

HP Operations Orchestration

Software Version: 10.21

Windows and Linux Operating Systems

Converting Flows using Multi-instance Steps from HP OO 9.0x to HP OO 10.x Format

Document Release Date: February 2015

Software Release Date: February 2015



Legal Notices

Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

Restricted Rights Legend

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright Notice

© Copyright 2005-2015 Hewlett-Packard Development Company, L.P.

Trademark Notices

Adobe™ is a trademark of Adobe Systems Incorporated.

Microsoft® and Windows® are U.S. registered trademarks of Microsoft Corporation.

UNIX® is a registered trademark of The Open Group.

This product includes an interface of the 'zlib' general purpose compression library, which is Copyright © 1995-2002 Jean-loup Gailly and Mark Adler.

Documentation Updates

The title page of this document contains the following identifying information:

- Software Version number, which indicates the software version.
- Document Release Date, which changes each time the document is updated.
- Software Release Date, which indicates the release date of this version of the software.

To check for recent updates or to verify that you are using the most recent edition of a document, go to: <http://h20230.www2.hp.com/selfsolve/manuals>

This site requires that you register for an HP Passport and sign in. To register for an HP Passport ID, go to: <http://h20229.www2.hp.com/passport-registration.html>

Or click the **New users - please register** link on the HP Passport login page.

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HP sales representative for details.

Support

Visit the HP Software Support Online web site at: <https://softwaresupport.hp.com/>

This web site provides contact information and details about the products, services, and support that HP Software offers.

HP Software online support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the support web site to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HP support contacts
- Review information about available services
- Enter into discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and sign in. Many also require a support contract. To register for an HP Passport ID, go to:

<http://h20229.www2.hp.com/passport-registration.html>

To find more information about access levels, go to:

http://h20230.www2.hp.com/new_access_levels.jsp

HP Software Solutions Now accesses the HPSW Solution and Integration Portal Web site. This site enables you to explore HP Product Solutions to meet your business needs, includes a full list of Integrations between HP Products, as well as a listing of ITIL Processes. The URL for this Web site is

<http://h20230.www2.hp.com/sc/solutions/index.jsp>

Contents

Background	4
Accessing Step and Global Context from a Multi-Instance Branch	4
How Steps with List Appender Operations are Converted	5
Using List Appender to Aggregate Branch Results	5
How Bad Practice Usages are Converted	8
Bad Practice Usages Automatically Converted to Best Practice by CUU	8
Bad Practice Usages That Require Manual Editing After Conversion	9
Special Cases	11

Background

The architecture of multi-instance steps has changed in HP OO 10.

The changes include:

- Steps that run within multi-instance step branches (internal steps) have access only to branch context and not to the overall step and global contexts of the flow. This affects, among other things, the work of **List Appender** operations used to aggregate branch results.
- Usages that were considered bad practice in HP OO 9.0x are not supported in HP OO 10.x, as follows:
 - Direct transitions from the internal steps of a multi-instance step to return steps are not supported. Only the **Group-done** transition of the multi-instance steps can be used for this purpose.
 - Parallel steps cannot be used as internal steps in a multi-instance step.
 - Other multi-instance steps cannot be used as internal steps in a multi-instance step.

Accessing Step and Global Context from a Multi-Instance Branch

In HP OO 10.x, there is a separation between the flow context and the branch contexts of multi-instance and parallel steps. When variables are put into the context from internal steps of a multi-instance step, these variables do not reach the flow context unless a special step scriptlet is added to the multi-instance step, which merges these variables from branch to flow context.

How Steps with List Appender Operations are Converted

Using List Appender to Aggregate Branch Results

It is common practice in HP OO 9.0x to use the **List Appender** content operation to aggregate the results of all branches of a multi-instance step, in order to handle them later on in the flow.

Since **List Appender** works by appending values to a variable located in the flow context, in HP OO 10.x by default this variable will not exist in the flow context after the **Group-done** transition of a multi-instance step. However, the step scriptlet of the multi-instance step can be used to merge the contents of each branch context into the flow (step or global) context.

For example:

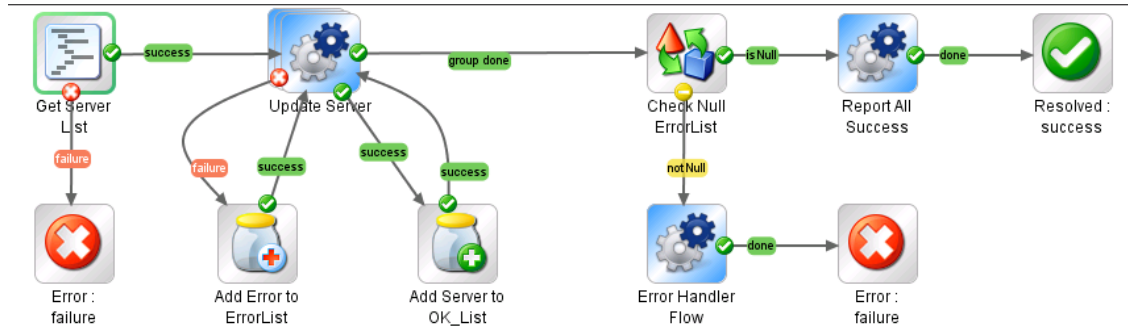
```
var appenderKeyName = scriptletBranchContext.get("appenderKeyName");
var appendedValue = scriptletBranchContext.get("appendedValue");
var delimiter = scriptletBranchContext.get("delimiter");
var currentVal = scriptletContext.get(appenderKeyName);
if(currentVal == null) {
  scriptletContext.put(appenderKeyName, appendedValue);
} else {
  scriptletContext.put(appenderKeyName, currentVal + delimiter +
  appendedValue);
}
```

Note: This scriptlet runs once at the end of each branch. The "appenderKeyName", "appendedValue" and "delimiter" variables should be created by the step scriptlet of the corresponding **List Appender** step from its respective inputs.

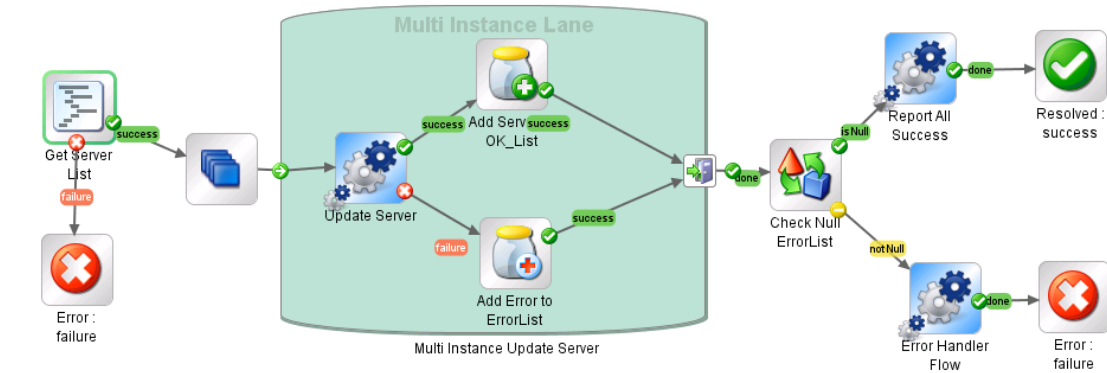
Important: In HP OO 10.21 and later, the Content Upgrade Utility creates similar scriptlets automatically. Therefore a 9.0x flow using **List Appender** to aggregate the results of multi-instance branches will work well after conversion to 10.x format, without a need for further editing, provided that only one **List Appender** instance runs per branch. (See [Special Cases](#)).

Examples

9.0x flow before conversion:



10.x flow after conversion:



Scriptlet added to the **Add Server to OK_List** step:

```
Inspector
Step Name: Add Server to OK_List
Inputs Results Display Description Advanced Scriptlet
1 // Generated by Content Upgrade Utility, do not remove:
2 scriptletContext.put("keyName_cuuGenerated",keyName);
3 scriptletContext.put("delimiter_cuuGenerated",delimiter);
4 scriptletContext.put("val_cuuGenerated",resultText);
```

Note: A similar script is added to the **Add Server to ErrorList** step. Since only one of these steps is executed for each branch of the multi-instance step, the corresponding variables put on the scriptletContext will not collide. Each **List Appender** step can have its own keyName, delimiter and appended value.

Scriptlet added to the multi-instance step:

```
Inspector
Step Name: Multi Instance Update Server
Inputs Results Display Description Advanced Scriptlet
1 //This scriptlet is for Multi-Instance step migration!
2 //It will merge the Global Contexts of each child Multi-instance branch with the current flow global context
3 global_names = scriptletBranchContext.getGlobalVariableNames();
4 var curName;
5 var curVal;
6 for (var i in global_names){
7 curName = global_names[i];
8 curVal = scriptletBranchContext.get(curName);
9 scriptletContext.putGlobal(curName,curVal);
10 }
11 // Generated by Content Upgrade Utility, do not remove:
12 keyName = scriptletBranchContext.get("keyName_cuuGenerated");
13 val = scriptletBranchContext.get("val_cuuGenerated");
14 delimiter = scriptletBranchContext.get("delimiter_cuuGenerated");
15
16 currentVal = scriptletContext.get(keyName);
17 if(currentVal == null) {
18     scriptletContext.put(keyName, val);
19 }
20 else {
21     scriptletContext.put(keyName, currentVal + delimiter + val);
22 }
```

- The fragment highlighted in red is generic for all multi-instance steps. It merges all variables from the branch context to the global context.
- The fragment highlighted in blue is specific for multi-instance steps using **List Appenders**.

How Bad Practice Usages are Converted

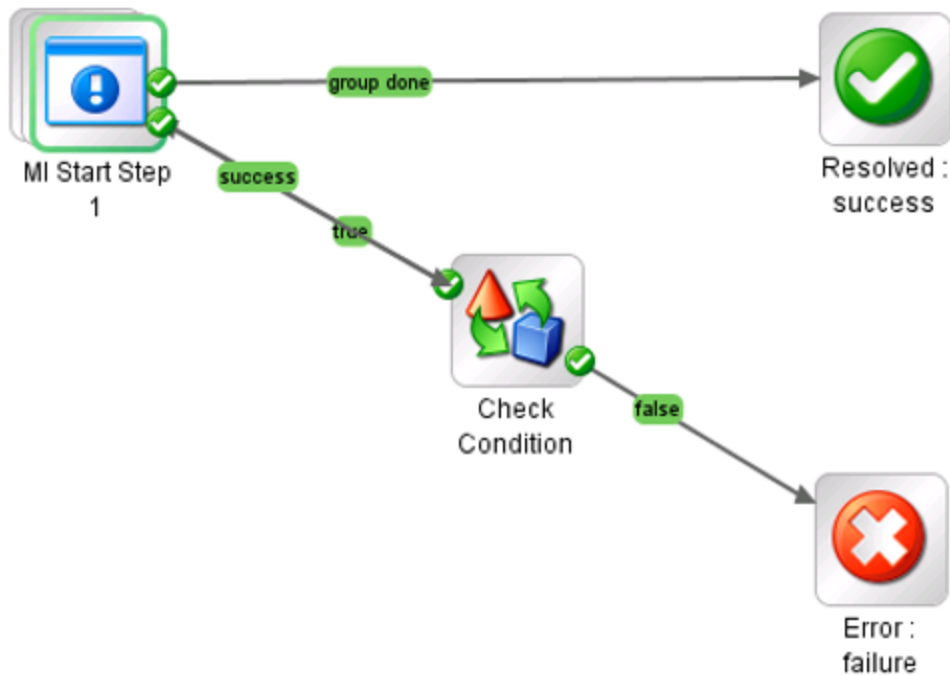
Bad Practice Usages Automatically Converted to Best Practice by CUU

HP OO 10.x does not support direct transitions from internal steps in a multi-instance step to return steps (**Resolved**, **Error**, **Diagnosed** and **No Action Taken**). However, starting from HP OO 10.21, the Content Upgrade Utility can convert multi-instance steps with direct transitions to **Error** steps into a format compatible with HP OO 10.x.

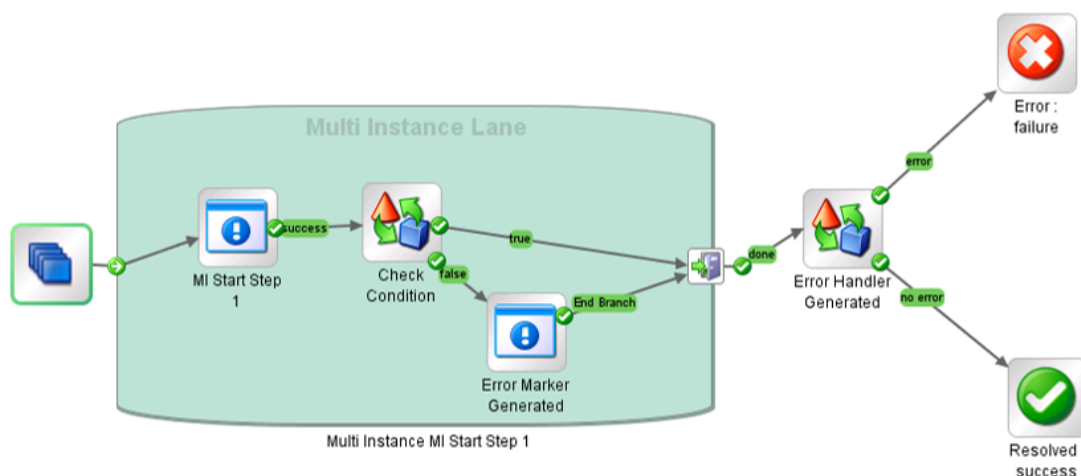
To accomplish this, the CUU creates a new internal **Error Marker** step for each multi-instance step with such a direct transition. This step is based on the **Do Nothing** operation and has a scriptlet that puts a variable on the branch context, marking that an error transition should be forced after the **Group-Done** transition. Additionally, a new **Error Handler** step is added after the **Group-done** transition, which checks this variable, and if it exists, chooses a transition to an **Error** step.

Examples:

9.0x flow before conversion:



10.x flow after conversion:



Bad Practice Usages That Require Manual Editing After Conversion

Direct transitions from internal steps of a multi-instance step to return steps (other than Error)

Such flows cannot be converted automatically to 10.x. The corresponding transition will be removed by the CUU and the resulting flow will be placed under the **Invalid Content** sub-folder in the CUU output folder.

Solution: Consider adding a new transition to the **Group-done** step of the multi-instance step, instead of the removed one.

Parallel steps used as internal steps of a multi-instance step

HP OO 10.x, does not support multi-instance steps using a parallel step as an internal step. Such 9.0x flows cannot be converted automatically to 10.x. The corresponding transition will be removed by the CUU and the resulting flow will be placed under the **Invalid Content** sub-folder in the CUU output folder.

Solution: To make the flow valid in HP OO 10.x, you have options:

- Wrap each internal parallel step in a sub-flow
- Refactor the flow, so that the parallel step is executed after the **Group-done** transition of the multi-instance step

Other multi-instance steps used as internal steps of a multi-instance step

HP OO 10.x does not support multi-instance steps having another multi-instance step as an internal step. The corresponding transition will be removed by the CUU and the resulting flow will be placed under the **Invalid Content** sub-folder in the CUU output folder.

Solution: To make the flow valid in HP OO 10.x, you have two options:

- Wrap each internal multi-instance step in a sub-flow
- Refactor the flow, so that the second multi-instance step is executed after the **Group-done** transition of the first one

Special Cases

Direct transitions to distinct Error steps from a single multi-instance step

Direct transitions to a single **Error** step from within a multi-instance branch can now be converted automatically to 10.x format by the CUU. However, in the case of direct transitions to multiple distinct **Error** steps from a single multi-instance branch, the flow will still be placed under the **Invalid Content** sub-folder and might require manual editing. The reason is that some **Error** steps may become orphan (without any incoming transitions) after the conversion.

Multiple List Appender steps used in the same branch of a multi-instance step

After conversion, such flows may behave differently from the way they behaved in 9.0x. The reason is that the scripts added by the CUU take into account the common case when a single instance of a **List Appender** is used in every branch (in the example above, either **Error Appender** or **Success Appender**, but not both).

