

# Updating Messages with Change Advisor

This topic explains how to use Parasoft Change Advisor in SOAtest and Virtualize. It describes how to update the request or response messages that your test or service virtualization assets send.

Sections include:

- [About Change Advisor](#)
- [Workflow](#)
- [Change Impact Analysis](#)
- [Refactoring](#)
- [What Changes Does it Cover?](#)
- [The Change Advisor UI](#)
- [Identifying Changes with Change Impact Analysis](#)
- [Refactoring Using a Change Template](#)
- [Defining a Change Template](#)
- [Sharing Change Templates](#)
- [Tutorial](#)

## About Change Advisor

Change Advisor automatically identifies how messages sent by your existing test and/or service virtualization assets are impacted by service changes (e.g., renamed operations, added elements or types, modified namespaces, etc.). It also automatically refactors impacted tools (individually or in bulk) based on a change template to either update your existing test or responder suite or create a new one. Change templates are created by reviewing and refining automatically-defined mappings that explain how elements/operations/resources and schemas from the original definition relate to those in the current definition. This ensures that your assets are updated as rapidly, accurately, and intelligently as possible.

### Did You Install SOAtest as Plugin or from an Update Site?

Change Advisor depends on an Eclipse plugin called EMF Compare. That plugin gets installed in the full SOAtest installation, but not when SOAtest is installed from update site or plugin installer.

You can install the EMF Compare plugin using the Eclipse update site mechanism; the update site is at <http://download.eclipse.org/modeling/emf/compare/updates/releases/2.0>

## Workflow

If you suspect that a service has changed, the typical workflow is:

1. Run a change impact analysis from the Parasoft Change Advisor perspective.
2. Create a change template.
3. Apply that change template—on a case by case basis (to resolve specific change impacts reported by the change impact analysis) or for batch refactoring (to rapidly update all impacted tools across a project, suite, etc.)—to update your existing suite or create a new one.

SOAtest can catch unexpected service changes as soon as they occur:

1. Update your preferred test execution Test Configuration to perform change impact analysis as it executes tests.
2. If change impacts are reported:
  - a. Create a change template.
  - b. Apply that change template—on a case by case basis (to resolve specific change impacts reported by the change impact analysis) or for batch refactoring (to rapidly update all impacted tools across a project, suite, etc.)—to update your existing test suite or create a new one.

## Change Impact Analysis

When a Change Impact Analysis is run, Change Advisor reports how assets' messages are impacted by a change in the related definition file (e.g., WSDL, schema, WADL, OpenAPI/Swagger, RAML). This analysis can either:

- Compare the assets' messages against the service definition with which they are currently configured.
- Compare the assets' messages against a different service definition.

The result of this analysis helps you assess the scope and nature of changes that will be required to update your assets' messages.

## Refactoring

After determining what should be updated, you can apply a change template to automatically refactor the messages to suit the modified service definition. The template can be applied in a number of ways:

- On a test-by-test or responder-by-responder basis
- On a selected set of test or responders
- On all applicable tests or responders in a suite
- On all tests or responders in a suite

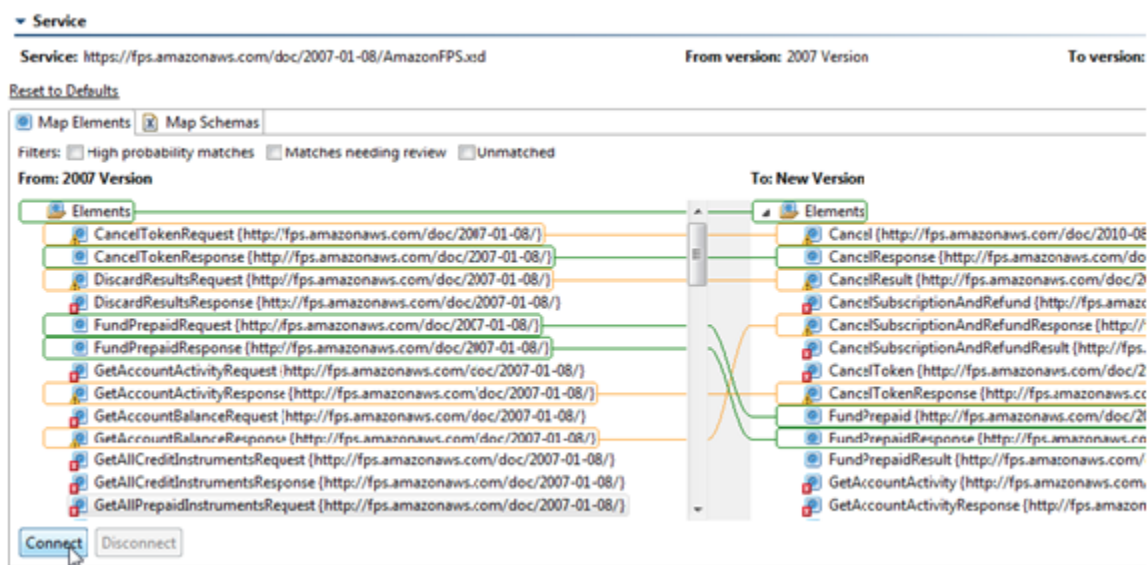
## Change Template Definition

Change templates can be automatically generated from the Service Catalog, which is populated during change impact analysis.

Once a change template is created, we recommend that you review and fine-tune the automatically-defined mappings that explain how elements /operations/resources and schemas from the original definition relate to those in the current definition.

- When the automated analysis determines that there is a high probability that an item from one definition correlates to an item in the other definition, we'll point that out and automatically set up that change to be applied when you're ready. Such mappings are considered "high probability matches."
- If the analysis identifies a possible match between the original service and the current one, we'll still preconfigure that mapping, but we'll also highlight this as a match that requires review.

Reviewing and updating the automatically-defined mappings is performed in the graphical Change Template editor:



Once a change template is generated, it can be applied on a case by case basis (to resolve specific change impacts reported by change impact analysis) as well as for batch refactoring (to rapidly update all impacted tools across a project, suite, etc.)

## What Changes Does it Cover?

Change Advisor can be applied to messaging tools (SOAP Client, REST Client, EDI Client, Fixed Length Client, Message Stub) in SOAtest. It can also be applied to Message Responders in Virtualize. Change Advisor operates on the messages specified via Form Input if the tool is set to be constrained to WSDL or Schema, or if the Service Definition has a RAML, WADL (SOAtest), or OpenAPI/Swagger specified. It operates on the messages specified via Form JSON if the Service Definition has a RAML, WADL (SOAtest), or OpenAPI/Swagger specified. If a tool's message is specified in a non-supported format, it will be skipped during analysis and refactoring. To update values in these formats, values in other non-message fields in these tools, or values in tools not supported by Change Advisor, use Parasoft Search and Replace. See

the following chapters:

- SOAtest: [Updating Tool Values with Search and Replace](#)
- Virtualize: [Updating Tool Values with Search and Replace](#)

Simple changes to the definition file (elements are reordered, types are changed, etc.) are updated automatically.

## The Change Advisor UI

The Change Advisor perspective is designed specifically for identifying and addressing change impacts.

To open the Change Advisor perspective:

- Choose **Window > Perspective > Open Perspective > Parasoft Change Advisor**.

This perspective provides:

- **The standard asset explorer:** This lets you select which assets to analyze or refactor (by applying the toolbar buttons described below).
- **Toolbar buttons:** Three buttons are added: **Run** (used to rerun change impact analysis to check that assets have been refactored properly) **Perform Change Impact Analysis** (described below), and **Refactor Using a Change Template** (described below).
- **The Service Catalog view:** This stores information about the different versions of each service, as well as the change templates that define how one service version's structure compares to another's.
- **The Properties view:** This provides details about the item selected in the Change Template editor.
- **The Change Impact progress view:** This reports the progress and results of change impact analysis.
- **The Quality Tasks view:** This is where you can review change impacts and refactor the related tools using any available change templates.

## Identifying Changes with Change Impact Analysis

### About Change Impact Analysis

Change impact analysis compares the service definition that was originally used to construct the tools' messages vs. either

- The live version of that same service definition (at the same URL as the original definition file).
- A different service definition (at a different URL than the original definition file).

Change impact analysis will report only changes that are relevant to the tools you're analyzing. For example, if the service no longer includes an optional element that your tools never used, a change impact will NOT be reported.

Typical use cases for running change impact analysis in SOAtest are:

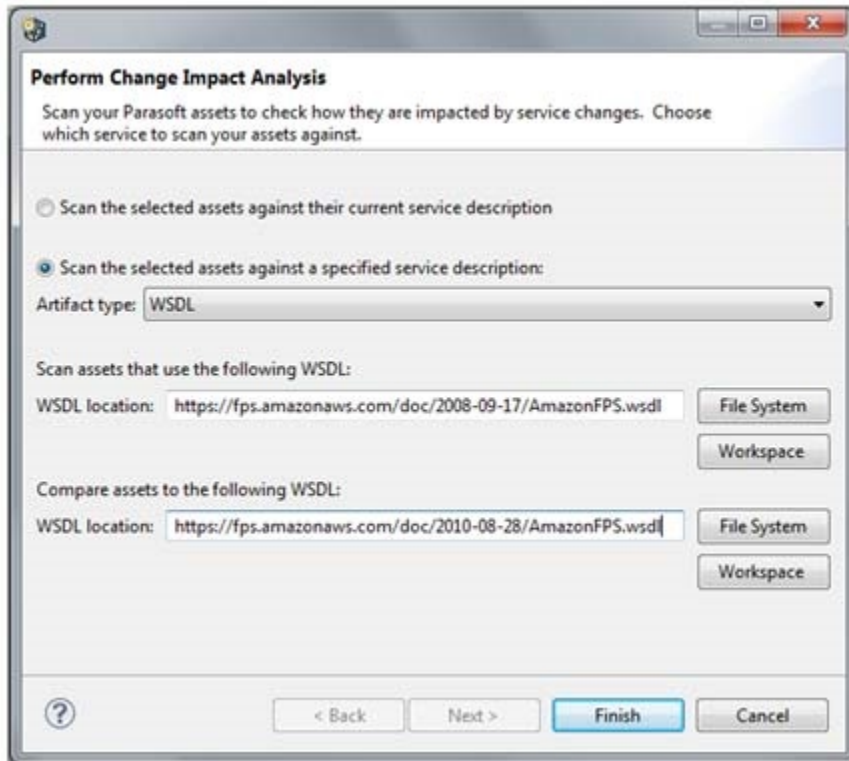
- If you suspect that the service changed, run it from Change Advisor to identify what changed and see the scope and nature of the updates that will be required to address that change.
- If you're not expecting any changes, run it alongside normal test execution. This way, if a test fails, you can immediately determine if that failure might be related to a change in the service.

### Running Change Impact Analysis

To run change impact analysis from Change Advisor:

1. In the explorer, indicate what assets you want to analyze. You can select individual tools, suites, project, etc. If you have both SOAtest and Virtualize installed, select *either* .tst files or .pva files.
2. In the Change Advisor toolbar, click the **Perform Change Impact Analysis** button.
3. Complete the dialog that opens:
  - **To analyze vs. existing definition(s):** If you want to analyze all selected assets vs the live version of their correlated service definitions, select the first button, then click **Finish**.

- **To analyze vs. a different definition:** If you want to analyze selected assets vs. a different version of the correlated service definition, select the second button, specify the location of both the current and updated service definitions, then click **Finish**.



For instance, assume you have selected a suite that includes assets based on 3 different service definitions. Analysis with the **Scan the selected assets against their current service description** option will compare all of those assets vs. the current version of their correlated service definitions. If you select **Scan the selected assets against a specified service description**—and you specify the location of the existing and updated version of 1 of the 3 service definitions used by those assets—the analysis will be applied to only the assets that use that specified service description. Assets based on the 2 other definitions will not be analyzed.

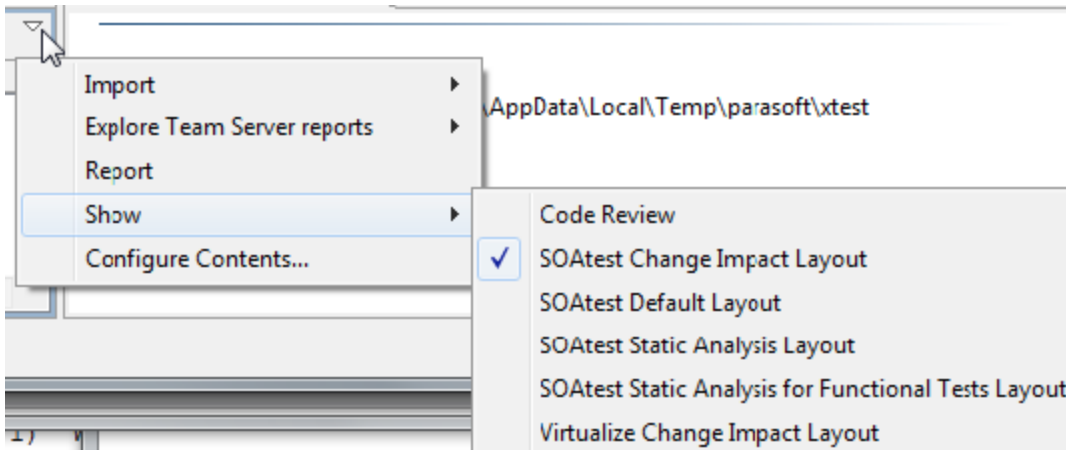
## Reviewing Analysis Results

As the analysis runs, analysis progress will be reported in the progress view as well as the console view. Note that if an asset is reported as "skipped," this indicates that it did not have messages specified in a supported format (e.g., Form Input).

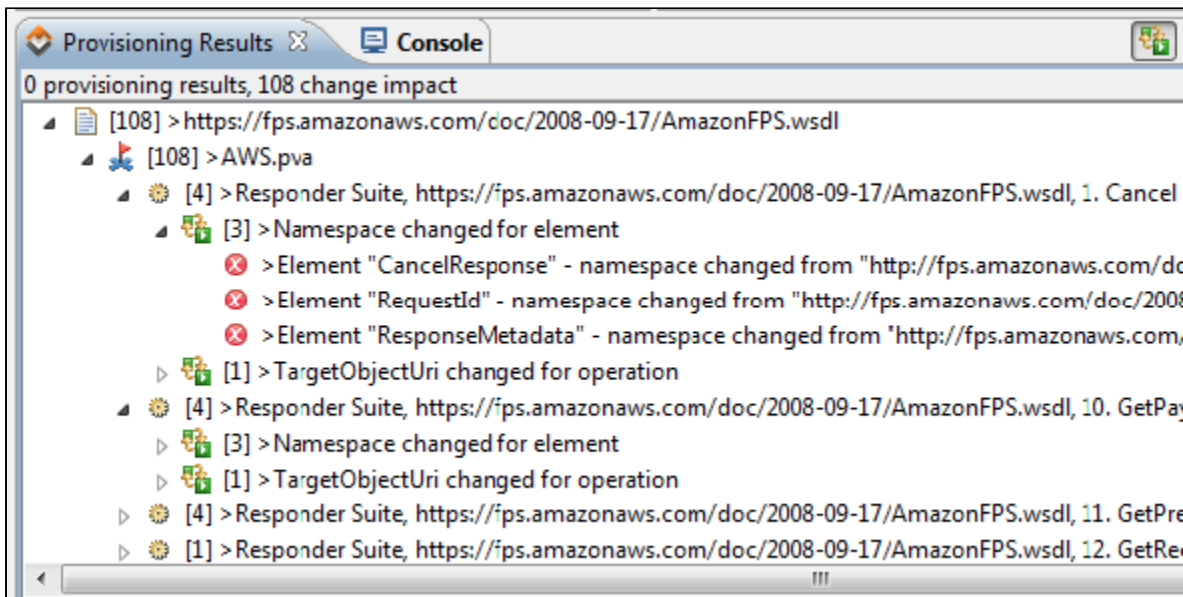
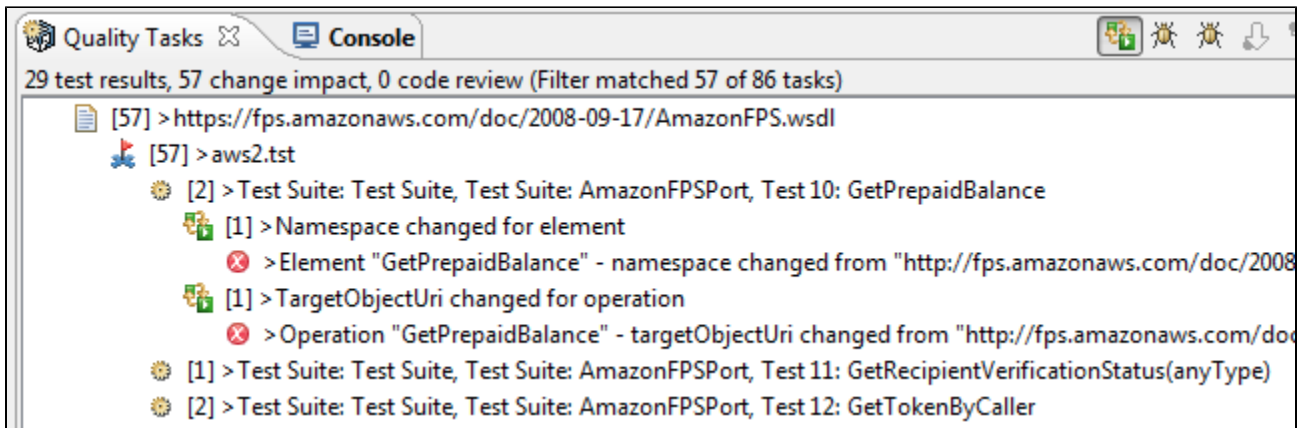
The results of the analysis will be reported in the Quality Tasks view. A Change Impact layout is provided to help you review change impacts. If the Quality View is not already using this layout, you can enable it by clicking the **Change Impact Layout** toolbar button in the results view:



You can also open this layout from that view's drop-down menu.



The results are organized by service, suite/location, and the type of change:



## Exploring the Change Impacts

To open the tool whose message needs to be updated:

- Right-click the change impact message, then choose **View Associated Editor**.

To see the tool's location within your projects:

- Right-click the change impact message, then choose **Show Source in Explorer**.

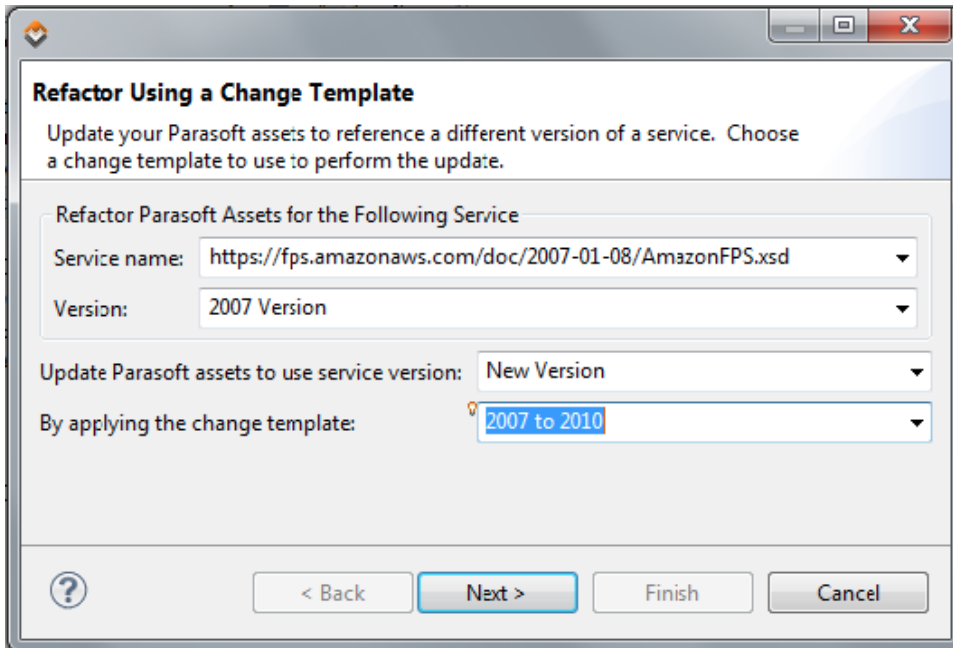
# Refactoring Using a Change Template

Change templates are created as described in [Defining a Change Template](#). There are two ways to apply a change template:

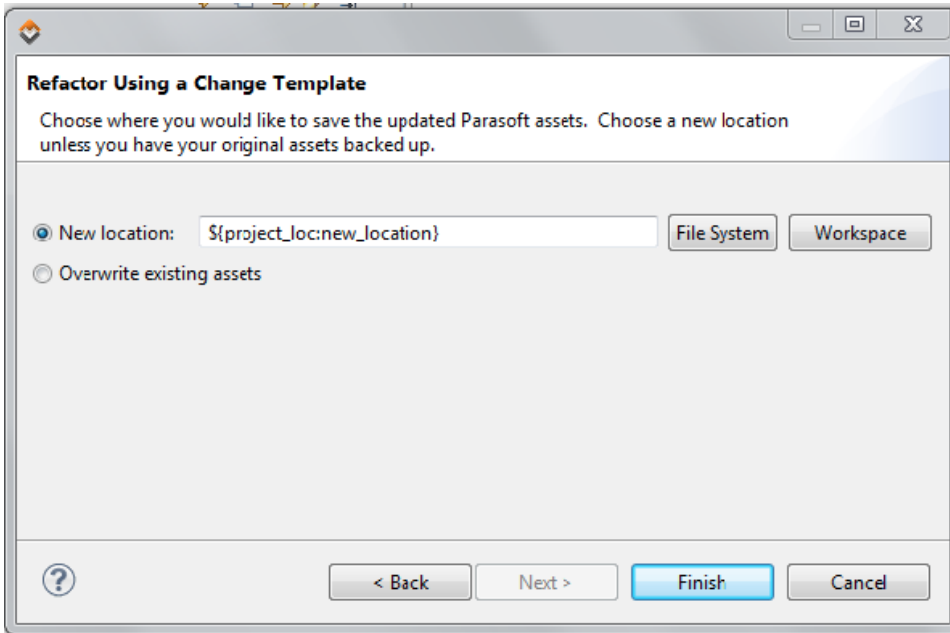
- **At the change-impact level:** Apply it to specific change impacts reported from the change impact analysis. This is often used if you just ran a change impact analysis and want to review and address the reported impacts one-by-one.
- **At the project or suite level:** Perform batch processing across suites, tools, or projects selected in your explorer. This is typically used if you want to rapidly apply the change template to a group of assets.

To apply a change template:

1. Open the refactoring wizard in one of the following ways:
  - Right-click an impact in the results view, then choose **Refactor Using Change Template**.
  - In the explorer, select the node(s) for the assets you want to update, then choose **Refactor Using Change Template**. If you have both SOAtest and Virtualize installed, select *either* .tst files or .pva files.
2. Complete the first page of the wizard as follows, then click **Next**
  - **Service name:** Specify the name of the service whose messages you want to refactor.
  - **Version:** Specify the version of the service whose messages you want to refactor. The version names that appear here are those that appear in the Service Catalog view.
  - **Update Parasoft assets to use service version:** Specify the version of the service that the updated messages will correspond to. The version names that appear here are those that appear in the Service Catalog view.
  - **By applying the change template:** Specify the name of the change template that you want to apply in order to update messages for the service version specified above.



3. In the second page of the wizard, indicate where you want to save the updated assets. The default is to save the updated assets in a new location so that you're creating a new set of tests or responders rather than overwriting your original assets.



The refactoring will update messages as defined in the change template. Tools for removed operations will be disabled, but not deleted.

## Defining a Change Template

### Creating a Change Template

A change template specifies how to map the elements/operations/resources and schemas (either XML or JSON) from one service version to those in another. Once a change template is defined, it can be applied to update assets impacted by a service change: either to remedy specific change impacts reported after a change impact analysis or to perform a batch update on selected tools, suites, or projects. For instance, if FundPrepaid is mapped to FundPrepaidRequest in a change template, applying this template will update FundPrepaid to FundPrepaidRequest in the selected tools' messages.

Change templates are created, reviewed, and modified from the Parasoft Change Advisor perspective.

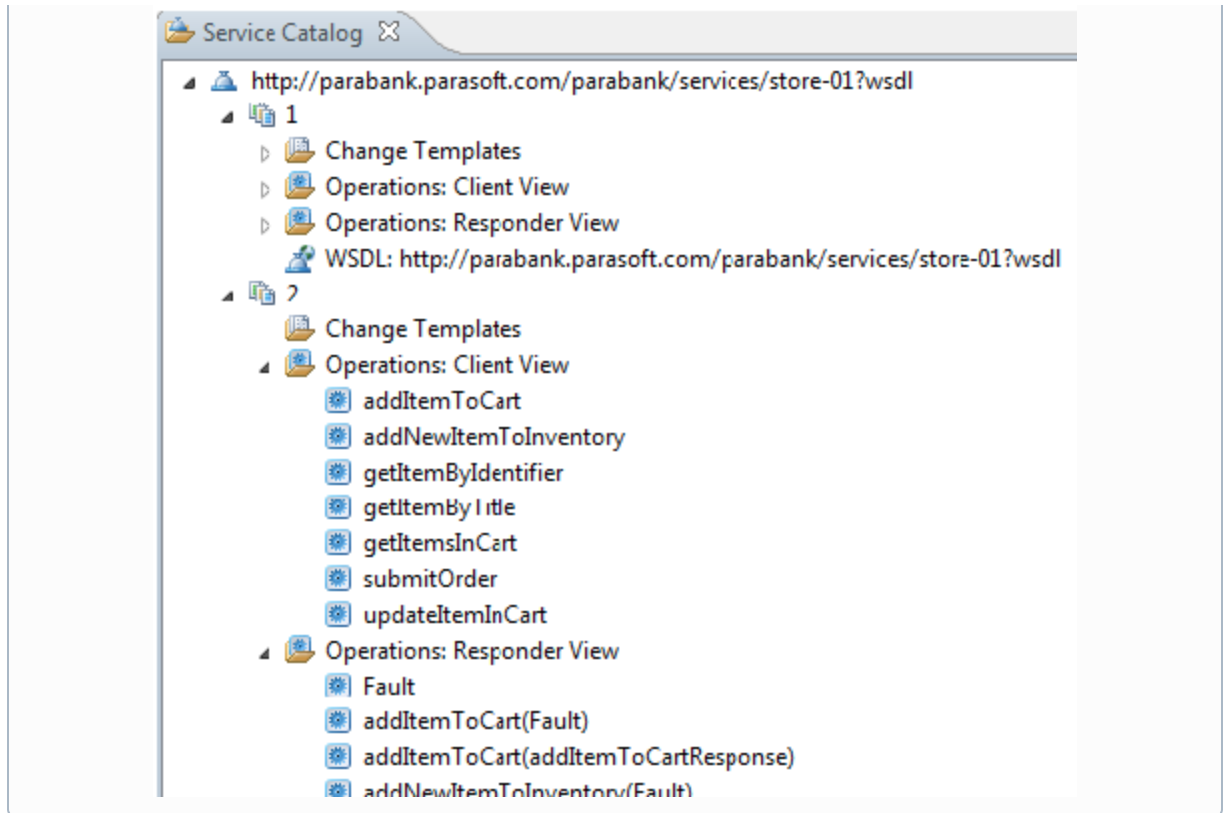
To create a change template:

1. If you have not already done so, run a change impact analysis as described in [Identifying Changes with Change Impact Analysis](#).
2. Ensure that the Service Catalog shows the service that has changed and contains "versions" for both the original service definition and the current service definition.
  - To add a service to the catalog, right-click any where in the Service Catalog view, choose **Add Service to Catalog**, complete the available fields (these vary based on the artifact type selected), then click **Finish**.
  - To add a version to the catalog, right-click on the related service, choose **Add Version to Catalog**, then specify the necessary version details.
  - To rename a service or version, right-click its Service Catalog node, then choose the appropriate **Rename** command. The names specified here will be used in the Refactor Using a Change Template wizard.

#### Why Are There Two Operations Nodes in the Service Catalog?

Note that if you have both SOAtest and Virtualize installed, the **Operations** node will be split into two nodes:

- **Client View:** For SOAtest messaging clients (covers request parameters)
- **Responder View:** For Virtualize message responders (covers response parameters)



3. Do one of the following:
  - Choose **File > New > Change Template**.
  - In the Service Catalog, select the nodes for the original version and the new version, right-click the selection, then choose the appropriate **Add Change Template** command.
  - In the Service Catalog, right-click the **Change Templates** node for the original version, then choose **Add ChangeTemplate**.
4. In the wizard that opens:
  - a. Specify basic information about the change template.
    - Note that if you have both SOAtest and Virtualize, you will see an option to create a change template for clients or responders. Choose **Client** if you want this change template to define how to update requests in SOAtest messaging client tools. Choose **Responder** if you want this change template to define how to update responses in Virtualize message responder tools.
  - b. Click **Next**.
  - c. Specify the name and location for the change template file that will be created.
  - d. Click **Finish**.

A change template will then be created and opened in the Change Template editor.

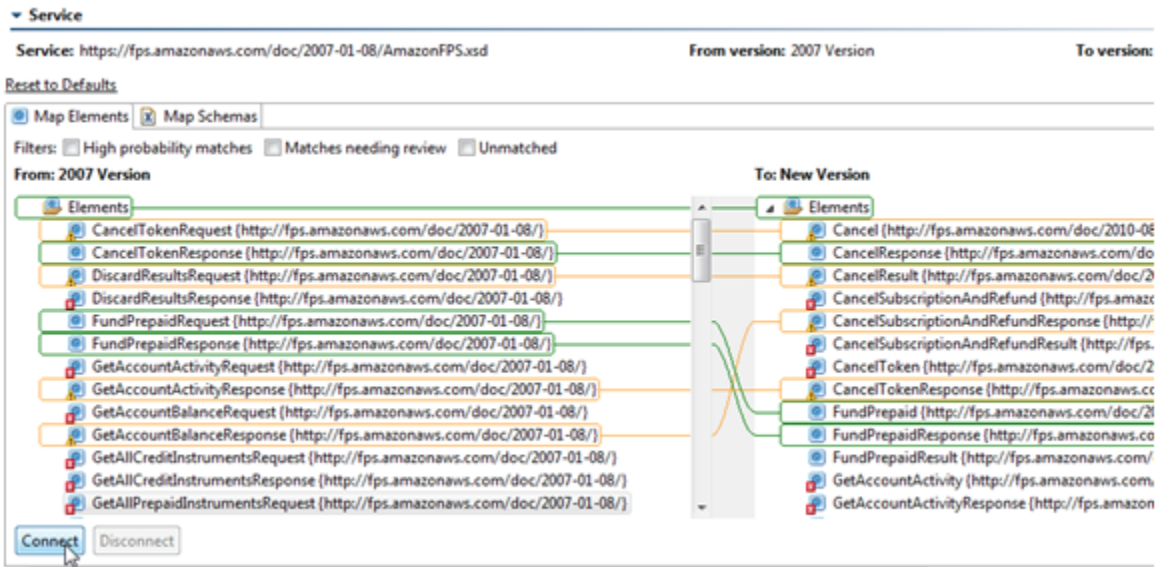
## Reviewing and Modifying the Change Template

The change template automatically maps elements/operations/resources and their schema from the original service definition to the new one according to Parasoft's heuristics:

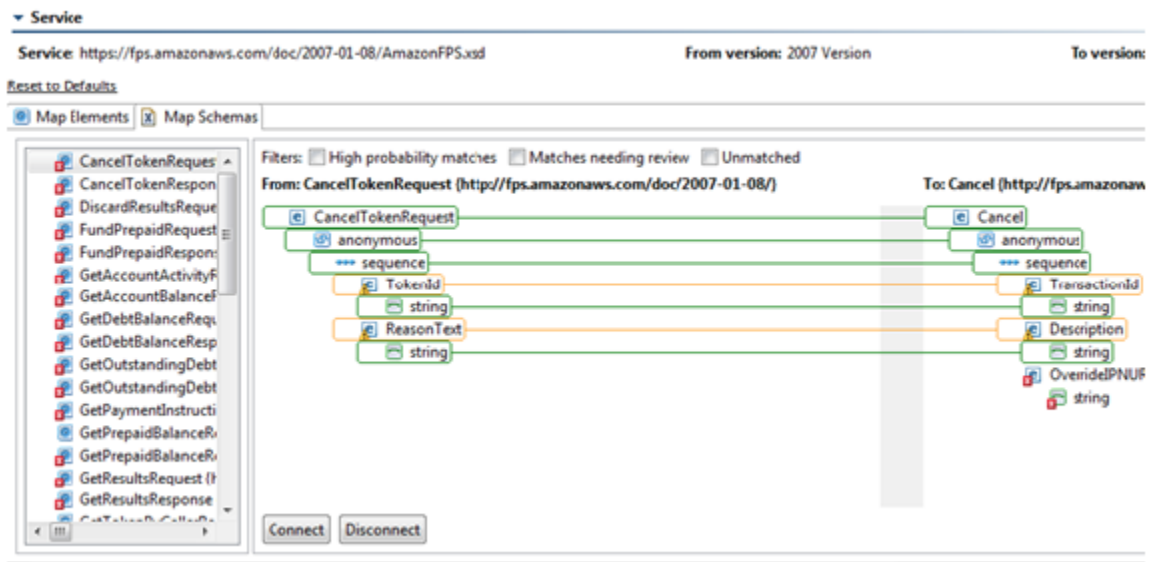
- If there is a perfect or high probability match between items in the original and new version, the related items will be mapped and marked with green.
- If there a close (but not definite) match between items in the original and new version, the related items will be mapped and marked with yellow. These items should be reviewed before you finalize and apply the change template.
- If no close match is detected for an item in the original version, it will not be mapped.

Elements/operations/resources are mapped in the **Map Elements/Operations/Resources** tab.

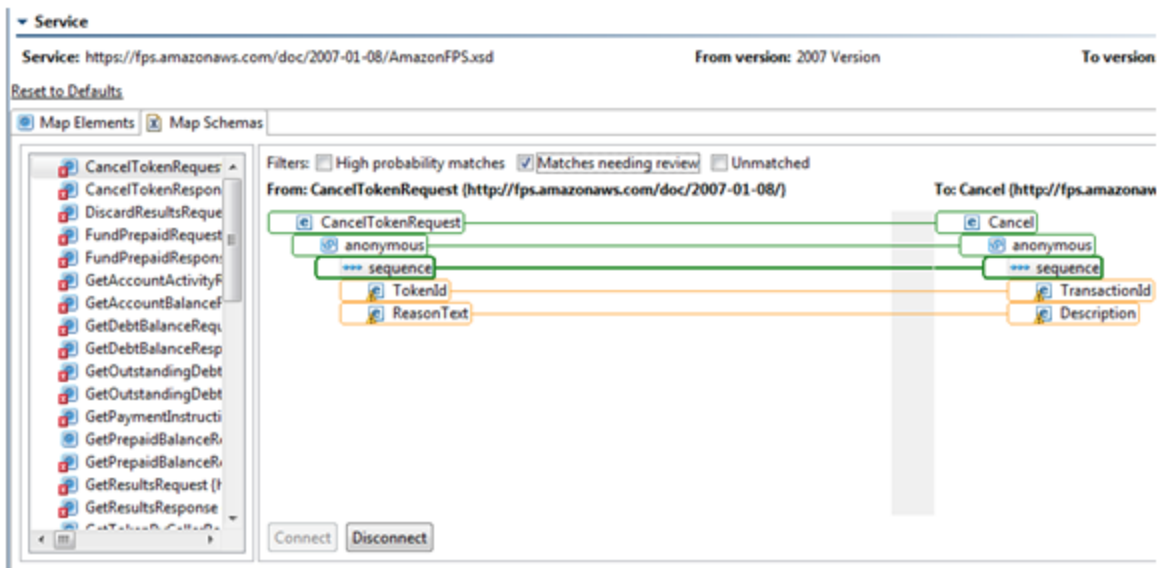




Each element's schemas are mapped in the **Map Schemas** tab.



As you are reviewing the mappings, you can use the filters available at the top of the editor to focus on high probability matches, matches needing review, or unmatched items.

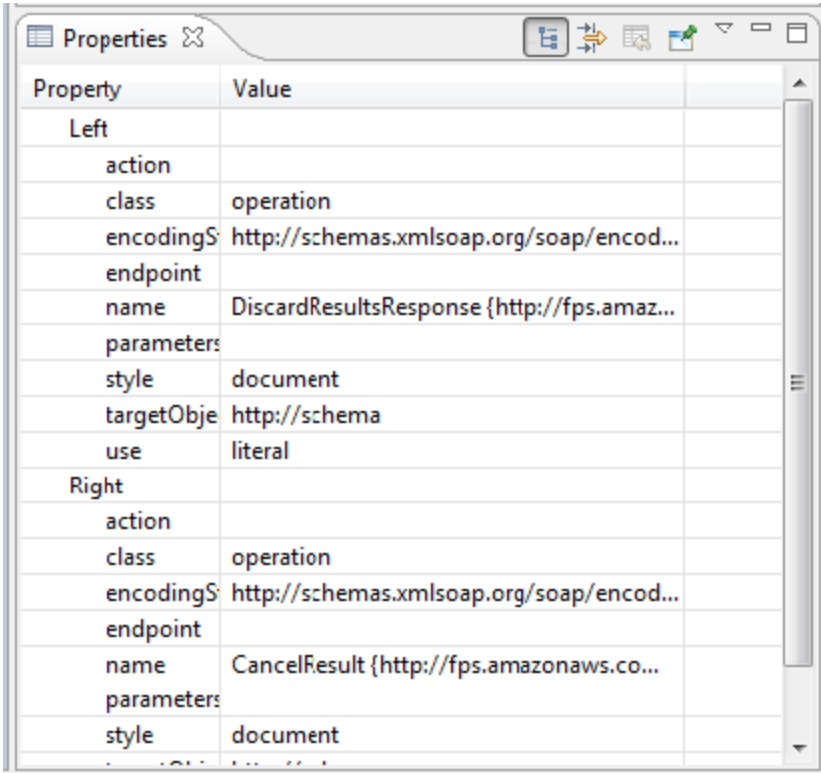


As you review and modify the change template, you can perform the following actions:

To achieve this...	Do this...
Specify that two unmapped items are correlated.	Select both items, then click <b>Connect</b> .
Specify that two mapped items are NOT correlated.	Select both items, then click <b>Disconnect</b> .
Specify the default value for an unmatched item.	Right-click that item, select <b>Configure Default Value for Added [item_name]</b> , then specify the appropriate value in the dialog that opens.
Indicate that a "match needing review" has been reviewed (turning a yellow mapping to a green one).	Right-click any area of the mapping, then choose either <b>Mark Match [mapping] as Reviewed</b> or <b>Mark Match [mapping] and Child Matches as Reviewed</b> .
Indicate that an unmatched item has been reviewed.	Right-click that element, then choose either <b>Mark [item_name] as Reviewed</b> or <b>Mark [item_name] and Children as Reviewed</b> .











## Examining Properties While Mapping

As you're mapping elements/operations/resources and schemas, you can refer to the Properties view for details about the item you're trying to map. For example:



## Change Template Icon Descriptions for XML















The following icons are used to mark XML items in the Change Template editor:

Icon	Description
	All
	Any type
	Attribute
	Choice
	Complex type
	Element
	Enumeration
	Operation
	Sequence
	Simple type

## Change Template Icon Descriptions for JSON

The following icons are used to mark JSON items in the Change Template editor:

Icon	Description
------	-------------

	Any type
	Array
	Array item
	Boolean
	Enum
	Enum value
	Integer/Number
	Item Container
	Null
	Object
	Object Property
	OneOf
	Property Container
	String

## Sharing Change Templates

If your team wants to share change templates, either share the project that contains that template, or copy the template from one workspace to another.

## Tutorial

For a tutorial on using Change Advisor, see [Updating Tests as Services Change](#).