

Web Services. Advanced topics

Publishing and consuming web services with GeneXus
Conceptual review



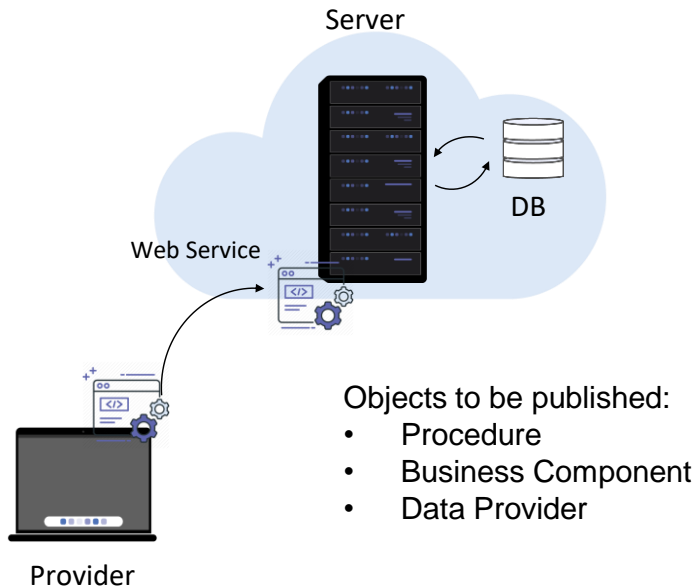
In other videos, we saw how to consume a web service created by a third party and how to publish web services with GeneXus in their most basic form.

In these videos, we will focus on advanced features that provide more flexibility when using web services, such as changing the server where they run or the authentication data; how to group web services; execution of several methods in a single web service; invocation of HTTP methods in REST services, or how to handle errors as a result of an invocation, among other topics.

Publishing web services with GeneXus

First, let's see the publication of web services with GeneXus.

Publishing a web service with GeneXus



BusinessComponent: Country	
Name	Country
Description	Country
Module/Folder	Transactions
Business Component	True
Qualified Name	Country
Object Visibility	Public
Network	
Connectivity Support	Inherit
Business Component	
Call protocol	Internal
Exposed name	Country
Exposed namespace	TravelAgency_AngularCourse
Interoperability	
Expose as Web Service	True
Web Service Protocol	REST Protocol
Generate OpenAPI interface	Use Environment property value

As we have already seen in other videos, in GeneXus we can expose procedures, business components, or data providers as web services.

To do so, we set the Expose as Web Service property to True, and then with the Web Service Protocol property we define if we want to use the SOAP protocol, REST, or both.

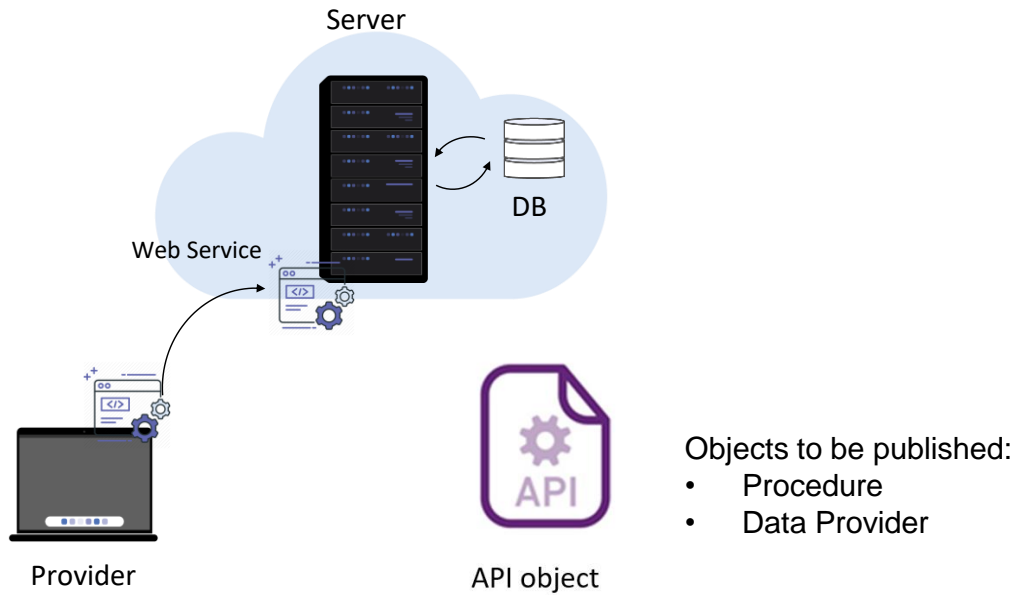
If we are exposing a SOAP service, another way to do this is to set the Call protocol property to the SOAP value and the properties we mentioned before will be set automatically.

Depending on the chosen protocol and the exposed object, we will have some more properties to configure the web service.

For example, if we publish a REST service and we want our service to comply with the OpenAPI protocol, we can generate the necessary interface information with the Generate OpenAPI interface property.

Another way to publish web services is by using the API object, which is valid for publishing procedures and data providers.

Publishing a web service with GeneXus



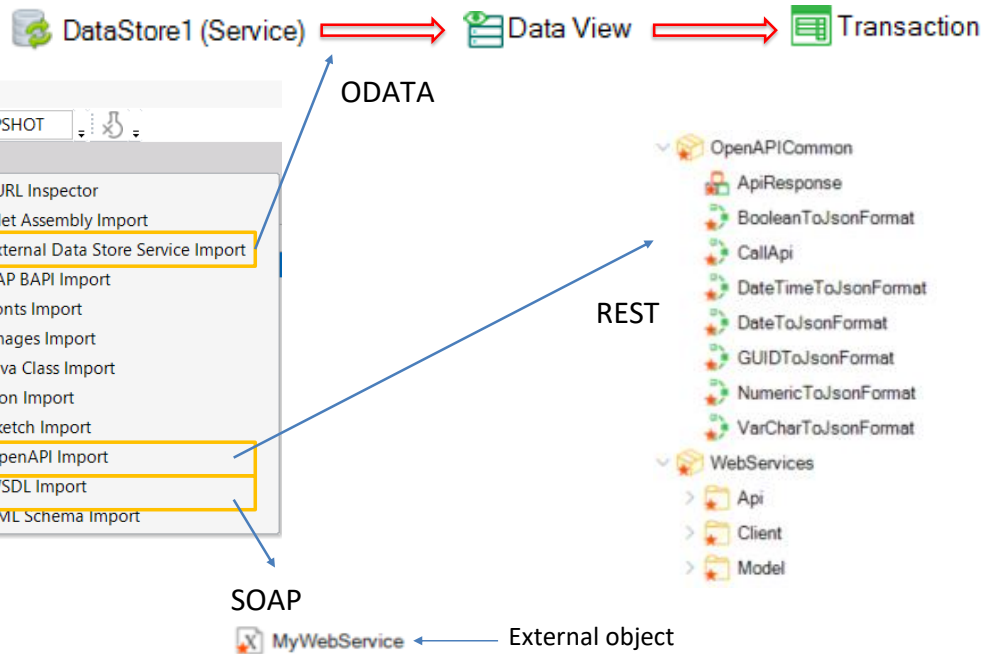
Another way to publish web services is by using the API object, which is valid for publishing procedures and data providers.

Consuming web services with GeneXus

So far, we have seen how to publish web services with GeneXus with both SOAP and REST protocols.

Now let's see how to consume services with both protocols, including the Odata protocol, from a GeneXus application.

Consuming web services with GeneXus



As shown in other videos, to invoke a web service from a GeneXus object we can create an object of External Object type and manually configure its properties. However, it is advisable to use the wizards accessed through the Tools / Application integration menu that will facilitate the task.

We can import SOAP web services (through the WSDL import option), REST (with the OpenAPI import option) and a special type of web services called OData services (with the External Data Store Service Import option), which provide operations to insert, modify, or delete records from a database, through a web site.

The wizard will vary depending on the type of web service selected. If the web service is SOAP, GeneXus will automatically create an External Object associated with the Web Service and the structured data types required for managing its data.

Otherwise, if it was REST, several GeneXus objects will be automatically created (and we will usually include them in a module, for example, WebServices). Also, they will allow us to automatically run the service through these objects in our KB. The wizard will leave in an API folder the programs to invoke, and in a Model folder the SDTs to manage the data.

If an OData service was selected, GeneXus will create a data store of Service type associated with the external data source. Also, as many Data View and Transaction objects will be created as entities were in the model; afterwards, we will be able to work with the transactions created, as with any other transaction of our

application.

In the following videos, we will see in detail how to consume and publish SOAP and REST services with GeneXus.

GeneXus™

training.genexus.com

wiki.genexus.com

training.genexus.com/certifications