

Articulation Transfer Clearinghouse **Implementation Guide**

for ReceiveDocumentBatch

8/2/2007

TABLE OF CONTENTS

INTRODUCTION	3
Project Identification	3
DOCUMENT CONTROL.....	4
Update History.....	4
ENVIRONMENTS	5
METHODS.....	6
JAVA IMPLEMENTATION DETAILS	8
.NET C# IMPLEMENTATION GUIDE.....	10
APPENDIX A: RECEIVEDOCUMENTBATCH WSDL.....	11



Introduction

This implementation guide provides information needed to connect to and receive documents awaiting pickup from OBR. These documents can include PESC postsecondary transcripts, PESC transcript acknowledgments, and OBR supplements. The batch nature of the response will support many document types which will be shown later in this document. Additionally, this document provides steps to follow when requesting documents from the ATC.

PROJECT IDENTIFICATION		
Project Name	Project Number	Date Created
Articulation Transfer Clearinghouse	EN-009	05/22/2007
Sponsor	Program Manager	
Paula Compton, David Barber	Jim Ginzer	
Project Manager		
Jim Ginzer		

**Document Control**

This section contains document change control and distribution information.

UPDATE HISTORY			
Version	Date	Author	Description
0.1	05/22/07	Alex Leader – OBR	Authored
0.2	05/25/07	Alex Leader – OBR	Revisions following ATC pilot discussion (CINC, CNST, NEOUCOM, SNCL, WSUN)
0.3	06/28/07	Alex Leader – OBR	Revision following initial development.
0.4	7/16/07	Eric Burke – OBR	Revision following final development



Environments

OBR provides both a testing and production implementation of ReceiveDocumentBatch for your convenience. Clients should develop against the test version of this service, then migrate their code accordingly to use the production version of the service. The following URLs correspond to these instances:

Environment	WSDL URL
User Acceptance Testing (UAT)	https://uat-atc.regents.ohio.gov/ccx/ReceiveDocumentBatch?wsdl
Production (PROD)	https://atc.regents.ohio.gov/ccx/ReceiveDocumentBatch?wsdl



Methods

Method Name: ReceiveDocumentBatch

Description

Provides the ability to acquire specified documents from OBR. If a BatchID is not specified then the documents this service will include in the response are those that have not been previously downloaded. Calling this method will change the status of the documents to reflect that they have been downloaded. If a BatchID is specified all documents originally sent to the institution as part of that batch will be in the response. This allows institutions to pull batches that have previously been created by OBR but have errored.

Request

Username – OBR supplied institutional username

Password – OBR supplied institutional password

BatchID – BatchID of batch to acquire.

or

DocumentType – Enumeration of document type options:

PescPostsecondaryTranscript – Includes transcript and corresponding supplement.

PescTranscriptAcknowledgment

All

Response

BatchID– OBR-defined batch id for this batch.

DocumentsRemaining - value that allows an institution to know if additional documents are available for download. An institution can call the service multiple times until no more documents are remaining. This allows OBR to limit the number of docs to be downloaded in a single batch.

DocumentCount - Number of documents in data batch used for verification purposes.

Checksum – Base 64 encoded string representing the md5 checksum value of the data payload.

Data - compressed and base 64 encoded batch of documents.

Fault

Reason – Detailed error message.

Method Name: ReceiveDocumentBatchList

Description

Method that provides an institution to search for batches created by OBR. This method may be used if an error occurs during acquisition of a batch via the ReceiveDocumentBatch method. In this scenario, an institution will know that an error has occurred with ReceiveDocumentBatch and will need to query OBR to get batches produced by OBR. Once an OBR batch id is found that doesn't correspond to the batch IDs stored at an institution that batch would be need to be rerun. The use of ReceiveDocumentBatch specifying BatchID provides this ability.



Request

Username – OBR supplied institutional username

Password – OBR supplied institutional password

SearchStartDate – Search criteria to limit returned batch information using start date time.

SearchEndDate – Search criteria to limit returned batch information using end date time.

Response

BatchID – Repeating element that gives back batches that were created during the given search criteria.

Fault

Reason – Detailed error message.



Java Implementation Details

Using the WSDL found in Appendix A, a top-down development approach can be implemented. Within your IDE of choice, create a project from the WSDL or set a reference to the WSDL. Your IDE should assist you in creating the required code to call the web service. The targetNamespace and soap:address location values will require modification dependent on the environment being connected to at OBR.

Assumptions:

1. IDE of your choice
2. Framework for Web Service development/deployment

Prerequisites:

1. Point your browser to <https://uat-atc.regents.ohio.gov/ccx/ReceiveDocumentBatch?wsdl>.
2. The document data and checksum string will need to be Base64 decoded. One way to accomplish this is by using Commons Codec from <http://jakarta.apache.org/commons/codec/>. This library provides Base64 encoding/decoding functionality

Implementation Steps:

- 1) To retrieve a batch of documents, call the ReceiveDocumentBatch method.
 - i) Use the OBR supplied username and password for the 'Username' and 'Password' elements of the ReceiveDocumentBatchRequest message. To retrieve a new batch, pass in one of the allowed document types in the 'DocumentType' element of the ReceiveDocumentBatchRequest message. To retrieve a previous batch, pass in the batch id in the 'BatchID' element of the ReceiveDocumentBatchRequest message.
 - ii) After calling the ReceiveDocumentBatch service, the web service client will receive a SOAP response or a SOAP fault. The response is of type ReceiveDocumentBatchResponse which contains an id for the batch retrieved (BatchID), a count of the documents still remaining to be retrieved (DocumentsRemaining), a count of the documents included in this batch (DocumentCount), a base 64 encoded string representing the Md5 digest of the batch data (CheckSum), and the compressed and base 64 encoded batch data (Data). Additionally, if a SOAP fault or other exception is returned, the web service client should catch that exception and take the appropriate action.
 - iii) Process the response. Base64 decode the checksum string using org.apache.commons.codec.binary.Base64 from the Commons Codec library. Create a new checksum string using java.security.MessageDigest against the encoded, compressed batch data. If the two checksums do not match, take appropriate action. If the checksums do match, proceed to decode the batch data using org.apache.commons.codec.binary.Base64 from the Commons Codec library. Next, decompress the batch data utilizing zlib. This can be accomplished using java.util.zip.Inflater.
- 2) To retrieve a list of previous batches, call the ReceiveDocumentBatchList method.
 - i) Use the OBR supplied username and password for the 'Username' and 'Password' elements of the ReceiveDocumentBatchListRequest message. Pass in a start and end date of the form yyyy-mm-dd in the SearchStartDate and SearchEndDate elements of the ReceiveDocumentBatchListRequest message.





.Net C# Implementation Guide

Using the WSDL found in Appendix A, a top-down development approach can be implemented. Within your IDE of choice, create a project from the WSDL or set a reference to the WSDL. Your IDE should assist you in creating the required code to call the web service. The targetNamespace and soap:address location values will require modification dependent on the environment being connected to at OBR.

Assumptions:

1. Visual Studio 2005
2. .Net 2.0
3. C#

Prerequisites:

1. Obtain zlib.net.dll from <http://www.componentace.com/ZLIB.NET>.
2. Point your browser to <https://uat-atc.regents.ohio.gov/ccx/ReceiveDocumentBatch?wsdl>.

Implementation Steps:

To retrieve a batch of documents, call the ReceiveDocumentBatch method.

- ii) Use the OBR supplied username and password for the 'Username' and 'Password' elements of the ReceiveDocumentBatchRequest message. To retrieve a new batch, pass in one of the allowed document types in the 'DocumentType' element of the ReceiveDocumentBatchRequest message. To retrieve a previous batch, pass in the batch id in the 'BatchID' element of the ReceiveDocumentBatchRequest message.
 - iii) After calling the ReceiveDocumentBatch service, the web service client will receive a SOAP response or a SOAP fault. The response is of type ReceiveDocumentBatchResponse which contains an id for the batch retrieved (BatchID), a count of the documents still remaining to be retrieved (DocumentsRemaining), a count of the documents included in this batch (DocumentCount), a base 64 encoded string representing the Md5 digest of the batch data (Checksum), and the compressed and base 64 encoded batch data (Data). Additionally, if a SOAP fault or other exception is returned, the web service client should catch that exception and take the appropriate action.
- 3) Process the response. Base64 decode the checksum string using `System.Convert.FromBase64String`. Create a new checksum string (Md5 digest) using the MD5CryptoServiceProvider class against the encoded, compressed batch data. If the two checksums do not match, take appropriate action. If the checksums do match, proceed to decode the batch data using `System.Convert.FromBase64String`. Next, decompress the batch data utilizing zlib: create a reference to the DLL in your project and then decompress the data.
 - 4) To retrieve a list of previous batches, call the ReceiveDocumentBatchList method.
 - i) Use the OBR supplied username and password for the 'Username' and 'Password' elements of the ReceiveDocumentBatchListRequest message. Pass in a start and end date of the form yyyy-mm-dd in the SearchStartDate and SearchEndDate elements of the ReceiveDocumentBatchListRequest message.



Appendix A: ReceiveDocumentBatch WSDL

```

<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions name="ReceiveDocumentBatch" targetNamespace="http://www.regents.ohio.gov/ReceiveDocumentBatch/"
xmlns:tns="http://www.regents.ohio.gov/ReceiveDocumentBatch/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:types>
    <xsd:schema targetNamespace="http://www.regents.ohio.gov/ReceiveDocumentBatch/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
      <xsd:element name="ReceiveDocumentBatch">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="in" type="xsd:string"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="ReceiveDocumentBatchResponse">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element maxOccurs="1" minOccurs="1" name="BatchID"
type="xsd:int"/>
            <xsd:element maxOccurs="1" minOccurs="1" name="DocumentsRemaining"
type="xsd:int"/>
            <xsd:element maxOccurs="1" minOccurs="1" name="DocumentCount"
type="xsd:int"/>
            <xsd:element maxOccurs="1" minOccurs="1" name="checkSum"
type="xsd:string"/>
            <xsd:element maxOccurs="1" minOccurs="1" name="Data"
type="xsd:string"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="ReceiveDocumentBatchList">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="in" type="xsd:string"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="ReceiveDocumentBatchListResponse">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element maxOccurs="unbounded" minOccurs="0" name="BatchID"
type="xsd:string"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="ReceiveDocumentBatchListFault">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element maxOccurs="1" minOccurs="1" name="Reason"
type="xsd:string"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="ReceiveDocumentBatchFault">
        <xsd:complexType>
          <xsd:sequence>

```



```
type="xsd:string">
    <xsd:element maxOccurs="1" minOccurs="1" name="Reason"
    </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="ReceiveDocumentBatchRequest">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element maxOccurs="1" minOccurs="1" name="Username"
type="xsd:string">
    </xsd:element>
      <xsd:element maxOccurs="1" minOccurs="1" name="Password"
type="xsd:string">
    </xsd:element>
      <xsd:choice>
        <xsd:element maxOccurs="1" minOccurs="0" name="BatchID"
type="xsd:int">
      </xsd:element>
        <xsd:element maxOccurs="1" minOccurs="0"
name="DocumentType">
      <xsd:simpleType>
        <xsd:restriction base="xsd:string">
          <xsd:enumeration
value="PescPostsecondaryTranscript">
        </xsd:enumeration>
          <xsd:enumeration
value="PescTranscriptAcknowledgment">
        </xsd:enumeration>
          <xsd:enumeration value="All">
        </xsd:enumeration>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:element>
  </xsd:choice>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:group name="ModelGroupDefinition">
  <xsd:sequence/>
</xsd:group>
<xsd:element name="ReceiveDocumentBatchListRequest">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element maxOccurs="1" minOccurs="1" name="Username"
type="xsd:string">
    </xsd:element>
      <xsd:element maxOccurs="1" minOccurs="1" name="Password"
type="xsd:string">
    </xsd:element>
      <xsd:element maxOccurs="1" minOccurs="0" name="SearchStartDate"
type="xsd:date">
    </xsd:element>
      <xsd:element maxOccurs="1" minOccurs="0" name="SearchEndDate"
type="xsd:date">
    </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
</xsd:schema>
</wsdl:types>
<wsdl:message name="ReceiveDocumentBatchList_faultMsg">
  <wsdl:part name="ReceiveDocumentBatchList_fault" element="tns:ReceiveDocumentBatchListFault"/>
</wsdl:message>
```



```
<wsdl:message name="ReceiveDocumentBatchResponse">
  <wsdl:part name="ReceiveDocumentBatchResponse" element="tns:ReceiveDocumentBatchResponse"/>
</wsdl:message>
<wsdl:message name="ReceiveDocumentBatchListRequest">
  <wsdl:part name="ReceiveDocumentBatchListRequest" element="tns:ReceiveDocumentBatchListRequest"/>
</wsdl:message>
<wsdl:message name="ReceiveDocumentBatch_faultMsg">
  <wsdl:part name="ReceiveDocumentBatch_fault" element="tns:ReceiveDocumentBatchFault"/>
</wsdl:message>
<wsdl:message name="ReceiveDocumentBatchListResponse">
  <wsdl:part name="ReceiveDocumentBatchListResponse" element="tns:ReceiveDocumentBatchListResponse"/>
</wsdl:message>
<wsdl:message name="ReceiveDocumentBatchRequest">
  <wsdl:part name="ReceiveDocumentBatchRequest" element="tns:ReceiveDocumentBatchRequest"/>
</wsdl:message>
<wsdl:portType name="ReceiveDocumentBatch">
  <wsdl:operation name="ReceiveDocumentBatch">
    <wsdl:input message="tns:ReceiveDocumentBatchRequest"/>
    <wsdl:output message="tns:ReceiveDocumentBatchResponse"/>
    <wsdl:fault name="fault" message="tns:ReceiveDocumentBatch_faultMsg"/>
  </wsdl:operation>
  <wsdl:operation name="ReceiveDocumentBatchList">
    <wsdl:input message="tns:ReceiveDocumentBatchListRequest"/>
    <wsdl:output message="tns:ReceiveDocumentBatchListResponse"/>
    <wsdl:fault name="fault" message="tns:ReceiveDocumentBatchList_faultMsg"/>
  </wsdl:operation>
</wsdl:portType>
<wsdl:binding name="ReceiveDocumentBatchSOAP" type="tns:ReceiveDocumentBatch">
  <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
  <wsdl:operation name="ReceiveDocumentBatch">
    <soap:operation soapAction="http://www.regents.ohio.gov/ReceiveDocumentBatch"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="ReceiveDocumentBatchList">
    <soap:operation soapAction="http://www.regents.ohio.gov/ReceiveDocumentBatchList"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
    <wsdl:fault name="fault">
      <soap:fault name="fault" use="literal"/>
    </wsdl:fault>
  </wsdl:operation>
</wsdl:binding>
<wsdl:service name="ReceiveDocumentBatch">
  <wsdl:port name="ReceiveDocumentBatchSOAP" binding="tns:ReceiveDocumentBatchSOAP">
    <soap:address location="http://www.regents.ohio.gov/ReceiveDocumentBatch/ReceiveDocumentBatch"/>
  </wsdl:port>
</wsdl:service>
</wsdl:definitions>
```