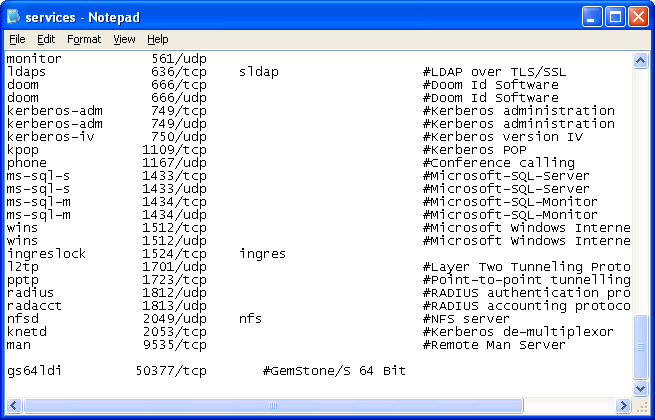
When asked what application to use to open the file, select ‘Notepad’ and click OK.



Add a line at the end with the IP address of the virtual machine (shown in the web browser in the virtual machine’s console) and the word ‘glass.’ In my case the IP address is 192.168.79.128 but in your case it might be different. Save the file and close.



Open the ‘services’ file in the same manner and add ‘gs64ldi 50377/tcp #GemStone/S’ to the end. Save the file and close Notepad.

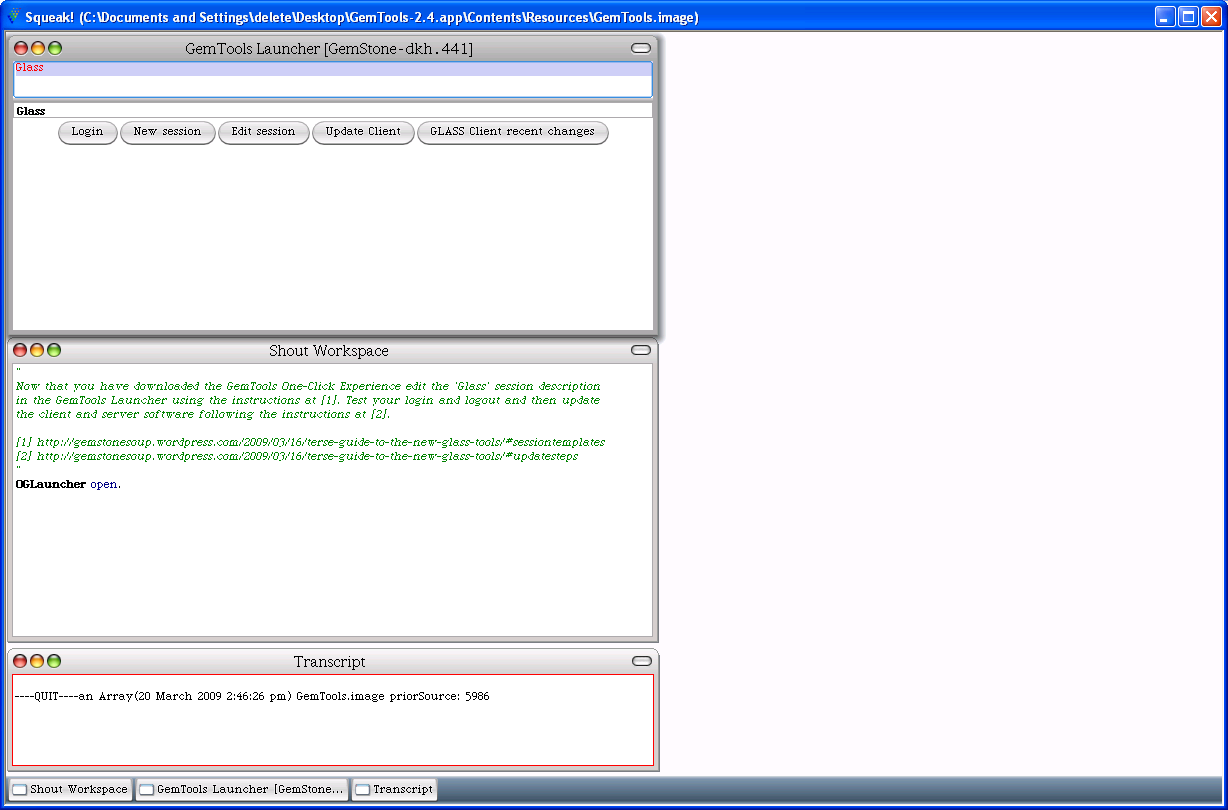


Now return to the GemTools application and launch it. This will launch Pharo with an image that contains three windows: (1) a GemTools Launcher; (2) a Shout Workspace; and (3) a Transcript. Instead of the earlier approach of using the Pharo ‘Seaside One-Click Experience’ to create and serve web pages, we are now using Pharo to run a set of tools that give us access to a GemStone system.

In the Shout Workspace we have the Smalltalk expression that was evaluated in this Pharo system to open the GemTools Launcher shown above the Workspace. Otherwise the Workspace has some instructions and a reference to Dale’s ‘Terse Guide to the New Glass Tools’ (which will give more detail than we will give here). Once you have copied the link, you probably should close the Workspace to avoid confusion (since you should not need to evaluate expressions in Pharo.

The Transcript will show notes about various operations that are happening.

The GemTools Launcher has a pre-defined session that connects to a host named ‘glass’ using a port named ‘gs64ldi’ and asks for a connection to a stone named ‘seaside.’ The reason we added ‘glass’ to the hosts file and ‘gs64ldi’ to the services file is so that these identifiers are properly defined. With ‘Glass’ selected in the list at the top of the GemTools Launcher, click the ‘Login’ button.

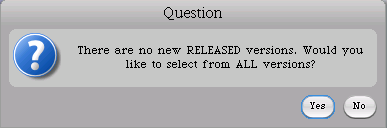


When prompted, enter your initials so that Monticello can track any edits to code.

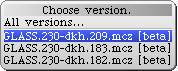
Once you are logged in to the GLASS system, the GemTools Launcher will change to show options available when logged in to the system. If you click the ‘Update’ button you will get a menu of options.



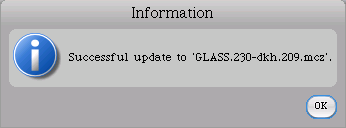
Select first command, ‘Update Client’ to update the GemTools code in Pharo. If the client is up-to-date then you will get a dialog offering you the option of selection from non-released version. Click ‘No’ to stay with the current code.



Repeat the update process and select the option to ‘Update GLASS.’ This should provide a list of released versions available. Select the most recent.



Once the update process completes, you should be informed of that fact.



At this point you can logout and quit the GemTools Pharo application.

If you have are running Macintosh OS X 10.6 or a 64-bit version of Linux you have the option of installing GemStone as a native application on your machine. This approach is a bit more complex to get right, but allows you to run GemStone without the virtual machine overhead. The following script shows how to do the install (see <http://seaside.gemstone.com/downloads.html> for details):

mkdir /opt/gemstone /opt/gemstone/locks /opt/gemstone/log

cd /opt/gemstone

curl http://seaside.gemstone.com/scripts/installGemstone.sh > \

installGemstone.sh

./installGemstone.sh

This script will prompt you for your password so it can update kernel memory settings (feel free to study the script if you have any concerns). After the script completes, continue with the following to install Seaside 3.0:

source /opt/gemstone/product/seaside/defSeaside

startnet

startGemstone

Now to login using a GUI.

curl http://seaside.gemstone.com/squeak/GemTools-1.0-beta.8-244x.app.zip > \

GemTools.zip

unzip GemTools.zip

open GemTools-1.0-beta.8-244x.app/

and the server is started, use 'startSeaside\_Hyper 8000' to start a web server. In a web browser go to http://localhost:8000/seaside.