How to – Share a WebSphere Application Server V6 installation among many Linux for zSeries systems

February 1, 2005

Steve Wehr
zSeries New Technology Center

Contents

Legal statement

Abstract

This document describes a process that enables you to share one installation of WebSphere® Application Server among many Linux® guests running under z/VM®. This is an update to the previous version of this document, which covered WebSphere v5.

As a platform for Linux, one of the strengths of zSeries® is the ability to centrally manage many Linux images. But as the number of Linux images grows, some of the typical problems of large server farms emerge. Software installed on the Linux images must be serviced or updated, and there is no way to do this other than servicing each image as if it were a standalone server. zSeries has the ability to share file systems as VM minidisks, so there ought to be a way to install WebSphere once and use that installation for many other Linux images.

This was previously impractical, because the installed WebSphere program files (which are read/only) and the user data files (which are read/write) were not identified and were intermixed in the same directories. WebSphere Version 6 introduced the concept of "Profiles", which are created by the **wasprofile** command. Profiles must be created for all installations of WebSphere v6, and contain all the user-writable files (logs, installed applications, etc) for WebSphere. All the profiles on a single server share the same WebSphere binaries from the WebSphere installation directory, so service can be applied to WebSphere and apply to all the profiles.

So with WebSphere version 6 the WebSphere binaries are separate from the user data files. We can use this separation to allow profiles to share the WebSphere binaries not only on the same server, but across many servers.

Who should read this paper

This document is intended for use by IBM field personnel or customers. It is written for z/VM system programmers or anyone who creates and maintains Linux guest systems using z/VM. This is a technical paper and requires detailed understanding of:
- How to setup Linux guests using z/VM.
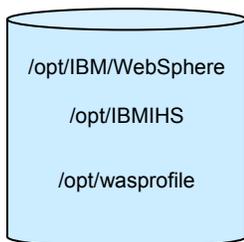- How to install WebSphere and apply fix packs.

Chapter 1: Concepts

The procedure described in this paper relies on two technologies:
- The WebSphere **wasprofile** command.
- The ability of z/VM to share minidisks R/O among many Linux guests.

The **wasprofile** command was designed to create a "profile", which holds all the user-writable files in a WebSphere installation and uses the WebSphere binaries in a separate directory. WebSphere designed this for use <u>on a single system</u>. This concept is adapted for zSeries Linux so that the WebSphere binaries appear to be locally installed on each Linux guest, but actually are links to R/O disks managed by z/VM.

Each Linux system sees this logical view.

But this is the physical view of the setup you will create.

WAS-MASTER   WAS-CLONE1   WAS-CLONE2

/opt/IBM/WebSphere

/opt/IBMIHS

/opt/wasprofile

/

/
/opt/wasprofile

/
/opt/wasprofile

/opt/IBM/WebSphere

/opt/IBMIHS
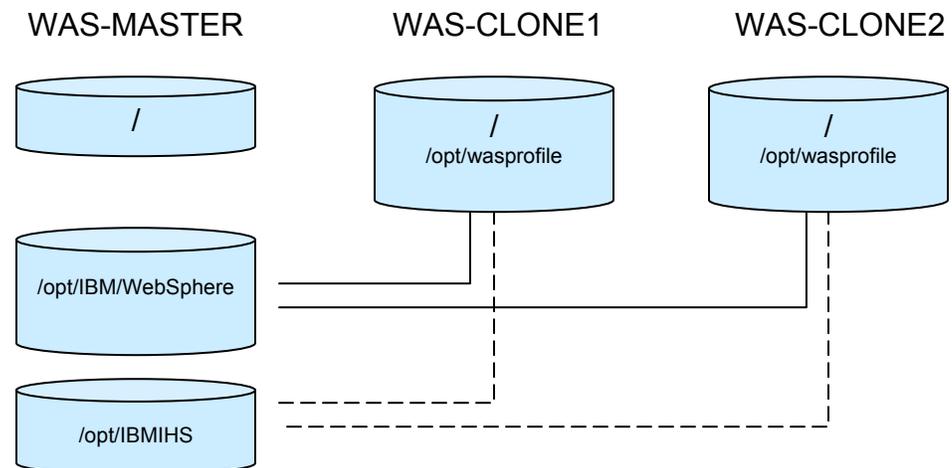
There is one Linux guest called WAS-MASTER where WebSphere will be installed. This Linux system always holds the master copy of the WebSphere binaries. It is only used for installation and maintenance of WebSphere.

Other Linux guests that want to run WebSphere application servers will share these binaries, and are called WAS-CLONEx. These guests will link to the WebSphere binaries that have been installed by WAS-MASTER.

Why use this procedure?

The advantages you get with this procedure are:
- Install WebSphere once for unlimited Linux guests within a VM LPAR.
- Install WebSphere maintenance only once.
- Save about 900MB of disk space per Linux guest.
- Move many guests easily between WebSphere maintenance levels.
- Install and maintain the IBM HTTP Server only once.

In WebSphere V5 there were possible downsides to this procedure:
- In WebSphere V5 you needed an NFS server to make the procedure work. With WebSphere V6, NFS is no longer needed and all those downsides no longer apply.
- In WebSphere V5 you could not use this procedure if you also wanted to use the integrated Java messaging support, since this installed separately from WebSphere. Now that WebSphere V6 uses an all-java solution for messaging, this restriction no longer applies.

Chapter 2: Overview

Software requirements

The following software levels are required for this procedure:
- WebSphere application server version 6.0 or above.

Note that this paper now covers the installation of the Integral JMS Provider (WebSphere messaging). In WebSphere v5 this procedure could not be used if you needed WebSphere messaging. But since WebSphere messaging is integrated with WebSphere now in V6, this restriction no longer applies.
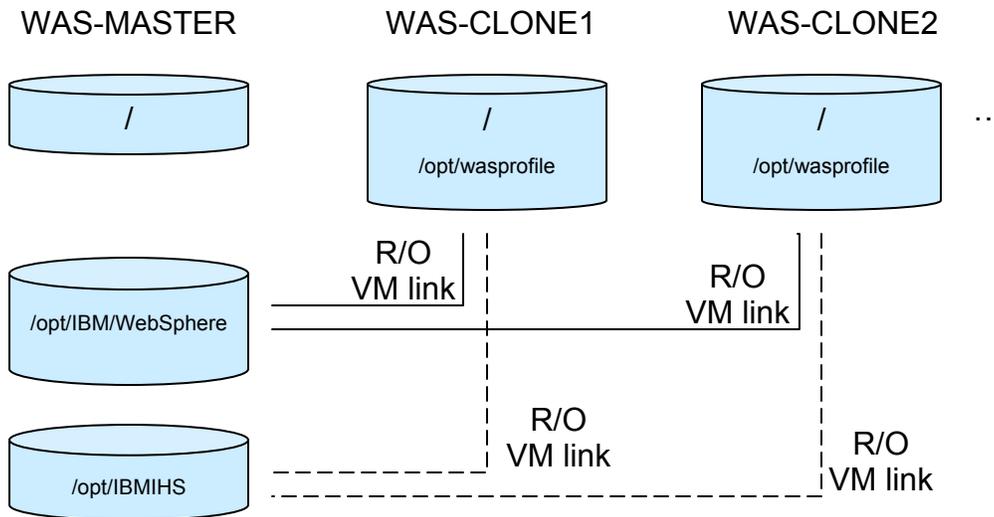
Overview of the procedure

This is an overview of the procedure to accomplish this sharing. Details for each step are found in the next chapter.

1. Create a Linux guest that will hold the master installation of WebSphere. Call this WASMASTER.

2. Create two minidisks:
   - One for holding the WebSphere binaries, /opt/IBM/WebSphere/. Call this the *base installation directory*. (As of this writing, the current version of WebSphere Portal is 5.1.0 and installs into the directory /opt/WebSphere/PortalServer by default. So this will not conflict with the directory on which we will mount WebSphere application server.)
   - (Optional) One for holding the HTTP Server binaries, /opt/IBMIHS. This is only needed if you also want to have separate HTTP servers on each Linux that will share the WebSphere binaries. If so then you can also share the HTTP server binaries.

3. Link both minidisks R/W on WASMASTER

4. Install WebSphere Application Server and IBM HTTP Server on WASMASTER. Do not create any profiles on this server.

5. Link the above minidisks to WASCLONEx. Both are mounted Read/Only.

6. Create a WAS profile using the **wasprofile** command.

7. (Optional) Configure the HTTP Server on the WASCLONEx system. If you are sharing the HTTP server binaries among several Linux servers, then the *httpd.conf* file must be unique and point to the unique WAS plug-in for that server.

8. WebSphere is now ready to run on the clone. You can test to make sure the installation is successful and WebSphere is running OK by running some of the samples applications.

9. Go on to create profiles on more Linux clone systems.

The following diagram shows the disks used and how they are attached to each Linux system.

WAS-MASTER   WAS-CLONE1   WAS-CLONE2

/   /   /   …

/opt/wasprofile   /opt/wasprofile

R/O VM link   R/O VM link

/opt/IBM/WebSphere

R/O VM link   R/O VM link

/opt/IBMIHS

To install WebSphere service:

1. Create a duplicate of the current base installation directory disk using DDR. This is called the "N+1" level of WebSphere.

2. Install the WebSphere fix pack on WASMASTER to the N+1 directories.

3. WASCLONEx users can then switch to link the new disk and test the new WAS level.

4. If everything tests out OK, then the new disk becomes the N level.

Chapter 3: The detailed setup procedure

Step 1: Create file systems and install WebSphere.

1. Create a minidisk to hold the WebSphere base installation directory (*/opt/IBM/WebSphere*). For WAS 6.0 you'll need at least 775 MB, and for the plugin you'll need at least 175MB. So if you will be installing both then create a minidisk of at least 950MB.

2. Create a minidisk to hold the HTTP server installation directory (*/opt/IBMIHS*). You'll need at least 120 MB.

3. Log in to the WASMASTER Linux guest. WASMASTER is the system that you will use to install the WebSphere base installation and any subsequent service.

4. Link both minidisks R/W to WASMASTER.

   The VM Directory entry should look like this:
   ```
   USER WASMASTER WASMASTER 512M 1G G
   INCLUDE LINDFLT1
   NICDEF 600 TYPE QDIO LAN SYSTEM VSW1
   MDISK 191 3390 0580 0010 VM5308 MR LINUX390 LINUX390 LINUX390
   ```

```
MDISK 100 3390 0001 3238 VM5322 MR LINUX390 LINUX390 LINUX390 (root)
MDISK 101 3390 0001 3238 VM5323 MR LINUX390 LINUX390 LINUX390 (/opt/IBM/WebSphere)
MDISK 102 3390 0475 0200 VM51U9 MR LINUX390 LINUX390 LINUX390 (/opt/IBMIHS)
MINIOPT NOMDC
```

/etc/fstab should contain entries like this:
```
/dev/dasdb1            /opt/IBM/WebSphere   ext2       noatime,nodiratime   1 0
/dev/dasdc1            /opt/IBMIHS          ext2       noatime,nodiratime   1 0
```

5. Install WebSphere following the normal installation instructions. However, do not create any profiles.
   - WAS ND install by default will launch the Profile Creation Wizard. Just skip this. We do this because we will not be running any profiles from the WASMASTER machine. (You may choose to create profiles and run them on the WASMASTER machine, but for the purposes of this procedure we treat the WASMASTER machine as only used for the installation and maintenance of WebSphere.)
   - WAS (non-ND) install by default will create a default profile. This won't hurt, we just won't use it.

6. Log in to WASCLONEx.

7. Mount the directories containing the WebSphere base installation on each of the WASCLONEx systems. The base installation directory (*/opt/IBM/WebSphere*) is linked R/O.

   The VM Directory entries should look like this:
```
USER WASCLONE1 WASCLONE1 512M 1G G
INCLUDE LINDFLT1
NICDEF 600 TYPE QDIO LAN SYSTEM VSW1
MDISK 191 3390 0610 0010 VM5308 MR PASSWORD PASSWORD PASSWORD
MDISK 100 3390 0001 3238 VM5328 MR PASSWORD PASSWORD PASSWORD (root)
link wasmaster 101 101 rr   (/opt/IBM/WebSphere)
link wasmaster 102 102 rr   (/opt/IBMIHS)
```

   /etc/fstab should contain entries like this:
```
/dev/dasdb1            /opt/IBM/WebSphere   ext2       defaults   0 0
/dev/dasdc1            /opt/IBMIHS          ext2       defaults   0 0
```
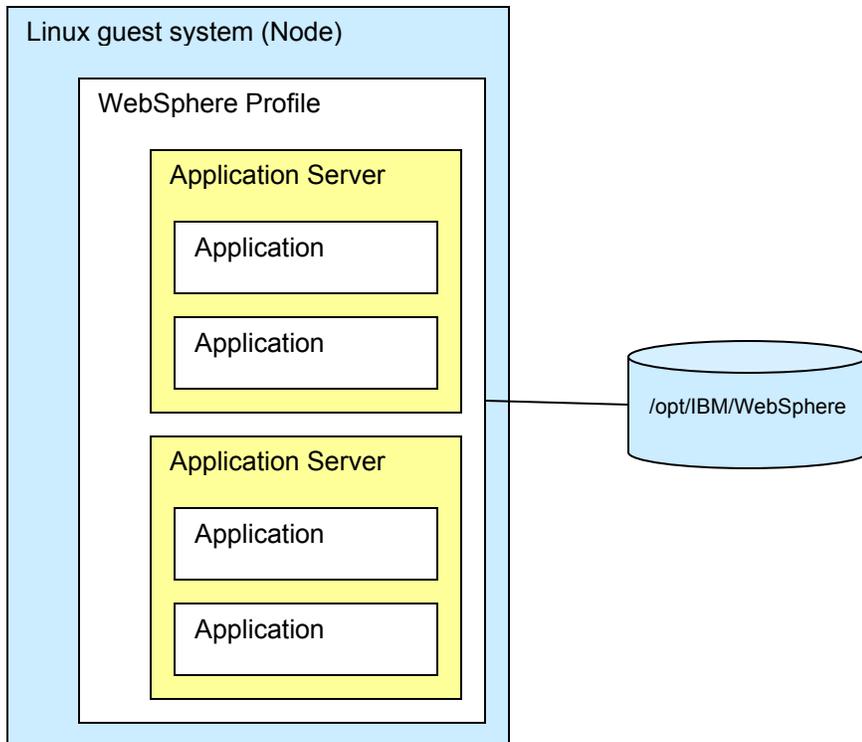
Step 2: Create a WebSphere Profile

Profiles are a new concept in WebSphere v6. Think of a profile as the final installation step in order to run WebSphere. Profiles can be of two types (there are three but we'll ignore the third for now) Application Server profiles and Deployment Manager profiles. We will create one application server profile on each Linux guest that wants to share the WebSphere installation. The only reason to have two profiles on a guest is if one of them is the deployment manager.

As in prior versions of WebSphere, you can create many Application servers on a single installation of WebSphere. In WebSphere V6 you can create many Applications Servers in a *profile*. A profile just represents the WebSphere installation on that Linux.

This diagram shows how nodes, profiles, application servers, and applications are related.

WAS-CLONE1

Linux guest system (Node)

WebSphere Profile

Application Server

Application

Application

/opt/IBM/WebSphere

Application Server

Application

Application

Setup for creating profiles

This is done just once, and will be used for all profiles you create in the future.

1. Log in to WASMASTER

2. Edit the file *<WAS_install>/properties/wasprofile.properties*
   The wasprofile command tries to write information about the profiles it creates into the WebSphere install directory (which is read-only from the clones). We will tell it instead to put those files into the local file system of the clone.

3. Change the default location of the logs written by the wasprofile command.
   Change: `WS_CMT_LOG_HOME=${was.install.root}/logs/wasprofile`
   To: `WS_CMT_LOG_HOME=/opt/wasprofile/logs`

4. Change the default location of the WebSphere profile registry.
   Change:
   `WS_PROFILE_REGISTRY=${was.install.root}/properties/profileRegistry.xml`
   To: `WS_PROFILE_REGISTRY=/opt/wasprofile/properties/profileRegistry.xml`

5. Remount this disk from all the clones to pick up these changes.

Create a profile

Do the following on every Linux guest on which you will be running WebSphere.

1. Log in to the WASCLONEx system.

2. Create the following directories:
   `/opt/wasprofile/profile`
   `/opt/wasprofile/logs`
   `/opt/wasprofile/properties`

An application server profile needs at least 200MB of disk space. They start out about 75MB with the sample applications, but will grow as applications and servers are defined in the profile. All the files for the profile will be in the `/opt/wasprofile/profile` directory.

3. Copy a file from the WASMASTER system. The WebSphere install program creates a file named `.WASRegistry` in the home directory of the root user, which is the userid used to install WebSphere. This file contains only one line:
`/opt/IBM/WebSphere/AppServer`
You need to create the same file on each Linux guest that will be running WebSphere. The easiest way to do this is to just edit the file and add that one line:
`# vi ~root/.WASRegistry`
insert the following line:
`/opt/IBM/WebSphere/AppServer`

4. Run the profile creation wizard to create an "application server" profile, or a "custom" profile. I used the wizard just because it was easier than entering all the parameters on the wasprofile.sh command. But that command will work just the same.
   - If you are creating profiles that will be immediately added to a cell, then create a "custom" profile. These profiles are not configured by default with all the default applications.
   - If you are creating profiles that will not be administered by a deployment manager cell, then create an "application server" profile.

   Start the profile creation wizard. It's an X-windows application. The command is located at
   /opt/WebSphere/AppServer/bin/ProfileCreator/pctLinux390.bin
   `# pctLinux390.bin`

4. In the wizard select the type of profile to create.

5. Specify the profile directory. This is where the files for the profile will be stored. I used `/opt/wasprofile/profile`. (When running the Profile Creation Wizard, this directory must be empty. But the wasprofile command does not enforce this.)

6. Specify your desired host name and node name.

7. Use the default ports. As long as each profile you create is on a separate Linux guest, they can all use the same ports because they all have different IP addresses. Remember that on the same Linux guest you can still create as many "application servers" as you want. All the application servers can run in the same profile. If you do choose to create another profile on the same Linux guest, then you'll need to change these port numbers for the new profile.

8. Let the Profile Creation Wizard run to completion.

Test the new profile you just created.

1. When the Profile Creation Wizard is done, you can let it run the First Steps Console, from which you can run an installation verification test. Or you can start the First Steps Console later from the command line.

2. All commands to work with the profile are found in `/opt/wasprofile/profile/bin`.

3. If you've created an Application Server profile you can test it by running:
   - The sample applications at `http://<your_host_name>:9080/WSsamples`
   - The administration console at `http://<your_host_name>:9060/admin`

If you make a mistake and need to create the profile again, do this first:

1. Remove the profile using the wasprofile command:
   `./wasprofile.sh -delete -profileName <profile_name>`

2. If you're going to use the Profile Creation Wizard to create the profile, then remember to remove all the files from the /opt/wasprofile/profile directory. The logs have been left in there from the former profile.

3. Now you're ready to create the profile again using the wasprofile command or the Profile Creation Wizard.

To create a profile on another Linux guest, just log in to the other Linux guest and repeat this step, using the host and node names of that guest.

Step 3: Configure IBM HTTP server (optional)

If you are using an HTTP server on a separate Linux guest to front-end your WebSphere servers, then install the HTTP server on that guest as normal. If your application servers are federated, then the WAS plug-in for this HTTP server can be generated by the Deployment Manager.

Follow these instructions if you want an HTTP server on each WASCLONEx Linux guest, and you want to share one HTTP server installation among these guests:

1. Log in to WASCLONEx

2. Create a new directory to hold the local files for the HTTP server:
   `/opt/wasprofile/http`

3. Copy the HTTP server configuration file to your local directory.
   `cp /opt/IBMIHS/conf/httpd.conf /opt/wasprofile/http/httpd.conf`

4. Edit *httpd.conf* to change the location of logs, etc that will be R/W on the clone.
   The following lines need to be changed:
   - Change: `PidFile /opt/IBMIHS/conf/httpd.pid`
     To: `PidFile /opt/wasprofile/http/httpd.pid`
   - Change: `ErrorLog /opt/IBMIHS/logs/error_log`
     To: `ErrorLog /opt/wasprofile/http/error_log`
   - Change: `CustomLog /opt/IBMIHS/logs/access_log common`
     To: `CustomLog /opt/wasprofile/http/access_log common`
   - There are several other log files specified in the *httpd.conf*. If you are using any of these change their locations. You can just search within *httpd.conf* for "logs" and see if you need to change any of these directives.
   - Change: `WebSpherePluginConfig /opt/WebSphere/AppServer/config/cells/plugin-cfg.xml`
     To: `WebSpherePluginConfig /opt/wasprofile/http/plugin-cfg.xml`

5. Copy the *plugin-cfg.xml* file from the plugins directory to the WASCLONEx profile directory.
   `cp /opt/IBM/WebSphere/Plugins/config/webserver1/plugin-cfg.xml /opt/wasprofile/http/plugin-cfg.xml`

6. Edit your copy of the *plugin-cfg.xml* file and change the following lines:
   - Change: `<Log LogLevel="Error" Name="/opt/IBM/WebSphere/AppServer/logs/http_plugin.log"/>`
     To: `<Log LogLevel="Error" Name="/opt/wasprofile/http/http_plugin.log"/>`
   - Change: `<Transport Hostname="<was_master hostname>" Port="9080" Protocol="http"/>`

> **To:** `<Transport Hostname="<was_clonex hostname>" Port="9080" Protocol="http"/>`
- **Change:** `<Transport Hostname="<was_master hostname>" Port="9443" Protocol="http"/>`
  > **To:** `<Transport Hostname="<was_clonex hostname>" Port="9443" Protocol="http"/>`

7. Set up the apachectl script to start the HTTP server with your httpd.conf file.
   a. Make a copy of the startup script.
      `cp /opt/IBMIHS/bin/apachectl /etc/init.d/apachectl`
   b. Edit `/etc/init.d/apachectl` and change the following lines:
      1. Point to the new location of the *httpd.conf* file.
         Change: `HTTPD='/opt/IBMIHS/bin/httpd -d /opt/IBMIHS'`
         To: `HTTPD='/opt/IBMIHS/bin/httpd -d /opt/IBMIHS -f /opt/wasprofile/http/httpd.conf'`
      2. You may also want to change the –d option to specify a new serverRoot.
   c. Start the webserver by issuing "apachectl start".

8. Test by going to the home page of the HTTP server for your Linux, which is just the IP address or hostname of your Linux.

Step 4: Create a Deployment Manager Profile (Optional)

If you will be using Deployment Manager then you will need to create a Deployment Manager Profile. You can create this profile on a Linux guest where no other WebSphere profiles exist, or you can create it on a Linux guest that is already running an Application Server profile.

Read and follow the instructions in section **Create a profile** on page 8. And do these additional steps:

1. Log in to the WASCLONEx system where you want the Deployment Manager to run.

2. Create a directory to hold the deployment manager profile: `/opt/wasprofile/dmgr`. A deployment manager profile needs a least 20MB of disk space.

3. Start the Profile Creation Wizard, or run the **wasprofile** command to create the profile.

4. In the wizard select the type of profile to create to be a Deployment Manager profile.

5. Specify the profile directory to be `/opt/wasprofile/dmgr`.

6. Make sure the ports are unique. The Profile Creation Wizard should ensure that the ports for this profile do not conflict with the ports used for any other profiles already created on this Linux guest.

7. Let the Profile Creation Wizard run to completion.

8. The dmgr server should be running. If it's not use the command:
   `# /opt/wasprofile/dmgr/bin/startManager.sh`

Step 5: Federate the profile

If you want the servers in your profile to be part of a Deployment Manager cell, then follow these instructions to federate the profile. (These are the standard instructions for federating a node, there is nothing unique about these instructions because you are sharing WebSphere binaries.)

1. Log in to WASCLONEx

2. Run the **addNode** command to federate all the application servers in this profile. This command stops any running application servers in the profile.
   ```
   # /opt/wasprofile/profile/bin/addNode <dm_host_name> <dm_port>
   ```
   (See the WebSphere InfoCenter for more details on this command)

3. The node agent should start at the end of the addNode command. If it doesn't:
   ```
   # /opt/wasprofile/profile/bin/startNode.sh
   ```

4. If the node agent still will not start then this is probably due to a port conflict. Check the logs for errors about the same ports being used by both the dmgr and the server you just federated.

5. Repeat these steps for every node you want to federate.

6. Now go to your Deployment Manager and test that you can work with the new node(s).

Chapter 4: Apply service to WebSphere

There are two ways to apply service to WebSphere:

- <u>Simple</u>: Just like you would install WebSphere service on a standalone system, you can just install the WebSphere fix pack onto the live systems on WASMASTER.
  1. Log in to WASMASTER
  2. Install the fix pack
  3. Go to each of the WASCLONEx systems and reboot each, or stop all the WAS servers and remount the R/O disks to pick up the changes.
  4. If the fix pack causes problems, then uninstall it from WASMASTER and have all the clones remount the disks again.

- <u>Robust</u>:
  1. Install the WebSphere service onto two new minidisks. Current WASCLONE systems still access the two "old" minidisks running the last level of WAS.
  2. Install the fix pack
  3. You decide which WASCLONEx systems mount the new minidisks, to test out the fix pack.
  4. When you decide that the fix pack is stable, then have *all* WASCLONEx systems mount it.

The "Robust" procedure

The following diagram shows the setup of minidisks for the "Robust" method of applying service to WebSphere.
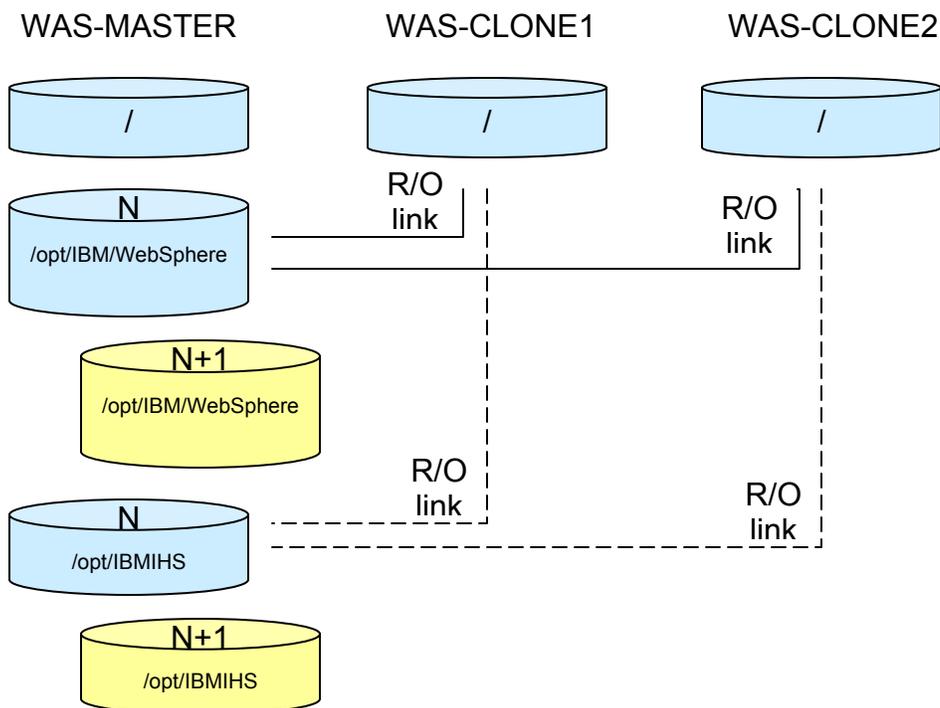
Log in to WASMASTER

1. Create two new minidisks to hold the N+1 versions of WebSphere and (optionally) IBM Http Server. (Note that WebSphere fix packs can ship service to the IBM HTTP Server also. If you are not sharing HTTP server binaries also then you'll need to install service onto each installation of the HTTP server.)

2. Copy the contents of the "N" version minidisks to their corresponding "N+1" minidisks.

3. From WASMASTER unlink the N disks and link the N+1 disks in their place. This leaves the N disks untouched so that the WASCLONEx systems can continue to use them.

4. Install the WebSphere fix pack (and optional HTTP server maintenance) onto the N+1 disks that are now mounted to the same mount points formerly used by the N disks.

Log in to the WASCLONEx system, which will be testing the new fix pack level.

1. Shutdown all WAS servers

2. Unlink the previous "N" versions of /opt/IBM/WebSphere and (optionally) /opt/IBMIHS.

3. Link the N+1 versions of these.

4. Restart your WAS servers and test.

5. Repeat this procedure for all other WASCLONEx systems that will be testing the new fix pack.

6. If the testing uncovers a problem with the fix pack, then unlink the N+1 minidisks and link the N minidisks.

Once you are done migrating to the N+1 fix pack level, those disks become the "N" level. You can keep the minidisks for the N level to be used the next time you need to install a fix pack. They now become the new N+1 level minidisks.

Appendix 1: Contact information

Steve Wehr
zSeries New Technology Center, Poughkeepsie NY
swehr@us.ibm.com

Carlos Ordonez
zSeries New Technology Center, Poughkeepsie NY
carloso@us.ibm.com

Kevin Curley
zSeries New Technology Center, Poughkeepsie NY
curley@us.ibm.com