GEOSOFT

Geosoft DAP XML

Technical Note

Manual release date: May-31-13.

Please send comments or questions to info@geosoft.com

Geosoft Incorporated
Queen's Quay Terminal
207 Queen's Quay West
Suite 810, PO Box 131
Toronto, Ontario
M5J 1A7
Canada
Tel: (416) 369-0111
Fax: (416) 369-9599

Website: www.geosoft.com
E-mail: info@geosoft.com

Support

For obtaining technical support, email support@geosoft.com

If you wish to speak to a technical support representative in your region, please visit the Geosoft Support page at: www.geosoft.com/about-geosoft/contact-us/world-offices for contact information.

Geosoft DAP XML

Geosoft XML Reference

This chapter is a reference manual for developers wishing to communicate with a Geosoft DAP server using XML. A fundamental level of knowledge of XML and DAP Server technology is assumed. For more information on XML, consult the online resources found at www.w3.org/XML.

Geosoft XML is an XML-compliant protocol used to communicate with a data access protocol (DAP) server. Basic XML services include support for catalog information, metadata information, imaging, coordinate translation, and extraction services.

The communication process between a client and a DAP server through Geosoft XML is illustrated by the following simple example. In this example, the client retrieves a catalog of all the datasets stored on the DAP server.

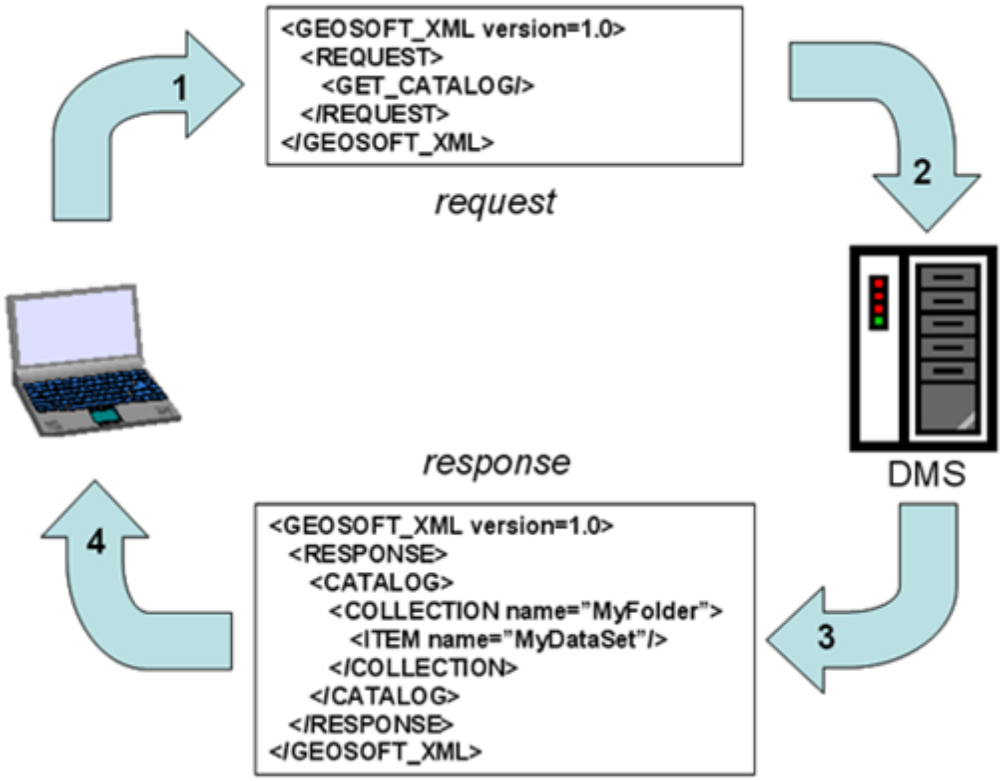
⚠ *Note: DAP XML has been replaced with a REST-based API in DAP 12.2.*

Developers wishing to communicate with a Geosoft DAP server using WCF can find information here: http://dap.geosoft.com/wcf/DAPservice.svc

XML Document Notation

Various techniques are used to guide the user through the different syntax elements present in Geosoft XML. Some of these elements are listed below:

- ()* – 0 or more elements contained within the braces may be present
- ()+ – 1 or more elements contained within the braces may be present
- [] – The content is optional
- (a|b) – Either a or b is required but both cannot be present



- Step 1: A “get_catalog” request is sent to the Geosoft DAP DMS server.
- Step 2: The request is received and processed by the server.
- Step 3: The server sends an XML response containing all the datasets it has stored. In this example, it has one dataset called “MyDataset” in the directory “MyDirectory”.
- Step 4: The catalog response is received by the client and displayed to the user.

Request – Response

The geosoft_xml tag is the first tag in any request or response. It informs the client or server of the version of Geosoft XML to use.

Format

```
<geosoft_xml version=”…”>
</geosoft_xml>
```

version	The version of the protocol the following elements are encoded with. It has the Major_Version.Minor_Version format.

Each request is enclosed by a start and end request tag, and each response is also enclosed with its own start and end tag.

Request

```
<request namespace="...">
</request>
```

namespace	Optional parameter used to direct an XML request to DAP server plug-in.

Response

```
<response>
</response>
```

Example

The following is a request for the catalog:

```
<geosoft_xml version="1.0">
<request>
<catalog/>
</request>
</geosoft_xml>
```

The DAP server sends the following response:

```
<geosoft_xml version="1.0">
<response>
<catalog>
(the catalog information is returned here)
</catalog>
</response>
</geosoft_xml>
```

XML Errors

If an error occurs, an error tag is sent back instead of the expected response tag. The error tag notifies the client that an error occurred while processing a particular request. The client matches the error to the request by the handle attribute in the request. The error is reported in the format described below.

Error Response

```
<error handle="..." code="...">
<![CDATA[...]]>
```

```
</error>
```

handle	Optional client-defined string used by the client to keep track of request and response pairs.
code	An integer specifying the particular error that occurred.

Example

```
<geosoft_xml version="1.0">
<error code="502">
<![CDATA[Error processing EXTRACT_DATA request, operation not supported by this server.]]>
</error>
</geosoft_xml>
```

Capabilities

Gets capabilities of a server.

Request

```
<capabilities handle="...">
</ capabilities >
```

handle	Optional client-defined string used by the client to keep track of request and response pairs.

Response

The response provides information about what is allowed for each supported dataset type.

```
<capabilities handle="...">
(<dataset_type name="...">
<commands>
(<command name="...">
(<parameter name="...">
(<attribute name="...">
(<value name="...">)*
</attribute>)*
</parameter>)*
</command>)*
</commands>
```

```
</dataset_type>)*
</capabilities>
```

handle	Optional client-defined string used by the client to keep track of request and response pairs.
DATASET_ TYPE	The spatial dataset type supported by DAP.
name	The name of the spatial dataset type.
COMMAND	The command that can be executed on a dataset of this type
name	The name of the command that maps to the corresponding XML tag.
PARAMETER	The child tag within a command that is further restricted by the following child nodes.
name	The name of the parameter, which maps to a corresponding XML tag that is a child node of the specified command.
ATTRIBUTE	The attribute of the parameter node that is restricted to the given list of values.
name	The name of the attribute within the parameter tag.
VALUE	The value that the attribute is allowed to have.
name	The value of the attribute.

First, the response outlines each command that is valid for the spatial dataset type. Only dataset-specific commands are listed. These include: dataset_edition, metadata, image, extract, extract_status, extract_cancel, extract_data and default_resolution. All other commands – catalog, catalog_edition, translate_coordinates, translate_bounding_box, and coordinate_system_list – are independent of the spatial dataset type, and are not included in the list of supported commands for a spatial dataset type.

Second, the response lists constraints for particular values that attributes can have in a request. Normally, all child nodes of a request are supported for all spatial dataset types. However, the response may return a list of allowable values for a particular attribute of a given tag. For instance, the “image” request has a “format” child with the “type” attribute. This attribute specifies the format of the image to return. The response can use the “parameter”, “attribute”, and “value” tags to restrict the allowable values for the “type” attribute. The following sample response limits the allowable values for the “type” attribute within the “format” tag to “image/jpeg” and “image/png”.

Example

```
<geosoft_xml version=”1.0”>
<request>
<capabilities/>
</request>
```

```
</geosoft_xml>
```

```
<geosoft_xml version="1.0">
<response>
<capabilities>
<dataset_type name="Grid">
<commands>
<command name="IMAGE">
<parameter name="format">
<attribute>
<value name="image/jpg"/>
<value name="image/png"/>
</attribute>
</parameter>
</command>
</commands>
</dataset_type>
</capabilities>
</response>
</geosoft_xml>
```

Configuration

Gets the configuration details of a DAP server.

Request

```
<configuration handle="..." />
```

handle	Optional client-defined string used by the client to keep track of request and response pairs.

Response

```
<configuration handle="...">
<meta>
<class name="...">
...
```

```
</meta>
</configuration>
```

handle	Optional client-defined string used by the client to keep track of request and response pairs.
META	See the Metadata section.

Example

```
<geosoft_xml version=”1.0”>
<request>
<configuration/>
</request>
</geosoft_xml>
```

```
<geosoft_xml version=”1.0”>
<response>
<configuration>
<meta>
<class name=\"Geosoft\">
<class name=\"Core\">
<class name=\"DAP\">
<class name=\"Configuration\">
…
</class>
</class>
</class>
</class>
</meta>
</configuration>
</response>
</geosoft_xml>
```

Catalog

Gets the list of datasets currently stored on a DAP server.

Request

```
<catalog handle="..." index ="..." max_results="..." keywords="..." count="true|false">
<bounding_box maxX="..." maxY="..." minX="..." minY="...">
<coordinate_system datum="..." protection="..." units="..."/>
</bounding_box>
<filter path="…" depth="..."/>
</catalog>
```

handle	Optional client-defined string used by the client to keep track of request and response pairs.
index	Optional; the number of datasets to skip before starting to return the catalog content.
max_results	Optional; the maximum number of results to return governed by an in-order tree traversal.
keywords	Optional; a search string.
count	Optional; if True, only the number of datasets that matchs the query is returned.
FILTER (Optional)	
path	A folder path variable that can be used to filter the list of datasets returned (in addition to the spatial and/or keyword criteria).
depth	Optional; the number of levels to return beginning at the path location.
	Examples:/Global/ – returns all datasets under the Global directory. */Global/@Global_Coastlines.hmap* – returns the specific dataset, Global_ Coastlines.hmap, if it exists.
BOUNDING_ BOX (optional)	
maxX	The maximum X coordinate.
maxY	The maximum Y coordinate.
minX	The minimum X coordinate.
minY	The minimum Y coordinate.
COORDINATE_ SYSTEM	See the Coordinate Systems section.

Response

```
< catalog handle="...">
<collection name="...">
```

```
(<item name="…" title="…" type="…" edition="…">
<bounding_box maxX="…" maxY="…" minX="…" minY="…"/>
</item>)+
(<collection name="…">…</collection>)*
</collection>
<count value="…"/>
</catalog>
<configuration version="…"/>
```

handle	Optional client-defined string used by the client to keep track of request and response pairs.
COLLECTION	A directory containing datasets and/or other directories.
name	The name of the directory.
ITEM	A dataset.
name	A unique name for the dataset used by the DAP server. It is used to specify a particular dataset in any request that has a dataset parameter.
title	User-defined name for the dataset.
type	Type of the dataset, which corresponds to one of the spatial dataset types returned by the capabilities response. E.g., image, grid, or document.
edition	The current edition of the dataset.
BOUNDING BOX	The bounding box for the dataset, always returned in WGS 84.
maxX	The maximum x coordinate.
maxY	The maximum y coordinate.
minX	The minimum x coordinate.
minY	The minimum y coordinate.
COUNT	Present if the catalog request only asked for a dataset count.
value	The number of datasets that match the search criteria.
CONFIGURATION	The current version of the server. Used to invalidate the client cache.
version	The version of the server.

Example

```
<geosoft_xml version="1.0">
```

```
<request>
<catalog/>
</request>
</geosoft_xml>
```

```
<geosoft_xml version="1.0">
<response>
<catalog edition="dog">
<collection name="Global">
<item name="Global_coastlines.hmap"
 title="World Costlines" type="Image">
<bounding_box maxX="180" maxY="90"
 minX="-180" minY="-90"/>
</item>
</collection>
</catalog>
</response>
</geosoft_xml>
```

Catalog Hierarchy

Gets the list of folders that have datasets currently stored on a DAP server.

Request

```
<catalog_hierarchy handle="..." keywords="...">
<bounding_box maxX="..." maxY="..." minX="..." minY="...">
<coordinate_system datum="..." protection="..." units="..."/>
</bounding_box>
</catalog_hierarchy>
```

handle	Optional client-defined string used by the client to keep track of request and response pairs.
keywords	Optional search string.
BOUNDING_BOX (optional)	
maxX	The maximum X coordinate.

maxY	The maximum Y coordinate.
minX	The minimum X coordinate.
minY	The minimum Y coordinate.
COORDINATE_ SYSTEM	See the Coordinate Systems section.

Response

```
< catalog_hierarchy handle="…">
<collection name="…">
(<collection name="…">…</collection>)*
</collection>
</catalog_hierarchy>
<configuration version="…"/>
```

handle	Optional client-defined string used by the client to keep track of request and response pairs.
COLLECTION	A directory containing datasets and/or other directories.
name	The name of the directory.
CONFIGURATION	The current version of the server. Used to invalidate the client cache.
version	The version of the server.

Example

```
<geosoft_xml version="1.0">
<request>
<catalog_hierarchy/>
</request>
</geosoft_xml>
```

```
<geosoft_xml version="1.0">
<response>
<catalog edition="dog">
<collection name="Global">
</collection>
```

```
</catalog>
</response>
</geosoft_xml>
```

Catalog Edition

Gets the current catalog edition.

Request

```
<catalog_edition handle="..."/>
```

handle	Optional client-defined string used by the client to keep track of request and response pairs.

Response

```
<configuration version="..."/>
<catalog_edition handle="..." edition="..." />
```

handle	Optional client-defined string used by the client to keep track of request and response pairs.
edition	The current edition of the catalog.
CONFIGURATION	The current version of the server. Used to invalidate the client cache.
version	The version of the server.

Example

```
<geosoft_xml version="1.0">
<request>
<catalog_edition/>
</request>
</geosoft_xml>
```

```
<geosoft_xml version="1.0">
<response>
<configuration version="432"/>
<catalog_edition edition="dog"/>
</response>
```

```
</geosoft_xml>
```

Dataset Edition

Gets the edition of a dataset.

Request

```
<dataset_edition handle="..." name="..."/>
```

handle	Optional client-defined string used by the client to keep track of request and response pairs.
name	The name of the dataset as given by the name attribute in the item tag of a catalog response.

Response

```
<dataset_edition handle="..." edition="..." />
```

handle	Optional client-defined string used by the client to keep track of request and response pairs.
edition	The current edition of the requested dataset.

Example

```
<geosoft_xml version="1.0">
<request>
<dataset_edition name="global_coastlines.hmap"/>
</request>
</geosoft_xml>
```

```
<geosoft_xml version="1.0">
<response>
<dataset_edition edition="dog"/>
</response>
</geosoft_xml>
```

Metadata

Retrieves metadata for a dataset.

Request

```
<metadata handle="..." name="...">
</metadata>
```

handle	Optional client-defined string used by the client to keep track of request and response pairs.
name	The name of the dataset for which to retrieve metadata, as given by the name attribute in the item tag of a catalog response.

Response

```
<meta handle="...">
<class name="..." type="...">
(<attribute name="..." type="..." value="..."/>)*
(<table>
(<item name="X">
(<attribute name="..." type="..." value="..."/>)*
</item>)*
</table>)*
(<class name="..." type="...">)*
</class>
</meta>
```

handle	Optional client-defined string used by the client to keep track of request and response pairs.
CLASS	Class containing a collection of attributes, tables, and/or other classes.
name	The name of the class.
type	The type of the class
ATTRIBUTE	An attribute of the class.
name	The name of the attribute.
type	The type of the attribute.
value	The value of the attribute.
TABLE	A collection of attributes in table format.
ITEM	A row of attributes in the table.
name	The name of the row.

Example

```
<geosoft_xml version=”1.0”>
<request>
<metadata name=”Global_coastlines.hmap”/>
</request>
</geosoft_xml>
```

```
<geosoft_xml version=”1.0”>
<response>
<metadata>
<class name="Geosoft">
<attribute name="ReferenceURL"
 type="/Predefined/Types/String"
 value="www.geosoft.com/schemas/geosoft.schema"/>
<attribute name="Description"
 type="/Predefined/Types/String"
 value="GEOSOFT metadata objects"/>
<class name="Data">
<attribute name="CentreY"
  type="/Predefined/Types/R8" value="2.531514"/>
<attribute name="CentreX"
 type="/Predefined/Types/R8" value="0" />
</class>
</class>
</metadata>
</response>
</geosoft_xml>
```

Image

Renders a series of datasets into a layered picture.

Request

```
<image handle=”…”>
<format type=”…” transparent=”(true|false)” background=”…”/>
<bounding_box maxX=”…” maxY=”…” minX=”…” minY=”…”>
```

```
<coordinate_system datum="..." protection="..." units="..."/>
</bounding_box>
<resolution height="..." width="..."/>
<layers base_map="..." index_map="...">
(<dataset name="...">)+
</layers>
</image>
```

handle	Optional client-defined string used by the client to keep track of request and response pairs.
FORMAT	
type	The requested image type. The valid types are defined in the capabilities response.
transparent	An optional True or False value; specifies whether the image should be transparent.
Background	Optional background colour for the image, specified in RGB (0xRRGGBB).
BOUNDING_ BOX	
maxX	The maximum X coordinate.
maxY	The maximum Y coordinate.
minX	The minimum X coordinate.
minY	The minimum Y coordinate.
COORDINATE_ SYSTEM	See the Coordinate Systems section.
RESOLUTION	
height	The height of the picture in pixels.
width	The width of the picture in pixels.
LAYERS	
base_map	An optional True or False value; specifies whether to draw the base map as a layer.
index_map	An optional True or False value; specifies whether to draw the index map as a layer.
DATASET	
name	The name of the dataset to draw as one of the layers of the image, as given by the name attribute in the item tag of a catalog response.

Response

```
<image handle="...">
<picture>
...
</picture>
</image>
```

handle	Optional client-defined string used by the client to keep track of request and response pairs.
PICTURE	The picture encoded in base64.

Example

```
<geosoft_xml version="1.0">
<request>
<image>
<format type="image/jpeg"/>
<bounding_box maxX="180" maxY="90"
 minX="-180" minY="-90">
<coordinate_system datum="WGS 84"/>
</bounding_box>
<resolution height="300" width="600"/>
<layers>
<dataset name="Global_coastlines.hmap"/>
<dataset name="Global_rivers.hmap"/>
</layers>
</image>
</request>
</geosoft_xml>
```

```
<geosoft_xml version="1.0">
<response>
< image>
<picture>
abcdefgdvafldafhfdkvczkhfdsafdaj#
```

```
</picture>
</ image>
</response>
</geosoft_xml>
```

Extract Data

Retrieves data for a dataset. Because the data extraction process may take time, extracting data involves first requesting the server to begin the extraction process. The process is then checked periodically to determine its status. This enables an application to provide periodic status information. Once the process is completed, the data can be requested.

Begin Extract Request

This section details the extraction process for a particular dataset.

```
<?xml version="1.0" encoding="utf-8" ?>
<geosoft_xml version="1.1">
<request>
<extract>
<bounding_box minX="-180" minY="-56" maxX="180" maxY="60">
<coordinate_system datum="WGS 84" />
</bounding_box>
<datasets>
<dataset name=”16” format="GEOSOFT/UNCOMPRESSED" resolution="1.44" />
</datasets>
</extract>
</request>
</geosoft_xml>
```

BOUNDING_ BOX	
maxX	The maximum X coordinate.
maxY	The maximum Y coordinate.
minX	The minimum X coordinate.
minY	The minimum Y coordinate.
COORDINATE_ SYSTEM	See the Coordinate Systems section.

DATASET	
name	The name of the dataset to extract, as given by the name attribute in the item tag of a catalog response.
FORMAT	
type	The requested image type. The type valid types are given in the capabilities response and depend on the type of the dataset.
RESOLUTION	
value	The resolution to extract the data at.

Begin Extract Response

The parameters returned represent the actual parameters that the extracted data conforms to. These may differ from the requested parameters due to the format of the raw data.

```
<extract handle="...">
<key name="..."/>
<dataset name="..."/>
<format type="..."/>
<bounding_box maxX="..." maxY="..." minX="..." minY="...">
<coordinate_system datum="..." projection="..." units="..."
 local_datum="..."/>
</bounding_box>
<resolution value="..."/>
</extract>
```

handle	Optional client-defined string used by the client to keep track of request and response pairs.
KEY	
name	A unique identifier of the extraction process.
DATASET	
name	The name of the dataset that is being extracted.
FORMAT	
type	The format the data is being extracted to. One of the formats listed in the capabilities response for this spatial dataset type.
BOUNDING_ BOX	

maxX	The maximum X coordinate.
maxY	The maximum Y coordinate.
minX	The minimum X coordinate.
minY	The minimum Y coordinate.
COORDINATE_SYSTEM	See the Coordinate Systems section.
RESOLUTION	
value	The actual data extraction resolution.

Extract Status Request

Gets the status of an extraction process.

```
<extract_status handle="..." key="...">
</extract_status>
```

handle	Optional client-defined string used by the client to keep track of request and response pairs.
key	The key given to the client on an extract response that specifies the particular data extraction process.

Extract Status Response

Returns the status of an extraction process.

```
<extract_status handle="..." key="...">
<status value="..." progress="..." status="..."/>
</extract_status>
```

handle	Optional client-defined string used by the client to keep track of request and response pairs.
key	The key given to the client on an extract response that specifies the particular data extraction process.
STATUS	
value	The current state of the extraction process: CANCELLED, IN_PROGRESS, or COMPLETED.
progress	The completion percentage of the extraction process
status	The current task the extraction process is working on.

Cancel Extraction

Cancels an extraction process.

```
<extract_cancel handle="..." key="...">
</extract_cancel>
```

handle	Optional client-defined string used by the client to keep track of request and response pairs.
key	The key given to the client on an extract response that specifies the particular data extraction process.

The response to the extract_cancel request is an extract_status response with status=CANCELLED.

Extract Data Request

Gets the data from an extraction process that has completed.

```
<extract_data handle="..." key="...">
</extract_data>
```

handle	Optional client-defined string used by the client to keep track of request and response pairs.
key	The key given to the client on an extract response that specifies the particular data extraction process.

Extract Data Response

Gets the data from an extraction process that has completed. The data is returned encoded in base64.

```
<extract_data handle="..." key="...">
...
</extract_data>
```

handle	Optional client-defined string used by the client to keep track of request and response pairs.
key	The key given to the client on an extract response that specifies the particular data extraction process.

⚠ *The data is encoded in base64.*

Example

Begin extraction:

```
<geosoft_xml version="1.0">
<request>
```

```
<extract>
<format type=”geosoft/compressed”/>
<bounding_box maxX=”180” maxY=”90”
 minX=”-180” minY=”-90”>
<coordinate_system datum=”WGS 84”/>
</bounding_box>
<resolution value=”0.01”/>
<dataset name=”Global_globeDEM.hgd”/>
</extract>
</request>
</geosoft_xml>
```

```
<geosoft_xml version=”1.0”>
<response>
<extract>
<key name=” hafdsfsdnzjhfdsafldhvbzjnfdsahfdas”/>
<dataset name=”Global_globeDEM.hgd”/>
<format type=”geosoft/compressed”/>
<bounding_box maxX=”-100” maxY=”35”
 minX=”-96” minY=”32”>
<coordinate_system datum=”WGS 84”/>
</bounding_box>
<resolution value=”0.01”/>
</extract>
</response>
</geosoft_xml>
```

Periodically check the status of the extraction using the key returned by the extract request:

```
<geosoft_xml version=”1.0”>
<request>
<extract_status key=”hafdsfsdnzjhfdsafldhvbzjnfdsahfdas”/>
</request>
</geosoft_xml>
```

```
<geosoft_xml version=”1.0”>
```

```
<response>
<extract_status key=”hafdsfsdnzjhfdsafldhvbzjnfdsahfdas”>
<status value=”IN_PROGRESS” progess=”20”/>
</extract_status>
</response>
</geosoft_xml>
```

After the extract_status response returns “100” as the progress, you can begin the actual data extraction. The data is returned as a ZIP file using base64 encoding:

```
<geosoft_xml version=”1.0”>
<request>
<extract_data key=”hafdsfsdnzjhfdsafldhvbzjnfdsahfdas”/>
</request>
</geosoft_xml>
```

```
<geosoft_xml version=”1.0”>
<response>
<extract_data key=”hafdsfsdnzjhfdsafldhvbzjnfdsahfdas”>
fdaklfjdafhrueanklfzhsdfahdfkl;sfjafsafkl;hafd
</extract_data>
</response>
</geosoft_xml>
```

The extraction process can be stopped at any time by sending an extract_cancel request:

```
<geosoft_xml version=”1.0”>
<request>
<extract_cancel key=”hafdsfsdnzjhfdsafldhvbzjnfdsahfdas”/>
</request>
</geosoft_xml>
```

```
<geosoft_xml version=”1.0”>
<response>
<extract_status key=”hafdsfsdnzjhfdsafldhvbzjnfdsahfdas”>
<status value=”CANCELLED”/>
```

```
</extract_status>
</response>
</geosoft_xml>
```

Translate Coordinates

Translates a series of coordinates from one projection to another.

Request

```
<translate_coordinates handle="…">
<input>
<coordinate_system datum="…" projection="…" units="…" />
</ input >
<output>
<coordinate_system datum="…" projection="…" units="…" />
</ output >
(<point x="…" y="…" z="…"/>)*
</translate_coordinates>
```

handle	Optional client-defined string used by the client to keep track of request and response pairs.
INPUT	The input coordinate system.
OUTPUT	The output coordinate system
COORDINATE_ SYSTEM	See the Coordinate Systems section.
POINT	
x	The x coordinate to translate.
y	The y coordinate to translate.
z	The optional z coordinate to translate.

Response

```
<translate_coordinates handle="…">
(<point x="…" y="…" z="…"/>)*
</translate_coordinates>
```

handle	Optional client-defined string used by the client to keep track of request and response pairs.
POINT	
x	The x coordinate to translate.
y	The y coordinate to translate.
z	The optional z coordinate to translate.

Example

```
<geosoft_xml version=”1.0”>
<request>
<translate_coordinates>
<input>
<coordinate_system datum=”NAD83”/>
</input>
<output>
<coordinate_system datum=”[NAD27] MEAN CONUS”/>
</output>
<point x=”-100.5” y=”36.122”/>
<point x =”-100.5” y=”36.233”/>
</translate_coordinates>
</request>
</geosoft_xml>
```

```
<geosoft_xml version=”1.0”>
<response>
<translate_coordinates>
<point x=”-100.5000123” y=”36.1219884”/>
<point x =”-100.5000121” y=”-36.2329882”/>
</translate_coordinates>
</response>
</geosoft_xml>
```

Translate Bounding Box

Translates a bounding box from one projection to another.

Request

<translate_bounding_box handle="…">

<bounding_box maxX="…" maxY="…" minX="…" minY="…">

<coordinate_system datum="…" projection="…" units="…" />

</bounding_box>
<coordinate_system datum="…" projection="…" units="…"/>

<resolution value="…"/>

</translate_bounding_box>

handle	Optional client-defined string used by the client to keep track of request and response pairs.
COORDINATE_ SYSTEM	The first coordinate system (the one nested under the translate_bounding_box tag) is the coordinate system of the bounding box.
	The second coordinate system is the coordinate system required for the returned bounding box.
	See the Coordinate Systems section for more information on how to describe a coordinate system.
BOUNDING_ BOX	
maxX	The maximum X coordinate.
maxY	The maximum Y coordinate.
minX	The minimum X coordinate.
minY	The minimum Y coordinate.
RESOLUTION (optional)	
value	This distance in the input coordinate system is converted to an equivalent distance in the output coordinate system based on the scale difference at the centre of the bounding box.

Response

<translate_bounding_box handle="…">

<bounding_box maxX="…" maxY="…" minX="…" minY="…"/>

<resolution value="…"/>

</ translate_bounding_box>

handle	Optional client-defined string used by the client to keep track of request and response pairs.
BOUNDING_ BOX	
maxX	The maximum X coordinate.
maxY	The maximum Y coordinate.
minX	The minimum X coordinate.
minY	The minimum Y coordinate.
RESOLUTION	Optionally returned when a resolution was supplied in the input request.
value	The new resolution for the new bounding box.

Example

```
<geosoft_xml version=”1.0”>
<request>
<translate_bounding_box>
<bounding_box maxX=”-101” maxY=”49”
 minX=”-96” minY=”45”>
<coordinate_system datum=” NAD83 to WGS 84 (4)”/></bounding_box>
<coordinate_system datum=”[NAD27] MEAN CONUS”/>
<resolution value=”0.01”/>
</translate_bounding_box>
</request>
</geosoft_xml>
```

```
<geosoft_xml version=”1.0”>
<request>
<translate_bounding_box>
<bounding_box maxX=”-101.00023” maxY=”49.00014”
 minX=”-96.000192” minY=”45.000133”/>
<resolution value=”0.01”/>
</translate_bounding_box>
</request>
</geosoft_xml>
```

Default Resolution

Gets the default resolution for a given datatype and bounding box.

Request

<default_resolution handle="..." type="...">

<bounding_box maxX="..." maxY="..." minX="..." minY="..."/>

</default_resolution>

handle	Optional client-defined string used by the client to keep track of request and response pairs.
type	The type of data contained within the bounding box: Grid, Image, Data, or Map.
BOUNDING_ BOX	
maxX	The maximum X coordinate.
maxY	The maximum Y coordinate.
minX	The minimum X coordinate.
minY	The minimum Y coordinate.

Response

<default_resolution handle="...">

<resolution value="..."/>

</default_resolution>

handle	Optional client-defined string used by the client to keep track of request and response pairs.
RESOLUTION	
value	The default resolution to extract the data at.

Example

<geosoft_xml version="1.0">

<request>

<default_resolution type="Map">

<bounding_box maxX="-101" maxY="49"
minX="-96" minY="45"/>

```
</default_resolution>
</request>
</geosoft_xml>
```

```
<geosoft_xml version="1.0">
<request>
<default_resolution>
<resolution value="0.01"/>
</default_resolution>
</request>
</geosoft_xml>
```

List Supported Coordinate Systems

Gets a list of the supported coordinate systems.

Request

```
<coordinate_system_list handle="..." list_type="..." datum="...">
</coordinate_system_list>
```

handle	Optional client-defined string used by the client to keep track of request and response pairs.
list_ type	The type of list to return; one of: "datum" "local datum description" "local datum name" "projection" "units"
datum	Optional; specifies a datum to narrow down the list to the supported projections, units, or local datums for the particular datum. If a local datum description or local datum name is specified as list_type, a datum must be provided.

Response

```
<coordinate_system_list handle="..." >
(<item name="X"/>)*
</coordinate_system_list>
```

handle	Optional client-defined string used by the client to keep track of request and response pairs.

ITEM	
name	The name of a list item.

Example

```
<geosoft_xml version=”1.0”>
<request>
<coordinate_system_list list_type=”datum”/ >
</request>
</geosoft_xml>
```

```
<geosoft_xml version=”1.0”>
<response>
<coordinate_system_list>
<item name=”WGS 84”/>
<item name=”NAD27”/>
<item name=”NAD83”/>
</coordinate_system_list>
</response>
</geosoft_xml>
```

Coordinate Systems

Any Geosoft XML request that includes a coordinate (such as a bounding_box), also includes a coordinate_system definition. You can describe coordinate systems in one of two ways:

```
<coordinate_system datum=”…” projection=”…” units=”…”/>
```

OR

```
<coordinate_system esri=”…”/>
```

Coordinate systems can be either geographic or projected. Geographic coordinate systems use longitude, latitude coordinates in degrees, and only require the datum to be defined. Projected coordinates use a map projection to define map coordinates in certain units of length on a datum. For example, the “UTM Zone 15N” projection defines coordinates in metres using a specific Transverse Mercator projection.

The coordinate system description in Geosoft XML enables you to define systems by name, by EPSG code, or explicitly by specifying all parameters on the projection. Supported names for the various coordinate systems can be found in the projection CSV tables in your “Oasis montaj/csv” directory.

The above tables may also be found in the "…/csv" directory of each of the installed DAP servers. The projection.csv tables on all DAP servers should be identical.

datum	Specifies the datum of the coordinate system in one of the following three ways: • By datum name, which can be any name returned for list_ type="datum", list_type="local datum description", or list_type="local datum name" in response to a COORDINAT_SYSTEM_LIST request. • By an EPSG code number for the datum or for a local datum transform (refer to http://www.epsg.org/). • By the name and parameters specified for a datum as follows: "Datum=name,major_axis,flattening,prime_meridian" The major axis must be defined in metres, and the prime meridian is in degrees relative to Greenwich. *Note that if you expect the/a transformation to take place between datums, you must use the first option with local datum description or local datum name, or the second option using a local datum transform code. This will fully describe the local datum transform information required to translate coordinates between datums.*
projection	Specifies the projection of the coordinate system. This is only required for projected coordinate systems. The projection can be defined in one of the following three ways: • By projection name, which can be any name returned for list_ type="projection" in response to a COORDINAT_SYSTEM_LIST request. • By an EPSG code number for the projection (refer to http://www.epsg.org/). • By the name and parameters specified for a projection as follows: Projection=name,method,length_units, P1,P2,P3,P4,P5,P6,P7,P8 where name is your projection name, method is one of the methods listed in the Projection Transformation Methods table, length_units is one of the length units listed in the Length Units table, and the parameters P1 through P8 are the method-specific parameters listed for each method the Projection Transformation Methods table. A method may have fewer than 8 parameters, in which case the parameters must be provided in the order listed in the Projection Transformation Methods table. • Distance references must be specified in length_unit. • Geographic references (latitudes and longitudes) are specified in degrees. • Longitudes in the Western hemisphere are negative. • Latitudes in the Southern hemisphere are negative. • Longitudes are relative to the prime meridian of the datum.
units	Optional coordinate system units of length, if different from length_unit defined by Projection. The unit name can be any name returned for list_

	type=”units” in response to a COORDINAT_SYSTEM_LIST request. Length units may also be defined in the form: units=“name,factor” where name is your length unit name and factor is the multiplication factor that converts your length unit to metres.
esri	*The coordinate system may be described using an Esri coordinate system string as defined for ArcGIS version 8 or later. Refer to the Esri documentation for information on how to describe an Esri coordinate system.* *Note that the Esri coordinate system descriptions do not explicitly describe a preferred local datum transform method; although a local datum transform can be inferred from many datum names. If you intend your data to be transformed across datums, you should test to ensure that your results are as you expect.*

Examples

The following describes geographic coordinates on the “ARC 1960” datum, which is widely used in Africa:

```
<coordinate_system datum="Arc 1960"/>
```

The following describes a projected coordinate system on the same datum:

```
<coordinate_system datum="Arc 1960" projection="UTM zone 37S"/>
```

Although the definition of units is redundant, the following two examples also define units, first by name only, and then by name and conversion factor:

```
<coordinate_system datum="Arc 1960" projection="UTM zone 37S" units="m"/>
<coordinate_system datum="Arc 1960" projection="UTM zone 37S" units="m,1.0"/>
```

This example explicitly defines the “ARC 1960” spheroid parameters:

```
<coordinate_system datum="Arc 1960,6378249.145000,293.465,0.0" projection="UTM zone 37S"/>
```

This example explicitly defines the UTM zone 37S projection parameters:

```
<coordinate_system datum="Arc 1960" projection="UTM zone 37S,Transverse
Mercator,m,0,39,,,0.9996,500000,10000000"/>
```

All previous examples are appropriate only for coordinate translations on the same datum (ARC 1960). If the coordinates require a datum change, a local datum transformation method must be defined for the datum. The following two examples show the same local datum transform description. The first uses the area descriptive string (from ldatum.csv), and the second uses the EPSG standard name (from datumtrf.csv):

```
<coordinate_system datum="Arc 1960 to WGS 84 (3)" projection="UTM zone 37S"/>
<coordinate_system datum="[Arc 1960] Tanzania" projection="UTM zone 37S"/>
```

The following examples use the EPSG code number to describe the coordinate systems. The first example describes the datum only, and the second example describes the local datum transform:

```
<coordinate_system datum=4210 projection=16137/>
<coordinate_system datum=1285 projection=16137/>
```

The following is an Esri string example that describes geographic coordinates on the ARC 1960 datum:

```
<coordinate_system esri=GEOGCS["Arc_1960",
 DATUM["Arc_1960",
 SPHEROID["Clarke_1880_(RGS)",6378249.145,293.4650000298]],
 PRIMEM["Greenwich",0],
 UNIT["Degree",0.01745329251994329955]]/>
```

The following is an Esri string example that describes a projected coordinate system using UTM zone 37S:

```
<coordinate_system esri=PROJCS["Arc_1960_UTM_zone_37S",
 GEOGCS["Arc_1960",
 DATUM["Arc_1960",
 SPHEROID["Clarke_1880_(RGS)",6378249.145,293.4650000298]],
 PRIMEM["Greenwich",0],
 UNIT["Degree",0.01745329251994329955]],
 PROJECTION["Transverse_Mercator"],
 PARAMETER["False_Easting",500000],
 PARAMETER["False_Northing",10000000],
 PARAMETER["Central_Meridian",39],
 PARAMETER["Scale_Factor",0.9996],
 PARAMETER["Latitude_Of_Origin",0],
 UNIT["Meter",1]]/>
```

Projection Transformation Methods

Projection Method	Required Parameters
Hotine Oblique Mercator	Latitude of projection centre Longitude of projection centre Azimuth of initial line Angle from Rectified to Skew Grid Scale factor on initial line False Easting False Northing
Laborde Oblique Mercator	Latitude of projection centre

	Longitude of projection centre Azimuth of initial line Scale factor on initial line False Easting False Northing
Lambert Conic Conformal (1SP)	Latitude of natural origin Longitude of natural origin Scale factor at natural origin False Easting False Northing
Lambert Conic Conformal (2SP)	Latitude of first standard parallel Latitude of second standard parallel Latitude of false origin Longitude of false origin Easting at false origin Northing at false origin
Lambert Conformal (2SP Belgium)	Latitude of first standard parallel Latitude of second standard parallel Latitude of false origin Longitude of false origin Easting at false origin Northing at false origin
Mercator (1SP)	Latitude of natural origin Longitude of natural origin Scale factor at natural origin False Easting False Northing
Mercator (2SP)	Latitude of first standard parallel Longitude of natural origin False Easting False Northing
New Zealand Map Grid	Latitude of natural origin Longitude of natural origin False Easting False Northing
Oblique Stereographic	Latitude of natural origin Longitude of natural origin Scale factor at natural origin False Easting False Northing
Polar Stereographic	Latitude of natural origin Longitude of natural origin Scale factor at natural origin

	False Easting False Northing
Swiss Oblique Cylindrical	Latitude of projection centre Longitude of projection centre Easting at projection centre Northing at projection centre
Transverse Mercator	Latitude of natural origin Longitude of natural origin Scale factor at natural origin False Easting False Northing
Transverse Mercator (South Oriented)	Latitude of natural origin Longitude of natural origin Scale factor at natural origin False Easting False Northing
***Albers Conic**	Latitude of first standard parallel Latitude of second standard parallel Latitude of false origin Longitude of false origin Easting at false origin Northing at false origin
***Equidistant Conic**	Latitude of first standard parallel Latitude of second standard parallel Latitude of false origin Longitude of false origin Easting at false origin Northing at false origin
***Polyconic**	Latitude of false origin Longitude of false origin Scale factor at natural origin Easting at false origin Northing at false origin

Length Units

Unit	Description	Factor to Metres
m	Metre	1.0
ft	Foot	0.3048
ftUS	US survey foot	0.3048006096012
ftMA	Modified American foot	0.3048122529845

ftCla	Clarke's foot	0.3047972651151
ftInd	Indian foot (Clarke)	0.3047995102481
ftSe	foot (Sears)	0.3047994715387
lkCla	link (Clarke)	0.201166194976
lkBen	link (Benoit)	0.2011678249438
lkSe	link (Sears)	0.2011676512155
chBen	chain (Benoit)	20.1167824943759
chSe	chain (Sears)	20.1167651215526
ydSe	yard (Sears)	0.914398414616
ydInd	Indian yard	0.9143985539701
fathom	Fathom	1.8288
nautmi	nautical mile	1852.0
mGer	German legal metre	1.0000135965
dega	degrees (angular)	n/a