JPL D-XXXXX

Atmospheric Infrared Sounder (AIRS) Project

A Quick Guide for TDS Data Access

Version 1.11
Preliminary

October 3, 2001

National Aeronautics and Space Administration
California Institute of Technology
Jet Propulsion Laboratory
4800 Oak Grove Drive
Pasadena, California 91109

Prepared by:

_____
Albert Y. Chang
Cognizant Development Engineer, Job Planning Subsystem, TLSCF Data System

_____
Quentin Sun
System Engineer, TLSCF Data System (TDS)

Approved by:

_____
Navid Dehghani
Project Element Manager, AIRS Science Processing System

Document Log

Date	Page Number	Status
6/6/2001	All	Preliminary for internal review
6/14/2001	All	Version 0.2; directory hierarchy revision
6/15/2001	All	Version1.0; for distribution at AIRS Science Team Meeting 6/19-21/2001
9/13/01	7; Appendices	Updated Figure, Appendices
9/18/01	6, 8, 9	Errata
10/3/01	9	Errata: airs-dom -> airs-dom:0

TABLE OF CONTENTS

Appendices: TDSDataHierarchy.xls

1. INTRODUCTION

1.1 Identification

This document describes methods that can be used to query and retrieve AIRS science data products from the AIRS TLSCF Data System (TDS) for supporting the post-launch AIRS science validation activities.

This document describes the organization of the dynamic (temporal-based) data files that are either received from the external data sources or generated at the AIRS TLSCF. It does not discuss the organization of static ancillary files that are part of the AIRS Science Processing System (SPS) software delivery.

Since TDS is still under development as this document is written, the implementation details (e.g., the complete list of supported file types, metadata keywords, etc.) are kept in a separated working document entitled "*TDS Data Hierarchy*" which can be found in the TDS_INFORMATION area of the AIRS Science Team Link in the Project area of the AIRS web site (www-airs.jpl.nasa.gov).

1.2 Overview

TDS uses the Distributed Object Manager (DOM) to organize and catalog data files. The following web site provides more information regarding DOM: http://eis.jpl.nasa.gov/dom.

1.2.1 Data Organization

The AIRS science data archive is organized into six primary data Collection types as follows:

1. The **level0** Collection contains all level 0 raw dynamic input data files that the TLSCF receives on a continuing basis in order to perform L1a processing. It includes PM1EPHMD (ephemeris), PM1ATTNR (attitude), PMCO_HK (S/C carry out), and all AIRS L0 PDS files.
2. The **tlscf** Collection contains AIRS products generated at the TLSCF through routine baseline processing.
3. The **gdaac** Collection contains AIRS products generated at the Goddard DAAC and subscribed by the AIRS TLSCF for QA monitoring.
4. The **test** Collection contains AIRS products generated at the TLSCF with test software or test input data, based on production requests approved by the Change Control Board (CCB) appointed by the AIRS Project.
5. The **sim** Collection contains simulated AIRS data files generated off-line by the AIRS simulation software, as well as AIRS product files derived from that simulation data.
6. The **correl** Collection contains correlative data files that the TLSCF receives on a continuing basis. Some of these are inputs to processing at the TLSCF such as the AVN files, and radiosonde and surface marine observations. Other file types are archived solely for use by the validation team.

1.2.2 Data Access Methods

There are two methods to retrieve data from the TDS. They are:

1. Directory Access: data files are stored in the TDS based on primary Collection type, date, and data type. Data users can traverse the UNIX file directories directly to locate files if these parameters are known.
2. Catalog Access: when data files are stored in TDS, metadata pertaining to each file is generated and stored in the TDS catalog. Data users can query the catalog for all files that satisfy any combination of matching criteria based on the metadata available for each file

type. Data users have the choice of whether a physical local file, a symbolic link, or a fully qualified file name is returned. There are two ways to access the TDS catalog, namely:

a) GUI: data users can use the Catalog Navigator (**catnav**) to interactively navigate through the collection structure that is based on primary Collection type, date and data type; the Catalog Navigator also provides a tool to interactively build catalog queries which can be used to retrieve files or interrogate metadata.

b) Command Line: alternatively, data users may use the **dom_getfile** command to directly query the catalog to retrieve files. This is useful for incorporation into scripts.

These methods will be discussed in full detail later in this document.

1.3 User Account/Authorization

In order to access either the TDS file directories or catalog, the data user must first acquire a computer account to log onto a specific TDS client node, for instance, *weather.jpl.nasa.gov*.

2. Directory Access Method

TDS organizes data files using an open (non-hidden) file directory hierarchy. When a file is ingested into the system, it is placed into a directory following pre-defined rules. TLSCF policy states that once a file is placed in a directory, it is never logically moved from that directory. This assures that user-created file links are preserved.

2.1 The Root

The root of the operational TDS file directory tree is always **/dom/files/ops** regardless of which TDS client node that the user logs onto.

2.2 The Primary Collections

Figure 2.1 shows the operational TDS file directory hierarchy.

Three of the six primary Collections are directly mapped to directories of root (**test**, **sim**, and **correl**), while the remaining three are organized under the **airs** directory (**airs/level0**, **airs/tlscf**, and **airs/gdaac**). This structure ensures the same number of directory level are traversed to access any specific type of AIRS file, regardless of whether it is in the **test**, **sim**, **tlscf**, or **gdaac** Collection.

The data within all these Collection sub-trees are organized by date and data type. Each leaf node (lowest level directory) is based on a single data type, and that directory is named after the ESDT short name (lower cased). The date is organized as either yyyy/mm/ or yyyy/mm/dd. Currently, only AIRS 6 minute granule products are binned in daily directories; all other file types are binned monthly. This is to done to optimize the number of files that the catalog needs to manage in each leaf node.

The date used to determine the target directory is contained in the catalog metadata field DOMContainerDate. For most file types this is based on the data granule start date (the ECS metadata field RangeBeginningDate). The exceptions are AVN files (based on the predict date [though this is mapped internally to the catalog RangeBeginningDate]), Multi-day AIRS VIS map files (the RangeEndingDate), and all RaOb and PREPQC-related files (the RangeEndingDate , to handle properly the T00Z file). Note for AIRS products, the DOMContainerDate is designed to be exactly the same date as that used in the LocalGranuleId naming convention.

On ingest, AIRS Product files are renamed to the value of the LocalGranuleID entry in their ".met" files. Within the directories, each file is seen to be associated with a DOM metadata file (.hdr) and an ESDIS metadata file (.met). Note however, these metadata files are not directly visible when accessing the data using the catalog navigator.

Specifics of the different sub-trees are discussed below.

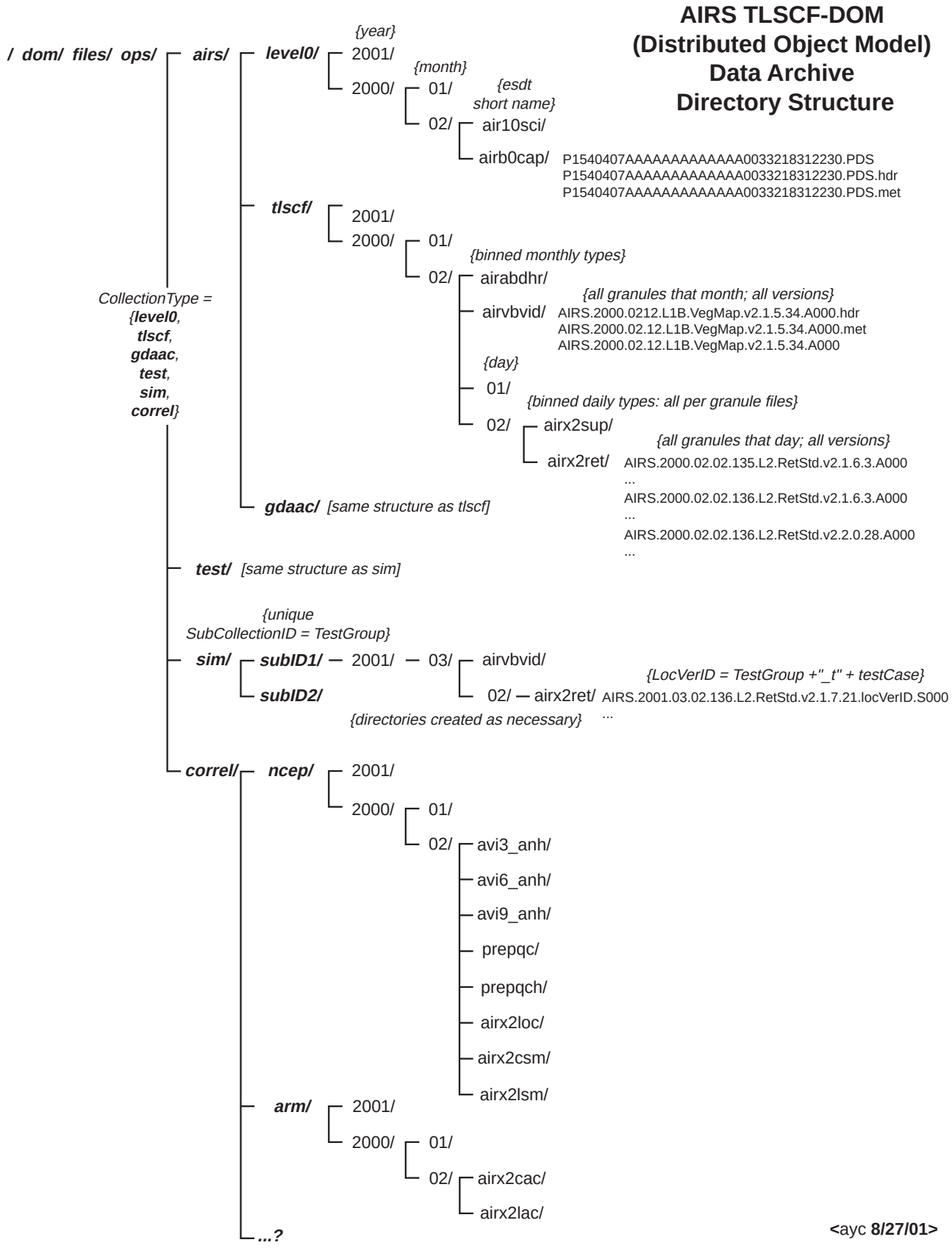

Figure 2.1 TLSCF Directory Structure

2.2.1 The **airs** Sub Tree

The **airs** sub-tree contains the **level0**, **tlscf** and **gdaac** Collections. The **level0** Collection contains instrument and spacecraft inputs to L1a processing and consequently is of little interest to most investigators.

Baseline AIRS products, processed at the TLSCF or the GDAAC, are found in the **tlscf** and **gdaac** Collections. These Collections have near-identical hierarchical directory structures, except that more data types are supported under **tlscf**.

All files in the **gdaac** Collection have their ProductGenerationFacility (the 4th to last character in the file name) as 'G', whereas those in **tlscf** files have 'A'. Furthermore, **gdaac**, files are distinguished by a 3-parameter PGE Version (major, minor, revision), whereas **tlscf** files include a fourth parameter (build number). Note multiple PGE versions of any given granule can coexist in the same directory. These files can be distinguished because the PGE Version of each AIRS product file is embedded in the filename.

Note for identification purposes, products generated in the **tlscf** collection can have a non-trivial LocalVersionId, but these are not propagated in to the LocalGranuleId filenames.

2.2.2 The **sim** and **test** Sub Trees

Test or Simulated AIRS products are found in the **test** and **sim** Collections. Files from the **test** Collection have their ProductGenerationFacility set to 'T', while those from **sim** have 'S'. As with **tlscf** files, **sim** and **test** files have 4-parameter PGE Versions.

At the top most level, these two Collections are organized by a unique **test** or **sim** ID (known in the catalog as SubCollectionID). To facilitate symmetry in the data access, these sub-trees are organized using the same yyyy/mm/dd convention as the baseline AIRS product trees, though only the minimum required directories are created as necessary to support each test run.

Before a set of simulated products can be archived in the TDS, a unique ID must be assigned to the data set. The simulation software must set the .met file field LocalVersionID to this identifier and follow the file naming rules for embedding this string into the LocalGranuleID.

Note, for cases of **test** or **sim**, the SubCollectionId is either equal to, or is a substring (starting at the first char) of, theLocalVersionId. For instance, one convention that may be adopted for organizing **sim** and **test** data is that the LocalVersionId = SubCollectionID + "_t" + testCaseId.

2.2.3 The **correl** Sub Tree

The correl directory contains correlative data files from other instruments needed for validation. This includes the "truth-location" files used for match-up processing (e.g, for RaObs and surface marine observations) generated at the TLSCF during the ingest process.

At the top-most level, the **correl** Collection is broken down into several directories depending on the data source. For the file types currently identified to be cataloged at the TLSCF, **ncep** and **arm** are the relevant data sources. Below this level, files are organized using the familiar yyyy/mm/esdt system (so far, no file types have been identified that need to be binned daily).

In some cases correlative data types will need to be cataloged that do not have official ESDT short names; in these cases, a TDS-defined short name will be assigned.

3. Catalog Access Method

3.1 Setting the UNIX Execution Environment

Upon logging on a TDS client node, the data users must reset their UNIX execution environments as follows:

1. Include **/dom/bin** in the execution path**.** This can be done by running the following command in the C shell command interpreter (or the equivalent in other shell command interpreter):

   "*setenv PATH ${PATH}:/dom/bin*"

2. Include **/dom/lib** in the dynamic library sarch path. This can be done by running the following C-shell command:

   "*setenv LD_LIBRARY_PATH ${LD_LIBRARY_PATH}:/dom/lib*"

3. Set CATALOG_SERVER to **airs-dom:0** by running the following C-shell command:

   *"setenv CATALOG_SERVER **airs-dom:0**"*

3.2 The GUI-based Catalog Navigator (catnav)

DOM supplies a GUI-based catalog navigator called *catnav* that can be used as-is by TDS data users to access the catalog. The manual page for the *catnav* command can be accessed from the following web site: http://eis.jpl.nasa.gov/dom/help/index.html.

Once the proper UNIX environments have been set up as outlined in the previous section, the TDS data users can invoke the GUI-based catalog navigator by simply typing "*catnav &*" at the UNIX command prompt.

Alternatively, the airs dom server can be explicitly specified, "*catnav -sairs-dom -p0 &*".

3.3 The Catalog Access Command (dom_getfile and cat_getfile)

The *dom_getfile* command is a wrapper program that wraps around the *cat_getfile* command provided by DOM to the TDS data users. They accept the same command line arguments. The only difference between these two commands is that the *dom_getfile* command returns the associated metadata file in addition to the data file where as the *cat_getfile* command returns only the data file. We will only discuss the cat_getfile command in this section.

Cat_getfile is a command-line utility that enables TDS data users to query the TDS data catalog. The result returned by *cat_getfile* is a set of files, which is either listed by name (by default), identified with symbolic links or physically copied to the user's directory.

This section describes a minimum set of command line parameters that need to be specified for a nominal query. The detailed manual page of the *cat_getfile* command can be found at the following web site http://eis.jpl.nasa.gov/dom/help/index.html.

The following is the syntax of a typical (super) query that makes use of the *cat_getfile* command to access the TDS data catalog:

cat_getfile –r airs –t FileType –x MaxResultEntries [–c|-l] –W'MetadataKeyword = "value"'

where:

- -r airs: specifies the root of a subtree where the search will start
- -t FileType: FileType specifies name of a TDS internal data file type that the user is interested in searching for in the query. For example: L1A_AMSU_T, L1A_HSB_T, Any_L1A_T, etc.

NOTE: All TDS-supported file types are documented in the working document entitled *TDS Data Hierarchy*, which can be found in the TDS_INFORMATION area of the AIRS Science Team Link in the Project Area of the AIRS web site.

- -x MaxResultEntries: MaxResultEntries is an integer that specifies the maximum number of files to be returned in the result file set. It provides a safety net for queries that may accidentally produce an excessive result set.

- [-c|-l]: These options are optional. If none is specified, the default is that a file list will be returned as the search result. Such a file list that contains one file entry per line could then be used as input for other application program.

  When the –c option is specified, the search result will contain files that are physically copied over to the data user's current (local) working directory.

  When the –l option is specified, the search result will contain a set of symbolic links created in the data user's current (local) working directory.

- -W'MetadataKeyword = "value"': the –W option allows data user to enter additional search criteria by specifying value for a predefined metadata keyword. In addition, multiple –W options can be specified in a query. In that case, all the metadata keywords will be joined with the logical "AND" operator.

  NOTE: All supported metadata keywords/parameters are documented in the working document entitled *TDS Data Hierarchy*, whose location is described above.

3.4 Query Examples Using cat_getfile

3.4.1 Example 1

The following catalog query creates symbolic links for the first 50 L1B AMSU Radiance product files found on the **tlscf** data Collection with data date equal to September 13, 1998:

cat_getfile –r airs –t L1A_AIRS_T –x50 –l –W'CollectionType="tlscf"' –W'DOMContainerDate="1998-09-13"'

3.4.2 Example 2

The following two examples are equivalent. They both are searching for any L2 data file in **tlscf** that has data date in the month of January, 2001 and NumClearMW>=10. Note the second version is more concise because it uses "inside" knowledge of the DOM server configuration (this, along with more useful insights, can be gleaned through using the catnav GUI).

cat_getfile -r airs -t Any_L2_T -W 'CollectionType = "tlscf" AND DOMContainerDate >= 2001-01-01 AND DOMContainerDate <= 2001-01-31 AND NumClearMW >= 10'

cat_getfile -r airs_tlscf_jan_2001 -t Any_L2_T -W 'NumClearMW >= 10'

3.4.3 Example 3

The following query searches for L1B AIRS radiance file that was produced by Goddard DAAC (gdaac) with LatGranuleCen <= 5 and AutomaticQualityFlag = suspect. If any file is found, a symbolic link is created in the user's current working directory.

cat_getfile -r airs -t L1B_AIRS_Rad_T -l -W 'CollectionType = "gdaac" AND LatGranuleCen <= 5 AND AutomaticQualityFlag = "Suspect"'

Appendix

For the hard-copy distributions of this document, the reference TDSDataHierarchy document is attached. This describes the directory and type hierarchies, and the AIRS-DOM catalog metadata.

The most-up-to-date copy of TDSDataHierarchy can be found in the TDS_INFORMATION area of the AIRS Science Team link in the Project area of the AIRS web site (www-airs.jpl.nasa.gov).