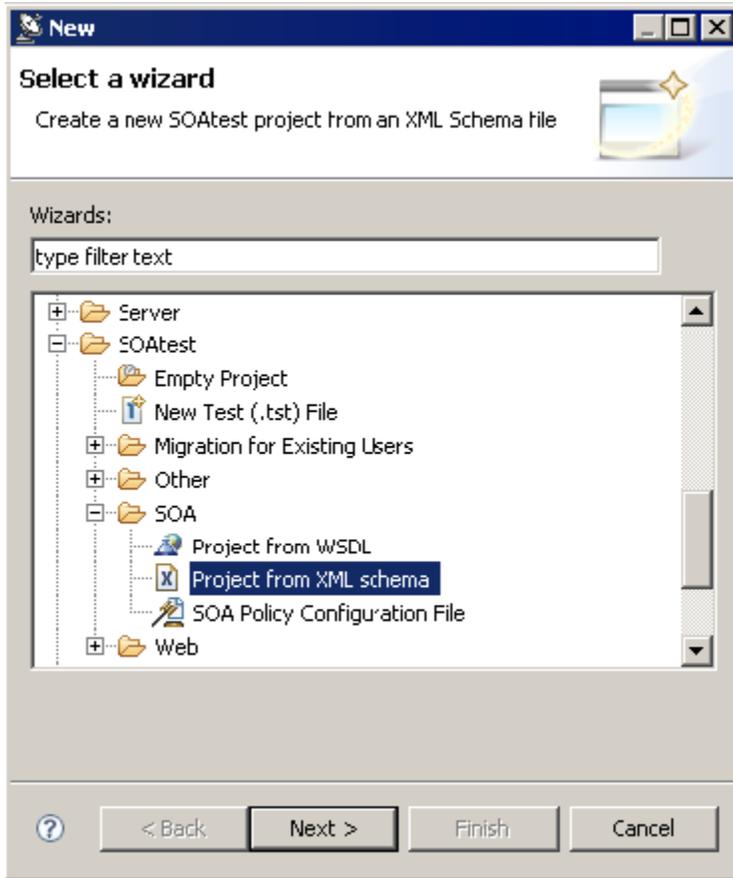


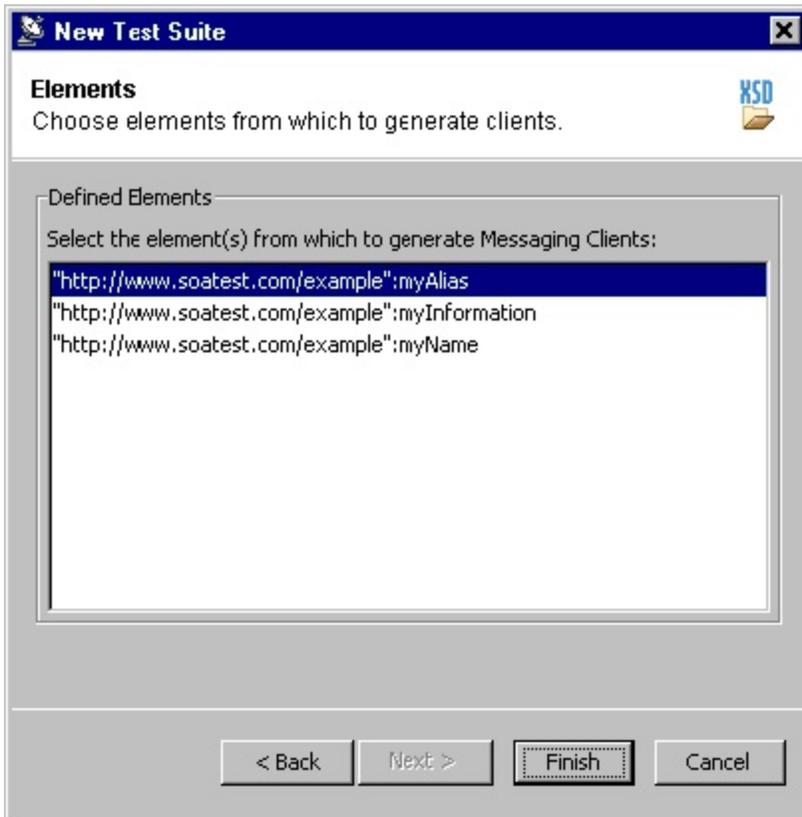
# Creating Tests From XML Schema

To automatically create a test suite from an XML schema, complete the following:

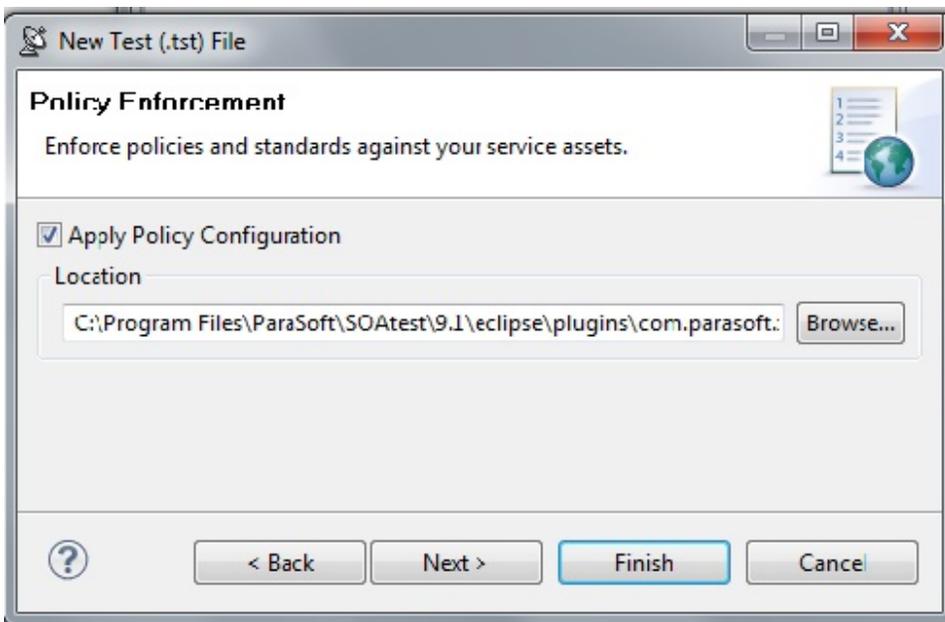
1. Choose the **SOA > XML Schema** option in one of the available test creation wizards. For details on accessing the wizards, see:
  - [Adding a New .tst File to an Existing Project](#)
  - [Adding a New Test Suite](#)



2. In the XML Schema wizard page, specify the location of the schema from which you want to generate tests.
3. Select the type of test you'd like to create:
  - Generate Messaging Client (non-SOAP XML messages). For more information, see [Messaging Client](#).
  - Generate SOAP Clients (SOAP Messages). For more information, see [SOAP Client](#).
4. For creating test clients that use HTTP, enter the URL of the web service endpoint in the **Endpoint** field, then click the **Next** button.
5. In the Elements page, select one or more elements from which to generate your tests, then click the **Next** button. SOAtest only recognizes element definitions defined at the top level (i.e.as the child of the root schema element).



6. Specify whether you want to reference an existing environment or create a new one.
  - To create a new environment:
    - a. Select the **Create a new environment for your project** checkbox
    - b. Enter an **Environment Name** and **Variable Prefix**
    - c. Select whether you want to create environment variables for **WSDL URI Fields**, **Client Endpoints**, or **Both**.
  - To reference an existing environment, select **Reference an existing environment** then specify the appropriate environment file. SOAtest will look for the Schema URL inside the referenced environment variable values. If a match is found, SOAtest will replace the relevant portions of the URL in the SOAP or Messaging Client with the environment variable name. If no match is found, then the environment will be referenced and added to the project, but the Schema URLs will be untouched.
  - For more information on environments, see [Configuring Testing in Different Environments](#).
7. Click **Next**. The **Policy Enforcement** dialog opens.



8. Select the **Apply Policy Configuration** check box. This will create WSDL and functional tests that will enforce the assertions defined in the specified policy configuration.

- The default policy configuration, `soa.policy`, is a collection of industry-wide best practices. To use a custom policy configuration, you can either use the **Browse** button to select a policy configuration or the policy configuration's path can be entered in the text field. For details on policy enforcement, see [SOA Policy Enforcement: Overview](#).
9. Click the **Next** button to advance to the **Layout** dialog.
  10. (Optional) Select the **Organize as Positive and Negative Unit Tests** checkbox to create both positive and negative tests for each operation since it is important to test situations where we send expected data as well as unexpected data to the server. The default value is configured to **Sort Tests Alphabetically**.
  11. (Optional) Select the **Asynchronous** radio button and choose **Parlay**, **Parlay X**, **SCP**, or **WS-Addressing** to create asynchronous test suites. For more information on asynchronous testing, see [Creating Asynchronous Tests](#).

Click the **Finish** button. SOAtest will generate a new test suite is created based on the XML Schema and functional test type you selected.