
Request Validation Web Service

User's Guide

USGS / EROS

March 13, 2013

Revision Sheet

Release No.	Date	Revision Description
	07/29/2009	Original
Rev. 1	11/04/2009	Added example graphics
Rev. 2	01/11/2010	1. Modification to thumbnail url format. 2. New Tiled example.
Rev. 3	02/22/2010	Eliminated Chunk Size requirement for Tiled requests.
Rev. 4	04/26/2010	Added optional JSON output.
Rev. 5	04/11/2011	1. Added method 'processAOI2' with examples 2. Added method 'getTiledDataDirectURLs' with examples
Rev. 6	06/03/2011	1. Added method 'getTiledDataDirectURLs2' with examples 2. Added REST information
Rev. 7	03/13/2013	1. Updated LAYER_ID information for Tiled products 2. Updated examples

Request Validation Web Service Users Guide

TABLE OF CONTENTS

1	<i>GENERAL INFORMATION</i>	4
1.1	Introduction.....	4
1.2	Organization of the Manual	4
1.3	Acronyms and Abbreviations.....	5
2	<i>REQUEST VALIDATION WEB SERVICE</i>	6
2.1	Overview	6
2.2	Notes on Request Limits	7
2.3	LAYER_IDS Tag Description.....	7
2.4	Error Messages	8
2.5	processAOI.....	8
2.6	processAOI2.....	9
2.7	getTiledDataDirectURLs.....	9
2.8	getTiledDataDirectURLs2	10
2.9	Sample Java code for calling the processAOI2() Method	11
2.10	Sample Python code for calling the processAOI2() Method.....	12
2.11	Request Validation Service Test Client.....	13
2.12	processAOI2 Response for Seamless Data Request	15
2.13	processAOI2 Response for Tiled Data Request	16
2.14	getTiledDataDirectURLs Response for Tiled Data Request	17
2.15	getTiledDataDirectURLs2 Response for Tiled Data Request.....	18
2.16	REST Information	20

1 GENERAL INFORMATION

1.1 Introduction

The USGS Seamless Data Warehouse (SDW) is comprised of two systems – a “Seamless” Data Server and a “Tiled” Data Server. “Seamless” datasets exist as large data mosaics from which a user can request to clip out any portion of the dataset using their own specific geographic limits. “Tiled” datasets are already clipped to a regular grid and exist as pre-packaged data bundles residing on either an on-line or a near-line storage system.

The USGS Seamless and Tiled Server functionality includes a set of web services for the developer that can be used to create custom data access applications. This document discusses how to use the Request Validation web service to generate valid data requests. There are two additional web services that can be used with the Request Validation service to obtain the full resolution data.

Inventory Service. This is a data discovery service. This service will provide information about what data is available over a particular area of interest. The addition of new datasets and demotion of older datasets from on-line systems to near-line systems is an ongoing process. Updates to the Inventory Service occur on a monthly basis. Efforts are currently underway to include in the Inventory Service all of the datasets currently available through the USGS Seamless and Tiled servers. Use this service to determine product keys, output formats available, metadata options, and bundling options, and whether your desired data resides on the Seamless Data Server, the Tiled Data Server, or both.

Request Validation Service. Utilizing dataset information obtained from the previous call to the Inventory Service and a user-defined area of interest, this service verifies and validates the information, and then returns to the user either a fully parameterized URL(s) that can be used by the Download Service, or a direct URL to the data in the case of certain tiled datasets.

Download Service. This service initiates a request for data, queries the system to obtain a job status, and returns the requested data to the user.

Orthoimagery data that is no longer considered the “best available” is removed from on-line systems and map services on a periodic basis. When this occurs, the dataset is clipped into pre-packaged zip files and stored on near-line systems so that it can still be obtained by the public. These datasets currently show up in the Inventory Service with a STATUS = Tiled.

1.2 Organization of the Manual

Section 1 provides a general description of the system.

Section 2 provides a description of the Request Validation Web Service.

1.3 Acronyms and Abbreviations

Acronym	Definition
WMS	Web Map Service
SDDS	Seamless Data Distribution Server
SDW	Spatial Data Warehouse
TDDS	Tiled Data Distribution Server
URI	Uniform Resource Identifier
REST	Representational State Transfer
WSDL	Web Service Description Language

2 REQUEST VALIDATION WEB SERVICE

2.1 Overview

The Request Validation Service is used to generate download requests for both Seamless and Tiled datasets. For Seamless datasets, the Request Validation Service performs the following tasks:

- a. It verifies that the total area of interest is of a reasonable size. For instance, if you were to submit to the Download Service the bounding coordinates for the entire contiguous U.S. you would receive an error because that area would exceed the service limit of an estimated 1.5 gigabytes of data. This is to prevent the service from creating an unmanageable number of download and thumbnail URLs at one time. There is no limit as to the number of times a user can call the Request Validation web service, but the size of each request is limited so that the service does not waste time building URLs that may never actually be used.
- b. The service also breaks up the area of interest into more manageable pieces for downloading and usability purposes. A 1.5 gigabyte file would take quite awhile to download and may be too large to load into a client application, so the area of interest is broken up into equal size pieces between 15 and 250 megabytes. Each resulting piece must be requested and downloaded separately when retrieving the data.
- c. The service validates that the product key, output format, metadata format and compression format are valid combinations for a particular dataset. All requests for Seamless data must be submitted to the Download Service for the actual download of the data.

For Tiled datasets, the Request Validation Service can be used in either of two ways:

- d. It can provide information on how to submit download requests to the Download Service for pre-packaged data bundles. You can also retrieve thumbnail and metadata URLs for each desired data bundle.
- e. The service can also provide a direct URL that can be used to retrieve the pre-packaged data without having to use the Download Service. This method frees the developer from having to manage all of the communication that is involved with using the Download Service.

The URL to the Request Validation Service WSDL page is:

<http://extract.cr.usgs.gov/requestValidationService/wsd/RequestValidationService.wsdl>

2.2 Notes on Request Limits

The maximum size of an individual Seamless download is currently set at 250 MB. You can request smaller pieces by using the “CHUNK_SIZE” parameter in the processAOI and processAOI2 methods. Requests larger than the desired chunk size will be recursively subdivided until each piece is less than the chunk size. You can estimate the size of your request (in MB) by multiplying the area (in square degrees) of your area of interest by the mbsqdeg value (obtained from the Inventory Service) per dataset/output format combination. Requests for tiled (pre-packaged) datasets do not utilize the CHUNK_SIZE parameter since the download bundles are already created and sitting on near-line systems.

There is also a limit as to the total size (in MB) that is allowed by the Request Validation service. If the estimated size for all products combined is greater than 1.5 GB, then you will be returned an error message. You can prevent this from occurring by estimating the total size as shown in the previous paragraph before submitting your request. The reason for this limitation is simply to prevent an extremely large and unmanageable number of links being returned by the service at one time.

2.3 LAYER_IDS Tag Description

Each of the four Request Validation Service methods require an input parameter called “LAYER_IDS”. This section describes how to construct this string.

For Seamless data download requests (STATUS=Seamless), the data is clipped in real time and the download bundle is constructed per the user’s desired options. Therefore, the LAYER_IDS string must be constructed based on the desired output format, metadata format, and file compression method. Once you have decided on which formats you want the data returned as, you append those codes onto the end of the productkey.

For example, if we want to download NAIP 4-Band UTM Zone 14N (P14) in GeoTiff (02) format and the metadata in xml (X) format, compressed in a zip (Z) file, our string would look like P1402XZ. The order is: productkey (always 3 char), output format (always 2 char), metadata format (always 1 char), and file compression format (always 1 char). Therefore, we would use **P1402XZ** for the LAYER_IDS string: <LAYER_IDS>P1402XZ</LAYER_IDS>

For Tiled download requests (STATUS=Tiled), the data has been pre-packaged into existing download bundles and must simply be retrieved. There are some products that have been tiled into multiple output formats, so you must build the LAYER_ID with a productkey (always 3 char) and an output format (always 2 char) such as: <LAYER_IDS>L6N02</LAYER_IDS>. Use the Inventory Web Service to determine which output formats are available for the product you desire.

2.4 Error Messages

This example shows the xml tags that will be used when returning any error from this web service:

```
<ERROR>
  <FIELD_NAME>Udaoi</FIELD_NAME>
  <ERROR_MSG>User defined area of interest greater than max allowed:
26366.855377122745 vs 1500.0</ERROR_MSG>
</ERROR>
```

This example shows the JSON string that will be used when returning any error from this web service:

```
{"REQUEST_SERVICE_RESPONSE":{"ERROR":{"FIELD_NAME":"Udaoi","ERROR_MSG
":"User defined area of interest greater than max allowed: 276847.13340001047 vs
1500.0"},"STATUS":false}}
```

2.5 processAOI

This is the first method that was originally provided in the service. The processAOI method only takes one parameter - an xml formatted string like this:

```
<REQUEST_SERVICE_INPUT>
  <AOI_GEOMETRY>
    <EXTENT>
      <TOP></TOP>
      <BOTTOM></BOTTOM>
      <LEFT></LEFT>
      <RIGHT></RIGHT>
    </EXTENT>
    <SPATIALREFERENCE_WKID/>
  </AOI_GEOMETRY>
  <LAYER_INFORMATION>
    <LAYER_IDS> </LAYER_IDS>
  </LAYER_INFORMATION>
  <CHUNK_SIZE></CHUNK_SIZE>
  <ORIGINATOR/>
  <JSON></JSON>
</REQUEST_SERVICE_INPUT>
```

This method returns a string containing two tags for each download piece – a `DOWNLOAD_URL` and a `THUMBNAI_URL`. The `DOWNLOAD_URL` is used to call the Download Service when you want to physically download the data. The `THUMBNAI_URL` can be used at any time to view a down-sampled image of the requested piece. Please note that this method will continue to be supported, but has been superseded by the newer processAOI2 method.

2.6 processAOI2

This method is an updated version of the original processAOI method. The input remains the same - an xml formatted string like this:

```
<REQUEST_SERVICE_INPUT>
  <AOI_GEOMETRY>
    <EXTENT>
      <TOP></TOP>
      <BOTTOM></BOTTOM>
      <LEFT></LEFT>
      <RIGHT></RIGHT>
    </EXTENT>
    <SPATIALREFERENCE_WKID/>
  </AOI_GEOMETRY>
  <LAYER_INFORMATION>
    <LAYER_IDS> </LAYER_IDS>
  </LAYER_INFORMATION>
  <CHUNK_SIZE></CHUNK_SIZE>
  <ORIGINATOR/>
  <JSON></JSON>
</REQUEST_SERVICE_INPUT>
```

This method returns a string containing an additional tag for each download piece – the METADATA_URL, to go along with the DOWNLOAD_URL and THUMBNAIL_URL tags.

2.7 getTiledDataDirectURLs

This method is valid only for Tiled datasets. For those application developers who do not want to use the Download Service, this method will provide a DOWNLOAD_URL that will not require any of the communication development effort that is normally required when using the Download Service. This method will produce the three usual tags for each piece – DOWNLOAD_URL, THUMBNAIL_URL, and METADATA_URL. The input for this method is identical to the other methods – an xml formatted string like this:

```
<REQUEST_SERVICE_INPUT>
  <AOI_GEOMETRY>
    <EXTENT>
      <TOP></TOP>
      <BOTTOM></BOTTOM>
      <LEFT></LEFT>
      <RIGHT></RIGHT>
    </EXTENT>
    <SPATIALREFERENCE_WKID/>
  </AOI_GEOMETRY>
  <LAYER_INFORMATION>
    <LAYER_IDS> </LAYER_IDS>
  </LAYER_INFORMATION>
  <CHUNK_SIZE></CHUNK_SIZE>
  <ORIGINATOR/>
```

```
<JSON></JSON>
</REQUEST_SERVICE_INPUT>
```

For Tiled datasets that currently reside on the SDW On-line Tiled Data Server, this method will provide a direct URL to the pre-packaged data bundle. For Tiled datasets that currently reside on the SDW Near-line Tiled Data Server, this method will provide a direct URL to a USGS java servlet that will manage the communication necessary for submitting the request, querying the system to determine when the download bundle has been moved to a USGS web server, and finally pushing the finished data bundle to the requestor.

2.8 getTiledDataDirectURLs2

This method is a newer version of method getTiledDataDirectURLs and is also valid only for Tiled datasets. This method will produce the three usual tags for each piece – DOWNLOAD_URL, THUMBNAIL_URL, and METADATA_URL, and in addition the new BBOX tags describing the footprint of the tile and the new ID tags which provide a unique tile identifier. This additional information can be used by the developer should they want to display the tile extents in their application.

The input for this method is identical to the other methods – an xml formatted string like this:

```
<REQUEST_SERVICE_INPUT>
  <AOI_GEOMETRY>
    <EXTENT>
      <TOP></TOP>
      <BOTTOM></BOTTOM>
      <LEFT></LEFT>
      <RIGHT></RIGHT>
    </EXTENT>
    <SPATIALREFERENCE_WKID/>
  </AOI_GEOMETRY>
  <LAYER_INFORMATION>
    <LAYER_IDS> </LAYER_IDS>
  </LAYER_INFORMATION>
  <CHUNK_SIZE></CHUNK_SIZE>
  <ORIGINATOR/>
  <JSON></JSON>
</REQUEST_SERVICE_INPUT>
```

For Tiled datasets that currently reside on the SDW On-line Tiled Data Server, this method will provide a direct URL to the pre-packaged data bundle. For Tiled datasets that currently reside on the SDW Near-line Tiled Data Server, this method will provide a direct URL to a USGS java servlet that will manage the communication necessary for submitting the request, querying the system to determine when the download bundle has been moved to a USGS web server, and finally pushing the finished data bundle to the requestor.

2.9 Sample Java code for calling the processAOI2() Method

```
String validationServiceEndpoint =
    "http://extract.cr.usgs.gov/requestValidationService/services/RequestValidationService";
String validationServiceMethod = "processAOI2";
String xmlRequestString = "<REQUEST_SERVICE_INPUT>" +
    "<AOI_GEOMETRY>" +
        "<EXTENT>" +
            "<TOP> 40.840</TOP>" +
            "<BOTTOM>40.815</BOTTOM>" +
            "<LEFT>-96.715</LEFT>" +
            "<RIGHT>-96.689</RIGHT>" +
        "</EXTENT>" +
        "<SPATIALREFERENCE_WKID/>" +
    "</AOI_GEOMETRY>" +
    "<LAYER_INFORMATION>" +
        "<LAYER_IDS>P1402XZ</LAYER_IDS>" +
    "</LAYER_INFORMATION>" +
    "<CHUNK_SIZE>250</CHUNK_SIZE>" +
    "<ORIGINATOR/>" +
    "<JSON>>false</JSON>" +
    "</REQUEST_SERVICE_INPUT>";

// Make request
try {
    Service validationService = new org.apache.axis.client.Service();
    Call validationCall = (Call) validationService.createCall();
    validationCall.setTargetEndpointAddress(new java.net.URL(validationServiceEndpoint));
    validationCall.setOperationName( new QName("http://edc.usgs.gov",
        validationServiceMethod) );
    String serviceResponse = (String) validationCall.invoke( new Object[] { new
        String(xmlRequestString) } );
} catch (Exception e) {
    System.out.println("Error calling ValidationService: " + e.toString());
}

// Process serviceResponse here
```

2.10 Sample Python code for calling the processAOI2() Method

```
wsdUrlValidationService =
'http://extract.cr.usgs.gov/requestValidationService/wsd/RequestValidationService.wsdl'

try:
    #Query the Validation Service
    server = ServiceProxy(wsd=wsdUrlValidationService)

    xmlRequestString =
"<REQUEST_SERVICE_INPUT><AOI_GEOMETRY><EXTENT><TOP>40.840</TOP><BO
TTOM>40.815</BOTTOM><LEFT>-96.715</LEFT><RIGHT>-
96.689</RIGHT></EXTENT><SPATIALREFERENCE_WKID/></AOI_GEOMETRY><LAY
ER_INFORMATION><LAYER_IDS>P1402XZ</LAYER_IDS></LAYER_INFORMATION>
<CHUNK_SIZE>250</CHUNK_SIZE><JSON></JSON></REQUEST_SERVICE_INPUT>"
    processAOI2ResponseDict = server.processAOI2(requestInfoXml=xmlRequestString)
    processAOI2Response = processAOI2ResponseDict['serviceResponse']

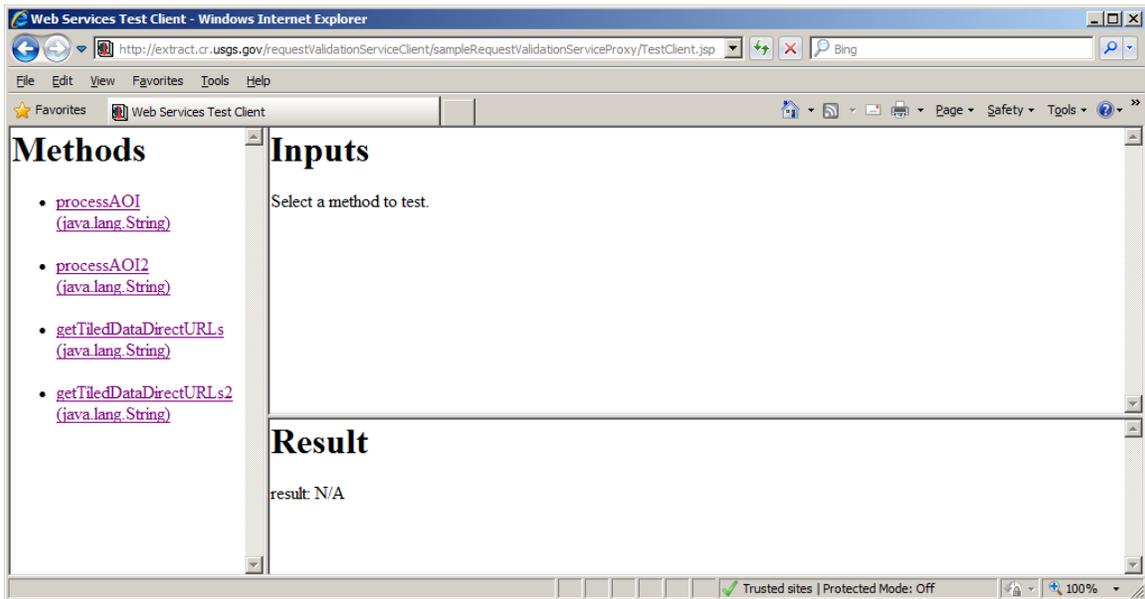
except:
    processAOI2Response = ""
    #raise

# Process processAOI2Response here
```

2.11 Request Validation Service Test Client

You can observe the operation of the Request Validation Service by using a test client that we have developed:

<http://extract.cr.usgs.gov/requestValidationServiceClient/sampleRequestValidationServiceProxy/TestClient.jsp>



Click on the `processAOI2(java.lang.String)` link in the Methods frame. Enter data into the required fields of the Inputs frame:

Inputs

TOP: (required)	<input type="text" value="40.840"/>
BOTTOM: (required)	<input type="text" value="40.815"/>
LEFT: (required)	<input type="text" value="-96.715"/>
RIGHT: (required)	<input type="text" value="-96.689"/>
SPATIALREFERENCE_WKID:	<input type="text" value="4326"/>
LAYER_IDS: (required)	<input type="text" value="P1402XZ"/>
CHUNK_SIZE: (required)	<input type="text" value="250"/>
ORIGINATOR:	<input type="text"/>
JSON:	<input type="text" value="true"/>
CALLBACK:	<input type="text"/>

<input type="button" value="Invoke"/>	<input type="button" value="Clear"/>
---------------------------------------	--------------------------------------

For Seamless datasets, valid values for `CHUNK_SIZE` are 15, 25, 50, 75, 100 or 250 megabytes. The smaller the number, the more pieces you will have to download to obtain the entire area of interest. This value is ignored when requesting Tiled datasets.

`SPATIALREFERENCE_WKID` indicates what projection the input coordinates are in. It defaults to 4326 (Geographic WGS84). If the coordinates were in some other projection such as polar coordinate then you'd want to put the correct code in this field.

`ORIGINATOR` is a client indicator. It is an optional mechanism that allows USGS to prioritize certain download requests. A valid entry can be worked out between USGS EROS and certain users when necessary as conditions dictate.

`JSON` indicates whether the client would like the response as XML or JSON. A value of `TRUE` will return a JSON string, all other values will return XML.

2.12 processAOI2 Response for Seamless Data Request

The output from the Request Validation Service is one or more “pieces”, each piece consisting of a `DOWNLOAD_URL` that is used to initiate the download via the Download Service for that piece of data, a `THUMBNAIL_URL` that can be used to display a small preview image in your application, and an associated `METADATA_URL` to view the metadata file.

This shows the response received back from the Request Validation Service using the example request from section 2.8:

```
<REQUEST_SERVICE_RESPONSE>
<PIECE>
<DOWNLOAD_URL>http://extract.cr.usgs.gov/axis2/services/DownloadService/initiateDownload?size=20&key=P14&ras=1&pfm=GeoTIFF&imsurl=-1&ms=-1&att=-1&lay=-1&fid=-1&dlpre=&lft=-96.715&rgt=-96.689&top=40.84&bot=40.815&wmd=1&mur=http://extract.cr.usgs.gov/distmeta/servlet/gov.usgs.edc.MetaBuilder&mcd=NAIP_Z14_P&mdf=XML&arc=ZIP&sde=NAIP_WGS84_WM_4band_201103&msd=NAIP.naip_z14_plygn_meta&zun=null&prj=26914&rsp=0&bnd=&bndnm=&csx=1.0&csy=1.0&ics=&ORIG=RVS
</DOWNLOAD_URL>
<THUMBNAIL_URL>http://isse.cr.usgs.gov/ArcGIS/services/Combined/USGS_EDC_Ortho_NAIP/MapServer/WMServer?servicename=&Version=1.1.1&SERVICE=WMS&request=map&layers=0&bbox=-96.715,40.815,-96.689,40.84&reaspect=false&width=40&height=38&format=jpeg&SRS=EPSG:4326&styles=</THUMBNAIL_URL>
<METADATA_URL>http://extract.cr.usgs.gov/distmeta/servlet/gov.usgs.edc.MetaBuilder?TYPE=HTML&DATASET=NAIP_Z14_P&YMAX=40.840&YMIN=40.815&XMIN=-96.689&XMAX=-96.715&coordsys=0</METADATA_URL>
</PIECE>
<STATUS>true</STATUS>
</REQUEST_SERVICE_RESPONSE>
```

This example shows the same response in JSON format:

```
{ "REQUEST_SERVICE_RESPONSE": { "PIECE": { "THUMBNAIL_URL": "http://isse.cr.usgs.gov/ArcGIS/services/Combined/USGS_EDC_Ortho_NAIP/MapServer/WMServer?servicename=&Version=1.1.1&SERVICE=WMS&request=map&layers=0&bbox=-96.715,40.815,-96.689,40.84&reaspect=false&width=40&height=38&format=jpeg&SRS=EPSG:4326&styles=", "METADATA_URL": "http://extract.cr.usgs.gov/distmeta/servlet/gov.usgs.edc.MetaBuilder?TYPE=HTML&DATASET=NAIP_Z14_P&YMAX=40.840&YMIN=40.815&XMIN=-96.689&XMAX=-96.715&coordsys=0", "DOWNLOAD_URL": "http://extract.cr.usgs.gov/axis2/services/DownloadService/initiateDownload?size=20&key=P14&ras=1&pfm=GeoTIFF&imsurl=-1&ms=-1&att=-1&lay=-1&fid=-1&dlpre=&lft=-96.715&rgt=-96.689&top=40.84&bot=40.815&wmd=1&mur=http://extract.cr.usgs.gov/distmeta/servlet/gov.usgs.edc.MetaBuilder&mcd=NAIP_Z14_P&mdf=XML&arc=ZIP&sde=NAIP_WGS84_WM_4band_201103&msd=NAIP.naip_z14_plygn_meta&zun=null&prj=26914&rsp=0&bnd=&bndnm=&csx=1.0&csy=1.0&ics=&ORIG=RVS"}, "STATUS": true } }
```

2.13 processAOI2 Request for Tiled Data Request

This example shows a request for a pre-packaged (Tiled) High Resolution Orthoimagery dataset (OBI) in GeoTIFF (02) format.

Inputs

TOP: (required)	<input type="text" value="39.825"/>
BOTTOM: (required)	<input type="text" value="39.824"/>
LEFT: (required)	<input type="text" value="-105.091"/>
RIGHT: (required)	<input type="text" value="-105.090"/>
SPATIALREFERENCE_WKID:	<input type="text" value="4326"/>
LAYER_IDS: (required)	<input type="text" value="OBI02"/>
CHUNK_SIZE: (required)	<input type="text"/>
ORIGINATOR:	<input type="text"/>
JSON:	<input type="text"/>
CALLBACK:	<input type="text"/>

<input type="button" value="Invoke"/>	<input type="button" value="Clear"/>
---------------------------------------	--------------------------------------

The resulting response shows one piece (or in this case data bundle), comprised of a DOWNLOAD_URL that utilizes the USGS Download Service, a THUMBNAIL_URL, and a METADATA_URL.

```
<REQUEST_SERVICE_RESPONSE>
<PIECE>
<DOWNLOAD_URL>http://extract.cr.usgs.gov/axis2/services/DownloadService/initiateDownload?PL=OBI&MSU=http://tddsinfo.cr.usgs.gov/ArcGIS/rest/services&MSS=USGS_EDC_TDD
S_Colorado&MSL=11&MSEA=FID&DLS=http://gisdata.usgs.gov/TDDS/DownloadFile.php?T
YPE=ortho%26FNAME=&FID=ZI&ARC=ZI&DLA=FILE_ID&EIDL=3547&siz=20&lft=-
105.09348499999999&bot=39.812941111111115&rgt=-
105.075941666666667&top=39.826468888888888&ORIG=RVS</DOWNLOAD_URL>
<THUMBNAIL_URL>http://tdds.cr.usgs.gov/browse/ortho/13S/DE/13SDE920070_200803_0x
6000m_CL_1.jpg</THUMBNAIL_URL>
<METADATA_URL>http://tdds.cr.usgs.gov/metadata/ortho/13S/DE/13SDE920070_200803_0
x6000m_CL_1.htm</METADATA_URL>
</PIECE>
<STATUS>>true</STATUS>
</REQUEST_SERVICE_RESPONSE>
```

2.14 getTiledDataDirectURLs Response for Tiled Data Request

This example shows the request and response for a pre-packaged (Tiled) High Resolution Orthoimagery dataset (OBI) in GeoTIFF (02) format.

Inputs

TOP: (required)	<input type="text" value="39.825"/>
BOTTOM: (required)	<input type="text" value="39.824"/>
LEFT: (required)	<input type="text" value="-105.091"/>
RIGHT: (required)	<input type="text" value="-105.090"/>
SPATIALREFERENCE_WKID:	<input type="text" value="4326"/>
LAYER_IDS: (required)	<input type="text" value="OBI02"/>
CHUNK_SIZE: (required)	<input type="text"/>
ORIGINATOR:	<input type="text"/>
JSON:	<input type="text"/>
CALLBACK:	<input type="text"/>

The resulting response shows a `DOWNLOAD_URL` that does not utilize the USGS Download Service and instead is a direct link to the data bundle, a `THUMBNAIL_URL`, and a `METADATA_URL`.

```
<REQUEST_SERVICE_RESPONSE>
<PIECE>
<DOWNLOAD_URL>http://gisdata.usgs.gov/TDDS/DownloadFile.php?TYPE=ortho&FNAME=
13SDE920070_200803_0x6000m_CL_1.zip&ORIG=RVS</DOWNLOAD_URL>
<THUMBNAIL_URL>http://tdds.cr.usgs.gov/browse/ortho/13S/DE/13SDE920070_200803_0x
6000m_CL_1.jpg</THUMBNAIL_URL>
<METADATA_URL>http://tdds.cr.usgs.gov/metadata/ortho/13S/DE/13SDE920070_200803_0
x6000m_CL_1.htm</METADATA_URL>
</PIECE>
</REQUEST_SERVICE_RESPONSE>
```

2.15 getTiledDataDirectURLs2 Response for Tiled Data Request

As shown in this example, the getTiledDataDirectURLs2 method provides additional tags in the response.

Inputs

TOP: (required)	<input type="text" value="39.825"/>
BOTTOM: (required)	<input type="text" value="39.824"/>
LEFT: (required)	<input type="text" value="-105.091"/>
RIGHT: (required)	<input type="text" value="-105.090"/>
SPATIALREFERENCE_WKID:	<input type="text" value="4326"/>
LAYER_IDS: (required)	<input type="text" value="OBI02"/>
CHUNK_SIZE: (required)	<input type="text"/>
ORIGINATOR:	<input type="text"/>
JSON:	<input type="text"/>
CALLBACK:	<input type="text"/>

Notice the additional BBOX and ID tags that would allow the developer to display the data bundle bounding box in their mapping application.

```
<REQUEST_SERVICE_RESPONSE>
<PIECE>
<DOWNLOAD_URL>http://gisdata.usgs.gov/TDDS/DownloadFile.php?TYPE=ortho&FNAME=
13SDE920070_200803_0x6000m_CL_1.zip&ORIG=RVS</DOWNLOAD_URL>
<THUMBNAIL_URL>http://tdds.cr.usgs.gov/browse/ortho/13S/DE/13SDE920070_200803_0x
6000m_CL_1.jpg</THUMBNAIL_URL>
<METADATA_URL>http://tdds.cr.usgs.gov/metadata/ortho/13S/DE/13SDE920070_200803_0
x6000m_CL_1.htm</METADATA_URL>
<BBOX>
  <TOP>39.82646888888888</TOP>
  <BOTTOM>39.812941111111115</BOTTOM>
  <LEFT>-105.09348499999999</LEFT>
  <RIGHT>-105.07594166666667</RIGHT>
</BBOX>
<ID>OBI_13SDE920070_200803_0x6000m_CL_1</ID>
</PIECE>
</REQUEST_SERVICE_RESPONSE>
```

Here is the response to the same request when the JSON input is set to true.

```
{"REQUEST_SERVICE_RESPONSE":{"PIECE":[{"BBOX":{"BOTTOM":39.81294111111115,"RIGHT":-105.07594166666667,"TOP":39.82646888888888,"LEFT":-105.09348499999999},"THUMBNAIL_URL":"http://tdds.cr.usgs.gov/browse/ortho/13S/DE/13SDE920070_200803_0x6000m_CL_1.jpg","METADATA_URL":"http://tdds.cr.usgs.gov/metadata/ortho/13S/DE/13SDE920070_200803_0x6000m_CL_1.htm","ID":"OBI_13SDE920070_200803_0x6000m_CL_1","DOWNLOAD_URL":"http://gisdata.usgs.gov/TDDS/DownloadFile.php?TYPE=ortho&FNAME=13SDE920070_200803_0x6000m_CL_1.zip&ORIG=RVS"}]}}
```

2.16 REST Information

REST commonly refers to a stateless client server architecture whereby web services, such as the USGS Request Validation Service, can be accessed via a simple URI. So instead of obtaining information from the Request Validation Service using the four methods listed above (processAOI, processAOI2, getTiledDataDirectURLs, getTiledDataDirectURLs2), you can retrieve the same information using a simple http request.

A. processAOI

Sample REST request for tiled data:

```
http://extract.cr.usgs.gov/requestValidationServiceClient/sampleRequestValidationServiceProxy/processAOI2.jsp?TOP=39.825&BOTTOM=39.824&LEFT=-105.091&RIGHT=-105.090&LAYER_IDS=OBI02&JSON=true
```

Response

```
{"REQUEST_SERVICE_RESPONSE":{"PIECE":[{"THUMBNAIL_URL":"http://tdds.cr.usgs.gov/browse/ortho/13S/DE/13SDE920070_200803_0x6000m_CL_1.jpg","DOWNLOAD_URL":"http://extract.cr.usgs.gov/axis2/services/DownloadService/initiateDownload?PL=OBI&MSU=http://tddsinfo.cr.usgs.gov/ArcGIS/rest/services&MSS=USGS_EDC_TDDS_Colorado&MSL=11&MSEA=FID&DLS=http://gisdata.usgs.gov/TDDS/DownloadFile.php?TYPE=ortho%26FNAME=&FID=ZI&ARC=ZI&DLA=FILE_ID&EIDL=3547&siz=20&lft=-105.09348499999999&bot=39.812941111111115&rgt=-105.07594166666667&top=39.826468888888888&ORIG=RVS"}],"STATUS":true}}
```

B. processAOI2

Sample REST request for tiled data:

```
http://extract.cr.usgs.gov/requestValidationServiceClient/sampleRequestValidationServiceProxy/processAOI2.jsp?TOP=29.426617875735545&BOTTOM=29.420069397917526&LEFT=-98.49394726686275&RIGHT=-98.48708695676768&LAYER_IDS=O4B&CHUNK_SIZE=250&JSON=true
```

Response

```
{"REQUEST_SERVICE_RESPONSE":{"PIECE":[{"THUMBNAIL_URL":"http://tdds.cr.usgs.gov/browse/ortho/13S/DE/13SDE920070_200803_0x6000m_CL_1.jpg","METADATA_URL":"http://tdds.cr.usgs.gov/metadata/ortho/13S/DE/13SDE920070_200803_0x6000m_CL_1.htm","DOWNLOAD_URL":"http://extract.cr.usgs.gov/axis2/services/DownloadService/initiateDownload?PL=OBI&MSU=http://tddsinfo.cr.usgs.gov/ArcGIS/rest/services&MSS=USGS_EDC_TDDS_Colorado&MSL=11&MSEA=FID&DLS=http://gisdata.usgs.gov/TDDS/DownloadFile.php?TYPE=ortho%26FNAME=&FID=ZI&ARC=ZI&DLA=FILE_ID&EIDL=3547&siz=20&lft=-105.09348499999999&bot=39.812941111111115&rgt=-105.07594166666667&top=39.826468888888888&ORIG=RVS"}],"STATUS":true}}
```

C. getTiledDataDirectURLs

Sample REST request for tiled data:

```
http://extract.cr.usgs.gov/requestValidationServiceClient/sampleRequestValidationServiceProxy/getTiledDataDirectURLs.jsp?TOP=39.825&BOTTOM=39.824&LEFT=-105.091&RIGHT=-105.090&LAYER_IDS=OBI02&JSON=true
```

Response

```
{"REQUEST_SERVICE_RESPONSE":{"PIECE":[{"THUMBNAIL_URL":"http://tdds.cr.usgs.gov/browse/ortho/13S/DE/13SDE920070_200803_0x6000m_CL_1.jpg","METADATA_URL":"http://tdds.cr.usgs.gov/metadata/ortho/13S/DE/13SDE920070_200803_0x6000m_CL_1.htm","DOWNLOAD_URL":"http://gisdata.usgs.gov/TDDS/DownloadFile.php?TYPE=ortho&FNAME=13SDE920070_200803_0x6000m_CL_1.zip&ORIG=RVS"}]}}
```

D. getTiledDataDirectURLs2

Sample REST request for tiled data:

```
http://extract.cr.usgs.gov/requestValidationServiceClient/sampleRequestValidationServiceProxy/getTiledDataDirectURLs2.jsp?TOP=39.825&BOTTOM=39.824&LEFT=-105.091&RIGHT=-105.090&LAYER_IDS=OBI02&JSON=true
```

Response

```
{"REQUEST_SERVICE_RESPONSE":{"PIECE":[{"BBOX":{"BOTTOM":39.812941111111115,"RIGHT":-105.07594166666667,"TOP":39.826468888888888,"LEFT":-105.09348499999999},"THUMBNAIL_URL":"http://tdds.cr.usgs.gov/browse/ortho/13S/DE/13SDE920070_200803_0x6000m_CL_1.jpg","METADATA_URL":"http://tdds.cr.usgs.gov/metadata/ortho/13S/DE/13SDE920070_200803_0x6000m_CL_1.htm","ID":"OBI_13SDE920070_200803_0x6000m_CL_1","DOWNLOAD_URL":"http://gisdata.usgs.gov/TDDS/DownloadFile.php?TYPE=ortho&FNAME=13SDE920070_200803_0x6000m_CL_1.zip&ORIG=RVS"}]}}
```