

Cluster Dispatcher

USERS' GUIDE

Xiaowei Wang

V1.0

2019-06-04

Table of Contents

Overview	3
Design	4
Architecture.....	4
Cluster Recognition.....	4
Dispatch Mode	5
Service Dispatch	5
Failure Signal	5
Configuration	6
Setup Hostname	6
Setup Cluster Nodes	6
Setup Remote Servers.....	6
Self-Checking	6
Service Dispatch Management	7
List Service Dispatch	7
Add Service Dispatch	7
View Service Dispatch.....	7
Edit Service Dispatch.....	8
Delete Service Dispatch	8
Enable Service Dispatch	8
Disable Service Dispatch	8
Keep in Mind	9
Clone-able Pipeline	9
Recursive Dispatch.....	9
Hidden Service Dispatch Tasks	9

Overview

To build a high availability environment, multiple webMethods Integration Server usually are configured as a cluster. It's expected to have all Integration Server to function exactly the same. If something is changed on one node, for example a customized configuration file, system administrator usually has to synchronize it across all the nodes manually.

Cluster Dispatcher is designed to dispatch service invocation in a cluster, which means if a service is invoked on one node, it would be invoked on all other nodes as well automatically. Cluster Dispatcher is a non-invasive solution, so the users don't have to change their service or to invoke any service explicitly.

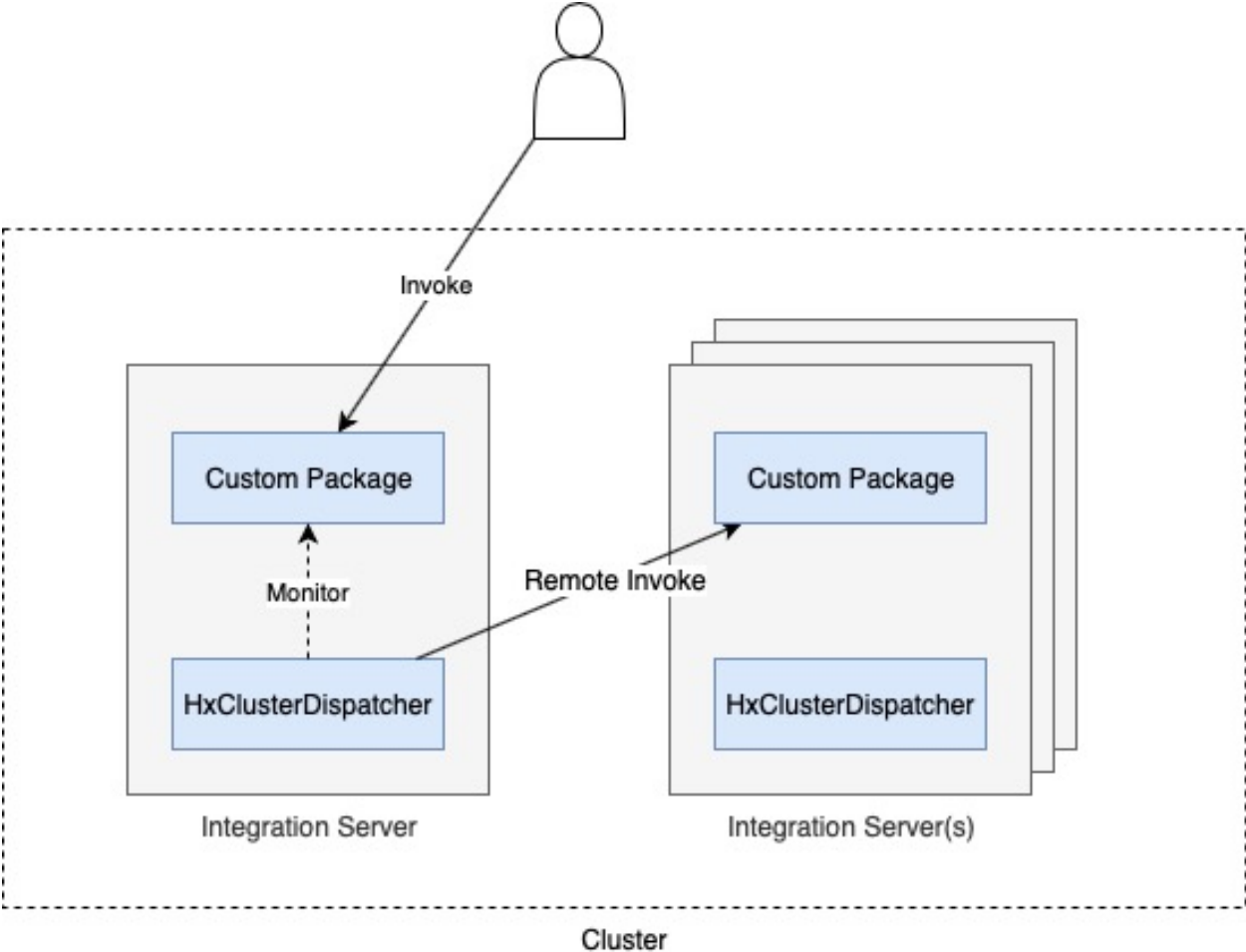
By using Cluster Dispatcher, the users could create Service Dispatch task for specific target service, for example a service to save customized configuration file. Once the target service is invoked on one node, it would be invoked on all other nodes with the same inputs, which makes the configuration file synchronized.

Cluster Dispatcher is released as a standalone Integration Server package HxClusterDispatcher, but it will rely on some Integration Server features such as global variables and remote servers. Cluster Dispatcher is built on webMethods 9.9 and compiled by Java 1.8.

Design

Architecture

Cluster Dispatcher would monitor service execution once it's registered as a Service Dispatch task. Cluster Dispatcher would dispatch this service execution on other nodes by remotely invoke the same service with the same input.



Cluster Recognition

The nodes in cluster is configured as Integration Server property "watt.server.cluster.aliasList". The node name of current Integration Server is configured by global variable "HxClusterDispatcher.hostname". The name of all other nodes are calculated based on them. Because the service is dispatched by remote invoke, so remote server alias has to be configured for each of all other nodes, and make sure the remote server alias has the same value of node name.

Unlike the webMethods' official definition of cluster, the nodes don't have to configured as a real cluster by sharing the same database and connecting to the same Terracotta Server Array, although that're recommended.

Dispatch Mode

Cluster Dispatcher supports two different dispatch mode Strict and Lax. For mode Strict, the service has to be invoked successfully on all nodes, otherwise a rollback service would be invoked, if there is one, on all successful nodes (including itself) to reverse the invocation. For mode Lax, the service will be invoked on as many nodes as possible if it's successful on local, no rollback service would be invoked.

Service Dispatch

When a service is invoked, the dispatching would happen only when all conditions below are met:

1. There are at least one other nodes configured in property "watt.server.cluster.aliasList".
2. The service is registered as target service in a Service Dispatch task.
3. The corresponding Service Dispatch task is enabled.
4. The service is invoked as a top level service if corresponding Service Dispatch task is configured as top service only.
5. The service is not invoked by Cluster Dispatcher on other nodes, which is recognized by a Dispatch Flag Field.
6. The service is invoked successfully on local without any failure signal.

Failure Signal

The service dispatch is not guaranteed, and there could be errors during the dispatching on local or on any other nodes.

A thrown exception would be definitely considered as a failure signal, or optional conditional expression could be configured for Service Dispatch task, which would follow the same pattern as evaluated label in Flow service. The conditional expression would be evaluated based on pipeline after service execution. Once the evaluated result is true, it would be considered as a failure signal.

When target service is invoked:

- If a failure signal is found on local, the service dispatching won't happen on other nodes. A rollback service would be invoked on local if there is one and dispatch mode is Strict.
- If a failure signal is found on remote node, the whole process would be marked as failure. A rollback service would be invoked on both local and all successful nodes if there is one and dispatch mode is Strict.

When rollback service is invoked:

- If a failure signal is found, no matter on local or on remote node, this failure would be logged if logging service is configured.

Configuration

Cluster Dispatcher has to be configured properly first, then it could function as expected.

Setup Hostname

Hostname would be used to calculate name of other nodes in cluster. You could setup hostname by following instruction below:

1. Login Integration Server Administrator UI via browser.
2. Click "Solutions" -> "HX Cluster Dispatcher...".
3. In the opened new window/tab, click "Setup Hostname" under "Management" menu item.
4. On the right side, create or edit global variable "HxClusterDispatcher.hostname" with proper value, for example "node1".

Setup Cluster Nodes

Cluster nodes would be used to calculate name of other nodes in cluster. You could setup hostname by following instruction below:

1. Login Integration Server Administrator UI via browser
2. Click "Solutions" -> "HX Cluster Dispatcher...".
3. In the opened new window/tab, click "Setup Cluster Nodes" under "Management" menu item.
4. On the right side, if you didn't see extended setting "watt.server.cluster.aliasList", click "Show and Hide Keys" and make it visible.
5. Click "Edit Extended Settings" and provide proper value (comma separated alias list), for example "node1,node2".

Setup Remote Servers

Remote servers would be used to dispatch the service invocation. You could setup remote servers by following instruction below:

1. Login Integration Server Administrator UI via browser.
2. Click "Solutions" -> "HX Cluster Dispatcher...".
3. In the opened new window/tab, click "Setup Remote Servers" under "Management" menu item.
4. On the right side, create remote server alias for each other nodes in cluster. Make sure the remote server alias name is the same as cluster node name.

Self-Checking

Because it's a little bit complicate to configure Cluster Dispatcher, so it provides self-checking feature to make sure it's ready for serve. You could do self-checking by following instruction below:

1. Login Integration Server Administrator UI via browser.
2. Click "Solutions" -> "HX Cluster Dispatcher...".
3. In the opened new window/tab, click "Service Dispatch List" under "Management" menu item.
4. On the right side, click "Self-Check" then you could see the self-checking status soon. Make sure all status is OK.

Service Dispatch Management

A service has to be configured as a target service of Service Dispatch task, then it could be monitored and dispatched by Cluster Dispatcher. One service could only be configured as one Service Dispatch task. If a service is configured twice, the original task would be replaced by new one.

List Service Dispatch

You could list all Service Dispatch tasks by following instruction below:

1. Login Integration Server Administrator UI via browser.
2. Click "Solutions" -> "HX Cluster Dispatcher...".
3. In the opened new window/tab, click "Service Dispatch List" under "Management" menu item.
4. On the right side, you would see all existing Service Dispatch tasks.


Add Service Dispatch

You could add a new Service Dispatch task by following instruction below:

1. Login Integration Server Administrator UI via browser.
2. Click "Solutions" -> "HX Cluster Dispatcher...".
3. In the opened new window/tab, click "Service Dispatch List" under "Management" menu item.
4. On the right side, click "Create" link.
5. Fill the form to configure Service Dispatch task properly.
 - a. Target Service, the name of service to be monitored and dispatched.
 - b. Dispatch Mode, please refer to section Dispatch Mode.
 - c. Top Service Only, whether Service Dispatch task would be triggered only for top level service.
 - d. Enabled, whether Service Dispatch task is enabled.
 - e. Dispatch Flag Field, a string field would be inserted for remote invoke, which would be used to recognize whether it's invoked by Cluster Dispatcher on other nodes. Make sure this field won't make any conflict with service input. By default, it's "\$clusterDispatchingInitiator".
 - f. Failure Signal Expression, an optional conditional expression to evaluate whether the target service is invoked successfully.
 - g. Logging Service, an optional service to log related Service Dispatch task. The logging service has to have the same service signature with "pub.flow:debugLog". By default, service "pub.flow:debugLog" would be invoked.
 - h. Log Pipeline, whether to log service input and output.
 - i. Rollback Service, an optional service to reverse service invocation. It would be invoked on local and any successful nodes if a failure signal is found on any other nodes.
 - j. Rollback Failure Signal Expression, an optional conditional expression to evaluate whether the rollback service is invoked successfully.
6. Click "Create" button.


View Service Dispatch

You could view Service Dispatch task detail by following instruction below:

1. Login Integration Server Administrator UI via browser.
2. Click "Solutions" -> "HX Cluster Dispatcher...".
3. In the opened new window/tab, click "Service Dispatch List" under "Management" menu item.
4. On the right side, click  icon in the same row of Service Dispatch to view.


Edit Service Dispatch

You could add a new Service Dispatch task by following instruction below:

1. Login Integration Server Administrator UI via browser.
2. Click "Solutions" -> "HX Cluster Dispatcher...".
3. In the opened new window/tab, click "Service Dispatch List" under "Management" menu item.
4. On the right side, click  icon in the same row of Service Dispatch to edit.
5. Fill the form to edit Service Dispatch task properly.
6. Click "Update" button.

Delete Service Dispatch

You could delete an existing Service Dispatch task by following instruction below:

1. Login Integration Server Administrator UI via browser.
2. Click "Solutions" -> "HX Cluster Dispatcher...".
3. In the opened new window/tab, click "Service Dispatch List" under "Management" menu item.
4. On the right side, click  icon in the same row of Service Dispatch to delete.
5. Click "OK" on popup prompt.

Enable Service Dispatch

You could enable a disabled Service Dispatch task by following instruction below:

1. Login Integration Server Administrator UI via browser.
2. Click "Solutions" -> "HX Cluster Dispatcher...".
3. In the opened new window/tab, click "Service Dispatch List" under "Management" menu item.
4. On the right side, click link "false" at column "Enabled" in the same row of Service Dispatch to enable.
5. Click "OK" on popup prompt.

Disable Service Dispatch

You could disable an enabled Service Dispatch task by following instruction below:

1. Login Integration Server Administrator UI via browser.
2. Click "Solutions" -> "HX Cluster Dispatcher...".
3. In the opened new window/tab, click "Service Dispatch List" under "Management" menu item.
4. On the right side, click link "true" at column "Enabled" in the same row of Service Dispatch to disable.
5. Click "OK" on popup prompt.

Keep in Mind

Clone-able Pipeline

To invoke a service remotely, the pipeline has to be clone-able. So do not try to dispatch service with a non-serializable data in pipeline.

Recursive Dispatch

Improper configuration would cause recursive dispatch, and it won't end.

Hidden Service Dispatch Tasks

To synchronize Service Dispatch tasks, there are 4 pre-defined Service Dispatch tasks which are hidden from users.