

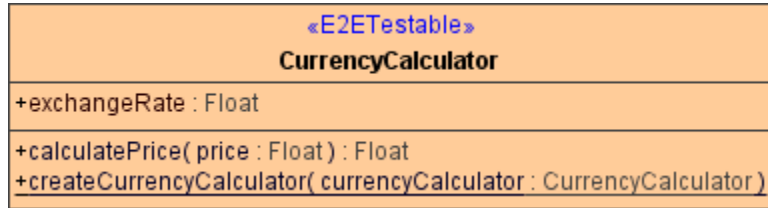
# Using Testable Classes

Example File (Builder project Testing Concepts):



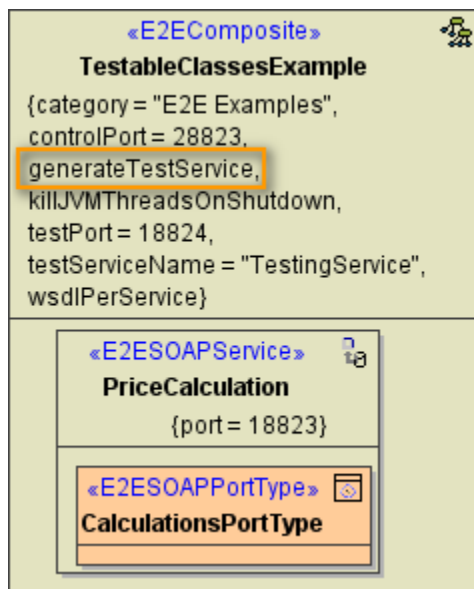
<your example path>\Testing Concepts\uml\priceCalculation.xml

The **priceCalculation** example - an example performing a very simple calculation of price, quantity and exchange rate - contains a class **CurrencyCalculator** having one attribute (**exchangeRate**) and two class operations (**calculatePrice** and **createCurrencyCalculator**). By applying the stereotype **<<E2ETestable>>** to the class this class is marked as testable.





- Added in Builder 6.0

If the flag **generateTestService** on the service composite is set to true, the compiler will generate test SOAP services for all testable classes of the model.




By the following tagged values you can configure the test settings:

Tagged Value	Description	Allowed Values	
<b>generateTestService</b>	Enables or disables testing.	true	test service generated
	If true, a test SOAP service will be generated for all <<E2ETestable>> classes.	false	no test service generated
<b>testEncoding</b>	Defines the encoding of the test SOAP operations. rpc/soap is recommended, because this matches well with the semantics of UML operations.   If this value is set and other services of the composite have different encodings, the tag <b>wsdlPerService</b> must be set to true.	rpc/soap	use rpc/soap encoding (default)
		doc/literal	use doc/literal encoding
<b>testPort</b>	Defines the number of the test service port.	any number (default: <b>55555</b> )	
<b>testServiceName</b>	Defines the name of the generated test service.	any string (default: <b>E2ETestService</b> )	

<b>testProxyHost</b>	Specifies the deployment host.	any string (default: <b>localhost</b> )
<b>testProxyProtocol</b>	Specifies the used protocol.	HTTP or HTTPS.
<b>testProxyPath</b>	Specifies the part of the overall URL that is mapped to the actual host name and port.  <div>  If this tagged value is used, the modeler must ensure, that this value is unique within the proxy. </div>	any string (default: <b>&lt;proxy URL&gt;/&lt;service host&gt;/&lt;configuration name&gt;</b> )
<b>testProxyPort</b>	Specifies the proxy port the test service is listening to.	any number (defaults: <b>80</b> (HTTP) and <b>443</b> (HTTPS))
<b>externalTestProxyHost</b>	Specifies the host name as seen by the client. If a server certificate is used, it must be issued for this name. If this value is not set, the name of the node hosting the proxy is being used.	any string

We recommend creating a separate service for all testable classes (e.g. **TestingService**) and to set the tagged value **wsdlPerService** to true on the **<<E2EComposite>>**. By that means, a mix-up between main and testing service is avoided. The testing features get a separate WSDL and a separate service entry on the Bridge and can be started and stopped within the Bridge by the **Activation** checkbox as needed.

 Please note that the compiler will not generate test SOAP services for **testable classes from imported modules**. Modules must be tested separately. Add a component diagram to the module in this case and test the classes from a deployed module service.

Composite Service
History
Logging
Dump
Settings
Version
Persistent State

**Overview**  
Node Instance: localhost  
Composite Service: TestableClassesExample  
Control Port: 28823  
Checksum: 49a5Agn8UdmEXDHZkqTaKWoGIXw=

**Status Information**  
Stopped
Start Stop Delete Kill Export

**Service Instances**  

Activation	Label	Stereotype	Port	WSDL
<input checked="" type="checkbox"/>	PriceCalculation:PriceCalculation	E2ESoapService	18823	WSDL
<input type="checkbox"/>	TestingService:TestingService	E2ESoapService	18824	WSDL

**Preferences**  

Tracing Enabled ☒  
Dump Context on Error ☒  
Bridge Server Log Level: Info  
Transaction Log Level: None

Automatic Startup ☐  
Automatic Restart ☐  
Owner: admin  
Timer Enabled: n/a  
Scheduler Enabled: n/a

Apply

For more information on the Bridge and service preferences refer to the [BRIDGE Integration Platform User's Guide](#) .

Once the class is part of the SOAP interface, it is possible to test all class operations with the **xUML SOAP Test Tool** (or any other SOAP test tool) and with the **Analyzer**. The testable class may even be part of regression tests defined with the Analyzer. Within each test case, attributes of the testable class are addressed via the self context, so that all information relevant for the execution of the test case can be provided.

priceCalculation.TestableClassesExample.TestableClassesDeployment

priceCalculation.TestableClassesExample.TestableClassesDeployment

- PriceCalculation
- TestingService
  - CurrencyCalculator
    - f() calculatePrice
      - calculatePrice
    - f() createCurrencyCalculator
      - createCurrencyCalculator

Request	Response
self: CurrencyCalculator	
0.01	exchangeRate: Base Types.Float = "0"
0.01	price: Base Types.Float = NULL

Preferences | Run | Debug | Remote Debug | Test Cases