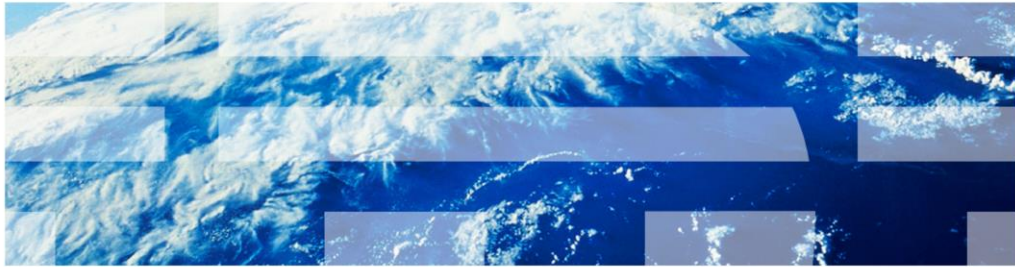# IBM Workload Deployer V3.1

Virtual application pattern policies

The presentation covers policies that you can apply to IBM Workload Deployer virtual applications.

## Table of contents

- Overview
- Policies
- Summary

The agenda for this presentation will provide an overview of the virtual application policy types provided by IBM Workload Deployer V3.1 and configuration and operational details on each.

Section

Overview

Virtual application pattern policies

© 2012 IBM Corporation

This section will provide an overview of the virtual application policies provided with IBM Workload Deployer V3.1.

Overview

- Policies are quality of service that can be applied to an individual component or an entire virtual application
  - JVM policy
  - Scaling policy
  - Routing policy
  - Log policy
- Load balancing and session persistence features use specific shared services
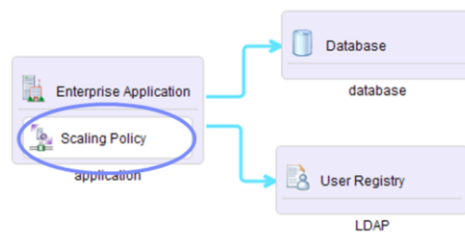
IBM Workload Deployer gives you the ability to augment your virtual applications with different qualities of service by way of policies. Four policies are included with the appliance: JVM, log, routing and scaling. You can apply policies at the virtual application or component level. A policy applied at the component level takes precedence over the same policy applied at the virtual application level.

Policies are represented in the Virtual Application Builder tool by a white box within a component.

To reduce the resource footprint and offer high availability, Workload Deployer allows virtual applications to share services for load balancing and session persistence. Load balancing and session persistence shared services are made available to your application through the use of the routing and scaling policies.
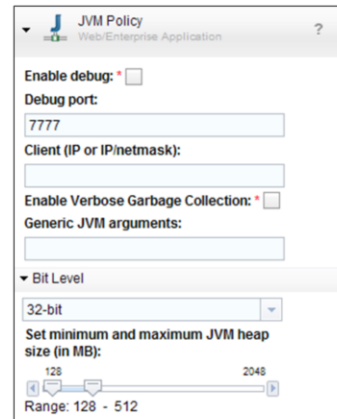
Section

Policies

Virtual application pattern policies

This section will cover the available policies in detail.

JVM policy

- JVM policy can be applied to these component types:
  - Enterprise Application
  - Web Application
  - OSGi Application
- JVM policy surfaces several JVM options
  - Heap size
  - Remote debug
  - Verbose Garbage Collection
  - 32-bit or 64-bit WebSphere® Application Server

Virtual application pattern policies   © 2012 IBM Corporation

You can apply a **JVM policy** to any of the application hosting environments: Enterprise, web and OSGi application components. It allows you to configure minimum and maximum heap size and enable verbose garbage collection and remote debugging. In addition, it allows you to choose 32-bit or 64-bit WebSphere Application Server. This has no impact on the operating system architecture, which is 64-bit.
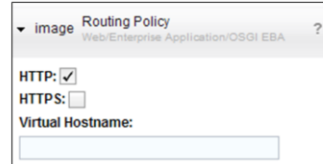
Logging policy

- Logging policy can be applied to these component types:
  - Enterprise Application
  - Web Application
  - OSGi Application
- Logging policy effects application server's startup trace string

Log Policy	?
Web/Enterprise Application	
Log Detail Levels:	
*=info	

You can apply a **logging policy** to any of the application hosting environments: Enterprise, web and OSGi application components.. This policy has a single property that sets the server startup trace string level for the application server. You can change the application server log level after deployment from the deployment inlet.

Routing policy

- Routing policy can be applied to these component types:
  - Enterprise Application
  - Web Application
  - OSGi Application
- HTTP or SSL over HTTP (HTTPS)
- Routing based on virtualhost name
- Shared proxy service must be deployed

The **routing policy** is used in conjunction with scaling policy to route requests to virtual applications based on a unique virtual host name. Communication to the application server can be limited to HTTP, HTTPS, or both.

When a virtual machine in the web application completes its initialization, it registers itself with the shared proxy service using the specified virtual host name. The proxy service then begins routing requests for the host name to the new server instance.
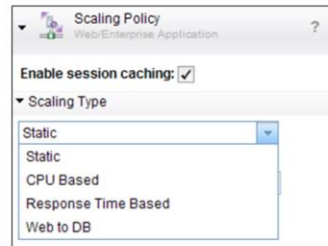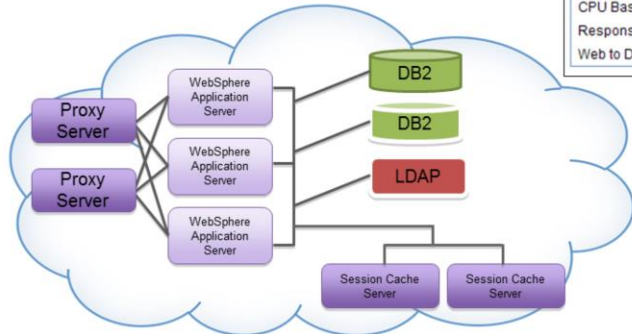
The routing policy is normally used in conjunction with a scaling policy, but can be applied without one.

If a virtual application includes a routing policy, an instance of the shared proxy service must be started within the same cloud group. If a shared proxy service is not deployed to the target cloud group the virtual application will not deploy.

You can apply a **routing policy** to any of the application hosting environments: Enterprise, web and OSGi application components.

The s**caling policy** gives you memory-to-memory session persistence, load balancing and dynamic scaling of your environment's virtual machines based on workload demand.

Dynamic scaling adds or removes virtual machines based on workload demand. When to dynamically scale up or down is based on certain triggers. These triggers are grouped by application scenario. Application scenarios group triggers that make sense in a given application architecture.

IBM Workload Deployer version 3.1 supports four application scenarios:

> **Static** disables the elastic scaling feature.
>
> **CPU Based** provides scaling based on processor usage.
>
> **Response Time Based** provides scaling based only on response times.
>
> **Web to DB** scaling is based on JDBC connection usage.

The **"Static"** application scenario is unique in that it disables elastic scaling. The number of instances that you defined in "Initial instance number" field will remain static no matter the workload demand.

If the scaling policy is enabled then a routing policy should also be enabled so the shared proxy service can route requests to servers in the web application as they scale in or out.

You can apply a **Scaling policy** to the any of the application hosting environments: Enterprise, web and OSGi Application components.

Scaling policy – CPU Based

- Scaling based on processor usage

The **"CPU Based"** application scenario scales based on processor usage within a defined threshold. If processor usage falls below the threshold for the specified time, virtual machines are scaled in; if processor usage exceeds the threshold the VMs are scaled out.

Scaling policy – Response Time Based

- Scaling based on web response times

The **"Response Time Based"** application scenario scales based on response times for web-based requests. If response times fall below the threshold for the specified amount of time, virtual machines are scaled in; if response times exceed the threshold then the VMs are scaled out.

Scaling policy – Web to DB

- Scaling based on:
  - Web response times
  - JDBC connection wait times
  - JDBC connection usage

Enable session caching: ✓

▼ Scaling Type

Web to DB

Scaling in/out when Web response time is out of threshold range(ms):

0   10000

Range: 1000 - 5000

Or when JDBC connections wait time is out of the threshold range(ms):

0   10000

Range: 1000 - 5000

Or when JDBC connection pools usage is out of the threshold range(%):

1%   100%

Range: 20% - 80%

Instance number range of scaling in/out:

1   50

Range: 1 - 10

Minimum time (sec) to trigger add/remove:

120

The **"Web to DB"** application scenario scales based on response times, JDBC connection wait times and JDBC connection usage. If response times, JDBC wait times or JDBC connections fall below defined thresholds virtual machines are scaled in and if thresholds are exceeded then VMs are scaled out.

JDBC wait time is the amount of time WebSphere Application Server waits for responses from your database.

JDBC connection usage is the number of connections that are used from the connection pool.

Scaling policy – shared caching service

- Enable session caching
  - Replicate HTTP session data to shared service
  - Shared caching service must be deployed to cloud group
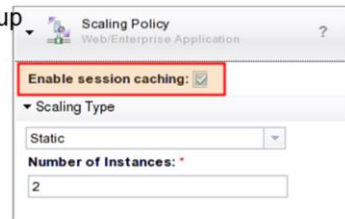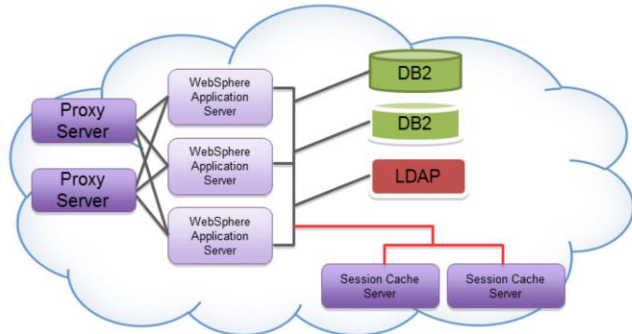
13   Virtual application pattern policies   © 2012 IBM Corporation

All of the scaling types provide the ability to use a shared service to replicate HTTP session data for high availability. If **"Enable session caching"** is checked then the shared caching service is used for HTTP session storage. If one of the virtual machines in your virtual application fails, requests can fail-over to another virtual machine which can retrieve the session data from the shared caching service.

If a virtual application enables session caching, an instance of the shared caching service must be started within the same cloud group. If a shared caching service is not deployed to the target cloud group the virtual application will not deploy.

Summary

Virtual application pattern policies © 2012 IBM Corporation

This section gives a brief summary.

Summary

- Policies covered:
  - JVM
  - Log
  - Routing
  - Scaling

Virtual application pattern policies

The following policies were covered: JVM, Logging, Routing and Scaling. By far the most involved policy is the Scaling policy with it's numerous options and use of shared services: caching and proxy.

Trademarks, disclaimer, and copyright information