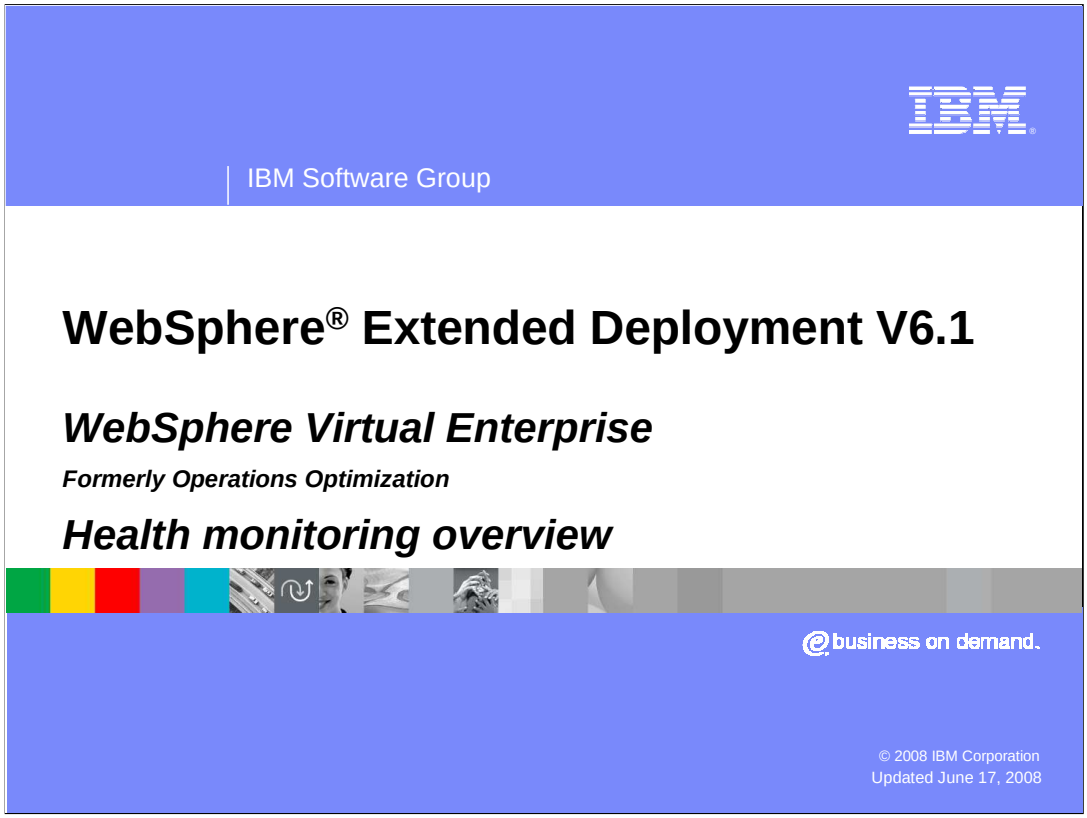

This presentation will cover the health monitoring capabilities of WebSphere Extended Deployment.

This module was originally recorded for WebSphere Extended Deployment Operations Optimization, which is now called WebSphere Virtual Enterprise. Though the module uses the previous names, the technical material covered is still accurate.

Agenda

- Health monitoring overview
- Health conditions
- Reaction lists
- Summary

The presentation will give an overview of WebSphere Extended Deployment's health monitoring capabilities, then will cover health conditions and actions lists.

Overview

- A WebSphere Extended Deployment environment can be monitored for various software health conditions
- Can monitor other middleware servers for most conditions
- Health conditions to monitor are defined by health policies
- Action lists are performed when policy is breached

WebSphere Extended Deployment's health monitoring features can monitor your environment for several common software health conditions. Some conditions, such as memory leaks, slow response times, or an excessive percentage of requests timing out, indicate a current problem in an application server. Other conditions, such as age or work performed, can be used to prevent problems in a server hosting an application with known minor errors. The conditions to monitor and the actions to take when they are encountered are defined in health policies.

WebSphere Extended Deployment 6.1 expands the ability to monitor health-related conditions beyond the WebSphere cell to other middleware server types.

Health policies

- A health policy defines
  - Unacceptable runtime behavior for an application server
  - Corrective actions to be taken when this behavior is violated
  - The set of servers to monitor

A health policy is a configuration object that defines a health condition to monitor, an action to be taken when the condition is breached, and the set of servers to monitor for the condition.

Health conditions

- Age-based
  - Target has been running longer than *X* hours or days
- Workload
  - Target has served more than *X* requests
- Excessive request timeout
  - At least *X*% of requests to the target time out
- Excessive response time
  - Average response time is greater than *X*

Several common health conditions can be monitored. The **age-based** condition generates an event when the target server has been running for longer than a specified number of hours or days. Similarly, an event can be generated after the target server has served more than a specified number of requests. Use these conditions to prevent problems in applications with known minor errors that surface after the server has been running for some period. The **excessive response timeout** condition triggers an event when at least a specified percentage of requests to the target server time out. And the **excessive response time** condition can monitor for a server that has an average response time greater than a specified value.

Health conditions

- Excessive memory usage
  - Java™ heap is *X*% full for longer than *Y* minutes
- Memory leak
  - Looks for consistent decrease in free memory
  - Three detection levels
    - Slower detection speeds reduce chance of false alarms
- Storm drain
  - Detects workload imbalance based on response times and cluster weights
    - Example: an error condition is causing a server to return requests immediately

The **excessive memory usage** condition monitors the Java heap and generates an event when the heap is more than a specified percent full for more than the specified time.

The **memory leak** condition looks for a trend of increasing memory usage, rather than just a heap that is more than a certain amount full. The detection level setting determines when these trends are detected. To more accurately detect a true leak, the system must wait for a longer memory growth pattern to develop. The three available detection levels give you the choice of balancing accuracy against detection speed.

The **storm drain** condition can detect a workload imbalance, and identify a server that is responding to requests more quickly than it should, possibly because an error condition is causing requests to return instantly.

Middleware server support

- Supported for all server types
  - Age-based
  - Workload
  - Excessive response time
  - Excessive request timeout
  - Storm drain
  - Custom policy
- Supported for WebSphere servers only
  - Excessive memory usage
  - Memory leak

The excessive memory usage and memory leak conditions are only available for servers that are running WebSphere Application Server or WebSphere Extended Deployment. The other conditions are available for all server types supported by WebSphere Extended Deployment.

Custom health policies

- You know your environment best
  - Trigger points to monitor
- Health policy custom conditions
  - Create custom health expressions monitor
- Create custom health condition definitions using
  - wsadmin createHealthPolicy AdminTask
  - Custom Health Policy Wizard
    - wsadmin AdminTask createHealthPolicy –interactive
  - **Administrative console support**

New in V6.1.0.1

V6.1 introduces the concept of a custom health policy that allows you to define a custom condition when the predefined health conditions do not fit your needs. Using a custom health policy lets you create expressions to define what “unhealthy” means in your environment, rather than having WebSphere Extended Deployment define it. In version 6.1, custom health condition definitions can only be created using the createHealthPolicy administrative task. V6.1.0.1 adds the ability to define custom health policies from the administrative console.

Custom condition operands

- WebSphere Extended Deployment supported PMI modules
- On demand router metrics
  - server level
  - cell level
- MBean invocations
  - operation metric
  - attribute metric
- URL return code metric

Custom health conditions can monitor metrics from a subset of PMI modules and server-level and cell-level metrics published by the on demand router. Custom health conditions can also invoke an MBean or query its attributes. In addition, you can ping any relative path, or Uniform Resource Identifier (URI), on the server that is the target of this policy.

Custom conditions are supported on all platforms, but with varying levels of support. For example, metrics from an on demand router can be used for a health policy for all server types. However, a custom health condition for a non-WebSphere server can not make use of the WebSphere PMI server metrics or MBeans.

Health management monitor reaction

- Reactions
  - Server restart (all conditions)
  - Dump threads (some conditions)
  - Dump JVM heap
    - For memory leak condition, IBM Virtual Machines for Java Platforms only
  - Place server in maintenance mode
  - Custom actions

For any health condition, you can specify that a server should be restarted. For certain conditions, you can specify that a Java thread dump or heap dump should be generated.

You can also place a server into maintenance mode and take it out of maintenance mode.

Maintenance mode is used to perform diagnostics, maintenance, or tuning on a node or server without disrupting incoming traffic. Putting a server into maintenance mode allows requests on the server to complete, but instructs the on-demand router to stop sending new requests to that server. When placing a server into maintenance mode you have the option of maintaining or breaking session affinity. When maintaining affinity, requests that have an open session on the server are routed to the server until the session ends or times out. Breaking affinity will force the session to fail over to another server immediately.

After the server has reached a healthy state, it can be reinstated to serve requests.

For example, if the health monitor detects excessive request timeouts, you might want to gather process dumps and traces to help diagnose the problem. You can accomplish this by placing the server into maintenance mode, running a custom health action to gather the information you need, then restarting the server. Once the server is restarted you can take the server out of maintenance mode.

You can specify multiple actions for any health policy, and the actions are run sequentially. One action must run to completion before the next action in the list starts.

Custom actions

- May need specific actions to occur when an unhealthy situation is detected.
- Extended Deployment allows customers to define custom actions in addition to a standard set Extended Deployment provides
- Allows custom action plan to be carried out when the unhealthy situation detected.

If the provided health actions do not meet your needs, you can create a custom health action. With a custom action, you define an external program to run when the health condition breaches.

A custom action can be an executable Java JAR or class file, shell scripts on UNIX platforms, batch files on Windows, or any executable file in the target environment.

You must define custom actions before you create the health policy that contains the custom actions.

Reaction modes

- Automatic
  - Action list is run immediately when policy is breached.
- Supervised
  - Health controller creates a task with the task management interface
  - Operator must explicitly approve before action list is run
  - Operator can deny the task or take manual action

The reactions can take place automatically as soon as the health policy breach is detected, or you can request to be notified first, with the ability to approve or deny the action.

When a health policy reaction mode is set to supervised, recommendations on actions are sent to the administrator through the task management interface when a policy is breached. If the administrator follows the recommendations, the only action required is selecting a button, and the actions are performed.

Cluster restart constraints

- Static clusters: keep at least one member up
- Dynamic clusters: placement controller interaction
  - Restart in place
  - Start new instance, stop old instance (placement controller determines location)
  - Just stop old instance (if demand is satisfied by remaining servers)
- Dynamic clusters, if placement controller disabled:
  - Keep minimum number of instances up

When a health action is “restart server”, the health controller attempts to ensure the server restart takes place with little or no loss of service. If the server being restarted is a member of a static cluster, the health controller will ensure that at least one other cluster member is running before restarting the unhealthy server. If a cluster only has one cluster member, or only one cluster member is running, then the server is not restarted.

When the server is a member of a dynamic cluster, the health controller will coordinate with the application placement controller to ensure that an adequate number of instances of the dynamic cluster are started to meet current demand. If the placement controller is disabled, the health controller will ensure that the number of running instances does not fall below the minimum value specified in the dynamic cluster definition.

Members to be monitored

- Single servers
- Static clusters
- Dynamic clusters
- Cluster members
- Entire cell

The monitoring targets for a health policy can be individual servers, static or dynamic clusters, individual cluster members, or everything within your cell.

Summary

- Health monitoring in WebSphere Extended Deployment enables actions to be taken automatically when certain software conditions are detected
- Can relieve administrative burden when dealing with applications that have minor problems
- Non-WebSphere nodes are supported
- Custom policies and actions are supported

In summary, the health monitoring features of WebSphere Extended Deployment can watch for certain software health conditions, and automatically take corrective or preventive action when such a condition is detected. These features can help reduce the administrative burden of managing applications that have minor problems.

Trademarks, copyrights, and disclaimers

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

IBM WebSphere

Java, JVM, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements or changes in the products or programs described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

Information is provided "AS IS" without warranty of any kind. THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (for example, IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. 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 in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products.

IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

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

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.