



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

Integrated Capture Points Guide

Apache CXF—Javascript Client

12/14/2025

Apache CXF—Javascript Client

This page provides an example of generating a Javascript client using the Apache CXFwsdl2js tool. See the [list of tools](#) used to generate clients in this document. You can also [generate a Java client in Apache CXF](#).

To generate a Javascript client using the Apache CXFwsdl2js tool:

```
wsdl2js -d <output directory> <WSDL URL>
```

For example:

```
wsdl2js -d c:\Temp\MyJSCClient
http://zoollander.us.int.genesyslab.com:10080/Genesys/Interaction/WSCP_812_zoo/
WebServiceCapturePoint?wsdl
```

The tool generates a single file that contains a proxy that can send requests to the service and receive replies asynchronously. You must also include the `cxf-util.js` file, which is part of Apache CXF.

The sample below does not require anything beyond HTML and Javascript (`wscp.js` is the file generated by `wsdl2js`):

```
<html>
<head>
<script type="text/javascript" src="http://ajax.googleapis.com/ajax/libs/jquery/1.7.1/
jquery.min.js"></script>
<script type="text/javascript" src="cxp-utils.js"></script>
<script type="text/javascript" src="wscp.js"></script>

<script language="JavaScript" type="text/javascript">

var gCounter = 0;

function print_list(list, indent)
{
    var r = '';

    for(var i=0; i < list._kvitem.length; ++i)
    {
        r += indent;

        var pair = list._kvitem[i];
        r += pair._key;
        if( pair._value._ValueString )
        {
            r += " [str] = ";
            r += pair._value._ValueString;
            r += " ";
            r += "<br>";
        }
        else if( pair._value._ValueInt )
        {
            r += " [int] = ";
            r += pair._value._ValueInt;
            r += "<br>";
        }
    }
}
```

```

        else
        {
            r += " [list] = ";
            if( pair._value._ValueList )
            {
                r += "<br>";
                r += print_list(pair._value._ValueList, indent + "....");
            }
            else
            {
                r += " EMPTY";
                r += "<br>";
            }
        }
    }
    return r;
}
function test()
{
    var svc = new _iWebServiceCapturePoint();
    svc.url = 'http://zoolander.us.int.genesyslab.com:10080/Genesys/Interaction/WSCP_812_zoo/
WebServiceCapturePoint';

    var extension = new _KVList();
    var items = new Array();

    var signature = new _KVPair();
    signature.setKey("signature");
    var signature_value = new _KVPairValue();
    signature_value.setValueString("JavaScript client generated with CXF");
    signature.setValue(signature_value);

    var counter = new _KVPair();
    counter.setKey("Request count");
    var counter_value = new _KVPairValue();
    counter_value.setValueInt(++gCounter);
    counter.setValue(counter_value);
    items.push(signature);
    items.push(counter);
    extension.setKvitem(items);

    svc.Ping(
        function(response)
        {
            var r = "Response timestamp: " + response.getEventTime() + ", ping info:<br>";
            r += print_list(response.getPingInfo(), "");
            $("#response_text").html(r);
        },
        function(status, statusText)
        {
            $("#response_text").html("Response failed: (" + status + ") " + statusText);
        },
        extension
    );
}
</script>
</head>
<body>
<p>Press the button to call the service...</p>
<p><input value="Ping the service" type="button" onclick="test()"/></p>
<p><div id="response_text"></div></p>
</body>

```

</html>