Archiving Task List Guide

Revision	Author	Scope
July-2016	DKRZ Datamanagement	public release

Intended Audience

DKRZ-LTA Data Providers, DKRZ-LTA Data Manager

Contents

Introduction

To perform the long-term archiving of data files, it is necessary that the elements to be archived and their assignment to metadata in the long-term archive are described clearly and completely.

To facilitate the process of archiving[1] (figure 1) a format for archiving task lists (ATL-format) is defined by DKRZ Long-Term Archive (DKRZ-LTA). This format is described in this document and further illustrated by examples.

We strongly recommend using this format to specify archival jobs to DKRZ_LTA.

[1] http://cera-www.dkrz.de/docs/DataSubmissionPreparationGuide.pdf

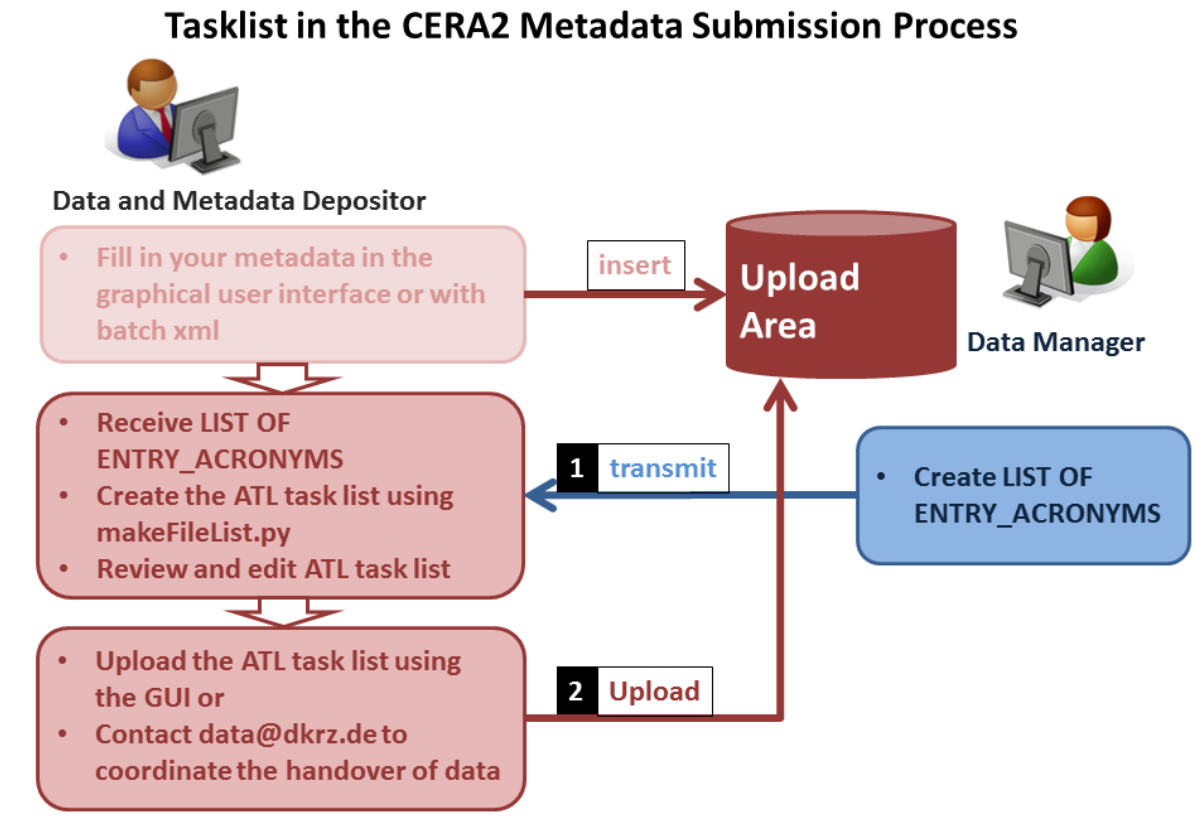
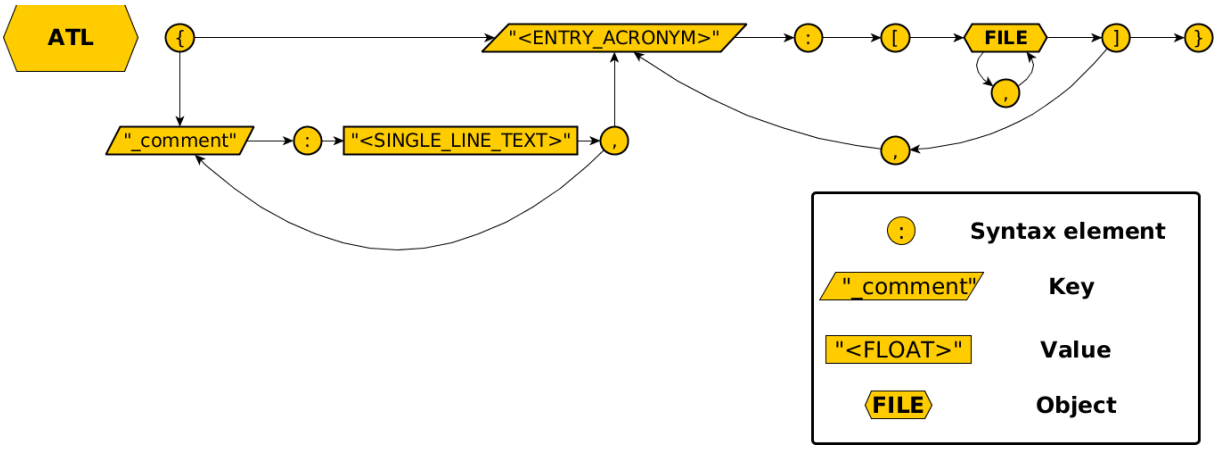


Figure 1

Format Definition

The archival task list used by DKRZ-LTA is formatted in JSON[2]. The list contains an assignment of one or more datasets named by their CERA entry_acronyms and an ordered list of data file information.

In figure 2 the ATL format is represented within a syntax diagram.



[2] http://www.json.org/

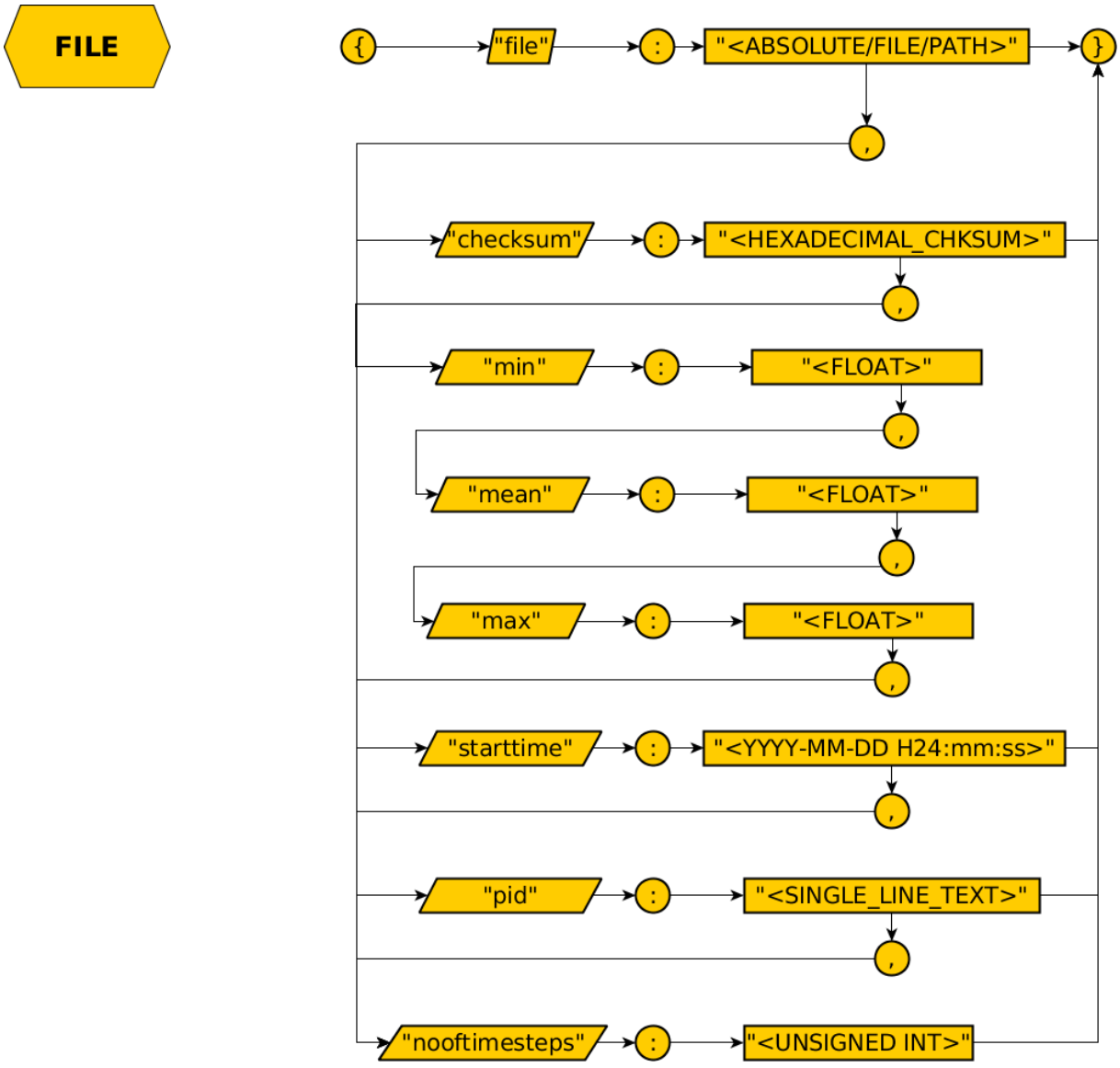


Figure 2: ATL syntax diagram

Important rules:

- The order or elements in the file list is exactly the order the specified files will be preserved within the archive.
- The order of entry_acronyms is not relevant.
- All specifications except the name of the files within file lists are optional.
- All optional specifications except checksum need to be specified for all elements in the same way as they are specified for the very first file in the archival order list.
- If one of min/max/mean is specified, all of them need to present.

Generation of archiving task lists

To ease the process of generating ATL file lists, DKRZ provides a simple python tool “makeFileList.py” which prepares an ATL file either for the complete contents of a directory or from a simple file containing only file names.

“makeFileList.py” is available for download[3] and already installed on mistral.dkrz.de as '/home/dkrz/k204206/bin/makeFileList.py' for those who already have the data available on DKRZ hosts. The python script requires python 2.7. On DKRZ nodes you may have to load python 2.7 first, by calling ‘module load python/default’. Not all options of the ATL format are supported by the script.

The syntax is as follows:

python makeFileList.py -h

== makeFileList.py ==

Usage

makeFileList.py -e <entry_acronym> {-d <src_dir> | -i <filelist>} [-c] -o <outputfile>

Parameters:

-d An existing directory which provides all files corresponding to the entry_acronym

-i An ordered list of files corresponding to the entry_acronym

-h Help

-c Calculates checksum of all files (MD5)

-o Outputfile as JSON

-e CERA entry acronym

When generating files we recommend to include the date and time of preparation of the archive task list as a comment in the very first line of the list.

Example for makeFileList.py:

python makeFileList.py –e DUMMY_ENTRY_ACRONYM –i file_list.txt –o out.txt

With more file_list.txt:

Data_file1.nc
Data_file2.nc
Data_file3.nc

[3] http://cera-www.dkrz.de/tools/makeFileList.py

ATL Examples

Basic example

```
{
    "MY_DUMMY_DATASET": [
        {
            "file": "/work/bm0123/m300456/MY_DATA/a.nc"
        }
    ]
}
```

Complete example

```
{
  "_comment" : "This is a comment",
  "_comment" : "This is another comment",
  "ANOTHER_DUMMY_DATASET" : [
    {
      "file" : "/work/bm0123/m300456/MY_DATA/a.nc",
      "min": 3,
      "max": 4,
      "mean": 3.5,
      "starttime": "1991-01-16 12:31:11",
      "checksum" : "1aa046db4bd1e82f668d4e0696724117",
      "nooftimesteps" : 2
    }
  ]
}
```

Example with two datasets

```
{
    "DATASET_1": [
        {
            "file": "/work/bm0123/m445778/MY_DATA/a.nc"
        },
        {
            "file": "/work/bm0123/m445778/MY_DATA/b.nc"
        }
     ],
    "DATASET_2": [
        {
            "file": "/work/bm0456/OTHER_DIR/abc.nc"
        }
     ]
}
```

Contact

In case any questions regarding preparation of the task list arise, please contact DKRZ-LTA at data@dkrz.de .

Appendix

These are the elements used by the ATL-format. All specifications in single quotes (‘) have to be present in the JSON list. All specifications in square brackets are optional.

- ARCHIVAL_TASK_LIST = '{' LIST_OF_ENTRY_ACRONYMS '}'
- LIST_OF_ENTRY_ACRONYMS = [COMMENT','] ENTRY_ACRONYM ':' FILELIST [',' COMMENT] [',' LIST_OF_ENTRY_ACRONYMS]
- COMMENT = '_comment' ':' '"'<any text>'"' [',' COMMENT]
- ENTRY_ACRONYM = '"' <alphanumeric string and underscores> '"'
- FILELIST = '[' FILEINFO [',' FILEINFO] ']'
- FILEINFO = '{' FILE [MINMAXMEAN] [STARTZEIT] [CHECKSUM] [TIMESTEPS][PID]'}'
- FILE = '"file"' ':' '"' <absolute path> '"'
- MINMAXMEAN = ',' '"min"' ':' FLOAT ',' '"max"' ':' FLOAT ',' '"mean"' ':' FLOAT
- STARTTIME = ',' '"starttime"' ':' '"' <timestamp: 'YYYY-MM-DD hh:mm:ss' hours 24h-format; time optional> '"'
- CHECKSUM = ',' '"checksum"' ':' '"' <hexadecimal, MD5 unless otherwise requested> '"'
- TIMESTEPS = ',' '"nooftimesteps"' ':' <unsigned integer>
- FLOAT = <floating point number; decimals separated by a dot>
- PID = ‘_pid' ':' '"'<any text>'"'

All specifications in single quotes (‘) have to be present in the JSON list. All specifications in square brackets are optional.