An Architectural Overview of IBM Workload Deployer

IBM® Workload Deployer is one of the foundational elements of the IBM private cloud strategy. This appliance provides rapid adoption and deployment of both Infrastructure and Platform as Service offerings. It provides a proven path to better use existing IT resources by improving the efficiency and flexibility of these infrastructures. IBM Workload Deployer is the correct solution for organizations that seek agility in response to a dynamic business environment.

This IBM Redpaper™ publication introduces the IBM Workload Deployer and its capabilities. In addition, it describes the primary tools available to create custom content to be provisioned and managed by the appliance.

This paper contains the following topics:

- IBM Workload Deployer V3.1
- IBM Workload Deployer patterns
- The cloud
- Administrative interfaces
- Appliance settings
- Tools for building custom assets

For more information: This paper is Chapter 1 of *IBM Workload Deployer: Pattern-based Application and Middleware Deployments in a Private Cloud*, SG24-8011. It is available separately for use by those customers who want a high level architectural view of IBM Workload Deployer. You can find the more information about building and deploying virtual systems and applications by downloading the full IBM Redbooks® publication at:

http://www.redbooks.ibm.com/abstracts/sg248011.html
IBM Workload Deployer V3.1

IBM Workload Deployer provides a solution to creating, deploying, and managing workloads in an on-premise or private cloud. It is rich in features that you can use to quickly build and deploy virtual systems from base images, to extend those images, and to customize them for future use as repeatable deployable units. IBM Workload Deployer also provides an application-centric capability that provides rapid deployment of business applications. By using either of these deployment models, an organization can quickly instantiate a complete application platform for development, test, or production.

Solution elements

The IBM Workload Deployer provisions both standard and customized middleware virtual images and patterns to the cloud. These virtual images come from “Hypervisor Editions”, pre-loaded on the appliance to help organizations to develop, test, and deploy business applications easily and quickly.

IBM Workload Deployer can receive and act upon operational data from the resource pools (IBM PowerVM®, IBM z/VM®, or VMware hypervisors) that make up the private cloud. It can also monitor application workload demand conditions and adjust resource allocation or prioritization as required to achieve established service level agreements.

When the virtual environment is no longer needed, the resources are returned to the shared resource pool automatically for future use and are logged for internal charge-back purposes. The appliance manages individual user and group access to resources, providing IT managers with the control needed to optimize efficiency at a fine-grained level.

IBM Workload Deployer integrates seamlessly with development and service management tools from IBM Rational® and IBM Tivoli® for architectural, design, development, management, and monitoring purposes.
Figure 1 shows the core components of an IBM Workload Deployer solution.

First, you have the physical appliance with its hardware configuration and management application firmware, pre-loaded and customizable middleware virtual images, configurable patterns, script packages, and administration interfaces.

Next, you have the on-premise or private cloud environment on which the virtual systems and applications run. The cloud environment consists of the hypervisors, networking infrastructure, and storage devices that are allocated to the appliance.

Finally, you have the instances of virtual systems and virtual applications that are deployed by the appliance into the cloud. These systems are dispensed into the cloud using the intelligent placement capabilities of IBM Workload Deployer, which guarantee efficient cloud resource usage coupled with high availability.
The hardware

IBM Workload Deployer is a 2U rack-mountable appliance based on the IBM DataPower® 7199 / 9005 product family. This appliance offering provides several benefits:

► Consumability
   After the initial setup of the appliance and accepting the user license agreement, the appliance console is immediately available. No extra installation steps are necessary, and you can start building private clouds in minutes.

► Security
   IBM Workload Deployer manages a shared and multi-tenant environment, where isolation and security are of utmost importance. The secure nature of the appliance is rooted in a self-disabling switch, which is triggered if the appliance cover is removed. This physical security allows IBM Workload Deployer to serve as a secure vault for credentials, which can be tied to virtual images throughout their entire lifecycle (in storage, being dispensed, running in the cloud, or being removed from the cloud).

► Storage
   IBM Workload Deployer contains a storage driver that streamlines the storage of image customizations. When an image is loaded on to the appliance, it is “shredded” into parts by the storage driver. When an image is later customized and reloaded on to the appliance, it is similarly shredded in a consistent and deterministic way. These collections of shredded images are then compared and only the new or modified ones are transmitted and stored.

   IBM Workload Deployer serves as a dedicated store for both the pre-loaded and customized middleware virtual images and patterns. The appliance includes advanced compression and storage techniques that allow a significant number of these sizeable virtual images to be stored. The appliance is backed up by the DataPower processing power that is needed to manage and provision these images to the cloud.

► Cost
   The total cost of ownership (TCO) that is associated with a physical appliance is low. With a single appliance, the expensive process of building, customizing, provisioning, and managing systems and applications to a cloud is streamlined and simplified, reducing the cost of operations and the skill level required. Hardware costs are also lowered through better utilization of existing systems. Support costs are reduced through repeatable and consistent deployments.

What is new in IBM Workload Deployer Version 3.1

IBM Workload Deployer V3.1 provides the following enhancements to Version 3.0:

► Support for custom images created by the IBM Image Construction and Composition Tool for deployment into cloud environments.

   The IBM Image Construction and Composition Tool is designed to provide efficient reuse and management of images and software in a cloud environment. It provides the capability to build and share images that are customizable and easily managed.

► Embedded IBM Workload Plugin Development Kit (PDK).

   IBM Workload Plugin Development Kit and detailed documentation is used by the developer to build custom virtual application plug-ins to extend virtual application patterns or create patterns, so that an administrator can import them and make available for other users.

4   An Architectural Overview of IBM Workload Deployer
High availability and failover of the IBM Workload Deployer appliance.

In IBM Workload Deployer V3.1, new active/passive failover capability is supported to use two IBM Workload Deployer appliances. One appliance is configured as the master while the second appliance as a slave. The slave has a replicated database and data so that it is ready to take over on failure of the master. Eventually, the administrator can restart replication between the two appliances where the slave becomes the master and a new appliance becomes the slave.

Support for broader programming models with IBM WebSphere® Application Server Hypervisor Edition V8.0.

These programming models include Enterprise Edition (Java EE) 6, Open Services Gateway initiative (OSGi) applications, Web 2.0 and Mobile, Java Batch, XML, Dynamic Scripting, Service Component Architecture (SCA), Communication Enabled Application (CEA), and Session Initiation Protocol (SIP).

Support for virtual applications running on IBM AIX®.

There is support for deploying virtual applications on AIX, providing the flexibility for platform of choice and using the virtualization benefits offered by IBM PowerVM technology using IBM Power Systems™.

Security enhancements.

To prevent the abuse of a single and all-powerful user account, such as cbadmin, IBM Workload Deployer V3.1 separates administration, auditing, and operation responsibilities and their associated privileges. Each role is defined for responsibilities, such as auditing, hardware administration, software administration, and operations.

IBM Workload Deployer patterns

IBM Workload Deployer bases its ability to provision virtual systems, applications, and databases to the cloud on the concept of patterns. Patterns are logical descriptions of both the physical and virtual assets that comprise a particular solution. The use of patterns allows an organization to construct an individual element or integrated solution one time, and then dispense the final product on demand.

IBM Workload Deployer provides two types of patterns to assist with the rapid deployment and integration of private cloud capabilities: virtual system and virtual application patterns. The appliance ships with preinstalled patterns of each type that represent varying degrees of automation and customization and are optimized with the most appropriate configurations and settings for the solutions that they support. The preinstalled patterns are based on industry-recommended practices.

You can also create your own patterns using a supplied pattern as a template, or create a pattern from scratch. After a pattern is created on the appliance, the pattern can be reused over and over to create multiple identical instances in the cloud. Custom patterns are stored on the appliance and can be reused as needed to ensure consistent and repeatable deployment environments.

Virtual system patterns

Virtual system patterns represent repeatable topology definitions based on various middleware virtual images and runtime configurations. Virtual system patterns provide flexibility and control over the middleware topology to be deployed.
A virtual system pattern typically consists of an operating system and additional IBM software solutions. An example of a virtual system pattern is one that contains Linux operating systems with WebSphere Application Server installed. The pattern might include a deployment manager, several custom nodes, an IBM DB2® database, and an IBM HTTP Server.

Using a virtual system pattern gives you more control over the middleware topology but also requires that you configure the middleware. Script packages can be added to the pattern to automate the customization of the virtual system topology after it is up and running. For example, to create WebSphere resources and to install an application.

These concepts are illustrated in Figure 2.

When a virtual system pattern is deployed, the appliance creates the topology, builds the relationships between the components (for example, federating the custom nodes to the deployment manager), and configures the middleware based on the script packages you provide. System administrators can log in to the system to perform additional customization.

**Virtual application patterns**

While virtual system patterns focus on the topology, virtual application patterns (as the name implies) take an application-centric approach. With virtual system patterns, you describe a middleware topology and IBM Workload Deployer builds that topology in the cloud. With virtual application patterns, you describe an application and IBM Workload Deployer builds the appropriate infrastructure and deploys the application to it. IBM Workload Deployer includes a set of pre-loaded web application and database patterns. You can also create your own patterns, from scratch or by using a supplied pattern as a template.
Virtual application patterns are highly optimized and are constructed solely for supporting a singular workload. This pattern requires the least amount of customization during deployment and provides the most direct method for obtaining a rapid return on investment. Virtual application patterns are application-centric. You provide the application files and describe the characteristics of how the application should be run and managed using policies. The appliance generates the middleware topology to meet your requirements, as shown in Figure 3.

You can use IBM Database Patterns to create and deploy DB2 databases in a Database-as-a-Service (DBaaS) cloud environment. With these patterns, you select the database requirements that meet your needs and IBM Workload Deployer builds the underlying topology to meet these requirements.

You can define the requirements by selecting a database workload standard. You can use these standards to choose from a predefined set of database configurations. The Departmental Transactional workload standard is appropriate for online transaction processing and is optimized for transactional applications. The data mart standard is primarily used for data warehousing and is optimized for reporting applications. Alternatively, you can choose to clone an existing database as a model.

When a workload standard is selected, a set of scripts runs to tune the operating system and instance configuration, create the database and accompanying objects, and load the initial data.

**Pattern elements**

You construct patterns by combining one or more elements together, and then performing a degree of integration. The integration activities can be as simple as standardizing the default location for software installation or as complex as automatic node federation within a WebSphere cell.
Figure 4 provides a high-level view of the elements that can be used to construct patterns and the characteristics that define them.

![Figure 4 Pattern elements]

**Virtual image: Hypervisor editions**

Virtual images are hypervisor edition images that provide the operating system and product binary files that are required to create a virtual system instance. IBM Workload Deployer supports a number of middleware Hypervisor Edition images, in the application infrastructure, business process management, connectivity, database, and portal arenas that are immediately available for use as is or can be customized to add extra functionality.

The appliance uses these virtual images to create and deploy virtual machines into the cloud. The virtual images follow the Open Virtualization format (OVF) specification, which is an industry standard specification for packaging and distributing virtual appliances that contain one or more virtual machines. Using OVF provides a standard mechanism to communicate virtual machine resource requirements to several hypervisors.

IBM Workload Deployer ships with a set of virtual images that includes:

- IBM WebSphere Application Server Hypervisor Edition virtual images for VMware ESX, IBM PowerVM, and IBM z/VM hypervisor technologies. This set of images also includes those images with the Intelligent Management Pack, which provides dynamic runtime capabilities similar to the capabilities that are present in WebSphere Virtual Enterprise.
- IBM Workload Deployer Image for x86 Systems for workload patterns.
- IBM operating system images for AIX.
- WebSphere MQ Hypervisor Edition virtual machine images.
- DB2 Enterprise and Express Hypervisor Edition virtual images.

In addition to this list, you can add your own custom virtual images. You can clone an existing image, deploy it to the cloud, update it, and recapture the new image. You can also create new virtual images based on your own operating systems using the IBM Image Construction and Composition Tool.
Figure 5 shows a snapshot of the Virtual Images catalog in the IBM Workload Deployer user interface. Selecting a hypervisor edition displays the characteristics. Some images provide the core operating system used in virtual application patterns. Others provide the core operating system and additional products and “parts” for use in virtual system patterns.

Patterns are associated with a specific image, supplying the parts that can be used in the pattern. In Figure 5, you can see the parts included in the WebSphere Application Server V8 Hypervisor Edition, for example, custom nodes, a deployment manager, and other elements required to build a topology. These parts can be selected by the pattern builder during the creation of virtual system patterns.

**Topologies**

Virtual system patterns provide the topology for the virtual system to be provisioned to the cloud, for example, a web server, WebSphere Application Server deployment manager and custom nodes, and a database. The topology is created from the parts available in the images, plus optional customization scripts that provide configuration after deployment, for example, the configuration of WebSphere Application Server for an application workload and the installation of the application. The ability to integrate standard aspects of high availability and fault tolerance are contained within the topology.
Figure 6 shows a virtual system pattern topology in the Pattern Editor. Parts, shown on the left, can be added to the topology on the right. Each part becomes a virtual system at deployment. Script packages, for example, the Installation DB2 drivers script package, are added to parts in the topology for execution at deployment. Add-ons can also be used in the pattern to customize a virtual machine, for example, to increase the disk size of the virtual machine or add a user. Multiple instances of a part can be defined, for example, the 2 in the upper left corner of the custom node in Figure 6 indicates that two custom nodes should be deployed.

![IBM Workload Deployer](image)

**Workloads**

You can achieve significant integration with middleware components and infrastructure resources optimize the components for a particular type of application workload. Little knowledge of the underlying components is required to deploy and use the solution. Dynamic and elastic capabilities are fully realized and the system can create or remove additional resources as required by the application demand.
Virtual application patterns are deployed using IBM-defined topologies, which are not exposed to the IBM Workload Deployer administrator. Rather, the topology is based on the application components defined by the virtual application pattern (Figure 7) and is adjusted in the run time to accommodate the workload.

Administrative access to the middleware infrastructure running the application workload is obtained through a simplified interface on the IBM Workload Deployer, not, for example, through an administrative console.

IBM Workload Deployer provides much of the decision-making actions related to deploying and running the application. However, policies can be configured by the pattern builder that define routing policies, policies for scaling workload, logging policies to specify the level of logging detail, and JVM policies. You can use these policies to configure the minimum and maximum heap sizes, JVM properties, verbose garbage collection, and other JVM related settings.

The cloud

The cloud that virtual systems are deployed to consists of a set of hypervisors that you provide. You configure the hypervisors to the IBM Workload Deployer, then create pools of IP addresses to assign to provisioned systems on those hypervisors. Each hypervisor belongs to a cloud group. When a virtual system or virtual application is deployed, it is deployed to a cloud group. The IBM Workload Deployer selects one or more hypervisors in the cloud group and assigns IP addresses to the provisioned systems from the IP group for the hypervisor. An alternative to having a static set of IP addresses is to use an environment profile to specify the IP address to be assigned by the pattern deployer.
The relationship of the cloud elements and a typical deployment flow is shown in Figure 8.

Hypervisors

A hypervisor is a software virtualization program that provides a layer of abstraction between operating systems and physical resources on a machine. This abstraction layer allows multiple operating systems and application stacks to run on a single physical entity and share resources, thus providing higher levels of resource utilization.

To set up the cloud, the administrator defines the location and login credentials for the hypervisors. These hypervisors host the virtual systems that IBM Workload Deployer dispenses. IBM Workload Deployer automatically detects the storage that is associated with the hypervisors and manages the placing of the middleware virtual systems across the set of hypervisors.

The following hypervisors are supported:

- VMware ESX
- IBM PowerVM
- IBM z/VM
IP groups

Pools of IP addresses, known as IP groups, are configured for use by the deployed virtual machines. The administrator defines this pool of IP addresses, and when new virtual machines are created, the appliance takes care of assigning each machine a unique value. IP addresses can then be added to and removed from the pre-configured pool as needed.

Cloud groups

A cloud group is a collection of related hypervisors. When deploying patterns to create virtual systems, you use a cloud group as the deployment target. One or more hypervisors of the same type make up a cloud group, for example, you can group all of your ESX hypervisors together or all of your high-end PowerVM hypervisors together. You can manage resource allocation thresholds, such as processor or memory usage, and also verify the runtime status of your configured hypervisors.

Environment profiles

Environment profiles group related deployment configurations, such as virtual machine names, IP address assignment, and cloud groups. Deploying patterns with environment profiles provide deployments across tiers from a single pattern.

In IBM Workload Deployer, environment profiles provide the functionality to:

- Define the operational environments, such as development, test, or quality assurance
- Define virtual machine naming conventions within the operational environment
- Specify whether the IP group or a pattern deployer provides the IP address on the deployment
- Segment the clouds, and IP groups within the clouds, to specific environments
- Assign aliases to the cloud resources, such as clouds and IP groups
- Assign sections within the clouds to specific users or groups
- Define limitations on the number of virtual processors, virtual memory, and storage

With environment profiles, you can also group multiple clouds to be used in the deployment. You can deploy a pattern to multiple cloud groups of the same hypervisor type. You might deploy a pattern to multiple PowerVM cloud groups, for example. However, you cannot deploy a single pattern to a z/VM cloud group and to a PowerVM cloud group. Environment profiles are platform-specific, so IBM Workload Deployer filters out the appropriate clouds.

Administrative interfaces

There are three ways to interact with IBM Workload Deployer:

- Web-based user interface
- Command-line interface
- Representational State Transfer API
Web-based user interface

The primary administrative access to the IBM Workload Deployer appliance is through the web-based user interface (Figure 9). This management console is enabled when the appliance is first initialized through the serial console.

Command-line interface

The IBM Workload Deployer command-line interface (CLI) provides a scripting environment based on Jython, the Java-based implementation of Python. In addition to commands that are specific to Workload Deployer, you can issue Python commands at the command prompt.

You can use the CLI to manage a Workload Deployer appliance remotely. It communicates with the Workload Deployer appliance over an HTTPS session. The Workload Deployer CLI can run in both interactive and batch modes.
The CLI can be downloaded from the IBM Workload Deployer user interface to a Windows or Linux system.

**Representational State Transfer API**

The IBM Workload Deployer appliance exposes a subset of its function using a Representational State Transfer (REST) API. The API is available on the same IP address or host name used to access the appliance user interface and CLI.

The REST API provides a means to interact with the appliance that is both language neutral and programming model neutral. When using the REST API, you interact with the resources of the appliance, such as the hypervisors, patterns, and script packages, just by using well-defined HTTP URLs and associated HTTP verbs (GET, POST, PUT, and DELETE).

Unlike the UI, the REST API is only supported over the HTTPS protocol. The appliance uses a self-signed certificate for its SSL sessions. The same certificate is used for the UI, CLI, and REST API sessions. You must configure your HTTPS client to either accept or ignore this certificate during the SSL handshake. You must use an HTTPS client that you can use to set the HTTP headers for each request.

Finally, the REST API supports only the sending and receiving of UTF-8 encoded data. Ensure that your HTTP client is appropriately set to encode and decode character data, including JSON data.

**Appliance settings**

After installation, the IBM Workload Deployer appliance settings can be performed through the administrative interfaces. This section provides a high-level overview of the administrative settings on the Workload Deployer appliance, covering networking, security, and basic appliance maintenance.

**Networking**

Networking settings can be configured for the appliance, including the Domain Name System (DNS), Network Time Protocol (NTP), and Simple Mail Transfer Protocol (SMTP) settings for the appliance. Although only a single Ethernet interface is required to be configured on the appliance for it to be functional, multiple Ethernet interfaces can be enabled. The most common reason for doing so is to add a level of redundancy to your environment. Another reason multiple Ethernet interfaces are used is so that the appliance can separate the virtual machines network from the administrative one.

**Security**

IBM Workload Deployer is designed with key features that establish and manage trust across the cloud. In addition to ready for use security on the appliance, you can also use a Lightweight Directory Access Protocol (LDAP) to authenticate users with the Workload Deployer appliance.
Appliance maintenance

Using the backup and restore process, you can capture a complete Workload Deployer environment at any point. You can then either restore that environment on the appliance from which it was taken or restore it on another appliance.

Upgrades to the Workload Deployer appliance are done using firmware updates. New firmware versions can be downloaded from the IBM Fix Central website and used to update your appliance. A firmware upgrade changes only the appliance application and does not affect the Hypervisor Edition virtual images on the appliance.

Power restart

Finally, the appliance can be restarted or powered down from the user interface.

Users and groups

Users and user groups are configurable so that you can manage the level of access for each individual to your Workload Deployer appliance.

User permissions are defined to determine which panels are viewable for each user and to determine a user's access to a particular object. Permissions provide the granularity to define the access and roles for each user. You specify access to patterns, virtual system instances, and catalog content at the object level.

The permissions assigned to users define which administrative tasks for Workload Deployer the users can perform. In addition to determining which of the administrative pages are shown, the content of the Welcome page is dynamically generated by the appliance to display distinct content for users that are assigned dissimilar levels of access. For example, you can define the following role-based groups to control user access to resources on the appliance:

- Pattern deployers: This group has permission to deploy patterns. Typically, these users have less middleware administration expertise and probably want to deploy constructed and configured environments.

- Pattern authors and catalog managers: This group has permission to create patterns, upload script packages, and create custom images. These users are typically seasoned middleware administrators who can build and configure application environments. They map their existing configuration knowledge to the various customization approaches in IBM Workload Deployer.

- Cloud and appliance administrators: This group has permission to administer the cloud infrastructure and the appliance. These users are familiar with the configuration and administration of the hardware components within the cloud. In addition, they have the skills necessary to manage and maintain the appliance.
**Tools for building custom assets**

When the pre-loaded images, plug-ins, and patterns do not meet all the requirements for your business workload, you can easily customize them using IBM Workload Deployer customization tools. The Pattern Editor in the IBM Workload Deployer user interface is used to create and customize virtual system patterns. You can customize virtual application patterns by using the Virtual Application Builder in the user interface. Both tools are available in the IBM Workload Deployer user interface.

In addition, two new tools for creating custom content are available and can be downloaded from the IBM Workload Deployer user interface Welcome window.

- The IBM Image Construction and Composition Tool is used to create virtual images to be used in virtual system patterns.
- The Plug-in Development Kit is used to create custom virtual application patterns.

**IBM Image Construction and Composition Tool**

IBM Workload Deployer includes IBM software product images with preinstalled middleware, such as WebSphere Application Server. Provisioning these pre-loaded images to your private cloud allows you to quickly build the virtual systems required to run your applications.

Reality, however, is never that simple. It is likely that the images provided with the IBM Workload Deployer require modification before deployment. For example, most organizations have corporate standards for security compliance across hardware and software platforms. The images deployed to the cloud must be updated to contain the software required to comply to those standards. Another example is the requirement for the installation of agents used by monitoring software. Finally, consider the case where third-party products are required to be installed on the system. These changes could be made after deployment, but would need to be performed after every deployment, making that option impractical. As the number of virtual systems in the cloud grow, more time and effort is required to keep these systems in compliance.

A better solution is to customize the images before they are deployed. The IBM Workload Deployer provides a “clone and extend” feature that you can use to deploy an image, update it, and then capture it as an image for future deployments. Although this feature has its advantages, you can take this concept one step further with IBM Image Construction and Composition Tool. With IBM Image Construction and Composition Tool, you prepare a software bundle that includes the information required to install and configure new software on a selected base image automatically at deployment time. These software bundles can be reused in IBM Image Construction and Composition Tool to create additional customized images by mixing and matching bundles with base operating system images.

These customized images can be provisioned to a cloud using IBM Workload Deployer, IBM SmartCloud™ Enterprise, IBM SmartCloud Provisioning, or VMware ESX. A direct integration process is now available between IBM Workload Deployer V3.1 and IBM Image Construction and Composition Tool V1.1.

**Integration with IBM Workload Deployer**

The new, tighter integration between IBM Workload Deployer and IBM Image Construction and Composition Tool takes place over a cloud provider connection defined in the tool interface. This link allows images in the IBM Workload Deployer image catalog to be viewable from the tool.
Images from the appliance can be “imported” into the tool, effectively creating a representation of the image in the tool. The actual image stays on the appliance. With the image now in the tool, it can be extended by adding software bundles. When an image is extended, a new image description is generated using the imported image as a template, and the software bundles are added to the new image.

Then the new image is synchronized. During the synchronization process, the tool requests that IBM Workload Deployer create a clone of the image and provision that clone to the cloud. Once provisioned, the tool communicates with the cloud image instance and performs the installation and configuration tasks defined by the software bundles. At completion of the software bundle execution, the synchronization is complete and the new virtual image can be tested. The last step is to use the tool to capture the extended image. The capture process cleans up the image to leave it in a state that is appropriate for redeployment. The captured image is defined in the tool, and is added to the catalog of virtual images in IBM Workload Deployer.

Figure 10 illustrates this process.

---

**Figure 10   Customizing IBM Workload Deployer images**

The steps shown in Figure 10 show the process to use IBM Image Construction and Composition Tool to build deployable images to IBM Workload Deployer. The steps are explained as follows:

1. An operating system specialist using IBM Image Construction and Composition Tool imports a base image directly from IBM Workload Deployer. The base image remains on IBM Workload Deployer and a description of the image is stored in IBM Image Construction and Composition Tool.
2. A software specialist defines the software bundles to be installed and configured on the base image. The software bundles contain installation and configuration parameters that define the middleware software to install to the virtual image:
   - **Installation**
     The installation parameters define how the IBM Image Construction and Composition Tool installs software into the virtual image. The parameters are stored as a script that is run by the IBM Image Construction and Composition Tool after the image is built. The installation of software occurs one time, and the software then becomes a permanent part of the image.
   - **Configuration**
     The configuration parameters contain a set of activation tasks that the IBM Image Construction and Composition Tool must perform to the installed software at deployment time.

3. The image builder creates a virtual image by specifying a base image to use and selecting the bundles to install. The image builder customizes the installation and configuration (deploy) parameters for the bundles as needed.
   When this new image is saved, the IBM Image Construction and Composition Tool instructs IBM Workload Deployer to create a copy of the base image, install the bundles, and perform the configuration.

4. The image builder verifies the new system, and then resets (cleans up) and captures the image.

5. The image is now available in the IBM Workload Deployer image catalog and can be included in virtual system patterns and deployed to the cloud.

**Adding new images to IBM Workload Deployer**

Figure 11 illustrates a second scenario for using IBM Image Construction and Composition Tool with IBM Workload Deployer to increase the flexibility of the process of building virtual systems for deployment to the cloud. In this case, the base image is not taken from an existing IBM Workload Deployer image. Instead, the base image is a virtual system that exists outside of the IBM Workload Deployer domain.
The steps shown in Figure 11 on page 19 show the process to add new images to the IBM Workload Deployer image catalog using IBM Image Construction and Composition Tool. The steps are explained as follows:

1. The operating system specialist defines the base operating system image to the IBM Image Construction and Composition Tool. In this case, the image is obtained from one of the following sources:
   - An existing virtual image (OVA file) with an operating system already installed is imported into IBM Image Construction and Composition Tool (Red Hat Enterprise Linux or SUSE Linux Enterprise Server only).
   - A running VMware virtual machine is captured into IBM Image Construction and Composition Tool.

2. The image builder then exports the new image for use with the IBM Workload Deployer. The export process creates an OVA package that contains the metadata necessary for importing the image into and deploying the image from the IBM Workload Deployer.

3. The IBM Workload Deployer administrator imports the image into the appliance.

4. The image builder reimports the image into IBM Image Construction and Composition Tool from the IBM Workload Deployer cloud provider. This action provides a new base image that can be extended.

5. A software specialist defines the software bundles to be installed and configured on the base image. The image builder creates a new virtual image by specifying the base image and selecting the bundles to install. The image builder customizes the installation and configuration (deploy) parameters for the bundles as needed. When this new image is saved, the IBM Image Construction and Composition Tool starts a copy of the base image, installs the bundles, and performs the configuration.

6. The image builder verifies the new system, and then captures the image.

7. The image is now available on the IBM Workload Deployer for inclusion in virtual system patterns.
Plug-in Development Kit

Virtual application patterns take a decidedly application-centric approach for building, deploying, and managing middleware application environments in a cloud. By supplying your application and specifying both functional and non-functional requirements for that application, IBM Workload Deployer transforms the input from you into an installed, configured, and integrated middleware application environment (Figure 12).

![Figure 12 IBM Workload Deployer Virtual application approach](image)

A deployer of virtual application patterns does not need to know how to install, configure, or integrate the middleware and applications because the pattern encapsulates all that knowledge in plug-ins. The plug-ins of each pattern type define components, links, and policies and their functionality in the virtual application pattern (Figure 13).

![Figure 13 IBM Workload Deployer example virtual application pattern](image)
In a virtual application pattern:

- Components represent functional profiles for a workload, such as enterprise application and database.
- Links express a connection point between components in a virtual application pattern. In the pattern shown in Figure 13 on page 21, a link specifies that the Enterprise Application has a dependency on the Database component in the pattern.
- You can use policies like the Scaling Policy shown in Figure 13 on page 21 to specify functional and non-functional requirements for your application environment.

By using IBM Workload Plugin Development Kit, user can create custom plug-ins (such as components, policies, and links) to extend an existing pattern or create a pattern type.

The team who wrote this paper

Zhi Xian Chen is a member of the system integration test team for WebSphere Application Server. She specializes in middleware and web services. Her area of expertise includes several WebSphere products, including IBM DB2 and IBM Rational Application Server. Zhi Xian is also a volunteer lecturer for Web Application Development in colleges and has a number of publications on IBM developerWorks® in China. Zhi Xian has a joint Master of Science degree from the University of Reading (UK), the Aristotle University of Thessaloniki (Greece), and the University Carlos III Madrid (Spain), and a Bachelor's degree from Zhejiang University in China.

Seiichiro Imazeki is an IT Engineer at IBM Japan Systems Engineering Co., Ltd. He has two years of experience in the IT field. He received his Bachelor of Science and Master of Science degrees in Mathematics from Waseda University, Japan. His areas of expertise include web-related technology. He currently provides technical support for WebSphere Application Server and IBM Workload Deployer.

Matthew Kelm is an Advisory Software Engineer for the IBM Software Group Application and Integration Middleware Division at Rochester, MN. His current focus is on delivering middleware solutions on IBM Cloud Platforms, such as IBM Workload Deployer. His experience includes technical roles on the WebSphere Application Server and Business Process Management product development teams. He holds a Bachelor of Science degree in MIS from North Dakota State University and Master of Science degree in Software Engineering from Southern Methodist University.

Simon Kofkin-Hansen is from IBM Australia and is the Global Technology lead at IBM for GTS Rapid Deployment Services. Simon has dealt with automation within the middleware space for over eight years and specifically within the cloud-computing space for the last two years. Simon continues to drive automation within this space in the creation of software bundles and automation assets that can be used within both physical and virtual environments.

Zhi Qiang Kou works at IBM China as a team lead for level 3 product support. His focus is on WebSphere Application Server installation technology. He holds a degree in Computer Science from Tianjin University, China. Before joining IBM CSDL in 2009, he had 9 years of experience in the IT / Telecom field and holds several professional certifications.
Bobby McChesney is an Advisory Software Engineer at IBM Software Group in the Application and Integration Middleware Division with 23 years of experience at IBM. He graduated with a Bachelor of Science in Computer Engineering from the University of Puerto Rico at Mayaguez and a Master of Science in Computer Engineering from Syracuse University. He is an IBM certified application developer and co-author of the Advanced Java EE Development for Rational Application Developer 7.5: Developers’ Guidebook, by McChesney, et al. He currently works on the IBM Workload Deployer product development team in WebSphere Cloud Initiatives. He lives in Austin, Texas.

Carla Sadtler is a Consulting IT Specialist at the ITSO, Raleigh Center. She writes extensively about WebSphere products and solutions. Before joining the ITSO in 1985, Carla worked in the Raleigh branch office as a Program Support Representative, supporting IBM MVS™ customers. She has a degree in Mathematics from the University of North Carolina at Greensboro.

Now you can become a published author, too!

Here’s an opportunity to spotlight your skills, grow your career, and become a published author—all at the same time! Join an ITSO residency project and help write a book in your area of expertise, while honing your experience using leading-edge technologies. Your efforts will help to increase product acceptance and customer satisfaction, as you expand your network of technical contacts and relationships. Residencies run from two to six weeks in length, and you can participate either in person or as a remote resident working from your home base.

Find out more about the residency program, browse the residency index, and apply online at:
ibm.com/redbooks/residencies.html

Stay connected to IBM Redbooks

- Find us on Facebook:
  http://www.facebook.com/IBMRedbooks
- Follow us on Twitter:
  http://twitter.com/ibmredbooks
- Look for us on LinkedIn:
  http://www.linkedin.com/groups?home=&gid=2130806
- Explore new Redbooks publications, residencies, and workshops with the IBM Redbooks weekly newsletter:
- Stay current on recent Redbooks publications with RSS Feeds:
  http://www.redbooks.ibm.com/rss.html
Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user’s responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.
This document REDP-4852-00 was created or updated on July 5, 2012.

Send us your comments in one of the following ways:
► Use the online Contact us review Redbooks form found at: ibm.com/redbooks
► Send your comments in an email to: redbooks@us.ibm.com
► Mail your comments to: IBM Corporation, International Technical Support Organization Dept. HYTD Mail Station P099 2455 South Road Poughkeepsie, NY 12601-5400 U.S.A.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at http://www.ibm.com/legal/copytrade.shtml

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

AIX®
DataPower®
DB2®
developerWorks®
IBM SmartCloud™
IBM®

MVS™
Power Systems™
PowerVM®
Rational®
Redbooks®
Redpaper™

Redbooks (logo) ®
Tivoli®
WebSphere®
z/VM®

The following terms are trademarks of other companies:

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

Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java, and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

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