Sample SSH-based Installation Scripts
for
WebSphere Application Server Network Deployment
and
WebSphere Extended Deployment

Table of Contents

Disclaimer

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS SOFTWARE ON AN "AS IS" BASIS AND IBM DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE WARRANTY OF NON-INFRINGEMENT AND THE IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. IBM SHALL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR OPERATION OF THIS SOFTWARE. IBM HAS NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS OR MODIFICATIONS TO THE SOFTWARE.

Purpose

This package provides a set of BASH scripts that can be used to automate the installation of the software stack required to run WebSphere Extended Deployment (XD) 5.1.0. This software stack includes the WebSphere Application Server for Network Deployment (WAS-ND) 5.1.0 product, fix pack 1 for WAS-ND, cumulative fix pack 1 for WAS-ND, a JDK Fix pack, and XD 5.1.0. In the end, it will provide a full installation of WAS-ND 5.1.1.1 with JDK 1.4.2 SR1a with WebSphere XD 5.1.0.

These scripts are NOT meant to run as is and are not supported by IBM. They are meant to serve as a starting point and set of example scripts that can be customized to fit your environment. These scripts have been used extensively in the testing of XD and are capable of installing all of the required software in a few minutes on multiple machines simultaneously. IBM has decided to provide these example scripts to help you get a jump start on using a similar solution to automate your product installations.

Machine Environment Assumptions

These scripts make two critical assumptions. The first is that you have a BASH environment to run these scripts. The second is that you have SSH connectivity configured for all of the machines involved. Furthermore, these scripts assume that SSH keys have been configured in order to enable no-password login between the machine running the scripts and the machines being installed. This enables the scripts transfer files to and execute commands on the target machines. SSH is provided by default with most Linux environments and is available for AIX, Solaris, and Windows. More information on SSH is available on the Internet.

Script Assumptions

The scripts themselves are designed to work with a couple of assumptions. First, most of the machine customization of these scripts is provided through a file called ***serverConfig***. This file provides the list of target machines and the location of the installation images. More

information about this file can be found below. The default **serverConfig** file assumes that the installation images for the base application server, the deployment manager, the fixpack, the cumulative fixpack, the JDK fixes, and the XD installer are all available in ZIP file format in the same directory as the script files themselves. Of course, if you place the images elsewhere, you can simply customize the **serverConfig** file to specify their location.

The final assumption made by the scripts is that the script is setting up a different machine than the machine the scripts are executing on. The scripts will connect remote to the target machine and execute the installation process. Any required files will be automatically transferred to the target machine.

It should be noted that all of these scripts are multi-platform enabled. They will detect the operating system of the target machine and install the correct platform version of the software.

Primary Scripts

There are two primary scripts that serve as the starting point for the set of provided scripts.

installWAS: **installWAS** is used to install the full stack minus XD. It will install a set of application server nodes, a single deployment manager node, and it will federate the application servers into the deployment manager. It will install the base product, a fix pack, a cumulative fixpack, and a JDK fixpack. If the product is already installed on the target node, it will stop all running servers, DELETE the existing installation, and reinstall from scratch.

installXD: **installXD** is used to install the XD product on top of an existing base install that was created with **installWAS**.

To install a full XD product stack, you should run **installWAS** followed by **installXD**.

Script Descriptions

Script Name	Description
appserver.response	This is the silent response file for the base application server installation. Make sure you edit this file and customize the application server install options and installation directory to suit your needs.
dmgr.response	This is the silent response file for the deployment manager installation. Make sure you edit this file and customize the deployment manager install options and installation directory to suit your needs.
federateNodes	This script will federate all of your application server nodes to the configured deployment manager.
installWAS	This is the main script for installing the base application server software stack. It will install a deployment manager and 1-n application server machines based on the configuration in

	serverConfig.
installWAS_dmgr.sh	This script is called by *installWAS* and installs the deployment manager on to a target machine.
installWAS_server.sh	This script is called by *installWAS* and installs an application server on to a target machine.
installXD	This is the main script for installing the XD product unto an existing application server or deployment manager. It will install XD to a single deployment manager and 1-n application server machines based on the configuration in *serverConfig*.
installXD_dmgr.sh	This script is called by *installXD* and installs the XD extensions to a deployment manager.
installXD_server.sh	This script is called by *installXD* and installs the XD extensions to an application server node.
serverConfig	This is the configuration file that enables the customization of all of these scripts. This script contains the desired cell name, the target deployment manager hostname, and the list of target application server node. It also contains the target installation location for the software and the location of the installation images for each platform. This should be customized to fit the particular deployment environment.
startAll	This script will start the deployment manager and the node agent on all of the target application server nodes.
startDMgr	This script is called by *startAll* and will start the deployment manager on a target machine.
startNode	This script is called by *startAll* and will start the node agent on a target machine.
stopAll	This script will stop the deployment manager and the node agent on all of the target application server nodes.
stopDMgr	This script is called by *stopAll* and will stop the deployment manager on a target machine.
stopNode	This script is called by *stopAll* and will stop the node agent and all application servers on target machine.
xd_dmgr.response	This is the silent response file for the XD extensions to the deployment manager installation. Make sure you edit this file and customize the XD install options. The installation directory should match the installation directory of the deployment manager.
xd_server.response	This is the silent response file for the XD extensions to the application server installation. Make sure you edit this file and customize the XD install options. The installation directory should match the installation directory of the application server.