



# Using Relative Paths to Local and Shared Resources

## Best Practices

Version: 1.0

**AN ACTIVE ENDPOINTS TECHNICAL NOTE**

© 2010 Active Endpoints Inc. ActiveVOS is a trademark of Active Endpoints, Inc. All other company and product names are the property of their respective owners.



2010

## Content

Best Practices .....	3
Use of Relative Paths .....	3
Use of a Common Repository Project for Shared Artifacts .....	3
Additional Guidance .....	4
Common Artifact Repository Archive Considerations.....	4
Avoiding the Deployment of WSDL Documents of the Same Namespace and WSDL Service QName to the ActiveVOS Resource Catalog.....	4
Relative Paths for Portability .....	5
About Active Endpoints .....	6

## Best Practices

### Use of Relative Paths

It is a best practice for processes, WSDL and XML Schema document to use relative paths when importing artifacts. The primary reason to do so is to ensure that BPEL processes, WSDL and XML Schema imports are portable between environments and tooling.

Use of the ActiveVOS 'project:/' Project Location URI is not required and recommended for use in XML Schema and WSDL imports. This URI is used by the Process Deployment Descriptor to identify the path to deployment artifacts based on their location relative to the root directory of the project. This location will be the location where artifacts are deployed to in the ActiveVOS Resource Catalog.

The Project Location URI is also used for ActiveVOS extension functions such as `getCatalogResource()` which requires a 'project:/' type Project Location URI as a hint to locate the resource in the catalog.

### Use of a Common Repository Project for Shared Artifacts

Organizations often standardize and create shared schema to house and consolidate approved and reusable schema definitions in one location in a schema repository. They do so to avoid introducing incompatible definitions requiring unnecessary transformation for example. These teams also often take on the responsibility to manage namespaces so as to avoid conflicts.

When this is the case the best practice is to create an ActiveVOS project in the workspace to hold these shared artifacts. An example might be a common repository project (e.g. 'project:/repository') where a flat list or a tree of schema and WSDL definitions are defined. If the tree of schema is used, all imports need to use a relative path.

When importing schema from a shared project, a BPEL process in 'project:/ProjectA' should use the relative path of '../..../repository/xsd/schema123.xsd' to import definitions of 'schema123.xsd'.

## Additional Guidance

### Common Artifact Repository Archive Considerations

A business process archive (BPR) can contain all artifacts that are required to support the execution of a process. A BPEL process may also import resources from a resolvable HTTP address, and in such cases it isn't necessary to include the resource in the BPR as it will be looked up at runtime.

In the case of shared resources such as schema and WSDL definitions, a BPR can be created to deploy these artifacts to the ActiveVOS Resource Catalog. This archive and the ActiveVOS project that contains the artifacts should be controlled by a team. Changes and deployment of this archive needs to be managed such as to avoid introducing non-backwards compatible updates that would negatively impact previously deployed processes that import its definitions.

Although it is possible for a single BPR archive to deploy common resources such as all WSDL definitions in one deployment there is a limitation in ActiveVOS 7: the public and private WSDLs that are generated by ActiveVOS Designer must be deployed in the same archive used to deploy the process definition that imports these. More specifically, any process that directly imports a public/private WSDL needs the WSDL in its BPR deployment archive. As such this does not apply only to the process that owns the public/private WSDL.

### Avoiding the Deployment of WSDL Documents of the Same Namespace and WSDL Service QName to the ActiveVOS Resource Catalog

ActiveVOS allows the ability to create copies of an artifact in the Resource Catalog that share the same namespace but are found at different locations because they originate from different ActiveVOS projects. While it is permissible to do so, these artifacts can evolve independently in their respective projects yet share the same namespace, but special care needs to be taken for WSDL artifacts.

Where conflict arises and becomes problematic is when a service being invoked shares a WSDL of the same namespace and WSDL service QName of another service. At runtime ActiveVOS looks up WSDLs by namespace and matches the service to invoke by WSDL service QName

when performing an invoke operation. At all other times ActiveVOS looks up artifacts based on the associated deployment plan.

What specifically must be avoided is an overlap of the service QName definition within a WSDL with others. Issues arise when for example two different versions of a WSDL document get deployed to the Resource Catalog each with the same namespace, and yet each defining an incompatible version of the `tns:MyWSDLService` service. At runtime this results in ambiguity when ActiveVOS looks up the service by WSDL Service QName. This gives rise to the common runtime “operation not found for PortType” error being reported because two or more developers have deployed WSDL documents sharing the same namespace with conflicting service QNames, each exposing entirely different operations.

If deploying multiple WSDL in the same namespace, care needs to be taken to define service definitions that do not have overlapping service QNames. Using separate target namespaces (as part of your interface/service versioning strategy) will help avoid this situation.

### **Relative Paths for Portability**

Using relative path imports allows developers to move projects in an ActiveVOS Workspace between machines and environment, and ensures that imports still resolve. This applies both at design time and also at runtime.

For example if a schema at location hint `'project:/myproj/xsd/schema1.xsd'` in the resource catalog (and a developer's workspace) has an import that is relative such `'../../repository/xsd/schema2.xsd'`, then at runtime ActiveVOS will resolve the import to location hint `'project:/repository/xsd/schema2.xsd'` that had been specified by the process deployment descriptor used to deploy `'project:/Common'` artifacts.

## About Active Endpoints

Active Endpoints' ([www.activevos.com](http://www.activevos.com)) ActiveVOS is the business process management system (**BPMS**) that development teams will love. ActiveVOS empowers project teams to create business process management (**BPM**) applications using services, making their businesses more agile and effective. ActiveVOS promotes mass adoption of SOA-enabled BPM applications by focusing on accelerating project delivery time with a complete, affordable and easy-to-use system. Active Endpoints is headquartered in Waltham, MA with development facilities in Shelton, CT.

To find out how Active Endpoints can help your business, visit <http://www.activevos.com>, call +1 781 547 2900 and press 1 for Sales, or email us at [info@activevos.com](mailto:info@activevos.com).