A Cookbook for the use of Installation Manager on z/OS with Websphere on z/OS

This document can be found on the web at:
www.ibm.com/support/techdocs
Search for document number WP102014 under the category of "White Papers"

Version Date: February 2, 2013
See "Document Change History" on page 3 for a description of the changes in this version of the document

Mike Loos
Consulting IT Specialist
mikeloos@us.ibm.com
(651) 270-3324

Written and provided by the WebSphere Application Server for z/OS team at the
Washington Systems Center
IBM Advanced Technical Skills
Gaithersburg, MD, USA
The WebSphere Application Server for z/OS support team at the Washington Systems Center consists of: John Hutchinson, Mike Kearney, Mike Loos, Louis Wilen, Lee-Win Tai, and Don Bagwell.

Mike Cox, Distinguished Engineer, serves as technical consultant and advisor.

Brian Pierce provides invaluable system programming support to the team and considerable input to this paper in particular.

Jeff Mierzejewski provided much guidance on the installation and use of Installation Manager on z/OS.

For questions or comments regarding this document, e-mail Mike Loos at mikeloo@us.ibm.com
# Table of Contents

Overview.................................................................................................................................4
Assumptions...............................................................................................................................4
Creation of the Installation Manager..........................................................................................5
  *Installation Manager mode selection*..................................................................................5
  *Setup of the Security System*..............................................................................................6
  *Surrogate setup for Batch and USS*.....................................................................................7
  *Creating the Installation Manager Filesystem*......................................................................7
  *Creating the Installation Manager*......................................................................................8
  *Verification of the Install*....................................................................................................9

Creation of the WebSphere on z/OS V8 Filesystem.................................................................9

Using the Installation Manager to Populate the WebSphere Filesystem............................10
  *Installation of the Base Code Level*..................................................................................10
  *Installation of Fix Pack 3*.................................................................................................12
    *Maintenance Overview*....................................................................................................12
    *A word or two about proxy servers*................................................................................13
    *Setup work to Use the IBM Hosted Repository*.............................................................14
    *Installing the Fix Pack Using the IBM Repository*.......................................................14
    *A Word or Two about iFIXs*..........................................................................................16

A Discussion About Maintenance.............................................................................................17

Frequently Asked Questions.................................................................................................18

Document Change History....................................................................................................20
Overview

One of the more significant changes introduced with WebSphere on z/OS Version 8 is the use of IBM Installation Manager to create the runtime executable code. This function was previously completed by the System Modification Program Extended (SMPE). This paper will provide a “cookbook” approach to the customization for use of Installation Manager for z/OS, and the use of it to complete the installation of WebSphere on z/OS Version 8 and its associated products.

*It is worth noting, that the procedures described in this document apply to WebSphere on z/OS V8.5 and other related products, as well as WebSphere on z/OS V8.*

In short, this will involve the following steps:

1. Creation of the Installation Manager using the Installation Manager Install kit.
2. Creation of the WebSphere on z/OS V8 filesystem components on a /Service mountpoint.
3. Using the Installation Manager to populate the filesystem.
4. A discussion of a suggested method for properly handling the installation of maintenance.

Since the configuration of WebSphere on z/OS using the zPMT component of the WebSphere Customization Toolbox is essentially unchanged from the previous version, this will NOT be covered by this paper.

Assumptions

Since the use of SMPE should hold no mysteries for the person(s) performing this work, we will assume that the Installation Manager for z/OS Install kit and the WebSphere on z/OS Installation Manager Repository have already been successfully installed using SMPE.

We will also assume that enough resources are available for this work to be performed. Some of these jobs require large region sizes and installation specific restrictions on those region sizes, usually implemented via an IEFUSI exit can impair the correct functioning of these jobs. The successful circumvention of these restrictions is left to the installation.

The format of this paper is to serve as an aid to getting WebSphere on z/OS properly installed in a maintainable fashion. We will not be discussing the intricacies of either WebSphere on z/OS or the Installation Manager on z/OS. For that information the reader is directed to the appropriate InfoCenter articles for each product.

Information Center for Installation Manager:

http://publib.boulder.ibm.com/infocenter/install/v1r5/index.jsp

Using a search argument of “z/os” will take you to the appropriate sections.

Information Center for WebSphere on z/OS V8.

http://publib.boulder.ibm.com/infocenter/wasinfo/v8r0/index.jsp?
topic=com.ibm.websphere.zseries.doc/info/zseries/ae/welcome_zseries.html

A further assumption will be made that an appropriate level of z/OS with z/OS APAR OA34228 installed is available as the driving system.
Creation of the Installation Manager.

Creation of the Installation Manager using the Installation Manager Install kit involves relatively few tasks.

1. You must decide which “mode” you wish to use for the install.
2. You should perform some setup on the security system.
3. Creation of the filesystem for the Installation Manager.
5. Verification of the install.

Installation Manager mode selection.

The Installation Manager may be created in any of three possible modes: admin, user, or group.

Admin mode: You must be in uid=0 to run in admin mode. A file in /etc/.ibm/registry points to the Installation Manager binaries, so there can only be one admin mode Installation Manager per system. Any user ID with uid=0 can invoke this Installation Manager.

User mode: The userid used when creating the Installation Manager is the only userid that may ever use it. A file in the $HOME/.ibm/registry points to the binaries, so there can only be one user mode Installation Manager per user.

Group mode: A group mode Installation Manager may be invoked by any userid connected to the USS group that owns the Installation Manager files. There can be any number of group mode Installation Managers and there can be any number of userids connected to the group.

Since many (most) installations restrict the assignment of uid=0 to actual z/OS users, admin mode is probably not the mode of choice for most installations.

The restriction to only one user using a user mode Installation Manager makes that mode undesirable for most z/OS installations.

That leaves us with group mode as the probable choice. Since this is what the majority of installations will find to be the most usable mode, this is the mode that this paper will use. The use of the other modes involve very simple and few changes, other than who has the ability to use the Installation Manager.

When using the Installation Manager in group mode, you need to either:

A. If it is a batch job, you can:
   I. Specify the GROUP= parameter on the JOB card.
   II. Specify the Installation Manager's USERID and PASSWORD on the JOB card.
   III. Specify the Installation Manager's USERID on the JOB card without the PASSWORD parameter if you have SURROGAT authority for the USERID.

B. If you are using OMVS:
   I. Use the su command to switch to the Installation Manager's USERID and respond to the password prompt.
   II. Use the su -s command to switch to the Installation Manager's USERID if you have SURROGAT authority for the USERID.

C. If you are using telnet or SSH:
   I. Use any of the methods in B above.
II. Change your group to the Installation Manager’s group with the newgrp command before invoking the Installation Manager.

Setup of the Security System.

The setup of the security system is simple. It involves setting up one userid connected to one new group. The userid, or more likely the group, needs to have access to the following profiles in the RACF FACILITY and UNIXPRIV classes with the stated level of access.

**FACILITY:**
- BPX.FILEATTR.APF READ
- BPX.FILEATTR.PROGCTL READ
- BPX.FILEATTR.SHARELIB READ

**UNIXPRIV:**
- SUPERUSER.FILESYS.CHOWN READ
- SUPERUSER.FILESYS.CHANGEPERMS READ

The following JCL fragment (which is available in the **GIN.SGINJCL** dataset member **GIN2ADMN**) contains a sample set of commands to accomplish the required setup.

```
//RACF    EXEC PGM=IKJEFT01,DYNAMNBR=20,REGION=0M
//SYSTSPRT DD  SYSOUT=*  
//SYSTSIN  DD  *  
ADDGROUP IMGROUP OMVS(GID(gid))
ADDDUSER IMADMIN DFLTGRP(IMGROUP) OMVS(UID(uid) HOME('/u/imadmin') + PROGRAM('/bin/sh') ) NOPASSWORD
PERMIT BPX.FILEATTR.APF CL(FACILITY) ID(IMADMIN) ACCESS(READ)
PERMIT BPX.FILEATTR.PROGCTL CL(FACILITY) ID(IMADMIN) ACCESS(READ)
PERMIT BPX.FILEATTR.SHARELIB CL(FACILITY) ID(IMADMIN) ACCESS(READ)
PERMIT SUPERUSER.FILESYS.CHOWN CL(UNIXPRIV) + ID(IMADMIN) ACCESS(READ)
PERMIT SUPERUSER.FILESYS.CHANGEPERMS CL(UNIXPRIV) + ID(IMADMIN) ACCESS(READ)
SETR RACLIST(FACILITY) REFRESH
SETR RACLIST(UNIXPRIV) REFRESH
/*
```
Surrogate setup for Batch and USS.

The RACF commands to set up the use of the IMADMIN USERID for Batch processing are as follows.

```
RACDEF SURROGAT IMADMIN.* UACC(NONE)
PERMIT IMADMIN.* CLASS(SURROGAT) ID(userid) ACCESS(READ)
SETR RACLIST(SURROGAT) REFRESH
```

The RACF commands to set up the use of the IMADMIN USERID for USS processing are as follows.

```
RACDEF SURROGAT BPX.SRV.IMADMIN UACC(NONE)
PERMIT BPX.SRV.IMADMIN CLASS(SURROGAT) ID(userid) ACCESS(READ)
SETR RACLIST(SURROGAT) REFRESH
```

Creating the Installation Manager Filesystem.

We need to create a filesystem and mount it before we can create the Installation Manager. The default mountpoint is at `/InstallationManager` but we will be doing all of our work in the `/Service` directory so our mountpoint will be `/Service/InstallationManager`. The following JCL fragment (which is available in the `GIN.SGINJCL` dataset member `GIN2CFS`) will create the required filesystem, mountpoint, set the permissions and ownership properly, and mount the filesystem at the mountpoint.

```
//CREATE   EXEC PGM=IKJEFT01,REGION=0M
//SYSTSPRT DD  SYSOUT=* 
//BPXOUT   DD  SYSOUT=* 
//STDOUT   DD  SYSOUT=* 
//SYSTSIN  DD  *
BPXBATCH SH +
    /usr/lpp/InstallationManager/V1R5/tools/zCreateFileSystem.sh + 
    -name WAS800.GRPMODE.GINHFS.ZFS + 
    -type ZFS +
    -volume NGIV8A +
    -cylinders 2500 250 +
    -mountpoint /Service/InstallationManager +
    -owner IMADMIN +
    -group IMGROUP
/
```

A successful run of this job should result in the following messages in the `STDOUT` output.

```
CWLCS9023I Defining file system WAS800.GRPMODE.GINHFS.ZFS.
IOEZ00248I VSAM linear dataset WAS800.GRPMODE.GINHFS.ZFS successfully created.
CWLCS9024I File system WAS800.GRPMODE.GINHFS.ZFS successfully defined.
```
Creating the Installation Manager.

We're now ready to create the Installation Manager that we will use for all further tasks involved with creating and maintaining the WebSphere on z/OS V8 binaries. We had previously decided to use group mode, so the sample jcl provided in the GIN.SGINJCL dataset member GIN2INST may require some editing.

The userid created previously should be used to run this job, so the job card will need to be modified appropriately.

The command that will be used determines the mode:

installc Admin mode
userinstc User mode
groupinstc Group mode

So the command will have to be changed from installc to groupinstc.

The paths specified for the -installationDirectory and -dataLocation arguments will need to be changed to reflect the pathname we created in the previous step.

This is what it should look like based on what we've done previously.

```bash
//BUILDIM EXEC PGM=IKJEFT01,REGION=0M
//SYSTSPRT DD SYSOUT=* 
//BPXOUT DD SYSOUT=* 
//STDOUT DD SYSOUT=* 
//SYSTSIN DD * BPXBATCH SH + 
/usr/lpp/InstallationManager/V1R5/groupinstc + 
   -installationDirectory /Service/InstallationManager/bin + 
   -dataLocation /Service/InstallationManager/appdata + 
   -acceptLicense
/*
```

A successful run of this job should result in the following message in the STDOUT output (The level numbers may vary).
Installed com.ibm.cic.agent_1.5.0.20110909_1200 to the /Service/InstallationManager/bin/eclipse directory.

Verification of the Install.
The most simple verification of the install is to run the imcl version command from either the OMVS or a telnet shell. This can be accomplished by:

1. Open the shell.
2. Either logon with a userid connected to the Installation Manager group or issue an su command to the Installation Manager userid or if running from telnet/ssh you may issue the newgrp command.
3. cd to the newly installed /Service/InstallationManager/bin/eclipse/tools directory.
4. Issue the imcl version command.

```
-> cd /Service/InstallationManager/bin/eclipse/tools
:/Service/InstallationManager/bin/eclipse/tools
-> ./imcl version
Installation Manager (installed)
Version: 1.5.0
Internal Version: 1.5.0.20110909_1200
```

The above messages would indicate a successful installation.

Creation of the WebSphere on z/OS V8 Filesystem.
We now need to create a filesystem to contain the WebSphere on z/OS V8 binaries at a mountpoint contained within the /Service mountpoint. This process is nearly identical to that which we used to create the filesystem for Installation Manager. The following JCL fragment (which is available in the BBO.SBBOJCL dataset member BBO1CFS) will create the required filesystem, mountpoint, set the permissions and ownership properly, and mount the filesystem at the mountpoint.

```
//CREATE   EXEC PGM=IKJEFT01,REGION=0M
//SYSTSPRT DD  SYSOUT=* 
//BPXOUT   DD  SYSOUT=*  
//STDOUT   DD  SYSOUT=*  
//SYSTSIN  DD  *
BPXBATCH SH + 
   /Service/InstallationManager/bin/eclipse/tools/+ 
zCreateFileSystem.sh 
   -name WAS800.V8R0BASE.SBBOHFS 
   -type ZFS 
   -volume WASV8A 
   -cylinders 3360 336 
   -mountpoint /Service/usr/lpp/zWebSphere/V8R0BASE 
   -owner IMADMIN 
   -group IMGROUP 
/*
Note the name of the filesystem, WAS800.V8R0BASE.SBBOHFS. The middle qualifier is used to tie the hfs to the level of WebSphere installed, in this case the base level of the product (keep in mind that all of this is being done in the /Service directory). Also, the last directory in the mountpoint is a match for the middle qualifier of the filesystem dataset name. These names will become more important when we start to apply maintenance to the product.

The following output messages in STDOUT would indicate a successful run.

```
CWLCS9023I Defining file system WAS800.V8R0BASE.SBBOHFS.
IOEZ00248I VSAM linear dataset WAS800.V8R0BASE.SBBOHFS successfully created.
CWLCS9024I File system WAS800.V8R0BASE.SBBOHFS successfully defined.
CWLCS9021I Formatting ZFS file system WAS800.V8R0BASE.SBBOHFS.
IOEZ00077I HFS-compatibility aggregate WAS800.V8R0BASE.SBBOHFS has been successfully created.
CWLCS9012I Creating mount point directory /Service/usr/lpp/zWebSphere/V8R0BASE.
CWLCS9013I Mount point directory /Service/usr/lpp/zWebSphere/V8R0BASE successfully created.
CWLCS9006I Mounting data set WAS800.V8R0BASE.SBBOHFS at mount point /Service/usr/lpp/zWebSphere/V8R0BASE.
CWLCS9007I WAS800.V8R0BASE.SBBOHFS successfully mounted at mount point /Service/usr/lpp/zWebSphere/V8R0BASE.
CWLCS9017I Setting owner and group for directory /Service/usr.
CWLCS9018I Owner and group successfully set for directory /Service/usr.
CWLCS9019I Setting permissions for directory /Service/usr.
CWLCS9020I Permissions successfully set for directory /Service/usr.
```

Using the Installation Manager to Populate the WebSphere Filesystem.

Installation of the Base Code Level.

The package name that we are installing: com.ibm.websphere.zOS.v80_8.0.0.20110503_0249 is the “fully qualified” package name, including the product level. The package name without the product level would have been: com.ibm.websphere.zOS.v80 which would have caused the product to be installed at the most current level available in the repository. Since we wanted the base level of the product, and two levels were available in the repository, we needed to specify the fully qualified package name. The list of available packages is obtained using another option on the imcl command. We also include the -long and -features keywords to get all available information.

```
:/Service/InstallationManager/bin/eclipse/tools
-> . /imcl listAvailablePackages -long -features -repositories
/InstallationManagerRepository/HBB0800
  /InstallationManagerRepository/HBB0800 : com.ibm.websphere.IHS.zos.v80_8.0.0.20110503_0303 : IBM HTTP Server for WebSphere Application Server for z/OS : 8.0.0.0 : core.feature
  /InstallationManagerRepository/HBB0800 : com.ibm.websphere.IHS.zos.v80_8.0.3.20120320_0548 : IBM HTTP Server for WebSphere Application Server for z/OS : 8.0.0.3 : core.feature
  /InstallationManagerRepository/HBB0800 : com.ibm.websphere.NDMS.zos.v80_8.0.0.20110503_0249 : DMZ Secure Proxy Server for IBM WebSphere Application Server for z/OS : 8.0.0.0 : core.feature,thinclient
  /InstallationManagerRepository/HBB0800 : com.ibm.websphere.PLG.zos.v80_8.0.0.20110503_0306 : Web Server Plug-ins for IBM WebSphere Application Server for z/OS : 8.0.0.0 : core.feature
  /InstallationManagerRepository/HBB0800 : com.ibm.websphere.zos.v80_8.0.0.20110503_0249 : IBM WebSphere Application Server for z/OS : 8.0.0.0 : core.feature,ejbdeploy,thinclient,embeddablecontainer,samples
  /InstallationManagerRepository/HBB0800 : com.ibm.websphere.zos.v80_8.0.3.20120320_0544 : IBM WebSphere Application Server for z/OS : 8.0.0.3 : core.feature,ejbdeploy,thinclient,embeddablecontainer,samples
```

© 2011 IBM Corporation
Americas Advanced Technical Skills
ibm.com/support/techdocs
Version Date: Saturday, February 02, 2013
The information is presented with one line for each package in the format:

- Repository name.
- Complete package name
- Package description
- List of features available for the package.

The two levels available for WebSphere on z/OS are **8.0.0.20110503_0249** and **8.0.3.20120320_0544**. The first three qualifiers (**8.0.0**) indicate the level and the rest of the number (**20110503_0249**) is the date/time of the package build.

The feature list includes: core.feature, ejbdeploy, thinclient, embeddablecontainer, and samples. If nothing is specified later on the install command, only the core.feature feature will be installed (by default).

You also can see other products in the output: (IHS – HTTP Server V8, NDDMZ – Secure Proxy Server, PLG – WebSphere V8 Plugins).

Now that we have a filesystem created and mounted to contain the WebSphere on z/OS V8 binaries, we can use the Installation Manager that we had previously created to populate it. Again, we'll use a batch job, a sample of which is contained in `BSOJCL` member `BSO1INST`.

The execution of this job may take quite a while. Since most of the work will be done by USS processes that are “spawned” off of the main task, the main task may be in a wait state for a time which could exceed your installation's SMF JWT value. This suggests that it may be a good idea to include the parameter, **TIME=1440** or **TIME=NOLIMIT** on the job card.

Since the Installation Manager we are using is a group mode Installation Manager, this job must be run either by a userid which is connected to the Installation Manager group (IMGROUP) and specifying **GROUP=IMGROUP** in the JOB card, or by the Installation Manager userid (IMADMIN).

```bash
//INSTALL  EXEC PGM=IKJEFT01,REGION=0M
//SYSTSPRT DD SYSOUT=*  
//BPXOUT   DD SYSOUT=*  
//STDOUT   DD SYSOUT=*  
//SYSTSIN  DD *  
BPXBATCH SH +
    /Service/InstallationManager/bin/eclipse/tools/imcl
      +
      install com.ibm.websphere.zOS.v80_8.0.0.20110503_0249,+
        core.feature,ejbdeploy,embeddablecontainer,thinclient+
        -installationDirectory /Service/usr/lpp/zWebSphere/V8R0BASE+
        -sharedResourcesDirectory+
        /Service/InstallationManager/sharedResources+
        -repositories /shared/InstallationManagerRepository/HBBO800+
        -preferences com.ibm.cic.common.core.preferences.preserveDownload+
          addedArtifacts=false+
        -acceptLicense
    /*
```

We specified the base level of the product, as well as a full list of features.

We also had to specify the installation directory, the mountpoint of the filesystem created in the previous step, the shared resources directory, which is a directory within the mountpoint of the filesystem for the Installation Manager, the location of the repository that was created by SMPE (in
this case, /shared/InstallationManagerRepository/HBBO800), a preference argument (this one causes the Installation Manager to not keep artifacts in the shared repository to save DASD space), and the acceptLicense keyword.

We can now use the `imcl` command to check which packages have been installed with this Installation Manager.

```bash
./Service/InstallationManager/bin/eclipse/tools
-> ./imcl listInstalledPackages -long -features
```

The output indicates that we have two packages installed (Installation Manager, and WebSphere on z/OS V8), with a list of installed features (core.feature is the package itself and is not listed), their install locations, and the installed level on the install location.

At this point, we could declare success, unmount or copy the WebSphere filesystem currently mounted at: /Service/usr/lpp/zWebSphere/V8R0BASE, mount it in read only mode at the mountpoint we would use it to actually run WebSphere from, and begin the configuration process with which we may be already familiar. As part of that configuration process we would specify the mount location as the target of an intermediate symbolic link. Let's defer this declaration of successful completion for a little while.

**Installation of Fix Pack 3.**

**Maintenance Overview.**

The installation of maintenance is just as important a process as the installation of the base level of the code and this paper will suggest one methodology for handling that process.

There are two ways to obtain maintenance for WebSphere on z/OS that is installed using Installation Manager.

1. One is to apply maintenance using SMPE to the SMPE installed repository.
2. The other is to use (in addition to the SMPE installed repository) the repository maintained and hosted by IBM.

The first method, maintaining the SMPE installed repository, has a limitation. To save space in the repository, each subsequent fix package will supersede all previous fix packages. This means that the only packages available in the SMPE installed repository will be the base level of the code, and the most current fix package level. As an example, let's say that we install the base level of the code into the repository using SMPE. Then we install fix package 1. We now have the base level and fix pack 1 level available for installation or the application of iFIXs (iFIXs are small fixes which have not yet been rolled up into a fix package, similar to an apar fix, but available in a more timely fashion). Then we install fix package 2. We now have the base level and fix pack 2 level available for installation or the application of iFIXs. And so on...

Using the second method, we would use a concatenated repository list, with both the SMPE installed (local) repository that we used to install the base, and the repository hosted and maintained by IBM (remote), and specified by a URL. The repository hosted by IBM has all fix pack levels available as well as all of the iFIXs available for each fix pack level. So if we want to keep
some configurations running (and maintainable) at base, fix pack 1, and fix pack 2, we have that capability.

There are some other possibilities, such as maintaining your own repository that is an image of the one that IBM hosts, either on z/OS or on a distributed server within your installation. Those are beyond the scope of this paper (they involve the construction and maintenance of repositories and the use of a packaging utility) so we’re going to ignore those options for the remainder of this paper.

Since the second method above provides the most flexibility and is basically the least amount of work, that is what we will concentrate on (and suggest).

A word or two about proxy servers.

If your installation uses a proxy server between your system and the internet, you will need to specify some additional parameters on the `imcl` and `imutilsc` commands discussed in the next sections.

On the `imutilsc` command, in addition to the `-userName` and `-userPassword` keywords, you will need to specify the `-proxyHost`, `-proxyPort`, `-proxyUsername`, and `-proxyUserPassword` keywords.

On the `imcl` command when you are accessing the IBM hosted repository, you will have to specify, in addition to the `-keyring` keyword, you will have to specify the `-preferences` keyword with the following entries.

```
<preferences com.ibm.cic.common.core.preferences.http.proxyEnabled=True,
com.ibm.cic.common.core.preferences.http.proxyHost=<proxyName>,
com.ibm.cic.common.core.preferences.http.proxyPort=<proxyPort>
```

where `<proxyName>` and `<proxyPort>` are replaced with the values specified on the `-proxyHost` and `-proxyPort` keywords of the `imutilsc` command used to setup the keyring.

The use of these proxy related specifications will be demonstrated on the next couple of examples, but eliminated on subsequent ones, simply to save space.
Setup work to Use the IBM Hosted Repository.

Accessing the IBM repository requires that you have an ID and password. This is the same IBM ID that you would use to download things from developerworks, alphaworks, and access other parts of the IBM network. They are freely available and easy to obtain.

You also need the correct URL. The URL can be found in the WebSphere V8 InfoCenter. It is: http://www.ibm.com/software/repositorymanager/com.ibm.websphere.zOS.v80

Since you probably don’t want to specify your IBM ID and password in batch jobs, you’ll need to set up a “keyring” for Installation Manager use. This is very easy to do:

```bash
:/$Service/InstallationManager/bin/eclipse/tools.
imutilsc saveCredential -keyring /u/mjloos/imkeyring -url https://www.ibm.com/software/repositorymanager/com.ibm.websphere.zOS.v80/repository.config -userName mikeloos@us.ibm.com -userPassword ********
-proxyHost your.proxyhost.com -proxyPort yourproxyportnumber
-proxyUserName yourproxyusername -proxyUserPassword ********
The keyring might contain a valid credential or no credential is required for the provided URL.
:/$Service/InstallationManager/bin/eclipse/tools
```

The `imutilsc saveCredential` command can be used to create a keyring for use with Installation Manager. The parameters necessary are:

1. a path to a keyring file,
2. a url for which it will be used,
3. a username
4. and a password.
5. Optional proxy parameters as described previously.

You’ll note that it may be necessary to add `/repository.config` to the end of the url and to specify `https` in place of `http` on the `saveCredential` command.

Installing the Fix Pack Using the IBM Repository.

The steps to actually install the level of WebSphere on z/OS V8 that includes the fix pack is very similar to those necessary to install the base level. First you need to create a new filesystem and mountpoint, then run the install command.

Create the Filesystem:

```bash
//CREATE   EXEC PGM=IKJEFT01,REGION=0M
//SYSTSPPRT DD  SYSOUT=* 
//BPXOUT DD  SYSOUT=* 
//STDOUT DD  SYSOUT=* 
//SYSTSIN DD  * 
BPXBATCH SH + 
.GetService/InstallationManager/bin/eclipse/tools/+
zCreateFileSystem.sh +
   -name WAS800.V8R0FP03.SBBOHFS +
   -type ZFS +
   -volume WASV8A +
   -cylinders 3360 336 +
   -mountpoint /Service/usr/lpp/zWebSphere/V8R0FP03 +
```

© 2011 IBM Corporation
Americas Advanced Technical Skills
ibm.com/support/techdocs
Version Date: Saturday, February 02, 2013

- 14 -
The only thing different from the first filesystem is the dataset name and the mountpoint have been updated to indicate Fix Pack 3 (V8R0FP03).

Install the package at the fix pack 3 level:

```
//INSTALL  EXEC PGM=IKJEFT01,REGION=0M
//SYSTSPRT DD  SYSOUT=*  
//BPXOUT   DD  SYSOUT=*     
//STDOUT   DD  SYSOUT=*     
//SYSTSIN  DD  *             
BPXBATCH  SH +
 /Service/InstallationManager/bin/eclipse/tools/imcl +
   install  com.ibm.websphere.zOS.v80_8.0.3.20120320_0544 +
   -installationDirectory /Service/usr/lpp/zWebSphere/V8R0FP03 +
   -sharedResourcesDirectory +
   /Service/InstallationManager/sharedResources +
   -repositories /shared/InstallationManagerRepository/HBB0800,+ 
     http://www.ibm.com/software/repositorymanager/com.ibm.websphere.zOS.v80,+ 
   -keyring /u/mjloos/imkeyring +
   -acceptLicense 
/*
```

Changes worth noting are the specification of the fix pack level of the package (com.ibm.websphere.zOS.v80_8.0.3.20120320_0544), the path to the fix pack 3 mount point (/Service/usr/lpp/zWebSphere/V8R0FP03), addition of the the remote repository (http://www.ibm.com/software/repositorymanager/com.ibm.websphere.zOS.v80), and the addition of the keyring argument.

Parameters within the preferences argument are separated by a comma and no space.

A successful installation will show a message similar to the one below in the STDOUT DD.

```
Installed com.ibm.websphere.zOS.v80_8.0.3.20120320_0544 to the 
/Service/usr/lpp/zWebSphere/V8R0FP03 directory.
```
Just as with the base level, we can now use the `imcl` command to check which packages have been installed with this Installation Manager.

```bash
./imcl listInstalledPackages -long -features
```

The output indicates that we now have three packages installed (Installation Manager, and WebSphere on z/OS V8 base and fix pack 1), their install locations, and the installed level on the install location, and the installed features.

At this point, similarly to what we could have done after installation of the base level, we could unmount or copy the WebSphere filesystem currently mounted at:

```
/service/usr/lpp/zWebSphere/V8R0FP03
```

mount it in read only mode at the mountpoint we would use it to actually run WebSphere from, and assuming that we had already been running configured WebSphere processes at the base level, we would simply change (delete and recreate) the mount location as the target of the intermediate symbolic link. At that point, the processes could be restarted and would be running at the fix pack 1 level.

This process could be repeated for each fix pack level as it becomes available.

**A Word or Two about iFIXs.**

When a fix is needed for a problem and the installation of a fix pack that includes it isn't possible or desirable, it may be necessary to install an iFIX. An iFIX can be thought of as the Installation Manager equivalent of the SMPE APARFIX. A list of available iFIXs for a particular level of a package can be gotten using the `imcl listAvailableFixes` command.

```bash
./imcl listAvailableFixes
```

Each line of output returned lists the repository where the iFIX exists, the iFIX identifier, and whether or not it is recommended.
To install one of these iFIXs a job such as the following could be used. We'll use the first iFIX in the output listing as an example.

```
//INSTALL  EXEC PGM=IKJEFT01,REGION=0M
//SYSTSPRT DD  SYSOUT=*  
//BPXOUT  DD  SYSOUT=*  
//STDOUT   DD  SYSOUT=*  
//SYSTSIN  DD  *  
BPXBATCH SH +
/Service/InstallationManager/bin/eclipse/tools/imcl              +
install 8.0.0.0-WS-WAS-IFPM38438_8.0.0.20110602_1431 +
-installDirectory /Service/usr/lpp/zWebSphere/V8R0BASE +
-repositories  +
http://www.ibm.com/software/repositorymanager/+  
com.ibm.websphere.zOS.v80 +
-preferences ce.zOS.v80c.common.core.preferences.preserveDownloadedArtifacts=false +
-keyring /u/mjloos/imkeyring +
-acceptLicense  
/*
```

As you can see, the command to install the iFIX (8.0.0.0-WS-WAS-IFPM38438_8.0.0.20110602_1431) is very similar to that used to install a fix pack. Only the name of what is being installed has changed.

### A Discussion About Maintenance.

The suggested methodology for performing maintenance on WebSphere nodes on z/OS has been for some time, to simply install the latest code level, then point to that level with the intermediate symlink specified at configuration time. This methodology need not change. Many installations already use SMPE in a /Service zone to build the code at a particular level, then they copy the filesystem to a test, qa, production, or whatever other system you might be supporting, possibly naming the filesystem and the mountpoint to match the maintenance level. Using Installation Manager, you continue to build levels of WebSphere code at different levels. The major difference, is that you can build them on different filesystems and different mountpoints in the /Service area and Installation Manager will keep track of them all. SMPE only allowed you to have one level at a time. Now we have the capability of applying fix packs and iFIXs to various levels of the code simultaneously. This can provide much additional flexibility.

In the text above, I mentioned that you could unmount the filesystem from /Service and remount it read only at whatever mount point from which you wish to execute. Using that methodology would require that you reverse that procedure if you wished to apply additional maintenance to that installed level. That saves DASD space. If space isn't a problem, there is nothing that would prevent you from leaving all of the levels mounted read/write at the /Service mountpoint and making a copy of the filesystem with a slightly different name to be mounted read only elsewhere. The tradeoff is between flexibility and space.

The use of intermediate symlinks is discussed at length in a couple of papers available on the techdocs website [http://www.ibm.com/support/techdocs](http://www.ibm.com/support/techdocs)
One, WP100653, describes the creation of a model ND cell from soup to nuts. There is a discussion of the use of the intermediate symlink in that paper. It is based on V6 but the part about intermediate symlinks is applicable to V8 as well.

Another, WP100396, is a description of how to perform rolling system maintenance on WebSphere on z/OS and is at the V5 level. Again the concepts haven’t changed through V8.

**Frequently Asked Questions.**

- **Can I use the IBM repository, or a copy of it somewhere within my own installation, in place of those installed by SMPE?**
  
  This is a two part answer. The Installation Manager Install kit may be download in a zip format and following the directions in the Installation Manager InfoCenter, extracted into a filesystem. The products (WebSphere on z/OS V8, Proxy Server, etc.) which are installed using the Installation Manager, must have at least the base product level installed into a repository by SMPE. That repository must be present to verify proper licensing during other operations using repositories on those products (installation of fix packs, iFixs, etc.).

- **Is SMPE maintenance the “recommended” methodology for the maintenance of the Installation Manager Install kit?**
  
  This is up to the installation. Either downloading updates (full replacements) or installing the fix pack via PTFs against the Installation Manager repository with SMPE (also a full replacement) will work equally well. For the Installation Manager repository, SMPE tends to be less error prone as SMPE handles the setting of extended attributes rather than the person performing the installation running special scripts.

- **Our installation currently uses SMPE as an audit tool for the the WebSphere service level and a copy of the SMPE CSI is maintained with the WebSphere filesystem. As we roll out copies of the product filesystems with Installation Manager to different sites within our installation, what will they need to determine the service level?**
  
  If all you want is the gross service level, the `versionInfo.sh` script in the `/usr/lpp/zWebSphere/bin` directory will tell you that. If you actually want to service the filesystem, then you will need to create a copy of:
  
  - The Installation Manager binaries.
  - The Installation Manager appdata
  - The product target filesystems

  all mounted at the same location as they were on the source system (this works for group mode installations), and create the same Installation Manager userid and group on the target system as was used on the source. You can then invoke the Installation Manager on the target system just as you would on the source system.

- **Do you feel that developers should be able to build there own code base? What prevents any user from creating their own code base?**

  I would recommend that the system programmer or system administrator who normally would install WebSphere maintenance be the one who creates and maintains all copies of the code base. Using a group mode Installation Manager, only those users connected to the Installation Manager RACF Group will be able to use the Installation Manager.

- **How do you install stack products that currently do not use Installation Manager?**

  The same way you do now, using SMPE.

- **The listAvailablePackages output is rather cryptic. Is there descriptive text associated with the package name?**

  Yes, using the `-long` operand on the command provides more information, including an
actual product name. For instance, the package name listed would be: com.ibm.websphere.zOS.v80_8.0.0.20110503_0249 while the actual product name listed would be IBM WebSphere Application Server for z/OS : 8.0.0.0.

- It appears that the path name used to build the service level is important. Is it necessary to remount the filesystem at that same location to apply an IFIX?
  Yes, or an exact copy of it. The installDirectory is the key that Installation Manager uses to keep track of multiple copies of the product.

- If after creating a product filesystem at a given service level, many copies were then created, would it be possible to apply the same IFIX to each or any of them by simply mounting the filesystem at the same correct service mountpoint?
  No, the way you would do that would be to install the IFIX to the service copy, then replace the other copies that you want updated with a copy of the serviced filesystem.

- We keep keyrings in RACF. Can I do that with the keyring created for use with Installation Manager?
  The keyring used for Installation Manager is different from those used in RACF. The Installation Manager keyring has nothing to do with certificates as do the RACF keyrings. The Installation Manager keyrings are simply files in the filesystem that have an encrypted (not particularly strong encryption) copy of the userid and password used to gain access to a particular URL on a command so that you won't have to pass the userid and password on the command.

- If I delete my Installation Manager keyring how do I recover?
  Simply recreate it using the imutilc command you used originally. The same process would be used if you changed the password for accessing a particular URL.

- What are the considerations for setting up an installation copy of the IBM repository?
  Is there any reason that WebSphere on z/OS and WebSphere on distributed systems couldn't share the same repository?
  The best reason for setting up an installation copy would be if it is difficult to gain access to the IBM repository from the system(s) on which you perform service. Otherwise, it is just an additional use of DASD space. If your installation has no problem accessing the IBM repository from the system(s) on which you perform service, then you might as well use the IBM repository. It isn't all that much different from using shop zSeries to obtain the original product.

  Since repositories can and usually do contain multiple packages, and each WebSphere distribution is a separate package, there is no problem with co-residency.
Document Change History

Check the date in the footer of the document for the version of the document.

November 7, 2011  Original document.

November 15, 2011  Updated with assigned WP102014 number, made several minor editing changes, and republished.

February 22, 2012  Corrected some incorrect dataset names.

February 2, 2013  Updated with more correct information on using group mode after installation. Also added information on use of Installation Manager when a proxy server is in place. Added information on feature installation.

End of WP102014