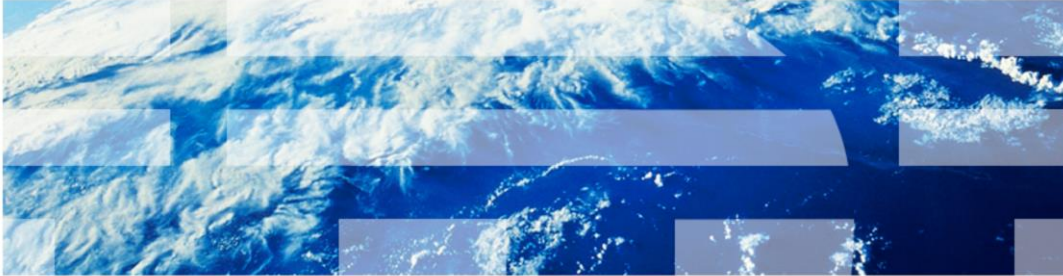


# IBM PureApplication System

## Virtual system patterns - Overview



© 2013 IBM Corporation

This presentation covers how to create the virtual system patterns in IBM PureApplication™ System.

---

## Table of contents

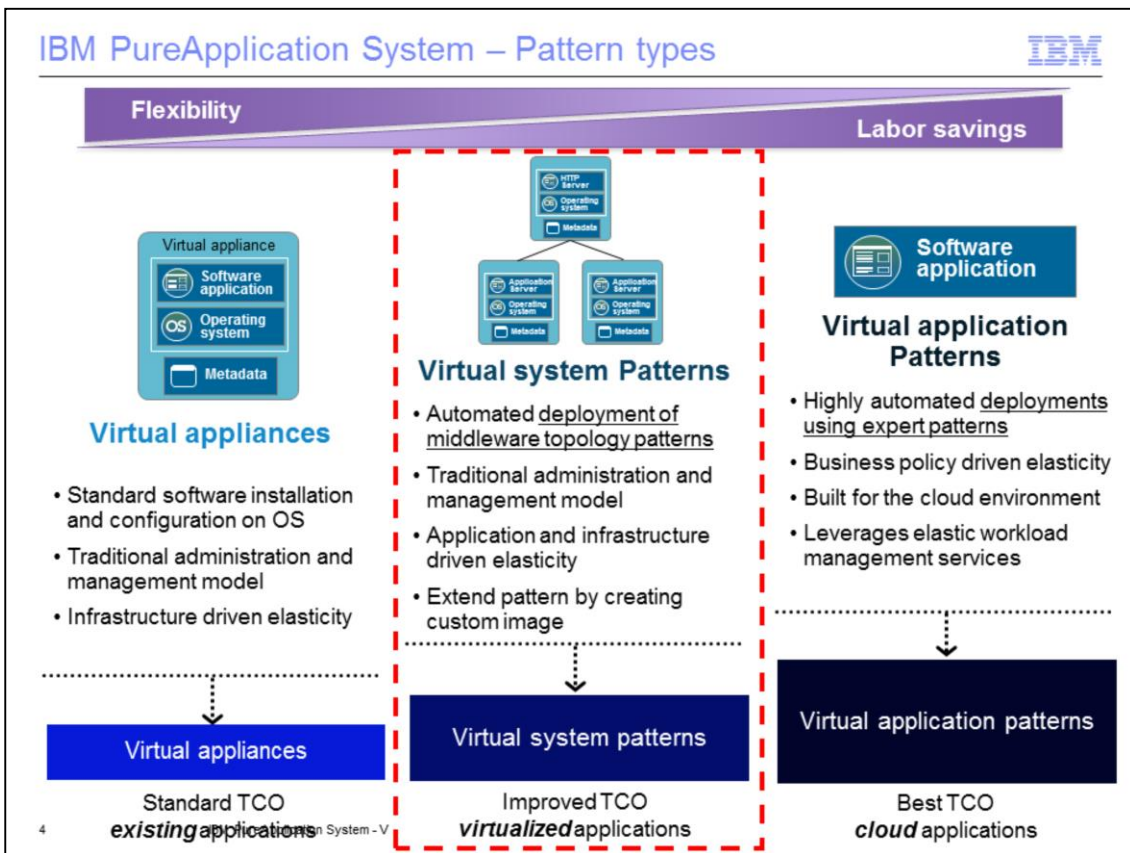
- Virtual system pattern – overview
- Virtual system pattern creation
- Virtual system pattern editing
- Virtual system pattern deployment
- Managing deployed patterns - virtual system instances
- Maintenance and troubleshooting
- Summary

This presentation covers the virtual system overview, followed by high level details of virtual system creation, editing, deploying, and managing deployed instances.

The presentation concludes with an overview of maintenance of patterns, and deployments and troubleshooting.

## ***Virtual system – Overview***

This section provides a quick overview of the virtual systems.



IBM PureApplication System supports three main deployment models, namely the virtual appliance, virtual system patterns, and virtual application patterns.

While other presentations focus on other deployment models, this presentation provides a general overview of the virtual system patterns. These patterns allow automated deployment and life cycle management of middleware topology patterns. It is similar to the traditional model. The administration and management model is the same as what you may already have within your current infrastructure. The time-to-deploy and the density of deployment provides a very high value due to the ease of creating these topologies as reusable patterns and deploying them in the cloud, when compared to traditional IT deployments.

Virtual System patterns support IBM provided middleware images and vendor- or client-created custom images using the tools provided by IBM, namely the Extend and Capture function within PureApplication System, or using IBM Image Construction and Composition tool.

## Virtual systems patterns - overview

- Virtual systems patterns are a logical grouping of virtual image parts combined together to create a specific topology for a given deployment needs
- Examples:
  - WebSphere® Application Server Cluster pattern containing deployment manager, one or more custom nodes, IBM HTTP server and configuration scripts to configure the environment
  - Business Process Manager pattern consisting of BPM Deployment manager, one or more custom nodes, DB2®
- PureApplication System provides a visual GUI editor called Virtual system pattern editor to create the pattern
- PureApplication System includes pre-loaded virtual system patterns based on years of best practice
- Client's focus is on the topology that they create for a given solution
- Clients have full administrative access to the deployed topology

Virtual systems patterns are a logical grouping of virtual image parts combined together to create a specific topology for a given solution and deployment needs. For example, you can create a WebSphere Application Server Cluster pattern containing a deployment manager, one or more custom nodes, IBM HTTP server, and configuration scripts to configure the environment. Or you can create a Business Process Manager pattern consisting of BPM Deployment manager and one or more custom nodes along with its DB2 database.

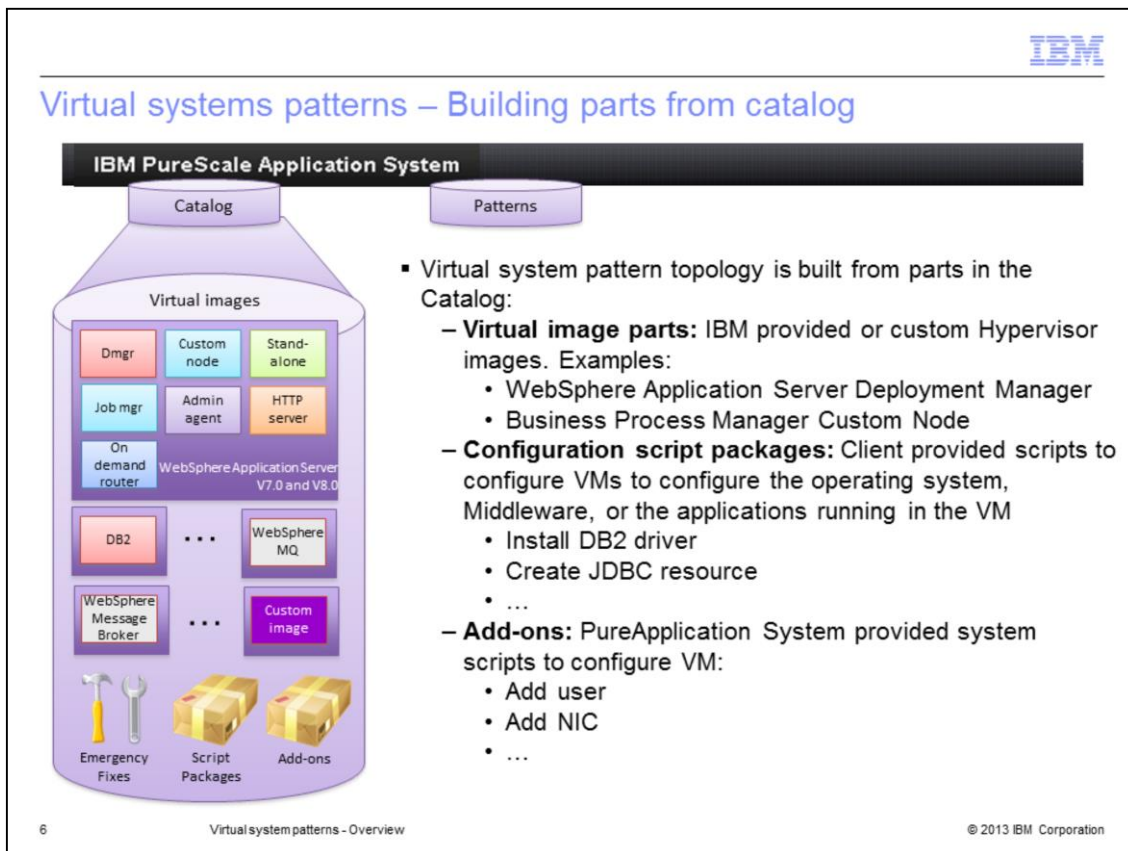
All this is done using a visual GUI editor called the virtual system pattern editor.

PureApplication System includes pre-loaded virtual system patterns based on years of best practice information.

In a virtual system deployment model, your focus is on the topology that you want to create for a given solution. Once the pattern is created and deployed, you have full administrative access to the deployed topology, just like your traditional deployments of the middleware topology that you are used to in your current environment.

You can add configuration script packages to the pattern to customize the pattern and the middleware.

## Virtual systems patterns – Building parts from catalog



Virtual system patterns are built from parts in the catalog. They include the virtual image parts, configuration scripts and add-ons.

The Virtual image parts are IBM-provided Hypervisor images of IBM middleware products or custom Hypervisor images built by someone using the tools provided with PureApplication System. Examples of virtual image parts are WebSphere Application Server Deployment Manager and Business Process Manager Custom Node.

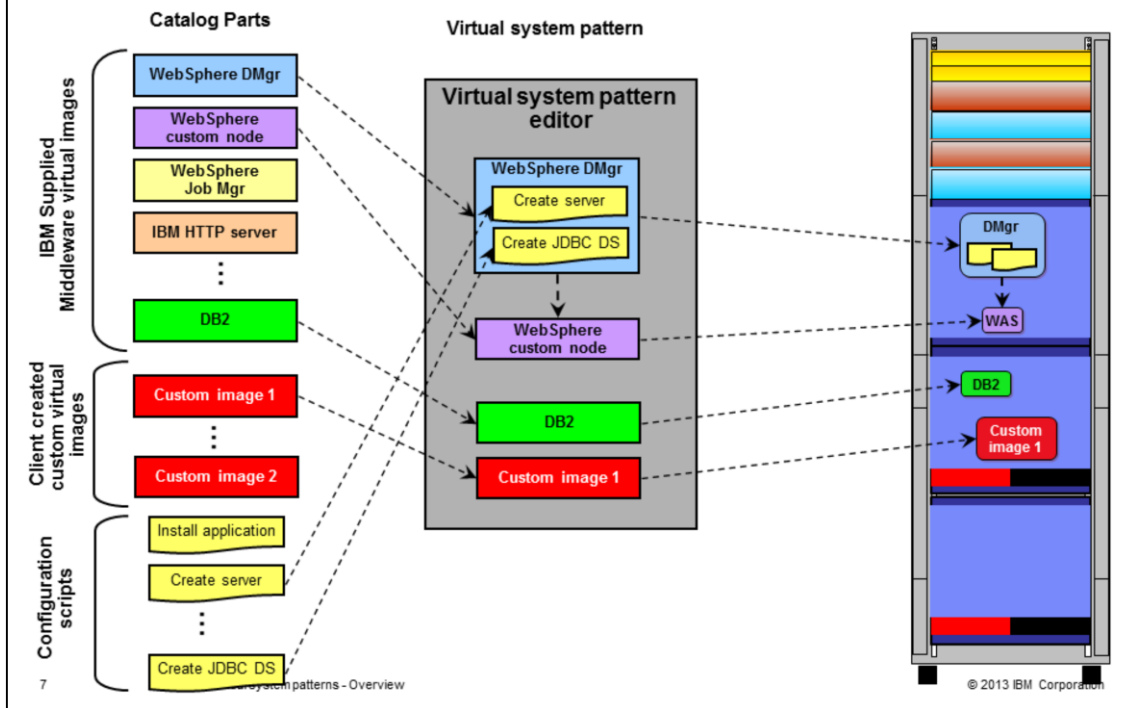
You can then add configuration script packages to these parts. The script packages are OS level or middleware scripts you have created to configure the operating system, middleware, or the applications running in the VM. Examples include installing a DB2 driver, creating a JDBC resource, or installing an application.

Configuration scripts provide total flexibility to configure the deployment the way you want.

Add-ons are PureApplication System-provided system scripts to configure the VM, for example, to add an user or a network interface card.

The picture shows the catalog with some of the virtual images. The list of virtual images is quite exhaustive in IBM PureApplication System.

Catalog parts → Virtual system pattern → Deployed VMs



Virtual systems patterns are a combination of virtual image parts, script packages, and add-ons, combined together to create a specific topology.

A pattern is a set of instructions to PureApplication System on how to create the VMs and configure them.

The example on this slide shows a typical WebSphere Application Server environment with DB2 and scripts to configure the topology.

Once the pattern is deployed, a VM is created for each image part within the pattern, and configured based on the pattern.

The virtual image parts in your pattern could be IBM-provided Hypervisor images. Or you could create your own custom image from the base OS image provided by IBM.

Section

## ***Virtual images***

This section covers virtual images.



## Virtual image parts from IBM

- IBM provides several entitled and non-entitled hypervisor editions of IBM middleware as virtual images for PureApplication System
- These virtual images used to build Virtual System patterns
- Entitled Hypervisor (HV) images :
  - WebSphere Application Server HV 7.0.0, 8.0.0 and 8.5.0
  - DB2 9.7 FP6 Enterprise HV, 10.1 Enterprise HV
  - Advanced Middleware Configuration (RAF 3.0.0.5) HV 1.0.0.3
  - OS Image for Red Hat Linux® Systems 2.0.0.1
- Non-entitled Hypervisor (HV) images (list continues to grow):
  - Business Process Management 8.0
  - Informix® 11.70.FC5
  - Portal (and Content Manager) 7.0.0.2
  - SOA Policy 2.0
  - SOA Policy Gateway 2.0
  - WebSphere MB 8.0.0.1
  - WebSphere MQ 7.0.1.8, 7.5
  - WebSphere Operational Decision Mgmt 8.0
  - WebSphere Transformation Extender 8.4.0.3
  - Commerce V7 FP6 (Virtual system scripts)
  - Maximo® 7.5.0.3
  - InfoSphere® Information Server 9.1
  - IBM Connections 4

- **Non-entitled HV images**
  - **Subject to change**
  - **Availability date varies**
  - **Release levels may change**
- **See PureSystems™ Centre for updated list**

IBM provides several entitled and non-entitled hypervisor editions of IBM middleware as virtual images for PureApplication System. These virtual images are the building blocks of the virtual system patterns.

Some of them are entitled software, meaning their licenses are included as part of PureApplication System. You can use unlimited licenses of entitled offerings.

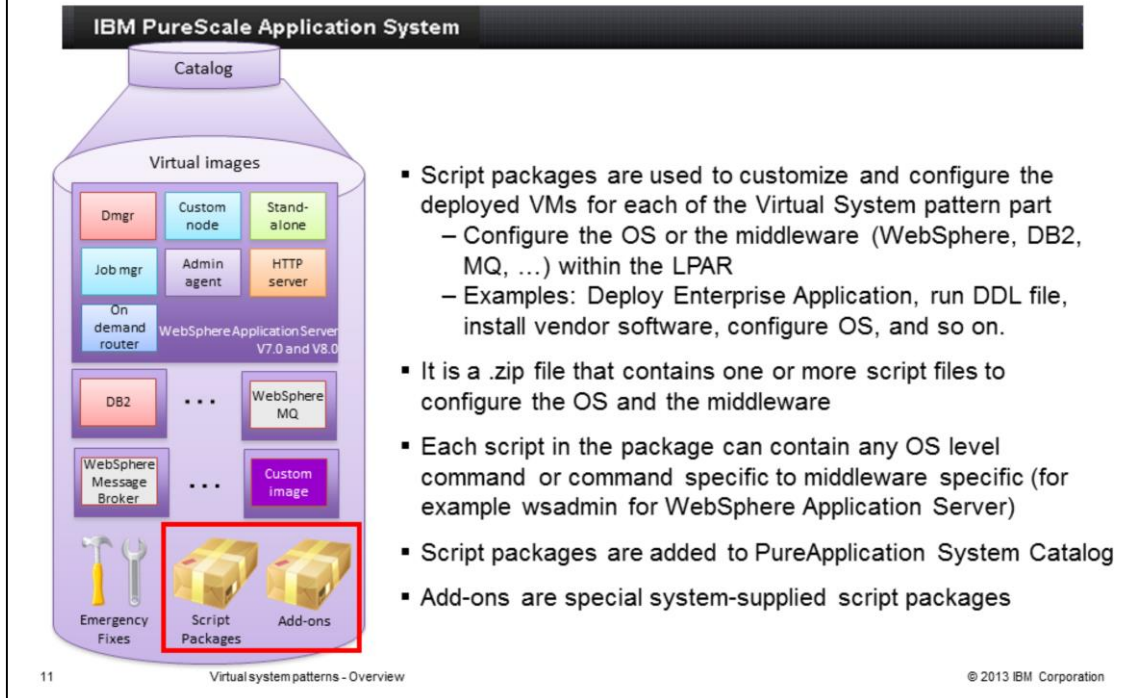
Some of them are non-entitled offerings which requires separate PVU licenses to use the images within the virtual system patterns. While this slide can only show you the snap-shot of the current offerings, the list continues to grow. Visit the PureSystems Centre web site for offerings from IBM and from ISVs and Business Partners.

Section

## ***Script packages***

This section covers script packages.

## Script packages (1 of 2)



As shown here, script packages exist in the PureApplication System catalog, along with the virtual image parts that they are used to customize. Script packages are very flexible and can do just about anything you want, including running wsadmin commands or operating system level commands. There are some examples that come pre-loaded in the PureApplication System catalog; however, most scripts will be user-supplied to do customization specific to your installation. Script packages are typically bundled up as an archive file but can also be a stand-alone script package.

## Script packages (2 of 2)

- Navigation: Workload Console → Catalog → Script Packages

The screenshot displays the configuration page for a script package. At the top, there are action buttons: Refresh, Clone, Lock, and Delete. The main configuration area is divided into several sections:

- Script package file:** A text input field with a "Browse..." button and an "Upload" button. A callout box points to this section with the text "Upload the script package .zip file".
- Environment:** A section containing a variable definition: `DB_IP = ${DB2_ESE.ipaddr}` with a "[remove]" link. Below it is an "Add variable" form with "name" and "value" fields and an "Add" button. A callout box points to this section with the text "Parameters used by script package".
- Working directory:** `/tmp/daytrader`
- Logging directory:** `${WAS_PROFILE_ROOT}/logs/`
- Executable:** `${WAS_PROFILE_ROOT}/bin/wsadmin.sh`
- Arguments:** `-connType SOAP -user ${WAS_USERNAME} -password ${WAS_PASSWORD} -lang jython -f /tmp/daytrader/daytrader_singleServer.py all`
- Timeout:** `0`
- Executes:** A dropdown menu currently set to "at virtual system creation". A callout box points to this dropdown with the text "Specify when the script is executed after deployment".
- Included in patterns:** `TS_Vsys_99`

At the bottom of the page, there is a footer with the number "12", the text "Virtual system patterns - Overview", and the copyright notice "© 2013 IBM Corporation".

The slide shows the panel of a script package added to the catalog. There are areas to upload the script package, specify the script, specify how to run and when to run the scripts. The scripts can run during deployment, during the removal of deployment instances, or on-demand.

## ***Virtual system pattern creation***

This section covers the steps involved in creating a virtual system pattern.

## Creating virtual system patterns

- Access Virtual system patterns from **Workload Console** → **Patterns** → **Virtual Systems**
  - Quick links also provided on the welcome page of the Workload console
- **Three ways to create virtual system patterns**
  1. From scratch, using parts in the catalog, and dragging and dropping on the pattern editor
  2. From cloning an existing virtual system pattern
  3. Importing a virtual system pattern from a previously exported pattern from the same or different PureApplication Systems
    - Using Command Line Interface and sample scripts
      - exportPattern.py and importPatterns.py



14

Virtual system patterns - Overview

© 2013 IBM Corporation


All virtual system patterns operations are managed from the patterns page. There are two ways to navigate to that page: from the “Patterns” menu on the top banner or from the main welcome page, as shown on the slide.

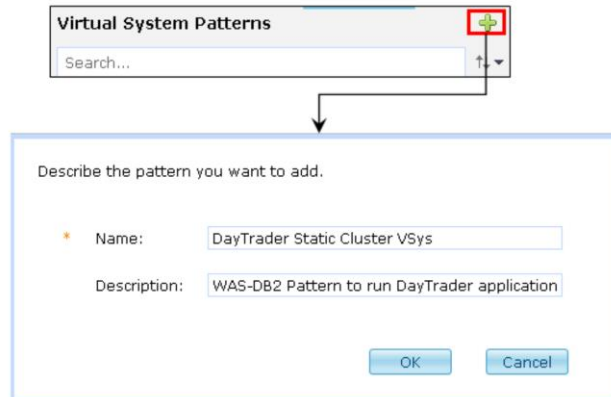
There are three ways to create new virtual system patterns. You can create a new pattern from scratch using the pattern editor and dragging and dropping parts from the catalog, namely, virtual images, script packages and add-ons.

Another way is to clone an existing pattern. This will allow you to modify or extend existing patterns. IBM-provided patterns for a given topology, representing best practices, come as locked patterns. You can use those patterns by cloning and extending the new pattern.

The third way is importing a previously exported pattern using command line options. The download CLI toolkit provides sample scripts for export and import of the patterns. This allows for migrating virtual system patterns between PureApplication Systems.

## Creating a new virtual system pattern

- From the **Workload console** → **Patterns** → **Virtual systems** screen, click the green plus  icon to create a new pattern “shell”
  - Provide a unique name for the pattern with optional description
  - After saving, edit the pattern to build the topology



The screenshot shows the 'Virtual System Patterns' interface. At the top, there is a search bar with the text 'Search...' and a green plus icon in a red box. Below the search bar, there is a dialog box titled 'Describe the pattern you want to add.' with the following fields:

- Name:
- Description:

At the bottom of the dialog box, there are two buttons: 'OK' and 'Cancel'.

To create a new pattern from scratch, click the green plus sign and provide the unique name for the pattern along with optional description.

Once the pattern is created, you can edit the pattern to add the necessary parts from the catalog and specify access rights and their parameters for the pattern.

## Virtual system pattern cloning

- Select pattern and click **Clone** icon
- Provide **Name**, and optionally select a different virtual image

The screenshot displays the 'WebSphere advanced cluster' interface. At the top, there are action buttons: Refresh, Deploy, Edit, Clone (highlighted with a red box), Lock, and Delete. Below these buttons, the cluster details are shown:

- Description:** Advanced cluster is a WebSphere Application Server Network Deployment topology with some of the Intelligent Management Pack features for larger scale development or production environments. The on demand routers reside on dedicated virtual machines.
- Created on:** Aug 3, 2012 1:34:01 AM
- Current status:** Read-only

Below the cluster details, a dialog box titled 'Describe the pattern you want to add.' is open. It contains the following fields and options:

- Name:** Copy of WebSphere advanced cluster (highlighted with a red box)
- Description:** Advanced cluster is a WebSphere Application S
- Virtual image:** A list of virtual images with radio buttons:
  - WebSphere Application Server 8.5.0.0 32-bit RHEL 6 x86-64 (VMWare)
  - WebSphere Application Server 8.5.0.0 64-bit RHEL 6 x86-64 (VMWare) (highlighted with a red box)
  - WebSphere Application Server 8.0.0.3 with Intelligent Management Pack 7.0.0.2
  - ...

At the bottom of the dialog box are 'OK' and 'Cancel' buttons. The footer of the screenshot shows '16 Virtual system patterns - Overview' and '© 2013 IBM Corporation'.

The second option of creating a new pattern is to clone the pattern from existing patterns. Open the pattern you want to clone and select the **Clone** function. Provide the pattern name and optionally select a different virtual image than what the pattern specifies.

This is a good way to test your patterns and applications with newer releases or service levels of the middleware with your applications in the pattern.



## ***Virtual system pattern editing***

This section covers editing for virtual system pattern.

## Virtual system pattern attributes and options

- From this panel, you can:
  - Deploy, Edit, Clone, Lock and Delete your pattern
  - View pattern topology
  - View the deployed systems using this pattern
  - View and modify access control – who can read or write your pattern
- Once locked, the pattern cannot be modified – you can clone the pattern and make changes to the clone

**DayTrader Static Cluster VSys** Refresh Deploy Edit Clone Lock Delete

**Description:** WAS-DB2 Pattern to run DayTrader application

**Created on:** Jun 7, 2012 11:59:04 AM

**Current status:** Draft

**Updated on:** Jun 7, 2012 3:37:44 PM

**In the cloud now:** (none)

**Access granted to:** Default Admin [owner]  
Add more...

**Topology for this pattern:**

Deploys to PureSystems\_ESX hypervisors.

Deployment manager 8.0.0.3

2 Custom nodes

18

Virtual system patterns - Overview

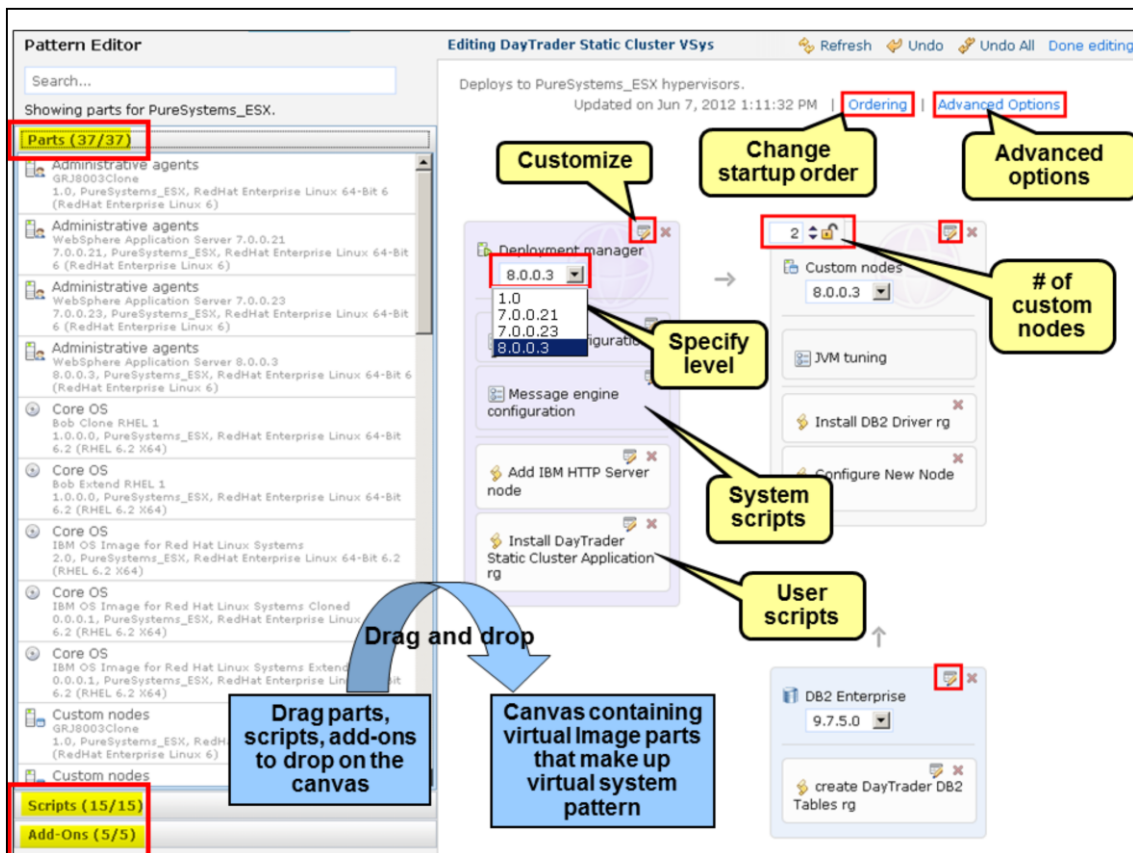
© 2013 IBM Corporation

The functions of virtual system patterns shown on the top of the panel are deploy, edit, clone, lock, and delete. The “In the cloud now” section shows which virtual systems were created from this pattern and “Access granted to” attribute shows who has permission to view, modify, and update this pattern.

There are two options that warrant a deeper look and those are the “Clone” and “Lock” pattern options.

Once you have created, tested, and verified your pattern, you can lock down the pattern to prevent any further changes. Locking a pattern is a permanent operation; once locked it cannot be unlocked. All is not lost if at a later time you decide that changes are required. You need to clone the pattern, make changes to the clone, test and verify. Once you are confident with your new pattern, you can delete the original pattern.

Cloning a pattern makes an exact copy of the original. One typical use of the clone operation is to take one of the predefined “best practices” patterns, clone the pattern, and use that clone as a basis for your own custom patterns.



If this is a new pattern to be built from scratch, you are presented with a blank canvas on the right and a palette of virtual image parts, script packages, and add-ons on the left. You create a pattern by dragging parts from the palette and dropping them onto the canvas. In order to add multiple parts at the same configuration point, where applicable, like in custom node, you increment the number located in the upper left corner of the part.

The part or script can be configured for parameters needed during deployment, for example, the root password of the operating system and credentials for WebSphere administration.

You can choose the release level and the service level of the virtual image part in the pattern. For example, in Deployment Manager, if you want to specify release 7.0.0.3, select that level from the pull down.

Clicking the Ordering link in the canvas allows you to order the sequence of when the different parts get deployed and when the scripts run. This ensures that the dependency order of startup of scripts can be handled. For example, a DB2 JDBC driver needs to be installed and the database needs to be created before installing the application that accesses the database.

Some virtual image parts allow advanced options to further customize those parts. One example is that you can customize messaging and clustering for WebSphere Application Server. Enabling the advanced options will automatically add some system scripts.

The example in this slide shows a WebSphere Application Server Network Deployment topology cell with DB2, totaling four VMs, a deployment manager node managing two custom nodes and DB2, along with the system and user scripts.

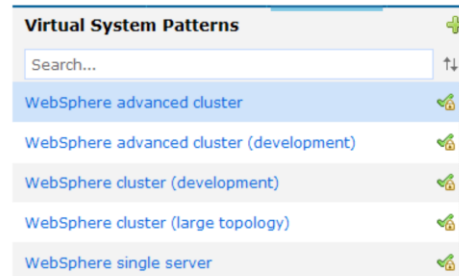
Also shown here are several editing features that allow you to customize the parts, specify the startup order, specify the service levels, system scripts, and user scripts.

## ***Pre-loaded virtual system patterns***

This section covers virtual system pattern deployment.

## Preloaded virtual system patterns

- PureApplication System comes preloaded with a set of “best practices” patterns from IBM for use at different application life cycle - development, test, and production.
- These patterns are locked but can be cloned
- Used as-is or used as a starting point when creating your new custom patterns (by cloning the preloaded patterns)



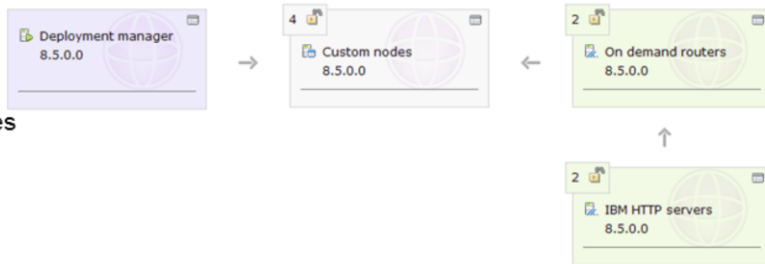
PureApplication System comes preloaded with a set of “best practices” patterns. The screen capture on this slide shows some of the patterns available on IBM PureApplication System. When you create your own custom pattern, you can use one of the preloaded “best practices” patterns as a starting point by cloning the pattern.

By starting with one of the PureApplication System’s existing patterns, you are starting with a “best practice, time tested approach”. The preloaded patterns range in complexity from a simple, single-server topology to a highly-available, cluster topology.

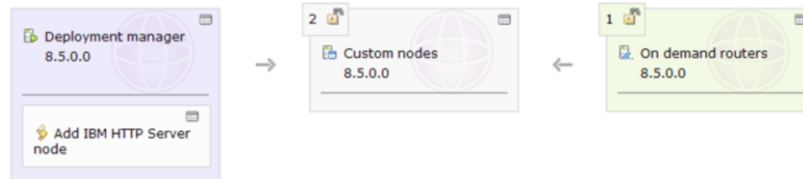
Preloaded patterns come for WebSphere Application Server, IBM Business Process Manager, and Operational Decision Manager, to name a few.

## Examples: Preloaded WebSphere Application Server patterns

- WebSphere advanced cluster
  - 1 deployment manager
  - 4 custom nodes
  - 2 On demand routers
  - 2 IBM HTTP server
  - Total of 9 virtual machines



- WebSphere advanced cluster (development)
  - 1 deployment manager
  - 2 custom nodes
  - 1 On demand router
  - Total of 4 virtual machines



22

Virtual system patterns - Overview

© 2013 IBM Corporation

Here are the “WebSphere advanced cluster” pattern and the “WebSphere advanced cluster (development)” pattern, making use of the Intelligent Management Pack or IMP and its on-demand router.

## Example: Pre-loaded Business Process Manager V8.0 patterns



Here is an example of the preloaded Business Process Manager Pattern for Process Server and Process Center.

## ***Virtual system pattern deployment***

This section covers virtual system pattern deployment.



**IBM**

## Virtual system pattern deployment steps

Daytrader Static Cluster Virtual System
Refresh **Deploy** Edit

- From the pattern attribute window, click **“Deploy”** to deploy the virtual system pattern
- Virtual system pattern deployment steps
  1. Specify a **unique** name for the virtual system (instance) that will be deployed
  2. Choose an **environment profile** to deploy pattern
  3. **Schedule** a deployment time (now or at future time) and optional end time
  4. **Configure virtual image parts** (if not already configured during pattern creation) – can modify any unlocked value at deployment time

Describe the virtual system you want to deploy.

Virtual system name:

Choose environment

Schedule deployment

Configure virtual parts

OK Cancel

Check mark indicate configuration is complete unless you need to modify

No check mark indicate more configuration needed

25      Virtual system patterns - Overview      © 2013 IBM Corporation

Deploying virtual system patterns is a simple process that has four steps. Step one; provide a unique name for the virtual system. Step two; choose an environment profile to deploy your pattern into. Through the environment profile, you are choosing the cloud group and the IP group to use for the deployment along with other parameters.

Step three; schedule a deployment time. Optionally, you can also specify an end time at which the deployment will be deleted.

And finally, step four; configure any remaining configurable options for your virtual image parts and script packages. The virtual image parts configuration could be included in the pattern with an option to allow a deployer to change the value, if not locked in the pattern. This gives greater flexibility to mix what becomes part of the pattern and what can be configured at deployment time.

## ***Managing virtual system instances***

This section covers virtual system pattern deployment.

**Virtual system instance panel**

Instances

- All Instances
- Virtual Applications
- Virtual Systems
- Virtual Appliances
- Shared Services
- Databases

Workload Console → Instances → Virtual Systems

Grant Access

History

Virtual System Instances

Search...

DayTrader Static Cluster VSys

Instance Operations

DayTrader Static Cluster VSys

Refresh Start Stop Store Service Delete

Created on: Jul 20, 2012 3:15:45 PM

From pattern: DayTrader Static Cluster VSys

Parent pattern

Using Environment profile: envProfile18

Deployment priority: Low

Current status: The virtual system has been deployed

Current Status

Updated on: Jul 20, 2012 4:19:12 PM

Deployment ID: d-f706f2cd-c7a0-4605-921a-14810e7ac0c4

Access granted to: deploy18 [owner]

Add more...

Create Snapshot of all VMs and restore from a snapshot

Snapshot: (none)

Create

History

The virtual system has been deployed

Virtual machines

4 total - 4 started

VM details and operations, SSH, VNC

Comments

There are no comments yet

27 Virtual sys

Managing your deployed patterns, called instances, is done by going to the “Virtual Systems” tab for that instance from the Workload Console, or from the welcome screen of the Workload console.

The virtual system instance panel allows you to view and manage the VMs of your virtual system instance.

In the details of the instance, the “From pattern” entry is the pattern from which the virtual system was created. The pattern name is a link to the management panel for the pattern.

“Current status” provides a high level view of the virtual system’s status. A green arrow indicates that the virtual system is available and ready for use.

“Access granted to” shows a list of users on the Workload Deployer appliance who can manage this instance. Only the owner of a virtual system instance can grant access to other users. In this example, Administrator is the owner of the virtual system and only Administrator or users with “Cloud administration” permission can view the virtual system.

Virtual systems deployed to PureApplication System provide the ability to save a copy of the entire current state of a virtual system. This copy is called a snapshot. Taking a snapshot of a virtual system will result in every virtual machine that makes up the virtual system having its own snapshot taken. Restoring from the snapshot takes you back to the state defined by the snapshot. PureApplication System calls out to the hypervisors hosting the virtual machines to create and restore snapshots. The snapshots are stored on the hypervisor and not in the appliance. Note that the restore button will not appear until the first snapshot is created.

History provides view of the history of the creation and operations of the instance.

The virtual machines section provides details and operations on individual VMs of the instance.

The comment section allows you to add documents as comments for later reference.

## Virtual system instance – virtual machines section

- View details of virtual machines of the instance
- Access the operating system using SSH (ID / password)
- Links to VM console (VNC) or middleware (WebSphere) console
- Perform operation on one VM or group action on one or more VMs – clone, stop, start and delete

The screenshot shows the 'Virtual machines' section of the IBM Virtual System Patterns Overview page. It displays a table of virtual machines with columns for Name, CPU, and Memory. Two VMs are listed: 'ipas-lpar-110-248-DMGR-DayTrader' and 'ipas-lpar-111-001-'. The second VM is expanded to show a login form with fields for 'User name' (virtuser) and 'Password'. Callouts highlight 'SSH login', 'VM operation action', 'Group action', 'SSH', 'Actions', 'Group Actions', 'Expand for details', 'Login', and 'Manage' buttons. A 'Group Actions' menu is also visible with icons for start, stop, and delete.

Each virtual system consists of one or more virtual machines. Expand virtual machines to see a list of the virtual machines that make up the virtual system. The pattern used to create this virtual system was a single server and DB2. Therefore, two virtual machines are included.

The initial virtual machine information indicates the amount of processor capacity currently used and the amount of allocated memory currently used. If the indicator is green then either the processor or memory is operating within normal limits. Yellow indicators tell you that the item should be monitored and possible adjustments made. A red indicator is a warning and action should be taken immediately.

You can SSH into the VM using the SSH login link as shown. Applicable operations like start, stop, delete, and clone can be performed on a single VM or can be performed on a group of VMs. Expand the VM tab for more details and additional operations on the VM.

Expanding a virtual machine will present you with information about the virtual machine. This page is divided into seven sections that provide varying levels of information and detail. It contains information such as the hypervisor where the virtual machine is installed, what IP was assigned and the cell and node name of the WebSphere Application Server environment.

**Virtual machine details**

**WebSphere configuration**

Cell name: CloudBurstCell\_1  
 Node name: CloudBurstNode\_1  
 Profile name: DefaultDmgr01

[Show all environment variables](#)

**Script Packages**

Script Package	Status	Time	Log Files
wasHVPatternConfiguration	✓	Jul 20, 2012 4:08:02 PM	remote_std_out.log remote_std_err.log cloudburst_collect1342818482648.zip
Install DayTrader Static Cluster Application 18	✓	Jul 20, 2012 4:11:47 PM	remote_std_out.log remote_std_err.log cloudburst_collect1342818707390.zip
ILMT Agent Install Package	✓	Jul 20, 2012 4:04:19 PM	remote_std_out.log remote_std_err.log
maestro	✓	Jul 20, 2012 4:16:52 PM	remote_std_out.log remote_std_err.log cloudburst_collect1342819011939.zip
Must Gather Logs	✓	Jul 20, 2012 4:18:17 PM	remote_std_out.log remote_std_err.log cloudburst_collect1342819093920.zip

**Execute script** → [Execute now](#)

**Consoles**

- [VNC](#) → VNC console
- [WebSphere](#) → Product specific links

**Product administrator user name and password**

User name:   
 Password:

OK Cancel

29 Virtual system patterns - Overview © 2013 IBM Corporation

The “middleware configuration” section displays information related to the middleware configuration, running of the script files and their log files, and a link to environment variables, with WebSphere configuration shown here as an example.

The “Script Packages” section is divided into log files and other important files generated when the script package runs and when the application server starts. The archive file contains a collection of log files, error files, and the activity log. This archive file is very useful for debugging and troubleshooting any problems with the initial deployment of the package.

For scripts configured to run “When I initiate it”, an additional link to “Execute Now” appears to allow running of the script at any time after the VM has been deployed. This allows you to control when to run the script rather than at the time when the VM is created. Ability to control execution of the script gives you the ability to debug the script in a more controlled fashion. Error flags appear when there are errors running the script.

## ***Maintenance and troubleshooting***

This section covers maintenance and troubleshooting.

## Virtual system – Pattern and instance maintenance

- Emergency fixes on instances
  - Covers existing deployments
  - Batch update of fix (apply to all image instances at a single time)
  - LIMIT: Can not be associated during deployment
- Update images fixes on patterns
  - IBM provided or client extend/captures updated images
  - Planned fixes for new patterns
    - As new service or release levels are installed, virtual system patterns can be edited to use the new levels, as shown



31

Virtual system patterns - Overview

© 2013 IBM Corporation

You can apply maintenance to the virtual systems either before or after deployment. After deployment, you normally apply fixes through the emergency fixes described earlier. The primary limitation of emergency fixes is they cannot be installed during virtual system deployment; they must be applied after the virtual system is successfully deployed and started. Some fixes may be required to fix problems with product initialization, or may be difficult to package in an emergency fix bundle. For these situations you can use the PureApplication System extend and capture feature to extend an existing image and install your fixes. Then specify the captured image in your virtual system pattern.

IBM will periodically provide new virtual image versions with updated middleware and other fixes. You can easily import these images to your PureApplication System catalog.

## Virtual system instance - Applying an emergency fix

The screenshot shows the IBM PureApplication System interface. The top navigation bar includes 'Welcome', 'Instances', 'Patterns', 'Catalog', 'Cloud', and 'System'. The main area displays a list of virtual system instances, with 'AIMCP\_DayTrader\_18' selected. A 'Service' button is highlighted in the top right of the instance details. A dialog box titled 'Describe your service request.' is open, showing options: 'Schedule service', 'Select service level or fixes' (with a dropdown menu), 'Apply emergency fixes' (checked), and 'Product administrator user name and password'. A callout box points to the 'Apply emergency fixes' option with the text 'Specify service level or apply emergency fixes'. The dialog also includes a note: 'A snapshot will be taken before applying service.' and 'OK' and 'Cancel' buttons.

When you apply an emergency fix or service level to a deployed virtual system, the maintenance is applied to all the virtual machines in the instance for which it applies.

Before maintenance is applied, the system will create a snapshot image of the virtual system instance. Only one snapshot image is stored for each virtual system instance. The snapshot can be used to restore your virtual system instance if a problem arises after the maintenance is applied.

When a service level is applied, the affected virtual machine is restarted. When an emergency fix is applied the affected virtual machine is not restarted.

You can schedule service now or at a future time. Any time maintenance is applied, the service history for the virtual system instance is updated. This is a record of all the emergency fixes and service levels that have been applied to this virtual system instance.



## Troubleshooting of virtual system deployed VMs

- Standard middleware troubleshooting techniques apply to the virtual system instance VMs
- Possible to SSH or VNC (if enabled) into the VM
- Many IBM middleware provides links to MustGather logs from the virtual system instance panel
- Script package log files can be viewed from the console

Troubleshooting of virtual system instance VMs uses the standard troubleshooting techniques you are used to from the middleware in the VM. You can SSH or VNC into the VM. Many of the IBM middleware products provide links to “MustGather” logs from the virtual system instance panel

Additionally, script package log files can be viewed from the console.

Section

## ***Summary***

This section provides the summary.

## Summary

- This presentation covered the overview of virtual systems
- What are virtual system patterns and how to create, edit, deploy and manage virtual system patterns and instances
- PureApplication System comes preloaded with industry “best practice” patterns
- PureApplication System has a unified maintenance story

In summary, this presentation covered the overview of virtual systems, how to create, edit, deploy, and manage virtual system patterns and instances.

PureApplication System comes preloaded with industry “best practice” patterns, and has mechanisms for applying updates and patches to the products in your virtual machine images.

## Trademarks, disclaimer, and copyright information

IBM, the IBM logo, ibm.com, DB2, Informix, InfoSphere, Maximo, PureApplication, PureSystems, and WebSphere are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of other IBM trademarks is available on the web at "[Copyright and trademark information](http://www.ibm.com/legal/copytrade.shtml)" at <http://www.ibm.com/legal/copytrade.shtml>

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

THE INFORMATION CONTAINED IN THIS PRESENTATION IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY. WHILE EFFORTS WERE MADE TO VERIFY THE COMPLETENESS AND ACCURACY OF THE INFORMATION CONTAINED IN THIS PRESENTATION, IT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. IN ADDITION, THIS INFORMATION IS BASED ON IBM'S CURRENT PRODUCT PLANS AND STRATEGY, WHICH ARE SUBJECT TO CHANGE BY IBM WITHOUT NOTICE. IBM SHALL NOT BE RESPONSIBLE FOR ANY DAMAGES ARISING OUT OF THE USE OF, OR OTHERWISE RELATED TO, THIS PRESENTATION OR ANY OTHER DOCUMENTATION. NOTHING CONTAINED IN THIS PRESENTATION IS INTENDED TO, NOR SHALL HAVE THE EFFECT OF, CREATING ANY WARRANTIES OR REPRESENTATIONS FROM IBM (OR ITS SUPPLIERS OR LICENSORS), OR ALTERING THE TERMS AND CONDITIONS OF ANY AGREEMENT OR LICENSE GOVERNING THE USE OF IBM PRODUCTS OR SOFTWARE.

© Copyright International Business Machines Corporation 2013. All rights reserved.