



esac

European Space Astronomy Centre
P.O. Box 78
28691 Villanueva de la Cañada
Madrid
Spain
T +34 91 8131 100
F +34 91 8131 139
www.esa.int

DOCUMENT

CSA Archive Inter-Operability (AIO) System User's Manual

Prepared by	Science Archives Team
Reference	
Issue	1
Revision	8
Date of Issue	03/02/2015
Status	Under review
Document Type	
Distribution	



APPROVAL

Title CSA Archive Inter-Operability (AIO) System User's Manual	
Issue 1	Revision 8
Author Science Archives Team	Date 03/02/2015
Approved by	Date
Head of the ESAC Science Archives Team	

CHANGE LOG

Reason for change	Issue	Revision	Date
Add information about Inventory Requests	1	1	21/06/2013
Added wild card option to product requests and asynchronous product requests	1	2	04/07/2013
Added missing information	1	3	26/07/2013
Archive renamed from Cluster Final Archive to Cluster Science Archive (CSA)	1	4	25/09/2013
Correct typos at product request and replace cfa instance at URLs by csa	1	5	31/10/2013
Included new functionality: data streaming	1	6	17/01/2014
Improve description of data download, specially with WGET	1	7	15/09/2014
Include Double Star example and updated IDL login routine	1	8	03/02/2015

CHANGE RECORD

Issue 1	Revision 8		
Reason for change	Date	Pages	Paragraph(s)
Updated IDL login routine to deal with special chars	03/02/2015	45-47	
Added example of Double Star data download	03/02/2015	17	



Table of contents:

1	INTRODUCTION	4
1.1	Use Cases	4
1.1.1	UC1 – General query to be executed immediately	4
1.1.2	UC2 – General query to be executed asynchronously	5
1.1.3	UC3 – List files	5
1.1.4	UC4 – Query by ingest date	5
1.1.5	UC5 – Query specifying download filesize limit	6
1.1.6	UC6 – List datasets without description	6
1.1.7	UC7 – List datasets with description	6
1.1.8	UC8 – Get dataset metadata information	6
1.1.9	UC9 – Include dataset reference documents	7
1.1.10	UC10 – Inventory information	7
2	METADATA REQUESTS	8
2.1	Metadata Request Parameters	8
2.2	User of QUERY parameter	10
3	INVENTORY REQUESTS	13
4	DATA REQUESTS	15
4.1	Product requests	15
4.2	Header requests	20
5	ASYNCHRONOUS DATA REQUESTS	21
5.1	Asynchronous Product requests	21
6	STREAMING DATA REQUESTS	24
6.1	Streaming Product requests	24
7	CREDENTIALS REQUESTS	27
7.1	Login requests	27
7.2	Logout requests	29
8	LIST OF CSA ENTITIES AND ATTRIBUTES	30
9	APPENDIX: HOW TO RUN THE DEFAULT CLIENT	33
9.1	Syntax for the Java client	33
9.1.1	Syntax for the shell script with the Java client	34
9.2	Examples of use	35
9.3	Equivalence to CAIO data requests	36
10	APPENDIX: HOW TO USE CAIO WITH WGET	38
10.1	Data download	38
10.1.1	Direct download	39
10.1.2	Asynchronous download	39
10.1.3	Data streaming	40
10.2	Inventory request	41
10.3	Metadata request	41
10.4	How to specify the output file?	41
10.5	How to create an authentication cookie?	42
10.6	Hint on descriptive error messages	43
11	APPENDIX: HOW TO LINK CAIO WITH MATLAB OR IDL	44
11.1	Matlab	44
11.1.1	For users with Matlab 2013a or later releases	44
11.1.2	For users with Matlab 2012b or earlier releases	44
11.2	IDL	45



1 INTRODUCTION

Within the ESA's Science Operations Department, the SAT in ESAC, Spain, is responsible of developing and maintaining ESA Scientific Archives. For the Cluster Science Archive (CSA), the standard way to access to this ESA archive is through a powerful Java interface (<http://csa.esac.esa.int/csa/csa.jnlp>).

To allow access to external applications and expert or normal users that need to retrieve data directly from the archive bypassing the Java User Interface, an integrated Archive Inter-Operability system (AIO) has been developed for CSA. This AIO system can be accessed using standard HTTP requests.

This subsystem shares the core system of the standard CSA User Interface.

It accepts three kinds of requests:

- Metadata request: To be used when we want to retrieve Inventory information, lists of Datasets or Files that fulfil a certain criteria or the metadata details of dataset(s) and file(s)
- Data request: To be used when we want to retrieve files of the data observed or dataset metadata header files
- Asynchronous Data request: To be used when we want to retrieve asynchronously files of the data observed
- Streaming Data request: To be use when we want to retrieve files of the data observed via the streaming functionality

Please note that for a synchronous data retrieval there is a limit of 1GB, while for an asynchronous data retrieval the limit is 50GB.

Ideally, this interface should be used by a script, application code or command line tools. First, we retrieve a list of datasets through a metadata request, and then we issue a data request for each one of them in order to retrieve the file(s).

Examples of using the CAIO interface from a Java client, wget , Matlab or IDL can be found at the appendix section. Meanwhile, at section 1.1 some use cases of CAIO pointing to its corresponding type of request are presented.

1.1 Use Cases

1.1.1 UC1 – General query to be executed immediately

This use case retrieves 2 CEF products for 1 day, split into 6 hour files, and the request will be executed immediately.

Type of request: **Data product request**



Example:

Simply replace `<username>` and `<pwd>` by your CSA username and password in the example below.

```
http://csa.esac.esa.int/csa/aio/product-
action?USERNAME=<username>&PASSWORD=<pwd>&DATASET_ID=C3_CP_ASP_IONC&DATASET_I
D=C1_CP_EFW_L1_P2&START_DATE=2004-06-18T00:00:00Z&END_DATE=2004-06-
19T00:00:00Z&DELIVERY_INTERVAL=hourly6&NON_BROWSER
```

1.1.2 UC2 – General query to be executed asynchronously

This use case retrieves 3 CEF products for 4 days, split into day files. The request will be executed asynchronously and will send an email when complete.

Type of request: **Asynchronous data request**

Example:

Simply replace `<username>` and `<pwd>` by your CSA username and password in the example below.

```
http://csa.esac.esa.int/csa/aio/async-product-
action?USERNAME=<username>&PASSWORD=<pwd>&DATASET_ID=C1_CP_ED1_EGD&DATASET_I
D=C1_CP_EFW_L3_P&DATASET_ID=C1_CP_FGM_FULL&START_DATE=2001-02-
01T00:00:00Z&END_DATE=2001-02-05T00:00:00Z&DELIVERY_INTERVAL=Daily&NON_BROWSER
```

1.1.3 UC3 – List files

This use case retrieves the list of files that match a search criteria. The response will be in CSV format. Note that the value of QUERY parameter should be URL encoded.

Type of request: **Metadata request**

Example:

```
http://csa.esac.esa.int/csa/aio/metadata-
action?SELECTED_FIELDS=FILE.FILE_NAME,FILE.START_DATE,FILE.END_DATE,FILE.CAA_INGES
TION_DATE&RESOURCE_CLASS=FILE&QUERY=(DATASET.DATASET_ID%20like%20'C1_CP_ED1_EG
D'%20OR%20DATASET.DATASET_ID%20like%20'C1_CP_EFW_L3_P'%20OR%20DATASET.DATASET_I
D%20like%20'C1_CP_FGM_FULL')%20AND%20FILE.START_DATE%20<=%20'2001-02-
05'%20AND%20FILE.END_DATE%20>=%20'2001-02-
01'%20&PAGE_SIZE=1000&PAGE=1&RETURN_TYPE=CSV
```

1.1.4 UC4 – Query by ingest date

This use case retrieves a particular product that have been ingested since a particular date, in daily files.

Type of request: **Data product request**

Example:

Simply replace `<username>` and `<pwd>` by your CSA username and password in the example below.



```
http://csa.esac.esa.int/csa/aio/product-
action?USERNAME=<username>&PASSWORD=<pwd>&DATASET_ID=C1_CP_EDI_EGD&START_DAT
E=2001-02-01T00:00:00Z&END_DATE=2001-02-
05T00:00:00Z&DELIVERY_INTERVAL=Daily&INGESTED_SINCE=2010-03-
10T10:21:00Z&NON_BROWSER
```

1.1.5 UC5 – Query specifying download filesize limit

This use case retrieves a particular product, with the additional criteria that the result cannot exceed a filesize limit.

Type of request: **Data product request**

Example:

Simply replace <username> and <pwd> by your CSA username and password in the example below.

```
http://csa.esac.esa.int/csa/aio/product-
action?USERNAME=<username>&PASSWORD=<pwd>&DATASET_ID=C1_CP_EDI_EGD&START_DAT
E=2001-02-01T00:00:00Z&END_DATE=2001-02-
05T00:00:00Z&DELIVERY_INTERVAL=Daily&MAX_FILE_SIZE=9216&NON_BROWSER
```

1.1.6 UC6 – List datasets without description

This use case retrieves the list of datasets that match a search criteria. The response in JSON format will contain only dataset ID, start date, end date and title.

Type of request: **Metadata request**

Example:

```
http://csa.esac.esa.int/csa/aio/metadata-
action?PAGE_SIZE=10&PAGE=1&SELECTED_FIELDS=DATASET.DATASET_ID,DATASET.START_DATE
,DATASET.END_DATE,DATASET.TITLE&RESOURCE_CLASS=DATASET&EXPERIMENT.NAME=CIS&R
ETURN_TYPE=JSON
```

1.1.7 UC7 – List datasets with description

This use case retrieves the list of datasets that match a search criteria. The response in CSV format will contain dataset ID, start date, end date, title and description.

Type of request: **Metadata request**

Example:

```
http://csa.esac.esa.int/csa/aio/metadata-
action?SELECTED_FIELDS=DATASET.DATASET_ID,DATASET.START_DATE,DATASET.END_DATE,DA
TASET.TITLE,DATASET.DESCRPTION&RESOURCE_CLASS=DATASET&QUERY=(DATASET.DATASET_
ID%20like%20'C%25_CP_EDI_EGD'%20OR%20DATASET.DATASET_ID%20like%20'%25RAP_%25')&R
ETURN_TYPE=CSV
```

1.1.8 UC8 – Get dataset metadata information

This use case retrieves the dataset headers in XML format that match a search criteria.



Type of request: **Data Header request**

Example:

Simply replace `<username>` and `<pwd>` by your CSA username and password in the example below.

```
http://csa.esac.esa.int/csa/aio/product-
action?USERNAME=<username>&PASSWORD=<pwd>&RETRIEVALTYPE=HEADER&DATASET_ID=C*
_CIS-CODIF_RPA_*&NON_BROWSER
```

1.1.9 UC9 – Include dataset reference documents

This use case retrieves particular products, with the additional criteria any documents referenced in the metadata are to be included in the delivery.

Type of request: **Data product request**

Example:

Simply replace `<username>` and `<pwd>` by your CSA username and password in the example below.

```
http://csa.esac.esa.int/csa/aio/product-
action?USERNAME=<username>&PASSWORD=<pwd>&DATASET_ID=C3_CP_RAP_DE&START_DATE
=2003-04-01T00:00:00Z&END_DATE=2003-04-02T00:00:00Z&REF_DOC=1&NON_BROWSER
```

1.1.10 UC10 – Inventory information

This use case retrieves the inventory information for a list of datasets that match a search criteria in JSON format. By default, inventory information includes the dataset ID, start/stop time for the interval, the number of instances and the version.

Type of request: **Inventory request**

Example:

```
http://csa.esac.esa.int/csa/aio/metadata-
action?SELECTED_FIELDS=DATASET_INVENTORY&RESOURCE_CLASS=DATASET_INVENTORY&QU
ERY=(DATASET_INVENTORY.DATASET_ID%20like%20'C1_CP_FGM_FULL'%20OR%20DATASET_INV
ENTORY.DATASET_ID%20like%20'C1_CP_FGM_SPIN')%20AND%20DATASET_INVENTORY.START_TI
ME%20<=%20'2008-02-
04T00:00:00Z'%20AND%20DATASET_INVENTORY.END_TIME%20>=%20'2008-02-
01T00:00:00Z'%20&RETURN_TYPE=JSON
```

2 METADATA REQUESTS

Metadata requests are used to retrieve the metadata details of one or many cluster datasets or cluster files. They are HTTP requests that the user will have to build using several parameters. Once this query is sent, the CSA backend will return the metadata that matches the given criteria in the requested format.

It is important to highlight that the default return type for all metadata requests is the VOTable¹ format.

2.1 Metadata Request Parameters

All metadata requests have the following HTTP prefix:

<http://csa.esac.esa.int/csa/aio/metadata-action>

After the HTTP prefix, the following parameters are used to build the request:

Parameter Name(s)	Description	Mandatory?	Possible Values
SELECTED_FIELDS	Single Entity ("Entity") or list of Attributes requested ("Entity". "Attribute" list separated by commas).	YES	Any CSA Entity or any combination of CSA attributes. (See List of CSA entities and attributes)
RETURN_TYPE	Output format of the response. The default response format is VOTable. Other formats are JSON or CSV	NOT	VOTABLE (default), CSV, JSON
RESOURCE_CLASS	Single Entity which defines the granularity level of the metadata query response.	YES	Any CSA Entity. (See List of CSA entities and attributes)
PAGE_SIZE & PAGE	Number of records per page & page number	NOT, but highly recommended	Any integer value
QUERY	Complex restrictions of the request	NOT	Any Jython valid query expression (see below)
ORDER_BY	Criteria to sort metadata results	NOT	Any CSA attribute. (See List of CSA entities and attributes)
NON_BROWSER	Indicates that the request is not performed from a Browser. When not specified (requests done from a browser), proper HTML pages will be displayed in order to present information to the user.	NOT	

Table 1: Metadata request parameters

¹ International Virtual Observatory standard for the interchange of data (<http://www.ivoa.net/Documents/VOTable>)

In addition to all these parameter names, simple attribute restrictions can be added as "Entity"."Attribute"="value" to the metadata request (for example EXPERIMENT.NAME=ASPOC). Once again, the possible values for this “Entity” and “Attribute” fields need to be compliant with the List of CSA entities and attributes.

Please note that:

- Any value that contains special chars or hyphens should be contained inside apostrophes ('). For example: dataset_id = ' C1_CP_CIS-CODIF_HS_H1_CS'
- Dates are expressed in yyyy-MM-dd format and are contained inside apostrophes (').
- HTTP prefix and parameters are separated by a question mark (?)
- Requests are build using *parameter=value* sentences separated by an ampersand (&)

Let's see the use of all the above through some examples:

- List Datasets 401 to 500 of 'PEACE' experiment in VOTable format:

http://csa.esac.esa.int/csa/aio/metadata-action?PAGE_SIZE=100&PAGE=4&RESOURCE_CLASS=DATASET&SELECTED_FIELDS=DATASET&EXPERIMENT.NAME=PEACE&RETURN_TYPE=VOTABLE

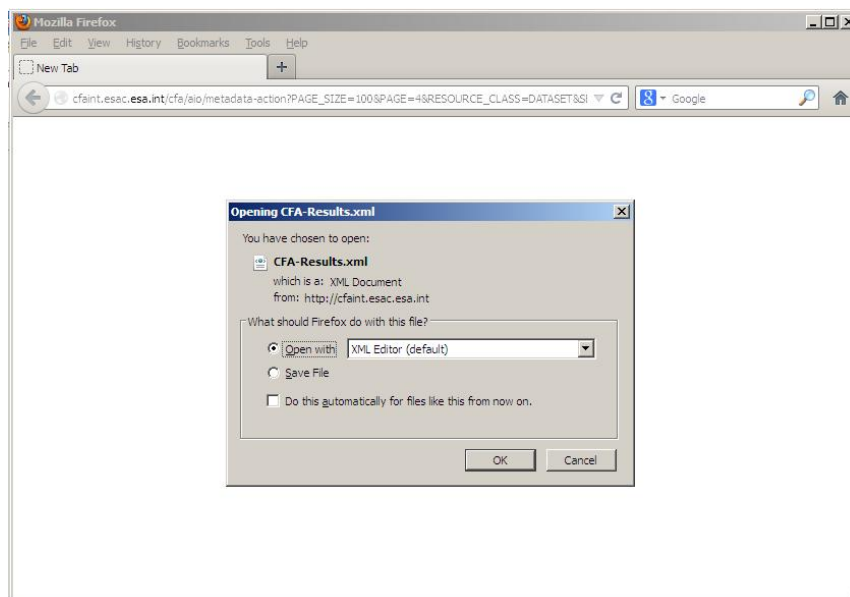


Figure 1: HTTP Metadata Request

- List first 10 datasets for experiment 'CIS' in CSV format selecting only ID, Start and End Date and Title:

http://csa.esac.esa.int/csa/aio/metadata-action?PAGE_SIZE=10&PAGE=1&SELECTED_FIELDS=DATASET.DATASET_ID

[DATASET.START_DATE,DATASET.END_DATE,DATASET.TITLE&RESOURCE_CLASS=DATASET&EXPERIMENT.NAME=CIS&RETURN_TYPE=CSV](http://csa.esac.esa.int/csa/aio/metadata-action?SELECTED_FIELDS=DATASET.DATASET_ID,DATASET.START_DATE,DATASET.END_DATE,DATASET.TITLE&RESOURCE_CLASS=DATASET&EXPERIMENT.NAME=CIS&RETURN_TYPE=CSV)

- List all datasets for experiment 'ASPOC' in VOTable format, selecting ID, Start and End Date, Title and Description:

http://csa.esac.esa.int/csa/aio/metadata-action?SELECTED_FIELDS=DATASET.DATASET_ID,DATASET.START_DATE,DATASET.END_DATE,DATASET.TITLE,DATASET.DESCRPTION&RESOURCE_CLASS=DATASET&EXPERIMENT.NAME=ASPOC&RETURN_TYPE=VOTABLE



```

--<VOTABLE version="1.1">
  --<RESOURCE type="results">
    --<DESCRIPTION>
      CFA Archive Inter-operability Service - Total number of records: 23
    --</DESCRIPTION>
    --<INFO name="QUERY_STATUS" value="OK"> <INFO>
    --</INFO>
    --<TABLE>
      <FIELD ID="DATASET.DATASET_ID" datatype="char" arraysize="*" />
      <FIELD ID="DATASET.START_DATE" datatype="char" arraysize="*" unit="iso-8601" />
      <FIELD ID="DATASET.END_DATE" datatype="char" arraysize="*" unit="iso-8601" />
      <FIELD ID="DATASET.TITLE" datatype="char" arraysize="*" />
      <FIELD ID="DATASET.DESCRPTION" datatype="char" arraysize="*" />
      <FIELD ID="DATA_LINK" />
      <FIELD ID="DETAILS_LINK" />
    --</TABLE>
    --<DATA>
      --<TABLEDATA>
        --<TR>
          <TD>C1_CQ_ASP_CAVEATS</TD>
          <TD>2001-01-01 00:00:00.0</TD>
          <TD>2006-01-01 00:00:00.0</TD>
          <TD>ASPOC instrument caveats</TD>
        --</TD>
          <TD>
            This dataset contains the instrument caveats from the ASPOC experiment on the Cluster C1 spacecraft.
          </TD>
        --<TD>
          http://cfaint.esac.esa.int/cfa/aio/product-action?DATASET_ID=C1_CQ_ASP_CAVEATS
        --<TD>
          http://cfaint.esac.esa.int/cfa/aio/metadata-action?DATASET.DATASET_ID=C1_CQ_ASP_CAVEATS&
            RESOURCE_CLASS=DATASET&SELECTED_FIELDS=DATASET&RETURN_TYPE=VOTABLE
        --<TD>
        --</TD>
        --<TR>
          <TD>C2_CP_ASP_ACTIVE</TD>
          <TD>2001-01-01 00:00:00.0</TD>
          <TD>2008-07-01 01:00:00.0</TD>
        --<TD>
        --</TD>
      --</TABLEDATA>
    --</DATA>
  --</RESOURCE>
--</VOTABLE>

```

Figure 2: Returned VOTable file

When requesting Datasets (RESOURCE_CLASS=DATASET) in VOTABLE or JSON format, additional information will be provided for each dataset:

- A "Data Link" to retrieve the files for this dataset
- An "Asynchronous Data Link" to asynchronously retrieve the files for this dataset
- A "Details Link" to retrieve more precise information about this dataset

2.2 User of QUERY parameter

In the metadata request, the *Keyword=value* paradigm creates simple queries using the AND argument as logical operator. If a user wants to create a complex query, by using >,



<, >=, <=, like (including SQL wildcards), and, or... operators he/she will have to use the QUERY parameter. The rest of the keyword parameters will be seen as other AND conditions.

The value of this QUERY field should be URL encoded².

Inside the value of the QUERY parameter, we allow any Jython³ valid query expression which makes use of CSA attributes (see List of CSA entities and Attributes).

Jython query operations are slightly different from the SQL ones, as stated in this table:

QUERY Operators	Type	Precedence	Symbol
PARENTHESIS	Grouping	1	()
EQUALS	Relational	2	==
NOT EQUALS	Relational	2	!=
GREATER THAN	Relational	2	>
LOWER THAN	Relational	2	<
GREATER OR EQUALS THAN	Relational	2	>=
LOWER OR EQUALS THAN	Relational	2	<=
LIKE: '%' => MULTIPLE CHARACTERS '_' => ONE CHARACTER	Relational	2	LIKE
AND	Logical	3	AND
OR	Logical	3	OR

Table 2: QUERY parameter operators

Let's see the use of all the above through some examples:

- First 100 datasets that fulfil the criteria:

```
QUERY=(DATASET.DATASET_ID like 'C*_CP_EDI_EGD' OR DATASET.DATASET_ID
like '*RAP_*')
```

Equivalent URL:

```
http://csa.esac.esa.int/csa/aio/metadata-action?SELECTED\_FIELDS=DATASET&RESOURCE\_CLASS=DATASET&QUERY=\(DATASET.DATASET\_ID%20like%20'C%25\_CP\_EDI\_EGD'%20OR%20DATASET.DATASET\_ID%20like%20'%25RAP\_%25'\)&PAGE\_SIZE=100&PAGE=1
```

- Dataset ID, Start Date, End Date, Title and Description of Datasets that fulfil the criteria:

² URL encoding reference (http://www.w3schools.com/tags/ref_urlencode.asp)

³ <http://www.jython.org/>



QUERY=(DATASET.DATASET_ID like 'C*_CP_EDI_EGD' OR DATASET.DATASET_ID like '*RAP_*')

Equivalent URL:

[http://csa.esac.esa.int/csa/aio/metadata-action?SELECTED_FIELDS=DATASET.DATASET_ID,DATASET.START_DATE,DATASET.END_DATE,DATASET.TITLE,DATASET.DESCRPTION&RESOURCE_CLASS=DATASET&QUERY=\(DATASET.DATASET_ID%20like%20'C%25_CP_EDI_EGD'%20OR%20DATASET.DATASET_ID%20like%20'%25RAP_%25'\)&RETURN_TYPE=CSV](http://csa.esac.esa.int/csa/aio/metadata-action?SELECTED_FIELDS=DATASET.DATASET_ID,DATASET.START_DATE,DATASET.END_DATE,DATASET.TITLE,DATASET.DESCRPTION&RESOURCE_CLASS=DATASET&QUERY=(DATASET.DATASET_ID%20like%20'C%25_CP_EDI_EGD'%20OR%20DATASET.DATASET_ID%20like%20'%25RAP_%25')&RETURN_TYPE=CSV)

- List File name, Start Date, End Date and Ingestion Date, in CSV format, of files that fulfil the criteria:

QUERY=(DATASET.DATASET_ID like 'C1_CP_EDI_EGD' OR DATASET.DATASET_ID like 'C1_CP_EFW_L3_P' OR DATASET.DATASET_ID like 'C1_CP_FGM_FULL') AND FILE.START_DATE <= '2001-02-05' AND FILE.END_DATE >= '2001-02-01'

Equivalent URL:

[http://csa.esac.esa.int/csa/aio/metadata-action?SELECTED_FIELDS=FILE.FILE_NAME,FILE.START_DATE,FILE.END_DATE,FILE.CAA_INGESTION_DATE&RESOURCE_CLASS=FILE&QUERY=\(DATASET.DATASET_ID%20like%20'C1_CP_EDI_EGD'%20OR%20DATASET.DATASET_ID%20like%20'C1_CP_EFW_L3_P'%20OR%20DATASET.DATASET_ID%20like%20'C1_CP_FGM_FULL'\)%20AND%20FILE.START_DATE%20<=%20'2001-02-05'%20AND%20FILE.END_DATE%20>=%20'2001-02-01'%20&PAGE_SIZE=1000&PAGE=1&RETURN_TYPE=CSV](http://csa.esac.esa.int/csa/aio/metadata-action?SELECTED_FIELDS=FILE.FILE_NAME,FILE.START_DATE,FILE.END_DATE,FILE.CAA_INGESTION_DATE&RESOURCE_CLASS=FILE&QUERY=(DATASET.DATASET_ID%20like%20'C1_CP_EDI_EGD'%20OR%20DATASET.DATASET_ID%20like%20'C1_CP_EFW_L3_P'%20OR%20DATASET.DATASET_ID%20like%20'C1_CP_FGM_FULL')%20AND%20FILE.START_DATE%20<=%20'2001-02-05'%20AND%20FILE.END_DATE%20>=%20'2001-02-01'%20&PAGE_SIZE=1000&PAGE=1&RETURN_TYPE=CSV)

3 INVENTORY REQUESTS

Inventory requests are a special case of Metadata Requests. This section explains how to use them to retrieve inventory information.

The default return type is the VOTable format, but it can return JSON and CSV as well.

The criteria for Inventory Requests are: Dataset Id(s), Start time and Stop time.

Below are outlined the parameters to create a basic inventory request:

Parameter Name(s)	Values
SELECTED_FIELDS	DATASET_INVENTORY
RESOURCE_CLASS	DATASET_INVENTORY
QUERY	Complex restriction using: - DATASET_INVENTORY.DATASET_ID: Unique identifier of the dataset. It can appear more than once, if so, include them between parenthesis and separated with 'OR' operator. It can contain wild cards using together 'LIKE' operator and '%' - DATASET_INVENTORY.START_TIME: Set it to lower or equals than the end of the time interval you are searching for - DATASET_INVENTORY.END_TIME: Set it to greater or equals than the beginning of the time interval you are searching for
RETURN_TYPE (optional)	VOTABLE (default), CSV, JSON
NON_BROWSER (optional)	No value needed. When specified, indicates that the request is not performed from a Browser. When not specified (requests done from a browser), proper HTML pages will be displayed in order to present information to the user.

Table 3: Inventory request parameters

Note that the value of the QUERY parameter must be URL encoded⁴.

Examples of retrieving inventory info

- Dataset(s): 'DWP UT Time-stamped PIOR'
- Time interval: [2013-03-01T00:00:00Z, 2013-03-02T02:00:00Z]
- Format: CSV

http://csa.esac.esa.int/csa/aio/metadata-action?SELECTED_FIELDS=DATASET_INVENTORY&RESOURCE_CLASS=DATASET_INVENTORY&QUERY=DATASET

⁴ URL encoding reference (http://www.w3schools.com/tags/ref_urlencode.asp)



[INVENTORY.DATASET ID%20like%20'CM CP DWP UT PIOR'%20AND%20DATASET INVENTORY.START TIME%20<=%20'2013-03-02T02:00:00Z'%20AND%20DATASET INVENTORY.END TIME%20>=%20'2013-03-01T00:00:00Z'&RETURN TYPE=CSV](#)

- Dataset(s): 'C1_CP_FGM_FULL' and 'C1_CP_FGM_SPIN'
Time interval: [2008-02-01T00:00:00Z, 2008-02-04T00:00:00Z]
Format: JSON

[http://csa.esac.esa.int/csa/aio/metadata-action?SELECTED_FIELDS=DATASET INVENTORY&RESOURCE CLASS=DATASET INVENTORY&QUERY=\(DATASET INVENTORY.DATASET ID%20like%20'C1 CP FGM FULL'%20OR%20DATASET INVENTORY.DATASET ID%20like%20'C1 CP FGM SPIN'\)%20AND%20DATASET INVENTORY.START TIME%20<=%20'2008-02-04T00:00:00Z'%20AND%20DATASET INVENTORY.END TIME%20>=%20'2008-02-01T00:00:00Z'&RETURN TYPE=JSON](http://csa.esac.esa.int/csa/aio/metadata-action?SELECTED_FIELDS=DATASET INVENTORY&RESOURCE CLASS=DATASET INVENTORY&QUERY=(DATASET INVENTORY.DATASET ID%20like%20'C1 CP FGM FULL'%20OR%20DATASET INVENTORY.DATASET ID%20like%20'C1 CP FGM SPIN')%20AND%20DATASET INVENTORY.START TIME%20<=%20'2008-02-04T00:00:00Z'%20AND%20DATASET INVENTORY.END TIME%20>=%20'2008-02-01T00:00:00Z'&RETURN TYPE=JSON)

- Dataset(s): 'C*_EDI_*_CDF'
Time interval: [2004-12-06T23:50:00Z, 2004-12-07T01:00:00Z]
Format: VOTABLE

http://csa.esac.esa.int/csa/aio/metadata-action?SELECTED_FIELDS=DATASET INVENTORY&RESOURCE CLASS=DATASET INVENTORY&QUERY=DATASET INVENTORY.DATASET ID%20like%20'C%25 EDI %25 CDF'%20AND%20DATASET INVENTORY.START TIME%20<=%20'2004-12-07T01:00:00Z'%20AND%20DATASET INVENTORY.END TIME%20>=%20'2004-12-06T23:50:00Z'

- Note that as an Inventory request is a metadata request, we can use at the QUERY parameter other restrictions, too. For example, 'experiment.name'. Below an example of how to retrieve info from ASPOC for a given time range:

Experiment: 'ASPOC'
Time interval: [2006-06-01T13:00:00Z, 2006-06-02T01:00:00Z]
Format: CSV

http://csa.esac.esa.int/csa/aio/metadata-action?SELECTED_FIELDS=DATASET INVENTORY&RESOURCE CLASS=DATASET INVENTORY&QUERY=EXPERIMENT.NAME%20==%20'ASPOC'%20AND%20DATASET INVENTORY.START TIME%20<=%20'2006-06-02T01:00:00Z'%20AND%20DATASET INVENTORY.END TIME%20>=%20'2006-06-01T13:00:00Z'&RETURN TYPE=CSV



4 DATA REQUESTS

Data requests are used to retrieve files. These files can be either, files that belong to a given dataset and that cover a given timespan, or files that contain the metadata header information of datasets. They are HTTP requests that the user will have to build using several parameters. Once this request is sent, the CSA backend will return one HTML link to download one specific file.

All the data requests have the following HTTP prefix:

<http://csa.esac.esa.int/csa/aio/product-action>

There are two types of data requests:

- Product requests
- Header requests

The kind of request you want to use is defined by the parameter RETRIEVALTYPE. Depending on its value, your request will be processed differently and different combination of parameters will be expected.

Please note that for data requests:

- It is mandatory that the user is logged in (see Chapter 6).
- HTTP prefix and parameters are separated by a question mark (?)
- Requests are built using *parameter=value* sentences separated by an ampersand (&)
- Any value that contains special chars or hyphens should be contained inside apostrophes (').

4.1 Product requests

These requests retrieve files of data observed. The maximum size of data to download in a single product request is 100MB.

For non-CEF files, the files delivered are direct copies of the files stored at the archive. For CEF files, new CEF file(s) is(are) created by extracting the data from the original files for the timespan given, the file format (CEF or CDF) can be specified, as the time interval that the delivered files will cover.

After the HTTP prefix, the following parameters are used to build the product request:

Parameter Name(s)	Description	Mandatory?	Possible Values
DATASET_ID	Unique identifier of the dataset whose data we want to retrieve. It can appear more than once, and can contain wild cards (*')	YES	Any of the Cluster datasets.



RETRIEVALTYPE	Type of retrieval. By default, PRODUCT (cluster data file)	NOT	PRODUCT or HEADER
START_DATE	Start time of the time range that the retrieved files should cover. By default, Cluster Mission start time	NOT (but recommended)	ISO time format: 'yyyy-MM-ddTHH:mm:ssZ'
END_DATE	End time of the time range that the retrieved files should cover. By default, Cluster Mission end time	NOT (but recommended)	ISO time format: 'yyyy-MM-ddTHH:mm:ssZ'
DELIVERY_FORMAT	Delivery format, applicable only to CEF files. By default, CEF	NOT	CEF, CDF or CDF_2_7
DELIVERY_INTERVAL	Specifies the delivery file interval in time. By default, "Daily".	NOT	All, Daily, TenMin, Hourly, hourly3, hourly6, hourly12
REF_DOC	When this is set, it delivers the files corresponding to the dataset requested AND the files of the available datasets that are referenced in the metadata of the requested dataset. Referenced documents are typically CAVEAT files. If the reference documents are CEF files only the requested interval is delivered, if they are non-CEF files, the file(s) containing the requested interval will be delivered. By default, 0	NOT	0 or 1
INGESTED_SINCE	Only download files that have been ingested since a particular date	NOT	ISO time format: 'yyyy-MM-ddTHH:mm:ssZ'
MAX_FILE_SIZE	The size of the maximum amount of uncompressed data that can be downloaded in Kbytes. If the query results in a delivery exceeds the specified filesize the user will receive an error message	NOT	A positive integer representing the maximum file size in Kbytes
NON_BROWSER	Indicates that the request is not performed from a Browser. If not specified, it means that the requests are done from a browser and proper HTML pages will be displayed in order to present information to the user.	NOT	

Table 4: Product request parameters

Although we **strongly** recommend the use of authentication session cookies, it is also possible to include the user credentials in the request:

Authentication Parameters	Description	Mandatory?
USERNAME	CSA user name	NOT (recommended use of cookie instead)
PASSWORD	CSA password	NOT (recommended use of cookie instead)

Examples:

- Request ASPOC products observed at Cluster-3 with dataset identifier 'C3_CP_ASP_ION*' for time range '2004-06-18 11:35:00 / 2004-06-19 18:35:00':

http://csa.esac.esa.int/csa/aio/product-action?DATASET_ID=C3_CP_ASP_ION*&START_DATE=2004-06-18T11:35:00Z&END_DATE=2004-06-19T18:35:00Z

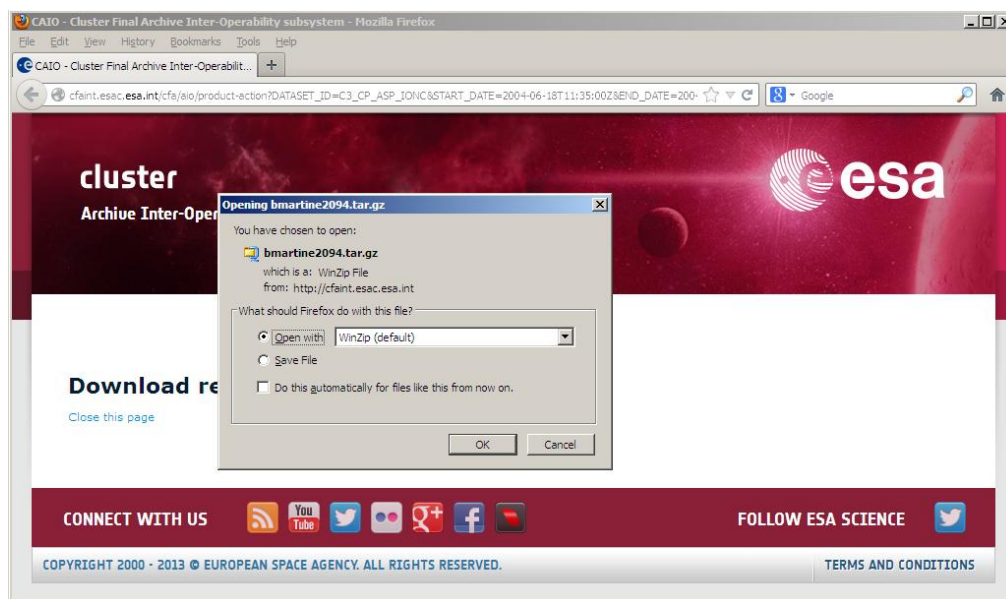


Figure 3: HTTP Product Request

- Use of user credentials for the previous request:
Simply replace <username> and <password> by your CSA username and password in the example below.

http://csa.esac.esa.int/csa/aio/product-action?USERNAME=<username>&PASSWORD=<password>&DATASET_ID=C3_CP_ASP_ION*&START_DATE=2004-06-18T11:35:00Z&END_DATE=2004-06-19T18:35:00Z

- Request DOUBLE STAR FGM products observed at D1 with dataset identifier 'D1_CP_FGM_SPIN' for time range '2004-01-08 04:00:00 / 2004-01-08 06:00:00':



http://csa.esac.esa.int/csa/aio/product-action?DATASET_ID=D1_CP_FGM_SPIN&START_DATE=2004-01-08T04:00:00Z&END_DATE=2004-01-08T06:00:00Z

- Request DWP products observed at Cluster-1 with dataset identifier 'C1_CP_DWP_CORR_ST' for time range '200-06-10 22:12:14 / 2001-06-11 06:12:14' in CDF format and file interval '3 hours':

http://csa.esac.esa.int/csa/aio/product-action?DATASET_ID=C1_CP_DWP_CORR_ST&START_DATE=2001-06-10T22:12:14Z&END_DATE=2001-06-11T06:12:14Z&DELIVERY_FORMAT=CDF&DELIVERY_INTERVAL=hourly3

- Request non-CEF products from WBD observed at Cluster-4 with dataset identifier 'C4_CG_WBD_GIFPLOT' for time range '2008-01-30 05:55:00 / 2008-01-30 05:57:00':

http://csa.esac.esa.int/csa/aio/product-action?DATASET_ID=C4_CG_WBD_GIFPLOT&START_DATE=2008-01-30T05:55:00Z&END_DATE=2008-01-30T05:57:00Z

- Request RAPID products observed at Cluster-3 with dataset identifier 'C3_CP_RAP_DE' and its caveats for time range '2003-04-01 00:00:00 / 2003-04-02 00:00:00':

http://csa.esac.esa.int/csa/aio/product-action?DATASET_ID=C3_CP_RAP_DE&START_DATE=2003-04-01T00:00:00Z&END_DATE=2003-04-02T00:00:00Z&REF_DOC=1

- Request EDI products observed at Cluster-1 with dataset identifier 'C1_CP_EDI_EGD' for time range '2001-02-01 00:00:00 / 2001-02-05 00:00:00' that have been ingested since '2010-03-10T10:21:00Z':

http://csa.esac.esa.int/csa/aio/product-action?DATASET_ID=C1_CP_EDI_EGD&START_DATE=2001-02-01T00:00:00Z&END_DATE=2001-02-05T00:00:00Z&DELIVERY_INTERVAL=Daily&INGESTED_SINCE=2010-03-10T10:21:00Z

- Request FGM products observed at Cluster-1 with dataset identifier 'C1_CP_FGM_FULL' for time range '2001-02-01 00:00:00 / 2001-02-05 00:00:00' that have been ingested since '2007-01-01T00:00:00Z'
Even when there are data for this time range, NO RESULTS are obtained since the ingestion date is previous to the one established at the parameters:



http://csa.esac.esa.int/csa/aio/product-action?DATASET_ID=C1_CP_FGM_FULL&START_DATE=2001-02-01T00:00:00Z&END_DATE=2001-02-05T00:00:00Z&DELIVERY_INTERVAL=Daily&INGESTED_SINCE=2007-01-01T00:00:00Z

- Request EDI products observed at Cluster-1 with dataset identifier 'C1_CP_EDI_EGD' for time range '2001-02-01 00:00:00 / 2001-02-05 00:00:00' whose total filesize is not greater than 9216 Kb (9.2Mb):

http://csa.esac.esa.int/csa/aio/product-action?DATASET_ID=C1_CP_EDI_EGD&START_DATE=2001-02-01T00:00:00Z&END_DATE=2001-02-05T00:00:00Z&DELIVERY_INTERVAL=Daily&MAX_FILE_SIZE=9216

- Request ASPOC and EFW products for the same time range:

http://csa.esac.esa.int/csa/aio/product-action?DATASET_ID=C3_CP_ASP_IONC&DATASET_ID=C1_CP_EFW_L1_P2&START_DATE=2004-06-18T11:35:00Z&END_DATE=2004-06-19T18:35:00Z

4.2 Header requests

After the HTTP prefix, the following parameters are used to build the header request:

Parameter Name(s)	Description	Mandatory?	Possible Values
RETRIEVALTYPE	Type of retrieval. For this request, "HEADER"	YES	PRODUCT or HEADER
DATASET_ID	Unique identifier of the dataset whose metadata header we want to retrieve. It can appear more than once, and can contain wild cards (*)	YES	Any of the Cluster datasets.
NON_BROWSER	Indicates that the request is not performed from a Browser. If not specified, it means that the requests are done from a browser and proper HTML pages will be displayed in order to present information to the user.	NOT	

Table 5: Header request parameters

Although we **strongly** recommend the use of authentication session cookies, it is also possible to include the user credentials in the request:

Authentication Parameters	Description	Mandatory?
USERNAME	CSA user name	NOT (recommended use of cookie instead)
PASSWORD	CSA password	NOT (recommended use of cookie instead)

Examples:

- Request dataset metadata header files for all CIS Dataset for RPA mode:

http://csa.esac.esa.int/csa/aio/product-action?RETRIEVALTYPE=HEADER&DATASET_ID=C* CIS-CODIF_RPA_*

- Use of user credentials in the previous request:
Simply replace <username> and <password> by your CSA username and password in the example below.

`http://csa.esac.esa.int/csa/aio/product-action?USERNAME=<username>&PASSWORD=<password>&RETRIEVALTYPE=HEADER&DATASET_ID=C*_CIS-CODIF_RPA_*`

- Request dataset metadata header files for EDI and RAPID:



http://csa.esac.esa.int/csa/aio/product-action?RETRIEVALTYPE=HEADER&DATASET_ID=C* CP EDI EGD&DATASET_ID=*RAP *

5 ASYNCHRONOUS DATA REQUESTS

Asynchronous data requests are used to retrieve files asynchronously. These files are files that belong to a given dataset and that cover a given timespan. They are HTTP requests that the user will have to build using several parameters. Once this request is sent, the CSA backend will return a Request ID and a link where the data will be available once the retrieval has finished.

All the asynchronous data requests have the following HTTP prefix:

<http://csa.esac.esa.int/csa/aio/async-product-action>

Only one type of data requests is available in asynchronous mode:

- Asynchronous Product requests

Please note that for asynchronous data requests:

- It is mandatory that the user is logged in (see Chapter 6).
- HTTP prefix and parameters are separated by a question mark (?)
- Requests are built using *parameter=value* sentences separated by an ampersand (&)
- Any value that contains special chars or hyphens should be contained inside apostrophes (').

5.1 Asynchronous Product requests

The maximum size of data to download in a single asynchronous product request is 50GB.

After the HTTP prefix, the following parameters are used to build the product request:

Parameter Name(s)	Description	Mandatory?	Possible Values
DATASET_ID	Unique identifier of the dataset whose data we want to retrieve. It can appear more than once, and can contain wild cards ('*')	YES	Any of the Cluster datasets.
RETRIEVALTYPE	Type of retrieval. For this request "PRODUCT" (default value)	NOT	PRODUCT or HEADER
START_DATE	Start time of the time range that the retrieved files should cover. By default, Cluster Mission start time	NOT (but recommended)	ISO time format: 'yyyy-MM-ddTHH:mm:ssZ'
END_DATE	End time of the time range that the retrieved files should cover. By default, Cluster Mission end time	NOT (but recommended)	ISO time format: 'yyyy-MM-ddTHH:mm:ssZ'
DELIVERY_FORMAT	Delivery format, applicable only to	NOT	CEF, CDF or CDF_2_7

	CEF files. By default, CEF		
DELIVERY_INTERVAL	Specifies the delivery file interval in time. By default, "Daily".	NOT	All, Daily, TenMin, Hourly, hourly3, hourly6, hourly12
REF_DOC	When this is set, it delivers the files corresponding to the dataset requested AND the files of the available datasets that are referenced in the metadata of the requested dataset. Referenced documents are typically CAVEAT files. If the reference documents are CEF files only the requested interval is delivered, if they are non-CEF files, the file(s) containing the requested interval will be delivered. By default, 0	NOT	0 or 1
INGESTED_SINCE	Only download files that have been ingested since a particular date	NOT	ISO time format: 'yyyy-MM-ddTHH:mm:ssZ'
MAX_FILE_SIZE	The size of the maximum amount of uncompressed data that can be downloaded in Kbytes. If the query results in a delivery exceeds the specified filesize the user will receive an error message	NOT	A positive integer representing the maximum file size in Kbytes
NO_NOTIFY	Indicates that the user does not want to be notified by email when the data is ready for download. If not specified, an email will be sent to the user.	NOT	
NON_BROWSER	Indicates that the request is not performed from a Browser. If not specified, it means that the requests are done from a browser and proper HTML pages will be displayed in order to present information to the user.	NOT	

Table 6: Asynchronous Product request parameters

Although we **strongly** recommend the use of authentication session cookies, it is also possible to include the user credentials in the request:

Authentication Parameters	Description	Mandatory?
USERNAME	CSA user name	NOT (recommended use of cookie instead)
PASSWORD	CSA password	NOT (recommended use of cookie instead)

Examples:

- Request EDI, EFW and FGM products for 4 days, split into day files:

http://csa.esac.esa.int/csa/aio/async-product-action?DATASET_ID=C1_CP_EDI_EGD&DATASET_ID=C1_CP_EFW_L3_P&DATA

[SET_ID=C1_CP_FGM_FULL&START_DATE=2001-02-01T00:00:00Z&END_DATE=2001-02-05T00:00:00Z&DELIVERY_INTERVAL=Daily](http://csa.esac.esa.int/csa/aio/async-product-action?SET_ID=C1_CP_FGM_FULL&START_DATE=2001-02-01T00:00:00Z&END_DATE=2001-02-05T00:00:00Z&DELIVERY_INTERVAL=Daily)

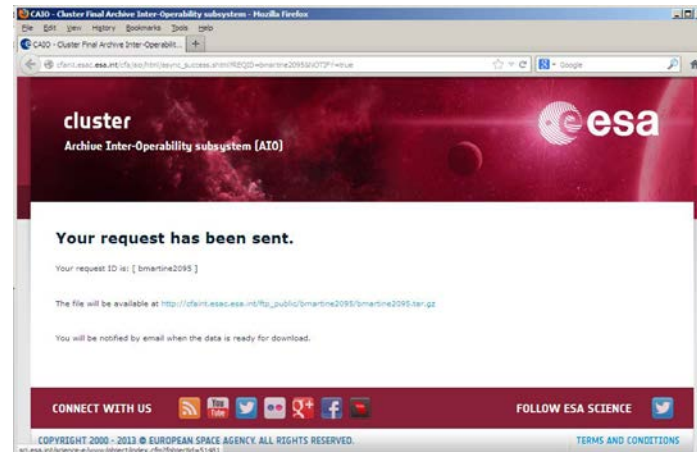


Figure 4: HTTP Asynchronous Data request

- Use of user credentials in the previous request:
Simply replace `<username>` and `<password>` by your CSA username and password in the example below.

http://csa.esac.esa.int/csa/aio/async-product-action?USERNAME=<username>&PASSWORD=<password>&DATASET_ID=C1_CP_EDI_EGD&DATASET_ID=C1_CP_EFW_L3_P&DATASET_ID=C1_CP_FGM_FULL&START_DATE=2001-02-01T00:00:00Z&END_DATE=2001-02-05T00:00:00Z&DELIVERY_INTERVAL=Daily

- Request ASPOC products observed at Cluster-3 with dataset identifier 'C3_CP_ASP_IONC' for time range '2004-06-18 11:35:00 / 2004-06-19 18:35:00':

http://csa.esac.esa.int/csa/aio/async-product-action?DATASET_ID=C3_CP_ASP_IONC&START_DATE=2004-06-18T11:35:00Z&END_DATE=2004-06-19T18:35:00Z

- Request ASPOC and EFW products for the same time range:

http://csa.esac.esa.int/csa/aio/async-product-action?DATASET_ID=C3_CP_ASP_IONC&DATASET_ID=C1_CP_EFW_L1_P2&START_DATE=2004-06-18T11:35:00Z&END_DATE=2004-06-19T18:35:00Z&NO_NOTIFY

6 STREAMING DATA REQUESTS

Data Streaming allows a faster delivery of the data. It indeed enables immediate streaming of one dataset to the users machine (instead of a gzip file being created on the CSA server and then sent).

However the following constraints apply:

- Only CEF products can be downloaded using these requests
- Only one dataset can be requested
- Only one file is delivered for the time period requested, i.e. delivery interval option is not available.
- Header only cannot be requested.

All the streaming data requests have the following HTTP prefix:

<http://csa.esac.esa.int/csa/aio/streaming-action>

Only one type of data requests is available in streaming mode:

- Streaming Product requests

Please note that for streaming data requests:

- It is mandatory that the user is logged in (see Chapter 6).
- HTTP prefix and parameters are separated by a question mark (?)
- Requests are built using *parameter=value* sentences separated by an ampersand (&)
- Any value that contains special chars or hyphens should be contained inside apostrophes (').

6.1 Streaming Product requests

After the HTTP prefix, the following parameters are used to build the product request:

Parameter Name(s)	Description	Mandatory?	Possible Values
DATASET_ID	Unique identifier of the dataset whose data we want to retrieve. Please note that wild cards ('*') are NOT allowed	YES	Any of the Cluster datasets.
START_DATE	Start time of the time range that the retrieved files should cover.	YES	ISO time format 'yyyy-MM-ddTHH:mm:ssZ'
END_DATE	End time of the time range that the retrieved files should cover.	YES	ISO time format 'yyyy-MM-ddTHH:mm:ssZ'
GZIP	Specifies if the retrieved file is gzip compressed or not. By default, "0"	NOT	0 or 1
NON_BROWSER	Indicates that the request is not performed from a Browser. If not specified, it means that the requests are done from a browser and proper HTML pages will be displayed in order to present information to the user	NOT	

Table 7: Streaming Product request parameters



Although we **strongly** recommend the use of authentication session cookies, it is also possible to include the user credentials in the request:

Authentication Parameters	Description	Mandatory?
USERNAME	CSA user name	NOT (recommended use of cookie instead)
PASSWORD	CSA password	NOT (recommended use of cookie instead)

Examples:

- Request DWP caveats for time range '2007-01-15 18:00:00 / 2007-01-16 00:00:00' with non-browser option:

```
http://csa.esac.esa.int/csa/aio/streaming-action?DATASET\_ID=C3\_CQ\_DWP\_INST&START\_DATE=2007-01-15T18:00:00Z&END\_DATE=2007-01-16T18:00:00Z&NON\_BROWSER
```

- Use of user credentials in the previous request:
Simply replace *<username>* and *<password>* by your CSA username and password in the example below.

```
http://csa.esac.esa.int/csa/aio/streaming-action?USERNAME=<username>&PASSWORD=<password>&DATASET_ID=C3_CQ_DWP_INST&START_DATE=2007-01-15T18:00:00Z&END_DATE=2007-01-16T18:00:00Z&NON_BROWSER
```

- Request ASPOC products for more than 1 day through the browser:

```
http://csa.esac.esa.int/csa/aio/streaming-action?DATASET\_ID=C3\_CP\_ASP\_IONS&START\_DATE=2004-06-18T11:35:00Z&END\_DATE=2004-06-19T18:35:00Z
```

- Request PEACE products for more than 1 month in compressed format:

```
http://csa.esac.esa.int/csa/aio/streaming-action?DATASET\_ID=C4\_CP\_PEA\_LERL\_PSD&START\_DATE=2004-09-18T10:00:00Z&END\_DATE=2004-10-20T22:00:00Z&GZIP=1
```

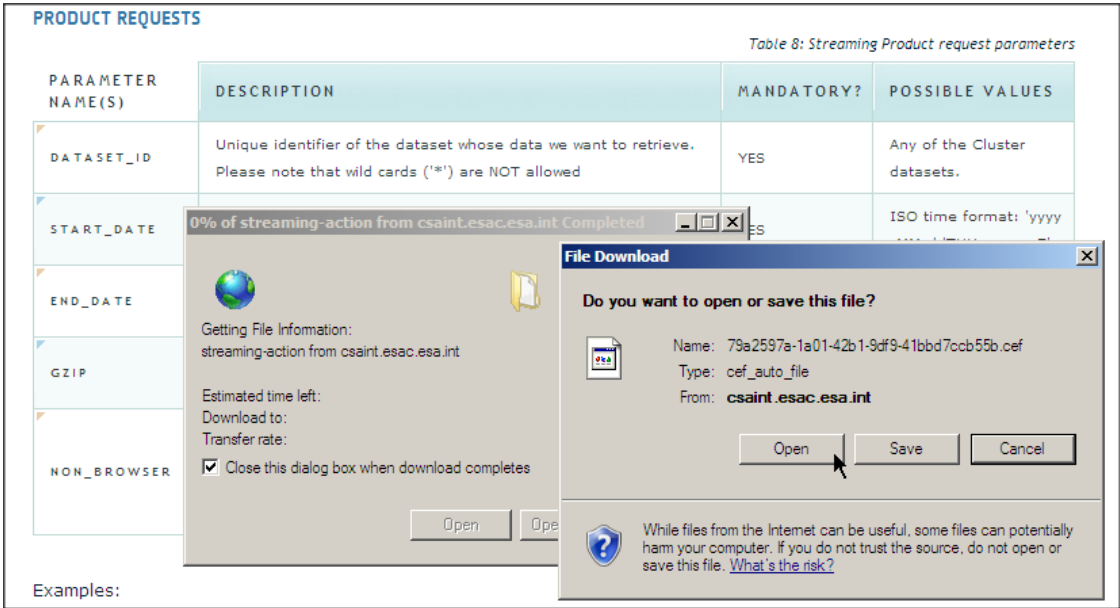


Figure 5: HTTP streaming product request

7 CREDENTIALS REQUESTS

Data requests and Asynchronous Data requests require the user to log in first. The user details will be stored in a session cookie. That means that the credential will be accessible as long as the user is using the web application. The cookie will be deleted when the connection is closed (for example, when the browser exits).

All credentials requests have the following HTTP prefix:

<http://csa.esac.esa.int/csa/aio/login-action>

There are two types of credentials requests:

- Login requests
- Logout requests

Please note that for credential requests:

- HTTP prefix and parameters are separated by a question mark (?)
- Requests are built using *parameter=value* sentences separated by an ampersand (&)
- Any value that contains special chars or hyphens should be contained inside apostrophes (').

7.1 Login requests

After the HTTP prefix, the following parameters are used to build the login request:

Parameter Name(s)	Description	Mandatory?	Possible Values
USERNAME	Unique identifier of the user at the CSA system.	YES	Username
PASSWORD	Password of the user at the CSA system	YES	Password
SUBMIT	LOGIN, as the request is for a login.	YES	LOGIN or LOGOUT
NON_BROWSER	Whether the request is performed from a Browser. If not specified, the requests is done from a browser and proper HTML pages will be displayed.	NOT	

Table 8: Login request parameters

If we are using a browser and we want to perform a Data or Asynchronous data request without being logged in, a pop-up window will be displayed requesting the username and password (no need to make explicitly the call to the login request):



Figure 6: Pop-up window for login

On the other hand, if we are executing a Data or Asynchronous data request with the NON_BROWSER parameter and we are not logged in, this will be the response of the system:

LOGIN_REQUESTED <http data or asynchronous data request>

Example:

- Login of a user in a browser:

Simply replace *<username>* and *<password>* by your CSA username and password in the example below.

<http://csa.esac.esa.int/csa/aio/login-action?USERNAME=<username>&PASSWORD=<password>&SUBMIT=LOGIN>

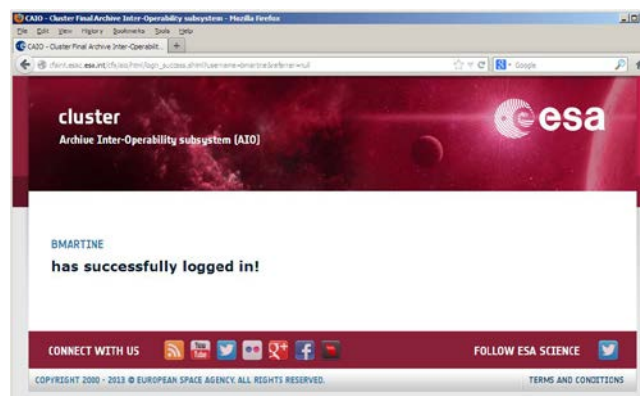


Figure 7: Successful login

7.2 Logout requests

When a user logs out, an “anonymous” user is set by default.

After the HTTP prefix, the following parameters are used to build the login request:

Parameter Name(s)	Description	Mandatory?	Possible Values
SUBMIT	LOGOUT, as the request is for a logout.	YES	LOGIN or LOGOUT
NON_BROWSER	Whether the request is performed from a Browser. If not specified, the requests is done from a browser and proper HTML pages will be displayed.	NOT	

Example:

- Logout of a user with no browser:

http://csa.esac.esa.int/csa/aio/login-action?%20SUBMIT=LOGOUT&NON_BROWSER

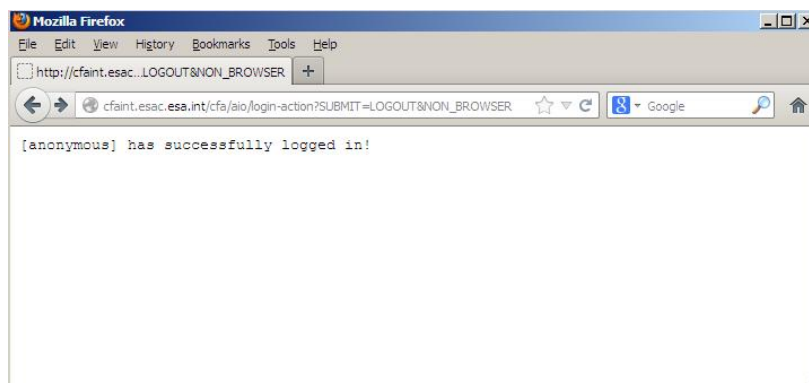


Figure 8: Successful logout

8 LIST OF CSA ENTITIES AND ATTRIBUTES

Below you can find the list of all the CSA entities and attributes. They can be used in the metadata request, related to certain parameters:

- Entity name can appear as **SELECTED_FIELDS** (only one at a time)
- Entity name can appear as **RESOURCE_CLASS** (only one at a time)
- Attribute names can appear as **SELECTED_FIELDS** (one or a combination of them)
- Attribute names can appear as **ORDER_BY** (one or a combination of them)
- Simple attribute restrictions can be added as “Attribute Name” = “value”

Entity Name	Attribute Name	Attribute Type
MISSION	MISSION.MISSION_OID	INTEGER
	MISSION.NAME	STRING
	MISSION.TIME_SPAN	STRING
OBSERVATORY	OBSERVATORY.OBSERVATORY_OID	INTEGER
	OBSERVATORY.ABBREVIATION	STRING
	OBSERVATORY.NAME	STRING
	OBSERVATORY.DESCRPTION	STRING
	OBSERVATORY.TIME_SPAN	STRING
EXPERIMENT	EXPERIMENT.EXPERIMENT_OID	INTEGER
	EXPERIMENT.NAME	STRING
	EXPERIMENT.LONG_NAME	STRING
	EXPERIMENT.DESCRPTION	STRING
	EXPERIMENT.ACTIVE	INTEGER
	EXPERIMENT.START_DATE	DATE
	EXPERIMENT.END_DATE	DATE
INSTRUMENT	INSTRUMENT.INSTRUMENT_OID	INTEGER
	INSTRUMENT.NAME	STRING
	INSTRUMENT.DESCRPTION	STRING
	INSTRUMENT.ACTIVE	INTEGER
INSTRUMENT_TYPE	INSTRUMENT_TYPE.INSTRUMENT_TYPE_OID	INTEGER
	INSTRUMENT_TYPE.TYPE	STRING
MEASUREMENT_TYPE	MEASUREMENT_TYPE.MEASUREMENT_TYPE_OID	INTEGER
	MEASUREMENT_TYPE.TYPE	STRING
DATASET	DATASET.DATASET_OID	INTEGER
	DATASET.DATASET_ID	STRING
	DATASET.START_DATE	DATE
	DATASET.END_DATE	DATE
	DATASET.DESCRPTION	STRING
	DATASET.DATASET_VERSION	STRING
	DATASET.RELEASED	INTEGER
	DATASET.ACTIVE	INTEGER
	DATASET.INVENTORY	INTEGER
	DATASET.DISPLAY_ORDER	INTEGER
	DATASET.FILE_EXTENSION	STRING



	DATASET.TITLE	STRING
	DATASET.TIME_RESOLUTION	DOUBLE
	DATASET.MIN_TIME_RESOLUTION	DOUBLE
	DATASET.MAX_TIME_RESOLUTION	DOUBLE
	DATASET.DATASET_CAVEATS	STRING
	DATASET.GUI_NAME	STRING
	DATASET.MAIN_GROUP	STRING
	DATASET.CATEGORY	STRING
	DATASET.INGESTION_DATE	DATE
	DATASET.UNIT_NAME	STRING
	DATASET.MULTI_INST	INTEGER
	DATASET.MULTI_SC	INTEGER
	DATASET.CROSSCAL	INTEGER
	DATASET.ACK	STRING
QUICK_LOOK_DATASET	QUICK_LOOK_DATASET.QL_DATASET_OID	INTEGER
	QUICK_LOOK_DATASET.QL_DISPLAY_ORDER	INTEGER
	QUICK_LOOK_DATASET.WIDTH	INTEGER
	QUICK_LOOK_DATASET.HEIGHT	INTEGER
	QUICK_LOOK_DATASET.MAX_WIDTH	INTEGER
	QUICK_LOOK_DATASET.MAX_HEIGHT	INTEGER
	QUICK_LOOK_DATASET.DATASET_OID	INTEGER
	QUICK_LOOK_DATASET.DATASET_ID	STRING
	QUICK_LOOK_DATASET.START_DATE	DATE
	QUICK_LOOK_DATASET.END_DATE	DATE
	QUICK_LOOK_DATASET.DESCRPTION	STRING
	QUICK_LOOK_DATASET.DATASET_VERSION	STRING
	QUICK_LOOK_DATASET.RELEASED	INTEGER
	QUICK_LOOK_DATASET.ACTIVE	INTEGER
	QUICK_LOOK_DATASET.INVENTORY	INTEGER
	QUICK_LOOK_DATASET.DISPLAY_ORDER	INTEGER
	QUICK_LOOK_DATASET.FILE_EXTENSION	STRING
	QUICK_LOOK_DATASET.TITLE	STRING
	QUICK_LOOK_DATASET.TIME_RESOLUTION	DOUBLE
	QUICK_LOOK_DATASET.MIN_TIME_RESOLUTION	DOUBLE
	QUICK_LOOK_DATASET.MAX_TIME_RESOLUTION	DOUBLE
	QUICK_LOOK_DATASET.DATASET_CAVEATS	STRING
	QUICK_LOOK_DATASET.GUI_NAME	STRING
	QUICK_LOOK_DATASET.MAIN_GROUP	STRING
	QUICK_LOOK_DATASET.CATEGORY	STRING
	QUICK_LOOK_DATASET.INGESTION_DATE	DATE
	QUICK_LOOK_DATASET.UNIT_NAME	STRING
	QUICK_LOOK_DATASET.MULTI_INST	INTEGER
	QUICK_LOOK_DATASET.MULTI_SC	INTEGER
	QUICK_LOOK_DATASET.CROSSCAL	INTEGER
	QUICK_LOOK_DATASET.ACK	STRING
DATA_TYPE	DATA_TYPE.DATA_TYPE_OID	INTEGER
	DATA_TYPE.ID	STRING
	DATA_TYPE.SHORT_NAME	STRING
FILE	FILE.FILE_OID	LONG



	FILE.LOGICAL_FILE_ID	STRING
	FILE.VERSION_NUMBER	INTEGER
	FILE.DATASET_VERSION	STRING
	FILE.METADATA_TYPE	STRING
	FILE.METADATA_VERSION	STRING
	FILE.FILE_NAME	STRING
	FILE.FILE_TYPE	STRING
	FILE.START_DATE	DATE
	FILE.END_DATE	DATE
	FILE.CSA_INGESTION_DATE	DATE
	FILE.CAA_INGESTION_DATE	DATE
	FILE.FILE_SIZE	LONG
	FILE.EXTENSION	STRING
	FILE.ACTIVE	INTEGER
	FILE.FILE_CAVEATS	STRING
	FILE.COMPRESSED	INTEGER
DATASET_INVENTORY	DATASET_INVENTORY. DATASET_INVENTORY_OID	LONG
	DATASET_INVENTORY.DATASET_ID	STRING
	DATASET_INVENTORY.START_TIME	DATE
	DATASET_INVENTORY.END_TIME	DATE
	DATASET_INVENTORY.NUM_INSTANCES	INTEGER
	DATASET_INVENTORY.VERSION	INTEGER



9 APPENDIX: HOW TO RUN THE DEFAULT CLIENT

We have implemented a code example (using the Java programming language) to be used as a simple AIO client. Download it and extract the content of the tar file on your local disk.

- Client example: <http://csa.esac.esa.int/csa/aio/aioclient-1.0.tar>

You will find both the source and the compiled code of our example.

We have implemented three different actions: retrieval from a single dataset, retrieval from multiple datasets and asynchronous retrieval from multiple datasets.

- For the single dataset option the unique identifier of the dataset (i.e. its dataset id) is required and optionally other restrictions can be added (see Table 4: Product request parameters). This option simply makes an http request to the CSA and retrieve the file(s).
- The multiple datasets options will retrieve all the datasets that match the result of a query. Internally, they will make several requests, one to execute the query and one for each of the dataset identifiers returned from it.

9.1 Syntax for the Java client

Windows users: Please be aware that the classpath declaration in Windows is different. The syntax below will be: `$java -cp .;.savot.jar .....`

The syntax for the different options is:

- Data retrieval from a single dataset:
Simply replace `<username>` and `<password>` by your CSA username and password in the example below.

```
$ java -cp ../savot.jar AioWEBClient -S <Server Name> -L "GET DATASET_ID=<value>" [-P
"<PARAMETER_NAME>=<value> ..."] [-AIOUSER <username> -AIOPWD <password>] [-O <output dir>]
```

- Data retrieval from multiple datasets:
Simply replace `<username>` and `<password>` by your CSA username and password in the example below.

```
$ java -cp ../savot.jar AioWEBClient -S <Server Name> -L "GET_ALL <complex query value>" [-P
"<PARAMETER_NAME>=<value> ..."] [-AIOUSER <username> -AIOPWD <password>] [-O <output dir>]
```

- Asynchronous Data retrieval from multiple datasets:
Simply replace `<username>` and `<password>` by your CSA username and password in the example below.



```
$ java -cp ../savot.jar AioWEBClient -S <Server Name> -L "GET_ASYNC <complex query value>" [-P
"<PARAMETER_NAME>=<value> ..."] [-AIOUSER <username> -AIOPWD <password>] [-O <output dir>]
```

Where:

- "Server name" is the http server location (csa.esac.esa.int/csa/ for the operational archive)
- "value" is the value of a parameter, either of a 'DATASET_ID' or of any of the product request parameters (see Table 4).
NOTE: if "value" contains white chars or hyphens (as in: C1_CP_CIS-CODIF_HS_H1_CS), then it must go between single quotes (').
- "complex query value" equivalent to the value of a 'QUERY' parameter within a Metadata request (see QUERY parameter)
- "username" and "password" are optional arguments to include the CSA login detail in the command when creating the query. If these are not provided, the user will be prompted to do so on the command line
- "output dir" is an optional argument for specifying where the user wants to store the files downloaded.

It is important to note that in the multiple datasets request we use pagination in order to prevent scalability problems in the client side. If the number of records that fulfil the query is greater than the defined page size (internally set to 1000), the system makes several different requests to retrieve them all.

9.1.1 Syntax for the shell script with the Java client

If you work on UNIX, we provide inside the package a shell script ('aioclient') with all the default values inside, so you do not need to enter them in the command line after you set them inside.

Edit the script file and enter your values for the variables:

- *AIOUSER* = <your CSA user name>
- *AIOPWD* = <your CSA password>
- *jdkhome* = <installation directory of your Java environment> (ex.:
 /usr/local/java1.6)

Example:

```
./aioclient -L "GET_ALL (DATASET.DATASET_ID=='%_CP_EDI_EGD' OR
DATASET.DATASET_ID=='%RAP_%DD_R') " -P "START_DATE=2001-02-01T00:00:00Z END_DATE=2001-02-
01T12:00:00Z" -O FebruaryDirectory
```

Where the files will be created in the folder: "FebruaryDirectory"



9.2 Examples of use

Data retrieval from a single dataset:

Retrieve 'ASPOC Ion current' data from C3 for time range [2004-06-18T11:35:00Z, 2004-06-19T18:35:00Z]

```
$ java -cp ../savot.jar AioWEBClient -S csa.esac.esa.int/csa -L "GET_DATASET_ID=C3_CP_ASP_IONC" -P
"START_DATE=2004-06-18T11:35:00Z END_DATE=2004-06-19T18:35:00Z"
```

Data retrieval from multiple datasets:

Retrieve data from all Science datasets from FGM between the time range [2002-02-12T23:30:00Z, 2002-02-13T01:00:00Z] and leave them at 'fgmDir'

```
$ java -cp ../savot.jar AioWEBClient -S csa.esac.esa.int/csa -L "GET_ALL (EXPERIMENT.NAME=='FGM' AND
DATASET.MAIN_GROUP=='Science')" -P "START_DATE=2002-02-12T23:30:00Z END_DATE=2002-02-
13T01:00:00Z" -O fgmDir
```

Retrieve from all 'EDI Electro Gyrotime' datasets and from all 'RAPID 3D distribution of Electrons' datasets the data between the time range: [2001-02-01T00:00:00Z, 2001-02-01T12:00:00Z] and leave the files at the 'clusterFiles' directory

```
$ java -cp ../savot.jar AioWEBClient -S csa.esac.esa.int/csa -L "GET_ALL
(DATASET.DATASET_ID=='_CP_EDI_EGD' OR DATASET.DATASET_ID=='_RAP_%DD_R')" -P
"START_DATE=2001-02-01T00:00:00Z END_DATE=2001-02-01T12:00:00Z" -O clusterFiles
```

Asynchronous Data retrieval from multiple datasets:

Retrieve asynchronously all 'EDI Electro Gyrotime' and all 'RAPID 3D distribution of Electrons' data between the time range: [2001-02-01T00:00:00Z, 2001-02-01T12:00:00Z] and leave the files at the 'asyncClusterFiles' directory

```
$ java -cp ../savot.jar AioWEBClient -S csa.esac.esa.int/csa -L "GET_ASYNC
(DATASET.DATASET_ID=='_CP_EDI_EGD' OR DATASET.DATASET_ID=='_RAP_%DD_R')" -P
"START_DATE=2001-02-01T00:00:00Z END_DATE=2001-02-01T12:00:00Z" -O asyncClusterFiles
```

The response to every asynchronous request will generate a text file containing information about the Request ID and the link where the data will be ready when finished. All those text files will be stored in the folder "asyncClusterFiles".



9.3 Equivalence to CAIO data requests

The equivalent command line syntax to the data product request:

http://csa.esac.esa.int/csa/aio/product-action?DATASET_ID=C3_CP_ASP_IONC&START_DATE=2004-06-18T11:35:00Z&END_DATE=2004-06-19T18:35:00Z

is:

```
$ java -cp ../savot.jar AioWEBClient -S csa.esac.esa.int/csa -L "GET DATASET_ID=C3_CP_ASP_IONC" -P "START_DATE=2004-06-18T11:35:00Z END_DATE=2004-06-19T18:35:00Z"
```

Data retrieval from multiple datasets:

A data retrieval from multiple datasets consists on a sequence of actions. First, a metadata request to search for the datasets that fulfil the criteria, afterwards a Data Product request per dataset found to retrieve its files:

Look for the datasets that contains files matching this particular criterion:

[http://csa.esac.esa.int/csa/aio/metadata-action?RESOURCE_CLASS=DATASET&SELECTED_FIELDS=DATASET.DATASET_ID&QUERY=\(DATASET.DATASET_ID+like+%27C%25_CP_EDI_EGD%27+OR+DATASET.DATASET_ID+like+%27%25RAP_%25DD_R%27\)&RETURN_TYPE=VOTABLE&PAGE=1&PAGE_SIZE=1000](http://csa.esac.esa.int/csa/aio/metadata-action?RESOURCE_CLASS=DATASET&SELECTED_FIELDS=DATASET.DATASET_ID&QUERY=(DATASET.DATASET_ID+like+%27C%25_CP_EDI_EGD%27+OR+DATASET.DATASET_ID+like+%27%25RAP_%25DD_R%27)&RETURN_TYPE=VOTABLE&PAGE=1&PAGE_SIZE=1000)

For each dataset returned in VOTABLE format, retrieve the data files (for the specific time range, if specified):

http://csa.esac.esa.int/csa/aio/product-action?DATASET_ID=C1_CP_EDI_EGD&START_DATE=2001-02-01T00:00:00Z&END_DATE=2001-02-01T12:00:00Z
http://csa.esac.esa.int/csa/aio/product-action?DATASET_ID=C3_CP_RAP_L3DD_R&START_DATE=2001-02-01T00:00:00Z&END_DATE=2001-02-01T12:00:00Z
http://csa.esac.esa.int/csa/aio/product-action?DATASET_ID=C4_CP_RAP_E3DD_R&START_DATE=2001-02-01T00:00:00Z&END_DATE=2001-02-01T12:00:00Z

this sequence of actions is equivalent to:



```
$ java -cp ../savot.jar AioWEBClient -S csa.esac.esa.int/csa -L "GET_ALL
(DATASET.DATASET_ID=='%_CP_EDI_EGD' OR DATASET.DATASET_ID=='%RAP_%DD_R')" -P
"START_DATE=2001-02-01T00:00:00Z END_DATE=2001-02-01T12:00:00Z" -O clusterFiles
```

All retrieved files will be stored in the folder "clusterFiles".

Asynchronous Data retrieval from multiple datasets:

In the same way as for a data retrieval, it is a sequence of actions. First, a Metadata request to search for the datasets that fulfil the criteria, afterwards an Asynchronous Data Product request per dataset found to retrieve its files:

Look for datasets that contains files matching this particular criterion:

```
http://csa.esac.esa.int/csa/aio/metadatas-action?RESOURCE\_CLASS=DATASET&SELECTED\_FIELDS=DATASET.DATASET\_ID&QUERY=\(DATASET.DATASET\_ID+like+%27C%25\_CP\_EDI\_EGD%27+OR+DATASET.DATASET\_ID+like+%27%25RAP\_%25DD\_R%27\)&RETURN\_TYPE=VOTABLE&PAGE=1&PAGE\_SIZE=1000
```

For each dataset returned in VOTABLE format, request asynchronously the data files (for the specific time range, if specified):

```
http://csa.esac.esa.int/csa/aio/async-product-action?DATASET\_ID=C1\_CP\_EDI\_EGD&START\_DATE=2001-02-01T00:00:00Z&END\_DATE=2001-02-01T12:00:00Z
http://csa.esac.esa.int/csa/aio/async-product-action?DATASET\_ID=C3\_CP\_RAP\_L3DD\_R&START\_DATE=2001-02-01T00:00:00Z&END\_DATE=2001-02-01T12:00:00Z
http://csa.esac.esa.int/csa/aio/async-product-action?DATASET\_ID=C4\_CP\_RAP\_E3DD\_R&START\_DATE=2001-02-01T00:00:00Z&END\_DATE=2001-02-01T12:00:00Z
```

this sequence of actions is equivalent to:

```
$ java -cp ../savot.jar AioWEBClient -S csa.esac.esa.int/csa -L "GET_ASYNC
(DATASET.DATASET_ID=='%_CP_EDI_EGD' OR DATASET.DATASET_ID=='%RAP_%DD_R')" -P
"START_DATE=2001-02-01T00:00:00Z END_DATE=2001-02-01T12:00:00Z" -O asyncClusterFiles
```

The response to every asynchronous request will generate a text file containing information about the Request ID and the link where the data will be ready when finished. All those text files will be stored in the folder "asyncClusterFiles".

Please feel free to take this code example and modify it to suit your application needs. This is a Java example but any other language could be used for the same purpose.



10 APPENDIX: HOW TO USE CAIO WITH WGET

Below some instructions on how to use "wget" when interacting with the CSA. Instructions for different kind of requests are explained, including two ways for user authentication (data download). Although username and password can be used directly on the http request for authentication, we **strongly** recommend the use of cookies that provide a higher level of security.

Recommendations

- For a smooth execution, add the flag "**NON_BROWSER**" to all CAIO requests when using 'wget'.
- Use the wget option '--content-disposition' when making requests using CAIO, otherwise the output filename will be valid but meaningless in terms of syntax.
- If your request contains special chars (e.g. password with exclamation mark, !), use bash shell and add single quote (') instead of the usual double quotes (") around the http request.
- Go to the last section of this chapter to know how to get more descriptive error messages.

These are the sections within this appendix:

- Data download
 - Direct download
 - Asynchronous download
 - Data streaming
- Inventory request
- Metadata request
- How to specify the output file?
- How to create a cookie?
- Hint on descriptive error messages

10.1 Data download

Data retrieval requires an authenticated user. Thus it is necessary either to provide user credentials in the request or to load the cookies file (recommended). For each type of data request (direct, asynchronous or data streaming), two ways of executing the same data request will be detailed. For CAA users upgrading their scripts, please note that the CAA "uname" keyword is simply USERNAME on CSA while "pwd" becomes "PASSWORD".

Any data retrieval request has as output a file. See *How to specify the output file?* (section 10.4) to get a meaningful filename. Bear in mind that --content-disposition is not yet bug-free, and sometimes the requests can store the output in a filename with the webpage name and query string parameters (default behaviour of wget). For example:



Simply replace `<username>` and `<password>` by your CSA username and password in the example below.

```
"async-product-
action?USERNAME=<username>&PASSWORD=<password>&DATASET_ID=C3_CP_ASP_IONC&START
_DATE=2004-06-18T11:35:00Z&END_DATE=2004-06-19T18:35:00Z&NON_BROWSER"
```

10.1.1 Direct download

See list of parameters at section 4.1.

Please find below two ways of executing the first example presented in the Data tab:

Credentials within the request

Simply replace `<username>` and `<password>` by your CSA username and password in the example below.

```
$ wget --content-disposition 'http://csa.esac.esa.int/csa/aio/product-
action?USERNAME=<username>&PASSWORD=<password>&DATASET_ID=C3_CP_ASP_IONC&START_DATE=20
04-06-18T11:35:00Z&END_DATE=2004-06-19T18:35:00Z&NON_BROWSER'
```

Authentication cookie

```
$ wget --load-cookies <cookies_file>.txt --content-disposition "http://csa.esac.esa.int/csa/aio/product-
action?DATASET_ID=C3_CP_ASP_IONC&START_DATE=2004-06-18T11:35:00Z&END_DATE=2004-06-
19T18:35:00Z&NON_BROWSER"
```

stores the retrieved data at a tar gzip file with a random name, for example: `"f73edec3-0cd9-4cd7-a203-bd413de5fd9e.tar.gz"`.

If a user cannot be authenticated, 'wget' will output an `"authentication.log[n]"` file instead.

10.1.2 Asynchronous download

See list of parameters at section 5.1.

Asynchronous download is necessary in particular for data request greater than 100MB compressed, i.e. between 200 and 900 MB once files are uncompressed. There are two ways of executing the second example presented in the Async Data tab:

Credentials within the request

Simply replace `<username>` and `<password>` by your CSA username and password in the example below.

```
$ wget --content-disposition 'http://csa.esac.esa.int/csa/aio/async-product-
action?USERNAME=<username>&PASSWORD=<password>&DATASET_ID=C3_CP_ASP_IONC&START_DATE=20
04-06-18T11:35:00Z&END_DATE=2004-06-19T18:35:00Z&NON_BROWSER'
```



Authentication cookie

```
$ wget --load-cookies <cookies_file>.txt --content-disposition "http://csa.esac.esa.int/csa/aio/async-product-
action?DATASET_ID=C3_CP_ASP_IONC&START_DATE=2004-06-18T11:35:00Z&END_DATE=2004-06-
19T18:35:00Z&NON_BROWSER"
```

stores the request information at a txt file with a random name, for example: "8cf8ae7f-7ab7-4af5-8557-11ff6498b4c8.txt".

Sometimes it can store the output in a filename with the following syntax of "async-product-

action?USERNAME=<username>&PASSWORD=<password>&DATASET_ID=C3_CP_ASP_IONC&START_DATE=2004-06-18T11:35:00Z&END_DATE=2004-06-19T18:35:00Z&NON_BROWSER"

Example of content of the file:

```
Your request has been sent.
Your request ID is: [bmartine1060]
The file will be available at http://csa.esac.esa.int/ftp\_public/bmartine1060/bmartine1060.tar.gz
You will be notified by email when the data is ready for download
```

If a user cannot be authenticated, 'wget' will output an "authentication.log[.n]" file instead.

10.1.3 Data streaming

See list of parameters at section 6.1.

Please find below two ways of executing the last example presented in the Streaming tab:

Credentials within the request

Simply replace <username> and <password> by your CSA username and password in the example below.

```
$ wget --content-disposition 'http://csaint.esac.esa.int/csa/aio/streaming-
action?USERNAME=<username>&PASSWORD=<password>&DATASET_ID=C4_CP_PEA_LERL_PSD&START_DATE=2004-09-18T10:00:00Z&END_DATE=2004-10-20T22:00:00Z&GZIP=1&NON_BROWSER'
```

Authentication cookie

```
$ wget --load-cookies <cookies_file>.txt --content-disposition "http://csa.esac.esa.int/csa/aio/streaming-
action?DATASET_ID=C4_CP_PEA_LERL_PSD&START_DATE=2004-09-18T10:00:00Z&END_DATE=2004-10-20T22:00:00Z&NON_BROWSER"
```

stores the retrieved data at a gzip file with a random name, for example: "1fee8280-0759-4d7b-9071-fd088ae49c37.ccf.gz".

If the parameter "GZIP" is not present or its value is "0", then this request stores the retrieved data at a cef file with a random name, for example: "4a47622a-5f49-4f9d-b689-



27a55822da86.cdf".

If a user cannot be authenticated, 'wget' will output an "authentication.log[.n]" file instead.

10.2 Inventory request

See list of parameters at section 3.

There is no need to authenticate when requesting inventory information. Please find below how to execute the first example presented in the Inventory tab:

```
$ wget --content-disposition "http://csa.esac.esa.int/csa/aio/metadata-
action?SELECTED_FIELDS=DATASET_INVENTORY&RESOURCE_CLASS=DATASET_INVENTORY&QUERY=DATA
SET.DATASET_ID%20like%20'CM_CP_DWP_UT_PIOR'%20AND%20DATASET_INVENTORY.START_TIME%20<=
%20'2013-03-02T02:00:00Z'%20AND%20DATASET_INVENTORY.END_TIME%20>=%20'2013-03-01T00:00:00Z'&
RETURN_TYPE=CSV&NON_BROWSER"
```

stores the result at a file named: "CSA-Results.csv[.n]" (where "n" is a number that increments in order to not overwrite the file).

10.3 Metadata request

See list of parameters at section 2.1.

No authentication needed, please find below how to execute the last example presented in the Metadata tab:

```
$ wget --content-disposition "http://csa.esac.esa.int/csa/aio/metadata-
action?SELECTED_FIELDS=FILE.FILE_NAME,FILE.START_DATE,FILE.END_DATE,FILE.CAA_INGESTION_DATE
&RESOURCE_CLASS=FILE&QUERY=(DATASET.DATASET_ID%20like%20'C1_CP_EDI_EGD'%20OR%20DATASET.
DATASET_ID%20like%20'C1_CP_EFW_L3_P'%20OR%20DATASET.DATASET_ID%20like%20'C1_CP_FGM_FULL')
%20AND%20FILE.START_DATE%20<=%20'2001-02-05'%20AND%20FILE.END_DATE%20>=%20'2001-02-
01'&PAGE_SIZE=1000&PAGE=1&RETURN_TYPE=CSV&NON_BROWSER"
```

stores the result at a file named: "CSA-Results.csv[.n]" (where "n" is a number that increments in order to not overwrite the file).

10.4 How to specify the output file?

Be aware of the following indication at the "wget" manual:

'--content-disposition'

If this is set to on, experimental (not fully-functional) support for Content-Disposition headers is enabled. This can currently result in extra round-trips to the server for a HEAD request, and is known to suffer from a few bugs, which is why it is not currently enabled by default. This option is useful for some file-downloading CGI programs that use Content-Disposition headers to describe what the name of a downloaded file should be.



Thus, maybe sometimes (especially with very old versions of `wget`), this may not work, in which case, you can always provide an output filename:

```
$ wget -O myRequest.tar.gz -load-cookies cookies.txt "http://csa.esac.esa.int/csa/aio/product-
action?DATASET_ID=C3_CP_ASP_IONC&START_DATE=2004-06-18T11:35:00Z&END_DATE=2004-06-
19T18:35:00Z&NON_BROWSER"
```

Please have in mind that if you do a sequence of `wgets`, the output filename must be changed, otherwise it will be overwritten.

10.5 How to create an authentication cookie?

In order to download data from Cluster Archive it is required to be able to authenticate the user. For this reason it is strongly advised to create and store a session cookie that identifies the user.

The syntax to create and store the cookie is:

```
$ wget --keep-session-cookies --save-cookies <cookies_file>.txt --content-disposition
'http://csa.esac.esa.int/csa/aio/login-
action?USERNAME=<username>&PASSWORD=<pwd>&SUBMIT=LOGIN&NON_BROWSER'
```

Where:

- `<cookies_file>` is the name of the file where the user wants to store the cookie. If no path is supplied, it will be created at the current directory, so if you will be executing '`wget`' from different directories, we recommend to provide an absolute path
- `<username>` is the CSA user identifier
- `<password>` is the CSA password of the user

It creates a cookies file with content similar to this one:

```
# HTTP cookie file.
# Generated by Wget on 2013-05-21 17:10:34.
# Edit at your own risk.

csa.esac.esa.int FALSE / FALSE 0 JSESSIONID 290552733F015D64370C02677A0B02
6D7A1450672A015D75650D4060284656722A465A6F3F1451642A1C416831465345391C5A6F2E12737228015C7138465
6722A465A6F3F
```

The login request will return a file named: "`credentials.log`" indicating whether the user has been successfully logged in or not.

The authentication cookies generated by CAIO are session cookies, meaning that they are not stored and are deleted when the browser exits. This is the reason for using "keep-



session-cookies" option at 'wget' when saving the cookie. As long as the cookies file is not modified, it can be used in all future requests with 'wget'.

10.6 Hint on descriptive error messages

When wget gets a response with an error status, it just displays the error code with the standard error message.

In some cases the error response contents some text explaining the reason of the failure. This text can be extracted with 'curl', using the syntax below:

```
$ curl -b <cookies_file>.txt -R "<url of failing request>"
```

For example, a request with an incorrect date format like:

```
$ curl -b <cookies_file>.txt -R "http://csa.esac.esa.int/csa/aio/streaming-  
action?DATASET_ID=C4_CP_PEA_LERL_PSD&START_DATE=2004-09-18T10:00:00Z&END_DATE=2004-10-  
2022:00800Z&NON_BROWSER"
```

returns the following response:

```
ERROR: 400  
The string [2004-10-2022:00:00Z] does not match an ISO Date format
```



11 APPENDIX: HOW TO LINK CAIO WITH MATLAB OR IDL

In this appendix some recommendations for users that want to access the Cluster Science Archive directly from Matlab and/or IDL are given.

11.1 Matlab

Please note that the Authentication option in `urlwrite.m` is ONLY available since Matlab release 2013a. You can check your matlab version by typing `ver` at the Matlab prompt. Hence, two examples are proposed here.

11.1.1 For users with Matlab 2013a or later releases

Simply replace `<username>` and `<password>` by your CSA username and password in the example below.

Example:

```
URL = 'http://csa.esac.esa.int/csa/aio/product-action';
fileName=tempname;
gzFileName = [fileName '.gz'];
[gzFileName,st]=urlwrite(URL,gzFileName, 'Authentication', 'Basic','Get', { 'Username', '<username>', 'password',
'<password>', 'DATASET_ID', 'C1_CP_WHI_ELECTRON_DENSITY', 'START_DATE', '2011-11-10T18:00:00Z',
'END_DATE', '2011-11-10T21:00:00Z', 'NON_BROWSER', '1'});
gunzip(gzFileName);
fileNames=untar(fileName);
for iFile = 1:numel(fileNames), disp(fileNames{iFile}); end
```

This set of commands creates in your local folder a CSA_Download type folder with the following folder/file inside:

`C1_CP_WHI_ELECTRON_DENSITY/C1_CP_WHI_ELECTRON_DENSITY__20111110_180000_20111110_210000_V120703.cdf`

11.1.2 For users with Matlab 2012b or earlier releases

Simply replace `<username>` and `<password>` by your CSA username and password in the example below.

Example:

```
URL_CSA='http://csaint.esac.esa.int/csa/aio/product-
action?USERNAME=<username>&PASSWORD=<password>&DATASET_ID=C2_CP_RAP_PAD_L3DD&START_DAT
E=2010-01-01T00:00:00Z&END_DATE=2010-01-01T23:59:59Z&NON_BROWSER';
fileName=tempname;
gzFileName = [fileName '.gz'];
[gzFileName,st]=urlwrite(URL_CSA,gzFileName);
gunzip(gzFileName);
fileNames=untar(fileName);
for iFile = 1:numel(fileNames), disp(fileNames{iFile}), end
```



These set of commands creates in your local folder a CSA_Download type folder with the following folder/file inside:

C2_CP_RAP_PAD_L3DD/C2_CP_RAP_PAD_L3DD__20100101_000000_20100101_235959_V110630.cef

11.2 IDL

Below you can find two IDL routines developed by Andrew Walsh, that use the CAIO to download data directly from IDL. They work with Linux, Mac and Windows enabling login and product actions. The error handling will be improved in the near future, together with the actions for unpacking the downloaded tar files.

csa_login

```
function csa_login,user,pass

;Function that logs in to the CSA AIO system.
;Parameters:
;
;           USER: String containing your RSSD LDAP user identifier
;           PASS: String containing your RSSD LDAP password
;
;Return Value:
;
;           If login is successful, returns a string containing a
JSESSIONID cookie
;           If login is unsuccessful, returns 0
;
;Example:
;
;           credentials = csa_login('username','password')
;
;Andrew Walsh, 2013-05-30, awalsh[AT]rssd[DOT]esa[DOT]int
;
;Updated to add support for reserved characters in passwords
;Andrew Walsh, 2015-01-28, awalsh[AT]sciops[DOT]esa[DOT]int

;Check for reserved characters in password and replace with URL-Encoded
equivalents.

    reserved =
["%", "!", "#", "$", "&", "'", "(", ")", "*", "+", ",", "/", ":", ";", "=", "?", "@", "[", "]" ]
    encoded =
["%25", "%21", "%23", "%24", "%26", "%27", "%28", "%29", "%2A", "%2B", "%2C", "%2F", "%3A", "%3B", "%3D", "%3F", "%40", "%5B", "%5D"]

    for i = 0, n_elements(reserved) -1 do begin
        startpos = 0
        foundpos = 0
        reserved_positions = -1
        while foundpos ne -1 and startpos le strlen(pass)-1 do begin
            foundpos = strpos(pass,reserved[i],startpos)
```



```

        if foundpos ne -1 then begin
            if startpos eq 0 then reserved_positions =
foundpos else reserved_positions = [reserved_positions,foundpos]
            startpos = foundpos+1
        endif
    endwhile
    if reserved_positions[0] ne -1 then begin
        reserved_positions = reverse(reserved_positions)
        for j = 0, n_elements(reserved_positions) -1 do begin
            if reserved_positions[j] eq strlen(pass)-1 then
begin
                pass_before =
strmid(pass,0,reserved_positions[j])
                new_pass = pass_before+encoded[i]
                pass = new_pass
            endif else if reserved_positions[j] eq 0 then
begin
                pass_after =
strmid(pass,reserved_positions[j]+1,strlen(pass)-(reserved_positions[j]+1))
                new_pass = encoded[i]+pass_after
                pass = new_pass
            endif else begin
                pass_before =
strmid(pass,0,reserved_positions[j])
                pass_after =
strmid(pass,reserved_positions[j]+1,strlen(pass)-(reserved_positions[j]+1))
                new_pass =
pass_before+encoded[i]+pass_after
                pass = new_pass
            endelse
        endfor
    endif
endfor

;construct host, path and query parts of URL
    csa_host = 'csaint.esac.esa.int'
    csa_login_path = 'csa/aio/login-action'
    csa_login_query =
'USERNAME='+user+'&PASSWORD='+pass+'&SUBMIT=LOGIN&NON_BROWSER'

;define successful login string for comparision with what's returned from CSA
AIO
    csa_login_success = '['+user+'] has successfully logged in!'

;create and set properties of an IDLnetURL object
    csa_login_obj = obj_new('IDLnetUrl')
    csa_login_obj->SetProperty, url_scheme = 'http'
    csa_login_obj->SetProperty, url_host = csa_host
    csa_login_obj->SetProperty, url_path = csa_login_path
    csa_login_obj->SetProperty, url_query = csa_login_query

;send login request to CSA AIO

```



```

        csa_login_response = csa_login_obj->get(/string_array)
;print response from CSA AIO
    print, csa_login_response
;check for successful login and extract session cookie from CSA AIO response
    if csa_login_response[0] ne csa_login_success then begin
        return, 0
    endif else begin
        csa_login_obj->getproperty, response_header=csa_login_header
        csa_cred_start = strpos(csa_login_header, 'JSESSIONID')
        csa_cred_end = strpos(csa_login_header, ';')
        csa_creds = strmid(csa_login_header, csa_cred_start, csa_cred_end-
csa_cred_start)
        return, csa_creds
    endelse
end

```

csa_product

```

function csa_product, csa_creds, csa_dataset_id, csa_start_date, csa_end_date,$
    delivery_format=csa_delivery_format, delivery_interval=csa_delivery_interval,$
    ref_doc=csa_ref_doc, ingested_since=csa_ingested_since, max_file_size = csa_max_file_size

;Function that Downloads data from the CSA AIO system.
;Parameters:
;  CSA_CREDS: String containing the JSESSIONID cookie returned from csa_login.pro
;  CSA_DATASET_ID: String array containing one or more valid dataset ids, one per element
;  CSA_START_DATE: String containing start date for data request, in the format 'yyyy-mm-
ddThh:mi:ssZ'
;  CSA_END_DATE: String containing end date for data request, in the same format as csa_start_date

;Keywords:
;  Optional parameters accepted by the CSA AIO product action, with the exception of NON_BROWSER and
RETRIEVALTYPE, are accepted as keywords.
;  Optional parameters MUST be passed as strings to the relevant keyword. See the CSA AIO User
Manual for accepted values.
;  Note START_DATE and END_DATE are mandatory and passed as parameters here.
;Return Value:
;  If download is successful, returns a string containing the downloaded file name.
;  If download is unsuccessful, returns 0 (or falls over gracefully).
;
;Example:
;  To download full resolution FGM data from C1 and C2 22:50-23:50 UT, 7 August 2004, including
caveats, in cdf format:
;
;  credentials = csa_login('username','password')
;  filename = csa_product(credentials,['C1_CP_FGM_FULL','C2_CP_FGM_FULL'],'2004-08-
07T22:50:00Z','2004-08-07T23:50:00Z',delivery_format='CDF',REF_DOC='1')
;
;Andrew Walsh, 2013-05-30, awalsh[AT]rssd[DOT]esa[DOT]int

```



```

;define URL host and path
csa_host = 'csa.esac.esa.int'
csa_product_path = 'csa/aio/product-action'

;construct URL query from supplied parameters and keywords
csa_product_query = ''
for i = 0, n_elements(csa_dataset_id) -1 do begin
    csa_product_query=csa_product_query+'DATASET_ID='+csa_dataset_id[i]+'&'
endfor
csa_product_query =
csa_product_query+'&START_DATE='+csa_start_date+'&END_DATE='+csa_end_date+'&NON_BROWSER'

    if keyword_set(csa_delivery_format) then
csa_product_query=csa_product_query+'&DELIVERY_FORMAT='+csa_delivery_format
    if keyword_set(csa_delivery_interval) then
csa_product_query=csa_product_query+'&DELIVERY_INTERVAL='+csa_delivery_interval
    if keyword_set(csa_ref_doc) then csa_product_query=csa_product_query+'&REF_DOC='+csa_ref_doc
    if keyword_set(csa_ingested_since) then
csa_product_query=csa_product_query+'&INGESTED_SINCE='+csa_ingested_since
    if keyword_set(csa_max_file_size) then
csa_product_query=csa_product_query+'&MAX_FILE_SIZE='+csa_max_file_size

;construct credentials string for http header
csa_cookie = 'Cookie: '+csa_creds

;Create IDLnetURL object and set properties
csa_product_obj = obj_new('IDLnetUrl')
csa_product_obj->SetProperty, VERBOSE=1
csa_product_obj->SetProperty, url_scheme = 'http'
csa_product_obj->SetProperty, url_host = csa_host
csa_product_obj->SetProperty, url_path = csa_product_path
csa_product_obj->SetProperty, url_query = csa_product_query
csa_product_obj->SetProperty, headers = csa_cookie

;send request to CSA AIO system, saving response in csa_buffer.dat
print, 'Getting data from CSA, please wait.'
csa_product_response = csa_product_obj->get(filename='csa_buffer.dat')
csa_product_obj->getproperty, response_header=csa_product_header

;check a .tar.gz file was downloaded and if so rename buffer to correct filename and return correct
filename, otherwise return 0
csa_filestart = strpos(csa_product_header,'filename=')
if csa_filestart ne -1 then begin
    csa_fileend = strpos(csa_product_header,'gz"')
    csa_filename = strmid(csa_product_header,csa_filestart+10,csa_fileend-csa_filestart-8)
    csa_dir_end = strpos(csa_product_response,'csa_buffer.dat')
    csa_working_dir = strmid(csa_product_response,0,csa_dir_end)
    file_move, csa_product_response, csa_working_dir+csa_filename
    print, 'Downloaded data to '+csa_working_dir+csa_filename
    outfile = csa_working_dir+csa_filename
    return, outfile
endif else begin

```



```
print, 'Something went wrong.'  
return, 0  
endelse  
end
```