Microsoft Visual Studio Add-In Users Guide

SOA | software™

VS Add-In

Microsoft Visual Studio Add-In Users Guide
SOA_VS_Users_v62

Copyright

Trademarks

SOA Software, Policy Manager, Portfolio Manager, Repository Manager, Service Manager, Community Manager, SOA Intermediary for Microsoft and SOLA are trademarks of SOA Software, Inc. All other product and company names herein may be trademarks and/or registered trademarks of their registered owners.

SOA Software, Inc.

SOA Software, Inc.
12100 Wilshire Blvd, Suite 1800
Los Angeles, CA 90025
(866) SOA-9876
www.soa.com
info@soa.com

Disclaimer

The information provided in this document is provided "AS IS" WITHOUT ANY WARRANTIES OF ANY KIND INCLUDING WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT OF INTELLECTUAL PROPERTY. SOA Software may make changes to this document at any time without notice. All comparisons, functionalities and measures as related to similar products and services offered by other vendors are based on SOA Software's internal assessment and/or publicly available information of SOA Software and other vendor product features, unless otherwise specifically stated. Reliance by you on these assessments / comparative assessments is to be made solely on your own discretion and at your own risk. The content of this document may be out of date, and SOA Software makes no commitment to update this content. This document may refer to products, programs or services that are not available in your country. Consult your local SOA Software business contact for information regarding the products, programs and services that may be available to you. Applicable law may not allow the exclusion of implied warranties, so the above exclusion may not apply to you.

Contents

Chapter 1 | Visual Studio Walk-Through

This guide provides information on the SOA Software Add-In for Visual Studio 2010 and 2012. The Add-In provides advanced support for runtime governance of a Microsoft environment within Visual Studio, and provides seamless support for products like SOA Software's Agent for WCF and Delegate for WCF.

Overview

The Microsoft Visual Studio Add-In provides the following functions:

- Ability to browse services managed by the Policy Manager runtime registry/repository.
- Ability to create governed WCF client endpoints from services registered in the SOA Software Policy Manager registry/repository. The Policy Manager registry/repository from within Visual Studio uses simple graphical drag and drop features to make service governance fast and easy.
- Ability to publish services from Visual Studio projects in the SOA Software Policy Manager registry/repository.

Version Support

The Add-In works for Visual Studio versions 2010 and 2012.

## Configure Existing Client Endpoint with soaDelegateBinding

1   Open the Visual Studio Add-In by selecting **View > Other Windows > SOA Software Policy Manager**. The SOA Software Policy Manager window within the Visual Studio Environment opens.
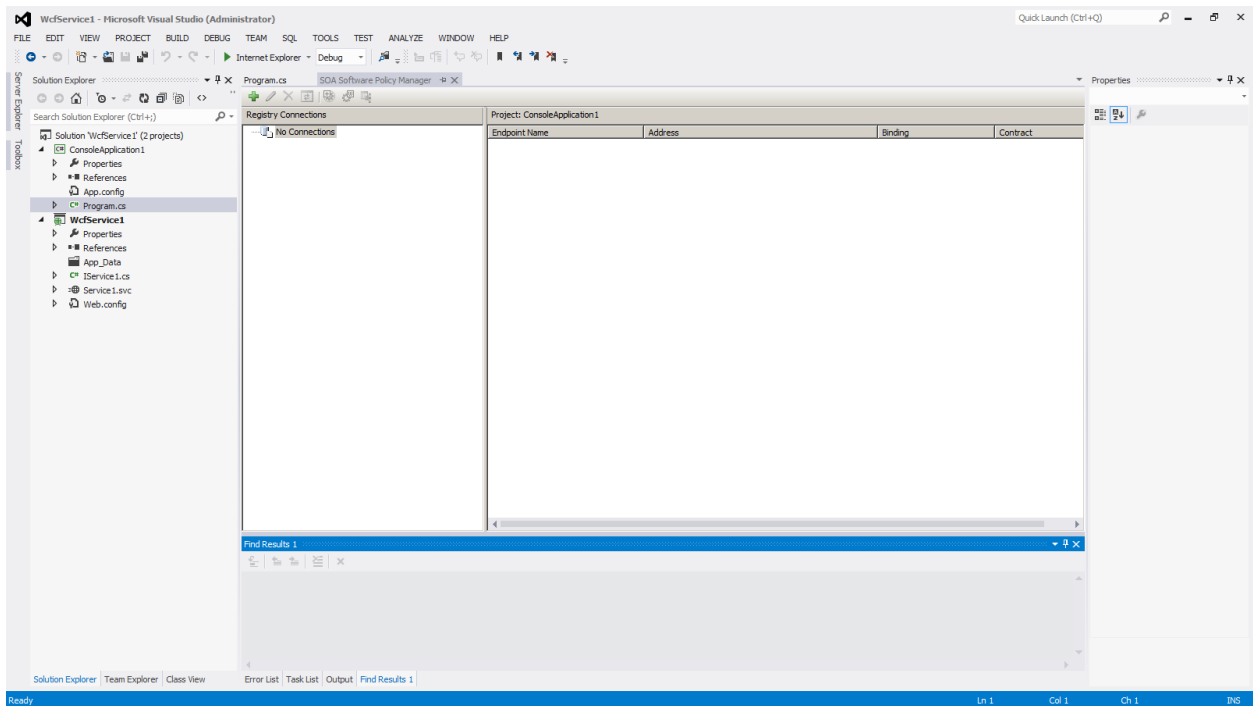
2   The Add-In window is split in two vertical panes.

    −   The left pane shows the Registry connections and allows browsing through them.

    −   The right pane shows client or service endpoints available in the currently selected Visual Studio project.

Browse Policy Manager Registry/Repository

1   Click the **Add Registry Connection** toolbar button.

    −   You can create one or more Registry connections.

    −   These connections can be associated with different instances of Registries or can provide different views into the same Registry depending on the user account configured to access a Registry.
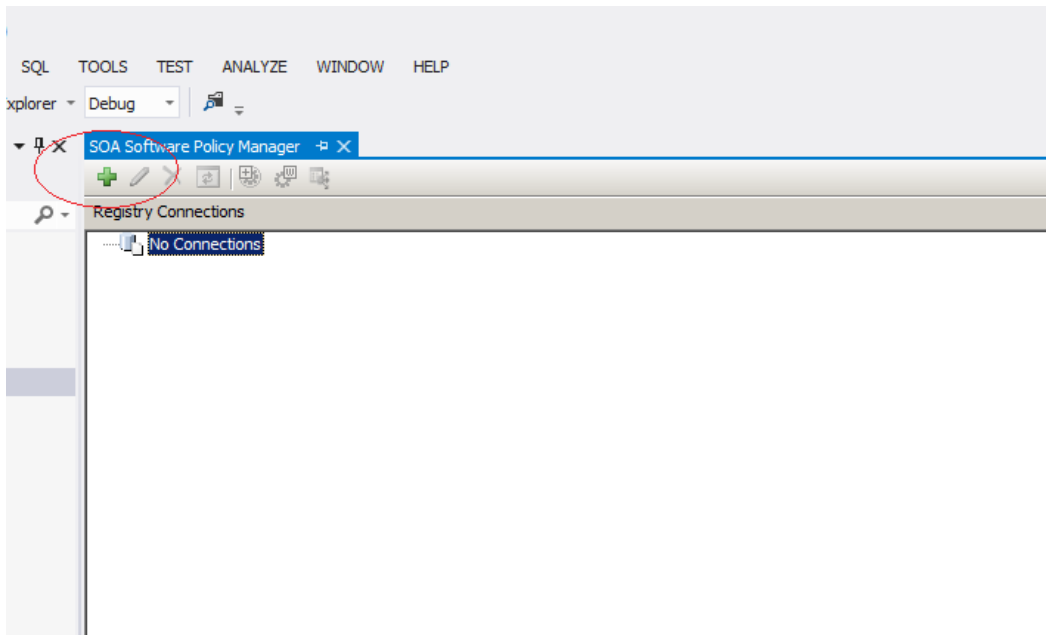
2 Enter the following in the New Registry Connection dialog box:

  – **Connection Name:** A user-friendly name.

  – **Connection Address:** The address of the Policy Manager location.

  – **User Name:** The User Name of the account that has access to the SOA Software Policy Manager registry/repository.

 – **Password:** The Password of the user account.

Note: User Name and Password information is stored on the local file system encrypted with Windows credentials of the current logged in user. Each Windows account user logged into this machine has its own persistent connection settings.
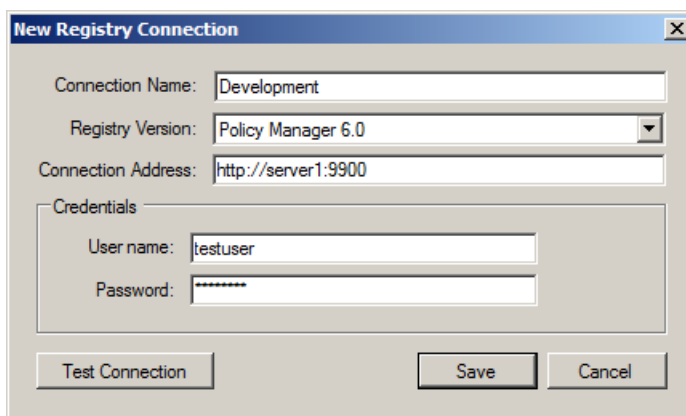
3 Test the connection to ensure the provided credentials and Connection address information is valid.

  – Click **Test Connection** and **Save**.

 – You will see the root element to the Registry connection in the left pane of the Add-In.
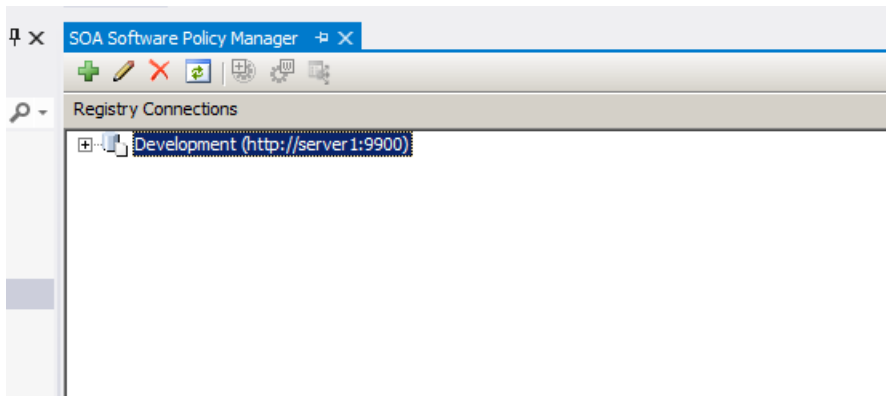
4   Expand the "Root" element. Observe that the Registry tree is loaded on demand and shows an animated loading image while the "Add-In" is communicating with the Registry.
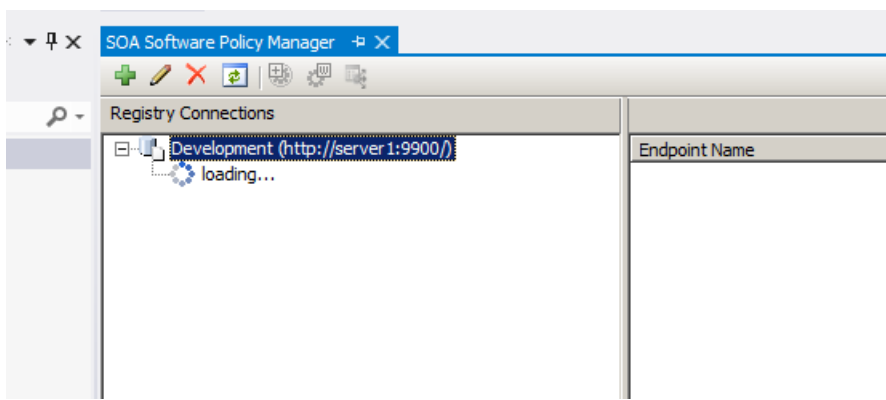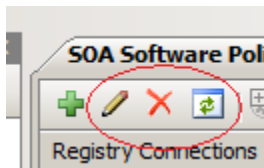

5   Expand the Registry tree nodes to view Organizations, Services, Service Interfaces, Service Operations, and Service Endpoints.

> Note that the standard Visual Studio Property windows will show Registry properties of the currently selected item.
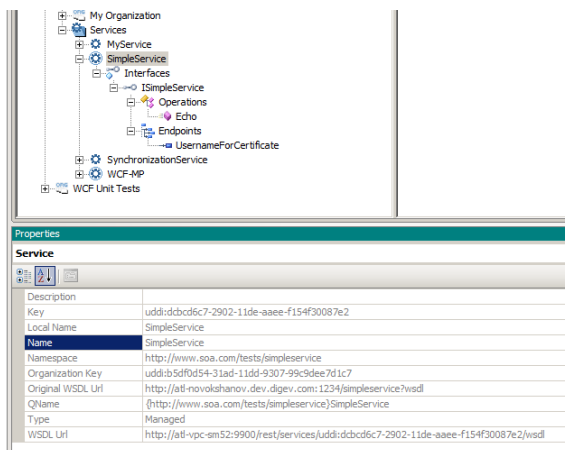
6   Manage your connections using the **Root** element and Modify, Remove, or Refresh toolbar options.

7 The View WSDL toolbar button is enabled when the Service element is selected in the Registry tree. The same option is available by performing a right-click on the Service element.
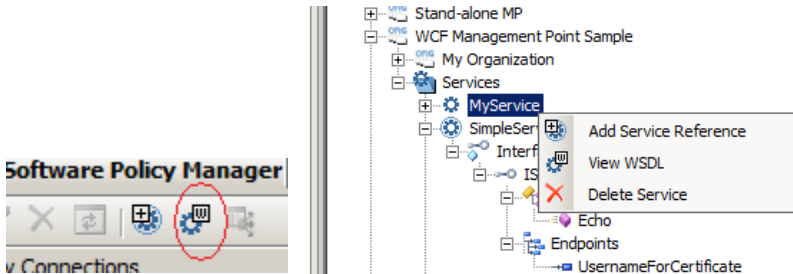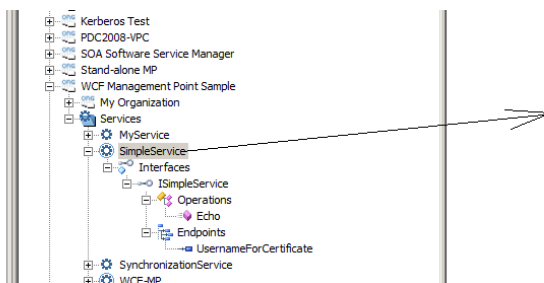

Add Service Reference to Selected Project

The following procedure illustrates how to add a service reference to a service selected in the Registry.

1 Select Service or Service endpoint in the left pane and use one of the following three methods to initiate the Service Reference generation:

   a) Right-click on a service or service endpoint element and select the **Add Service Reference** menu option (screenshot above).

   b) Click the **Add Service Reference** toolbar button.
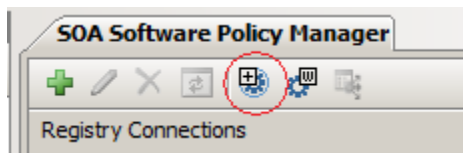
2 Drag-and-drop the selected service or service endpoint element from the left pane to the right pane.

Add Service Reference to a Service Selected in the Registry

1 Select Service or Service endpoint in the left pane and use one of the following three methods to initiate the Service Reference generation:
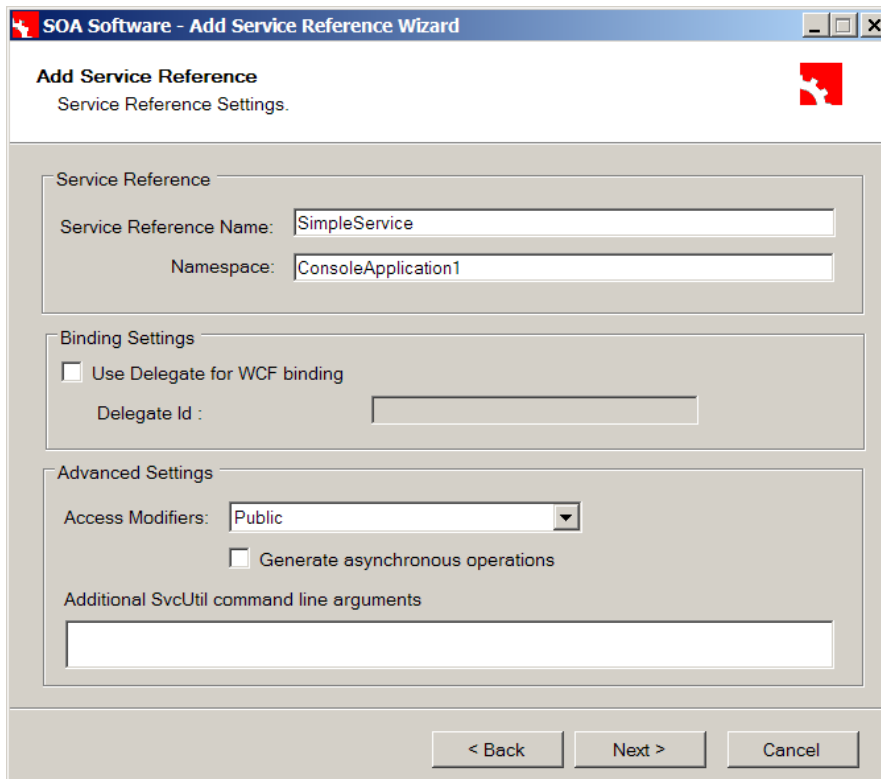
a) Right-click on a service or service endpoint element and select the **Add Service Reference** menu option (screenshot above).

b) Click **Add Service Reference** toolbar button.

c) Drag-and-drop the selected service or service endpoint element from the left pane to the right pane.
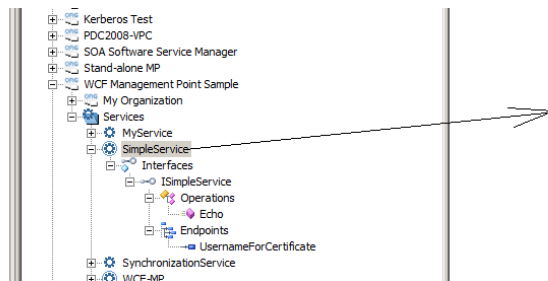
2　In the Add Service Reference Wizard select the service endpoints that will be created as client endpoints in the currently selected Visual Studio project. If one or more Delegate for WCF containers are configured on this machine there will be an option to automatically manage new client endpoints with SOA Software's Delegate for WCF. If the Management checkbox is checked then the new client endpoint(s) will automatically be configured with an soaDelegateBinding otherwise the endpoint will be created with a standard WCF binding.

3　Click **Next**.

4　Specify the Service Reference Name, Namespace and Advanced options. The Add Service Reference Wizard will be running in the background (i.e., `svcutil.exe` Microsoft utility that generates C# or VB.NET proxy class and creates client endpoints in the project's application configuration file). You can specify additional `svcutil.exe` command arguments if further customization to the proxy class generation is required. Refer to the Microsoft documentation for more details on the `svcutil.exe` utility, http://msdn.microsoft.com/en-us/library/aa347733.aspx

5   Click the "Next" button. Review Service Reference Settings and click the "Finish" button.

6   Observe the new proxy source file created in your project under the SOA Service References folder and application configuration file with the new client endpoint configured either with the soaDelegateBinding or a standard WCF binding.
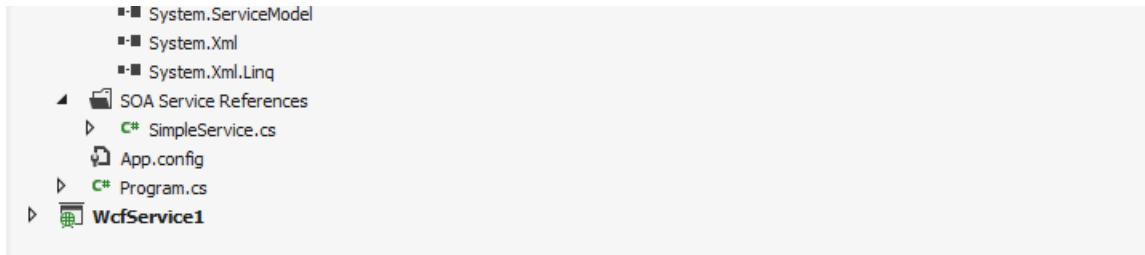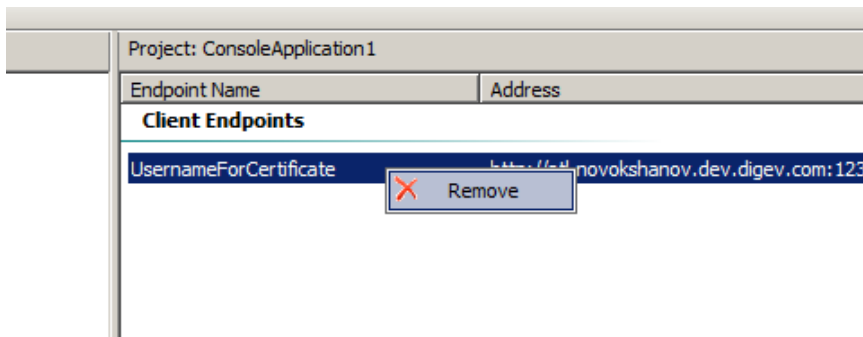
7   Observe the new client endpoint in the **Add-In** right pane.
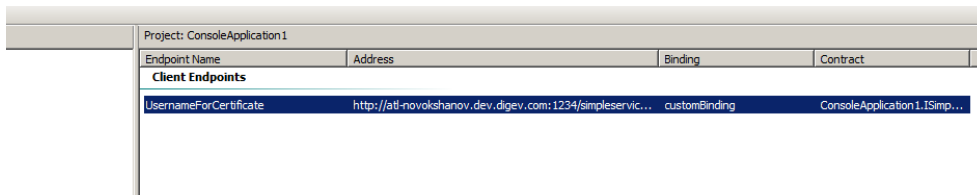
8   You can delete this endpoint and any other application client endpoint from this view by right-clicking on the endpoint.

Configure Existing Client Endpoint with soaDelegateBinding

If you already have project(s) with client endpoints configured with WCF standard bindings you can re-configure them to be governed using the Delegate for WCF with an soaDelegateBinding from within the Visual Studio **Add-In**.
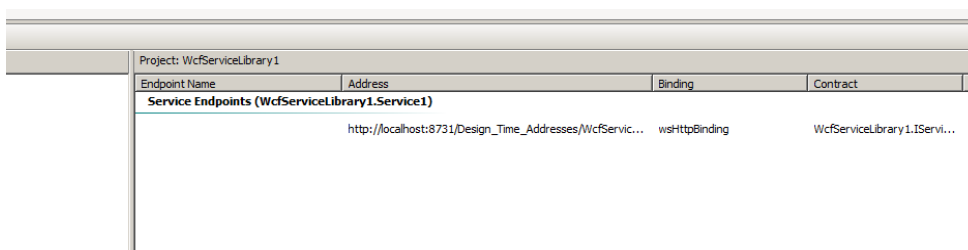
1   Select the service endpoint in the **Add-In** left pane and drag-and-drop it on the client endpoint in the right pane.

2   The Map Service Endpoints prompts to map Registry based Service endpoint to the Visual Studio client endpoint. This mapping effectively replaces current client endpoint binding with soaDelegateBinding configured with selected Delegate container. If you want to keep the original binding configuration in the application configuration file you can optionally select the **Keep original binding in application configuration file** checkbox.
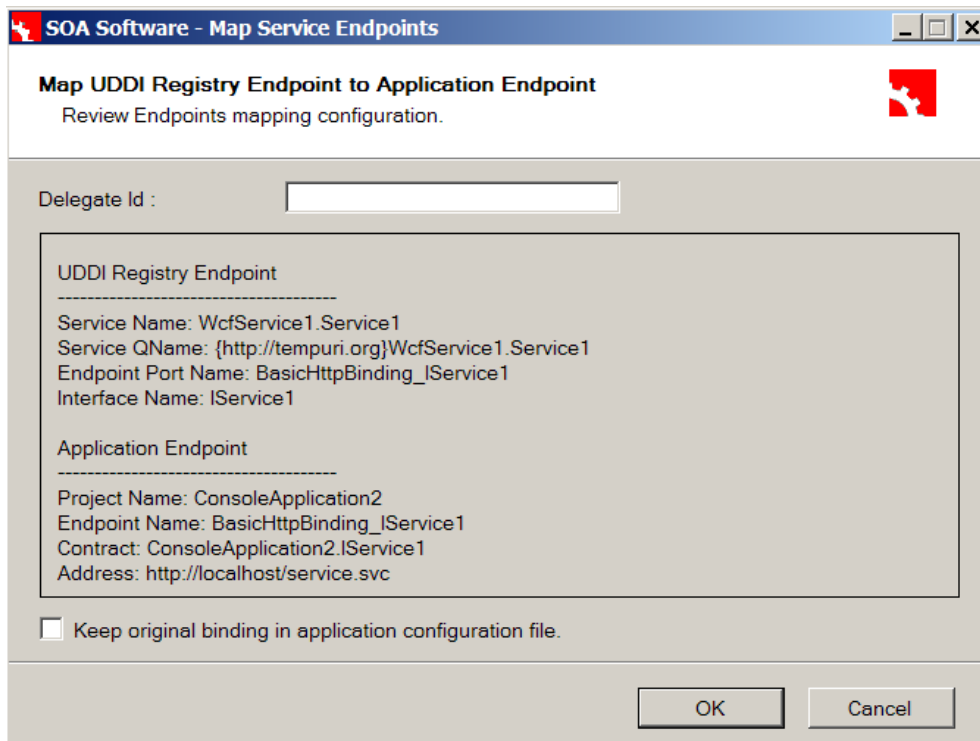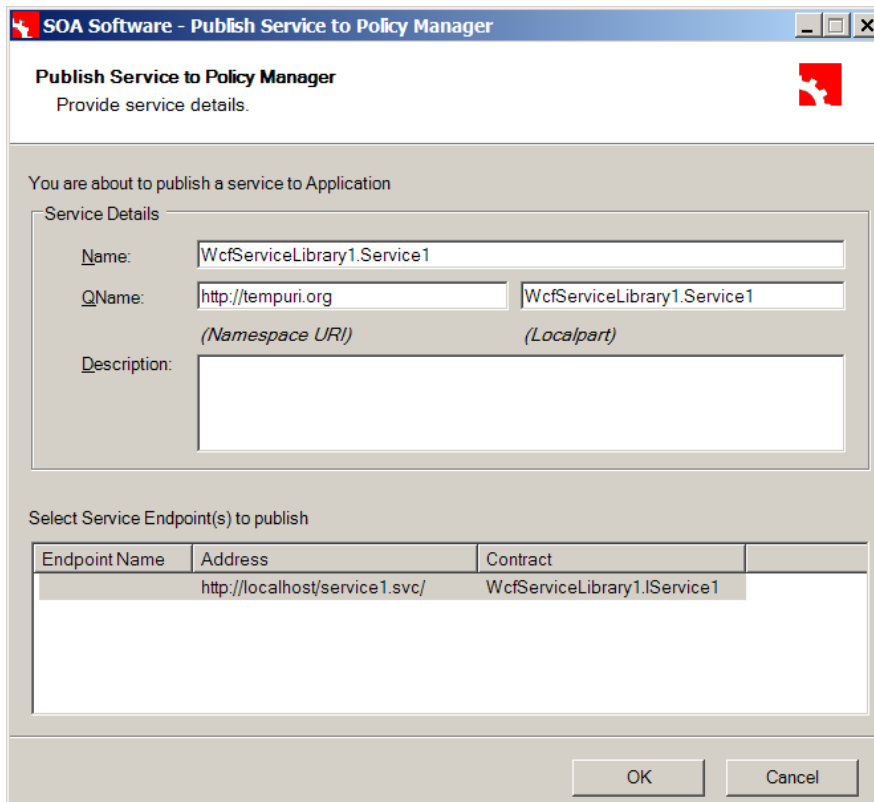
Publish Service to Policy Manager

The following procedure illustrates how to publish a service in the Policy Manager registry using Visual Studio.

1   Select a project with WCF services in the Visual Studio Solution Explorer view.

2   Observe the project service endpoints that appear in the **Add-In** right pane.

3   Identify which Registry Services folder the service will be registered in and make sure it is visible in the left **Add-In** pane.

4   Drag-and-drop the Visual Studio service endpoint from the **Add-In** right pane to the **Add-In** left pane. When dragging the service endpoint to the **Add-In** left pane, observe that the Registry Services folders are highlighted indicating a valid drop location. Dropping the endpoint in the Services folder brings up the Publish Service to Policy Manager dialog box.

5   In the Publish Service to Policy Manager Registry dialog box enter the Service Name, Namespace and Local Part that this service will be registered in the registry with. Select Service Endpoints that will be published in the Registry.

6   Click **OK**. Observe that the Services folder in the **Add-In** left pane is refreshed and the service is registered in the Policy Manager registry.

Configure Existing Service Endpoint with soaDelegateBinding

If you already have project(s) with service endpoints configured with WCF standard bindings you can re-configure them with soaDelegateBinding from within Visual Studio Add-In. The process of mapping Registry endpoints to the Visual Studio project service endpoints is similar to the same mapping for client endpoints. The only difference is that the mapping Service endpoints can be performed only if a Registry service is managed by the container that is also configured as an Agent container on this machine. Please refer to the Policy Manager Online Help for more information on how to use the Management Console and place a physical service under the management of a container.

Appendix A | Customer Support

SOA Software offers a variety of support services by email and phone. Support options and details are listed below.

Support Option	Details
Email	• support@soa.com • The Support section of the SOA Software website at https://support.soa.com/support provides an option for emailing product-related inquiries to our Support team.
Phone	1-866-SOA-9876 (1-866-762-9876)
Support Site	The Support section of the SOA Software website at https://support.soa.com/support includes many product-related articles and tips that might help answer your questions.
Documentation Updates	We update our product documentation for each version. If you're not sure you have the latest documentation, send an email request to support@soa.com. Specify the product and version you're using.

For more information, visit https://support.soa.com/support/.