

CEOS Working Group
on Information Systems and Services

WGISS Connected Data Assets

Error Handling Guide

Publication Date: 2019-04-04

Updated : 2019-04-09

Document version: V1.0

Category: WGISS Technical Document

Editors: WGISS CDA System-Level Team

Overview

Over the past several years, acting as a proxy between standards-based clients (OpenSearch and CSW) and a heterogeneous collection of remote data providers, CWIC has gained significant experience with various error conditions. The requirement to provide meaningful status messages to clients when exceptional conditions are encountered in request and response processing is a challenge. Moving forward with the WGISS Connected Data Assets will require additional coordination with both FedEO and with new remote data partners, while at the same time attempting to minimize the need for additional development initiatives.

The CSW protocol has very limited capabilities for reporting information about errors encountered in processing requests. These tend to be software exceptions, largely handled in the source code of the CWIC connectors and reported back to the client as HTTP status codes.

For OpenSearch, the capabilities for exception handling are even more meager, relying almost exclusively on HTTP status codes with little supplemental information.

The aim of this document is to provide information on how CWIC currently handles exceptional error situations and to present use cases as a starting point for discussion about strategies for handling additional issues.

Current CWIC Exception Handling

OpenSearch/Atom

CEOS-BP-017 - Exception codes [Recommended]

CWIC currently supports the CEOS Best Practices recommendations for handling errors in the request/response cycle. In particular, the OpenSearch Geo and Time Extensions [OGC 10-032r8] recommends the use of HTTP status codes as following, 4xx for client errors, and 5xx for server errors.

- 400 Bad Request: The request has an invalid syntax (i.e. badly formatted geometry)
- 413 Request Entity Too Large: The request originates too many returnable hits
- 415 Unsupported media type: Media type in the request is not available or valid.
- 500 Internal Server Error: Default code for the server side for an execution error.
- 501 Not Implemented: When requesting an unimplemented feature (e.g. relation operator not supported).
- 503 Service Unavailable: When the search service is temporarily not available (due to overload or other reasons).
- 504 Gateway Timeout: When the search engine is a broker or aggregator to other services that fail to produce an answer within a giving time frame.

CEOS OpenSearch implementations are recommended to support these codes.

CSW

Useful status and error messages help the Connector manage client sessions effectively. Any limitations on submitted search requests to the inventory systems should be noted in the response (e.g., “too many records requested”, “search timed out”) so that predictable error-handling can be managed by the Connector.

The CSW protocol itself has relatively limited capabilities for documenting errors which may arise during a transaction. The CWIC development team is investigating ways to enhance this functionality to provide better information to the end user or client. In order to support this eventuality, it would be useful for the inventory search system to attempt to return sensible and relevant http status codes (where applicable) if something goes wrong with the search or, perhaps even better, a small, descriptive response document (in XML or JSON or whatever the default format might be) providing error codes and error text. In this way, the CWIC connectors can distinguish the type of error arising at the inventory system from those arising elsewhere and take appropriate action. There are no specific recommendations at this time but this should be part of ongoing discussions between the Connector developers and the Data Partner’s support staff.

Use Cases

Server limits

Use Case: A user wants to request a search for granules obtained within the last calendar year. The remote server permits requests only for six months at a time. How should the user’s client be informed that the temporal range of the request is too large?

Discussion: Remote data servers can have internal limits on search dimensions for reasons of resource limitation, search time or other outside constraints. There is not a good way to communicate those limitations through the WCDA proxy back to remote clients within the constraints of the OpenSearch OSDD or CSW Capabilities document. These are not conventional limitations such as spatial or temporal boundaries. More commonly, they are constraints on spatial footprint, temporal range, or result set size, among others.

The issue for WCDA partners is to determine how to identify such limits from remote server and how to inform clients before submitting request so that requests can be well-formed and comply with the available limitations. Examples may include limits on the size of the spatial footprint (search area must be less than 20 square degrees), temporal range (search range must be less than 3 months) or result set size (search must yield fewer than 1000 results). These are not generally known a priori.

Time outs

Use Case: A user submits a valid search for granules to a remote data collection. The search has a large spatial footprint and seems to take a long time and then the remote server connection disappears.

Use Case: A user submits a valid search for granules and requests a maximum of 10000 hits. Before a response is received, the connection to the remote server disappears.

Discussion: Searches occasionally can take longer than the time the HTTP connection can be maintained and the connection may terminate before the search completes. The time it takes a search to run to completion cannot be anticipated by the remote data provider. To the client, a long running search may just disappear without indicating an error. How should the client be made aware that the error is due to a search timeout because the search took too long to complete?

Http status code extensions?

It is possible, although not advisable, to extend the standard HTTP status codes to indicate special types of errors. Since these are not standardized response codes, naïve clients may not recognize them although specialized WCDA clients may be aware of their meaning. Should WGISS data partners adopt specialized, non-standard HTTP status codes to provide more detailed information about possible errors.

Atom response text?

For OpenSearch, it may be possible to provide some predictable descriptive text in the (Atom) HTTP response body in addition to the HTTP status code in the HTTP response header. This might, at least in the case of OpenSearch, provide additional annotative text on the cause of errors that clients could incorporate to better inform users of problems with the request and give some guidance to formulating more successful requests. An implication of this is that

clients should not assume responses are Atom. In case of errors, the server might return an HTML exception report.

For non-fatal exceptions, the atom response may be extended with additional information (e.g. by including additional diagnostics information¹). In case of fatal exceptions, the HTTP message body may not contain an Atom feed but an alternative structure. See for instance the use of `ows:Exception Report` in FedEO (Annex B).

OpenSearch `<os:query>` Feedback

The Atom response typically contains an `<os:Query role="request" .../>` element which echoes the original user request parameters. In case the original request parameters are not acceptable to the server due to server limits, the server may decide to alter the parameters (e.g. reduce a temporal or geographical constraint) and inform the client about this by including one or more `<os:Query role="correction" totalResults=".." title=".." .../>`² response elements with a corrected set of search parameters. In addition the server can include a "title" attribute describing the modification, e.g. `title="Time interval should be less than 1 year."` and an indication of how many responses it will return for the modified query.

```
<os:Query eo:parentIdentifier="LANDSAT.ETM.GTC" role="request"
time:end="2010-01-31T00:00:00Z" time:start="2000-01-01T00:00:00Z"/>
```

```
<os:Query eo:parentIdentifier="LANDSAT.ETM.GTC" role="correction"
time:end="2001-01-31T00:00:00Z" time:start="2000-01-01T00:00:00Z"
totalResults="37695" title="Time interval should be maximum one year."/>
```

¹ <http://docs.oasis-open.org/search-ws/searchRetrieve/v1.0/os/part3-sru2.0/searchRetrieve-v1.0-os-part3-sru2.0.html#diagnostics>

² http://www.opensearch.org/Specifications/OpenSearch/1.1/Draft_5#OpenSearch_Query_element

Distributed Search

Use Case: The CWIC or FedEO broker/gateway may need to combine results from two remote metadata sources for responding to a client request. In case one of the data partners provides results and the other data partner raises an exception, the broker has to be able to return partial results and inform the client that the set of responses is incomplete. This requires a mechanism for non-fatal exceptions, thus returning an Atom feed with the partial results in combination with a diagnostic message informing the client about the exception.

Appendix A - CWIC Exceptions and HTTP Status Codes

While the CWIC CSW implementation is able to throw exceptions back to the client when unrecoverable errors occur, the accepted mechanism for handling errors in the OpenSearch environment is to use HTTP Status Codes for designating errors. It is, however possible to return supplementary, human-readable text along with the status code, so this document defines the HTTP Status Code associated with each CWIC exception defined in the CWIC Exception Handling document.

In general, 4xx status codes, i.e., client errors, will refer to the CWIC server (Mediator and Connectors) since the CWIC component do the basic syntax checking on incoming CSW requests. A few may be related to internal CWIC server errors.

The 5xx status codes, i.e, server errors, will generally refer to problems in connecting to or parsing responses from the remote data provider system, since these exceptions are generally not correctable nor controllable from the CWIC software.

XML Errors

Invalid XML Request

- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: UNABLE_TO_PARSE_REQUEST - Incoming request payload could not be parsed

Incoming CSW Request

- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: MISSING_REQUEST - Request type not specified
- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: UNRECOGNIZED_REQUEST - Only GetRecords and GetRecordById supported for HTTP POST method
- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: UNRECOGNIZED_REQUEST - Only GetCapabilities, DescribeRecord and GetRecordById supported for HTTP GET method
- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: MISSING_SERVICE - Service attribute not specified

- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: INVALID_SERVICE - Requested service value not valid (i.e., not CSW)
- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: MISSING_VERSION - CSW version not specified
- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: UNSUPPORTED_VERSION - Requested CSW version not supported (i.e., not v2.0.2)
- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: INVALID_OUTPUTSCHEMA - Requested outputSchema not valid, only <http://www.opengis.net/cat/csw/2.0.2> or <http://www.isotc211.org/2005/gmd> supported
- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: MISSING_TYPENAMES - No typeName attribute defined
- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: INVALID_TYPENAMES - Only typeName csw:Record and gmd:MD_Metadata
- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: INVALID_ELEMENTSETNAME – ElementSetName must be 'brief', 'summary' or 'full'
- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: MISSING_OUTPUTFORMAT - No outputFormat attribute defined
- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: INVALID_OUTPUTFORMAT - Only 'application/xml' is supported for outputFormat parameter
- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: INVALID_QUERYABLES - Requested Queryables parameter value not valid. Check Capabilities document to get the supported Queryables

GetCapabilities Request

- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: INVALID_SECTIONS – Requested section parameter value not valid
- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: INVALID_ACCEPTFORMATS - Requested AcceptFormats parameter value not valid

DescribeRecord Request

- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: INVALID_NAMESPACE – Requested NAMESPACE parameter value not valid. Please note that the value of the NAMESPACE parameter shall be properly escaped for URL encoding

- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: INVALID_SCHEMALANGUAGE – Requested schemaLanguage parameter value not valid. Only 'XMLSCHEMA' is supported for schemaLanguage parameter
- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: INVALID_TYPENAME - Only csw:Record and gmd:MD_Metadata are supported for typeName parameter
- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: INVALID_OUTPUTFORMAT - Only 'application/xml' is supported for outputFormat parameter

GetRecords Request

- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: MISSING_GETRECORDS – No {<http://www.opengis.net/cat/csw/2.0.2>}/GetRecords element defined in this request
- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: INVALID_RESULTTYPE – Requested resultType was not “hits” or “results”
- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: INVALID_STARTPOSITION – Requested startPosition was not a positive integer
- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: INVALID_MAXRECORDS – Requested maxRecords was not a positive integer
- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: MISSING_QUERY – <Query> not found in this request
- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: UNSUPPORTED_REQUEST – <ogc:SortBy> is not yet supported by CWIC
- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: MISSING_DATASET – No dataset dc:subject or dc:identifier defined in this request
- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: INVALID_DATASET – Unrecognized dataset
- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: INVALID_STARTDATE_VALUE – Start Date and Time not valid. Supported formats are 'yyyy-MM-dd', 'yyyy-MM-ddTHH:mm:ssZ' or 'yyyy-MM-dd HH:mm:ss'
- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: INVALID_ENDDATE_VALUE – End Date and Time not valid. Supported formats are 'yyyy-MM-dd', 'yyyy-MM-ddTHH:mm:ssZ' or 'yyyy-MM-dd HH:mm:ss'

- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: INCONSISTENT_START_END_DATE_VALUES – Start date and time must be before end date and time
- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: MISSING_BBOX – <BBOX> not found in this request and is required by data provider
- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: MISSING_CRS – Required srsName attribute was not found in <Envelope>
- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: INVALID_CRS – Only 'EPSG:4326' or 'http://www.opengis.net/def/crs/EPSG/0/4326' is supported for srsName attribute
- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: INVALID_QUERYABLE – Queryable specified in request is incorrect

GetRecordById Request

- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: MISSING_GETRECORDBYID – No {http://www.opengis.net/cat/csw/2.0.2}/GetRecordById element defined in this request
- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: MISSING_RECORDID – No csw:ID element specified in request
- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: MISSING_RECORDID_VALUE – No csw:ID value specified in request
- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: INVALID_RECORDID – Found unrecognized record identifier
- ❖ **(400 Bad Request)** REQUEST_EXCEPTION: INVALID_DATASET – Found unrecognized dataset from the record identifier

Request Limitations & Data Validation

Request Limitations

- ❖ **(400 Bad Request)** REQUEST_LIMITATION: TOO_MANY_RECORDS – The request asked for more records than can be handled
- ❖ **(400 Bad Request)** REQUEST_LIMITATION: TEMPEXTENT_TOO_BIG – The request asked for a larger temporal extent than can be handled
- ❖ **(400 Bad Request)** REQUEST_LIMITATION: SPATIAL_TOO_BIG – The request asked for a larger spatial extent than can be handled

- ❖ **(400 Bad Request)** REQUEST_LIMITATION: UNSUPPORTED_PARAMETER – The requested parameter was not supported
- ❖ **(400 Bad Request)** REQUEST_LIMITATION: UNSUPPORTED_MULTIPLE_DATASET – The request asked for multiple datasets in single GetRecords request
- ❖ **(400 Bad Request)** REQUEST_LIMITATION: MISSING_TEMPORAL_RANGE – Temporal range parameters are required by the requested catalog
- ❖ **(400 Bad Request)** REQUEST_LIMITATION: INVALID_CWIC_RECORDID_FORMAT – Found unrecognized record identifier

Data Validation

- ❖ **(400 Bad Request)** REQUEST_LIMITATION: BAD_LL_LON_VALUE – Longitude of lower left corner is not numeric
- ❖ **(400 Bad Request)** REQUEST_LIMITATION: INVALID_LL_LON_VALUE – Longitude of lower left corner should be between [-180.0, 180.0]
- ❖ **(400 Bad Request)** REQUEST_LIMITATION: BAD_UR_LON_VALUE – Longitude of upper right corner is not numeric
- ❖ **(400 Bad Request)** REQUEST_LIMITATION: INVALID_UR_LON_VALUE – Longitude of upper right corner should be between [-180.0, 180.0]
- ❖ **(400 Bad Request)** REQUEST_LIMITATION: BAD_LL_LAT_VALUE – Latitude of lower left corner is not numeric
- ❖ **(400 Bad Request)** REQUEST_LIMITATION: INVALID_LL_LAT_VALUE – Latitude of lower left corner should be between [-90.0, 90.0]
- ❖ **(400 Bad Request)** REQUEST_LIMITATION: BAD_UR_LAT_VALUE – Latitude of upper right corner is not numeric
- ❖ **(400 Bad Request)** REQUEST_LIMITATION: INVALID_UR_LAT_VALUE – Latitude of upper right corner should be between [-90.0, 90.0]
- ❖ **(400 Bad Request)** REQUEST_LIMITATION: INCONSISTENT_LAT_VALUES – Latitude of lower left corner cannot be greater than upper right corner

Remote System & Communication Errors

- ❖ **(500 Internal Server Error) COMM_EXCEPTION: REMOTE_SEARCH_FAILED** – Could not reach remote server
Generally, this means the remote server is down.
- ❖ **(500 Internal Server Error) COMM_EXCEPTION: REMOTE_SEARCH_FAILED** – Server returned error
The search failed and returned an HTTP error. The exception message contains the error text returned by the server.
- ❖ **(504 Gateway Timeout) COMM_EXCEPTION: NO_RESPONSE** – Failed to get response from remote server
The connection to remote server timed out.
- ❖ **(500 Internal Server Error) COMM_EXCEPTION: NO_CONNECTION_INFO** – No connection information is available. Contact CWIC staff.
- ❖ **(500 Internal Server Error) COMM_EXCEPTION: INVALID_CONNECTION_INFO** – Connection information is not valid for opening a connection to the remote server. Contact CWIC staff.
- ❖ **(502 Bad Gateway) COMM_EXCEPTION: UNABLE_TO_PARSE_RESPONSE** – Response from remote server could not be parsed (e.g., the connector received a bad response from the remote server)
- ❖ **(502 Bad Gateway) COMM_EXCEPTION: UNRECOGNIZED_RESPONSE** – Response from remote server could not be successfully converted to a valid CSW response.
- ❖ **(500 Internal Server Error) COMM_EXCEPTION: MISSING_REQUEST_TEMPLATE** – the template used to generate a SOAP request to the USGS/LSI system could not be found. Contact the CWIC support staff.

Other Exceptions

- ❖ **(500 Internal Server Error) NumberFormatException** – Conversion from string to numeric failed
- ❖ **(500 Internal Server Error) CONFIG_EXCEPTION: FILE_READ_ERROR** – unable to read contents of file
- ❖ **(500 Internal Server Error) CONFIG_EXCEPTION: FILE_OPEN_ERROR** – unable to open file for reading

- ❖ **(500 Internal Server Error) VALIDATION_ERROR** – error in validating XML (not user-accessible)
- ❖ **(500 Internal Server Error) CONFIG_EXCEPTION: MISSING_REQUEST_TEMPLATE** – the Velocity template for the USGS/LSI SOAP request could not be located. Contact the CWIC support staff
- ❖ **(500 Internal Server Error) CONFIG_EXCEPTION: CONFIGURATION_ERROR** – connection configuration could not be read. Contact the CWIC support staff

Appendix B - FedEO Exceptions and HTTP Status Codes

The HTTP message body may contain a standard `ows:ExceptionReport` element as defined in OGC 06-121r9, OGC Web Services Common Standard³ (chapter 8). An OpenSearch exception response has 4 elements:

- HTTP exception code 4XX or 5XX as per OGC 13-026r8
- `ows:Exception exceptionCode`: as per table 28 of OGC 06-121r9,
- `ows:Exception exceptionLocator`: as per OGC 06-121r9,
- `ows:ExceptionText` with a human readable message as per OGC 06-121r9

Example 1: OpenSearch response in case of exceptions

```
HTTP/1.1 400 Bad Request
Content-Type: application/xml
...

<?xml version="1.0" encoding="UTF-8"?>
<ows:ExceptionReport xmlns:ows="http://www.opengis.net/ows/2.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/ows/2.0
http://schemas.opengis.net/ows/2.0/owsExceptionReport.xsd" version="1.0.0" xml:lang="en">
  <ows:Exception exceptionCode="InvalidParameterValue" locator="httpAccept">
    <ows:ExceptionText>MIME type {application/rdf+xml} is not supported for dataset series
{urn:ogc:def:EOP:MDA-GSI:RSAT2_NRT}.</ows:ExceptionText>
  </ows:Exception>
</ows:ExceptionReport>
```

The following table lists the possible exceptions.

³ http://portal.opengeospatial.org/files/?artifact_id=38867

HTTP exception Code	OWS exceptionCode	OWS ExceptionText
400	InvalidParameterValue	Parameter {PARAM_NAME} has an invalid value.
400	InvalidParameterValue	Parameter {PARAM_NAME} should be a number.
400	InvalidParameterValue	Parameter {PARAM_NAME} should be an integer.
400	InvalidParameterValue	Missing parameter {PARAM_NAME}.
400	InvalidParameterValue	Parameter {PARAM_NAME} should be a number and between [UPPER_BOUND,LOWER_BOUND].
400	InvalidParameterValue	Parameter {PARAM_NAME} should be a range.
400	InvalidParameterValue	Parameter {PARAM_NAME} should have upper bound value.
400	InvalidParameterValue	Could not find dataset series {IDENTIFIER_VALUE} in the system configuration.
400	InvalidParameterValue	{time:end} must be after {time:start}
400	InvalidParameterValue	Parameter {PARAM_NAME} is not supported by dataset series {IDENTIFIER_VALUE}.
401	InvalidParameterValue	Missing parameter credentials.
403	InvalidParameterValue	Forbidden. Your credentials were checked, but you have no access.
415	InvalidParameterValue	MIME type {MIME_TYPE} is not supported for dataset series {IDENTIFIER_VALUE}.
500	NoApplicableCode	Internal Server Error.
500	NoApplicableCode	{Error message received from backend catalogue}

501	InvalidParameterValue	Parameter {PARAM_NAME} is not supported by dataset series {IDENTIFIER_VALUE}.
-----	-----------------------	---

Table 1: List of exception codes