AIRS Science Processing System (ASPS): A Description of Architecture and Capabilities

Navid Dehghani, Evan M. Manning, and Quentin Sun
California Institute Of Technology
Jet Propulsion Laboratory
4800 Oak Grove Drive
Pasadena, CA 91109-8099

ABSTRACT

The Atmospheric Infrared Sounder (AIRS) is a grating spectrometer designed to obtain high-resolution absorption spectra from the atmosphere. The AIRS instrument is scheduled for launch in late 2001 or early 2002 on the Earth Observing System (EOS) Aqua platform. The primary scientific objectives of AIRS are to investigate the dynamics of the atmosphere (including cloud properties), to look for climatological changes, and to improve weather prediction. These investigations will be carried out on a global scale. The instrument acquires data in the spectral range of 3.5 µm to 15.4 µm (650 - 2875 cm^{-1}) using 2378 IR spectral channels. In addition four medium to broad band visible and near-IR channels are primarily used for diagnostics involving the presence and extent of clouds.

This paper describes the architecture of the AIRS Science Processing System (SPS) used to process AIRS data. The AIRS SPS is a distributed parallel system designed to process AIRS data in a heterogeneous network environment. Various AIRS scientific products are generated by a set of independent Product Generation Executables (PGEs). The AIRS SPS will be operating at the Goddard Space Flight System (GSFC) Distributed Archive Center (DAAC) and also at the AIRS Team Leader Science Computing Facility (TLSCF) at the Jet Propulsion Laboratory (JPL). This paper also describes the architecture of the TLSCF Data System (TDS) which provides an environment for ingest, processing and archiving of the AIRS products at the JPL TLSCF.

INTRODUCTION

AIRS is a key facility instrument that is scheduled for launch on the EOS AQUA platform (formerly known as PM), in late 2001, or early 2002. The AIRS instrument on Aqua is collocated with the EOS Advanced Microwave Sounding Unit (AMSU–A) and the Humidity Sounder for Brazil (HSB).

The AIRS instrument measures upwelling Infrared radiances at 2378 frequencies ranging from 3.5 µm to 15.4 µm. A limited number of visible/near-infrared (VIS/NIR) channels are also present to provide diagnostic support for the temperature and humidity sounding.

The AIRS scan geometry is illustrated in Figure 1. AIRS employs ±49.5° crosstrack scanning with a 1.1°instantaneous field of view to provide twice-daily coverage of the globe from a 705-Km altitude, on a 1:30 P.M. sun synchronous orbit.

The scan rate of AMSU is once every eight (8) seconds. During this scan period AMSU scans 30 footprints. Each AMSU footprint covers a 45-Km diameter area. The scan rate of AIRS and HSB are three times faster (once every 8/3 seconds) than AMSU. During this scan period both AIRS and HSB scan 90 footprints. Both AIRS and HSB have 15-Km diameter footprints. This geometry, scan, and footprint characteristics leads to the alignment of these instruments on board the Aqua platform as illustrated in Figure 1.

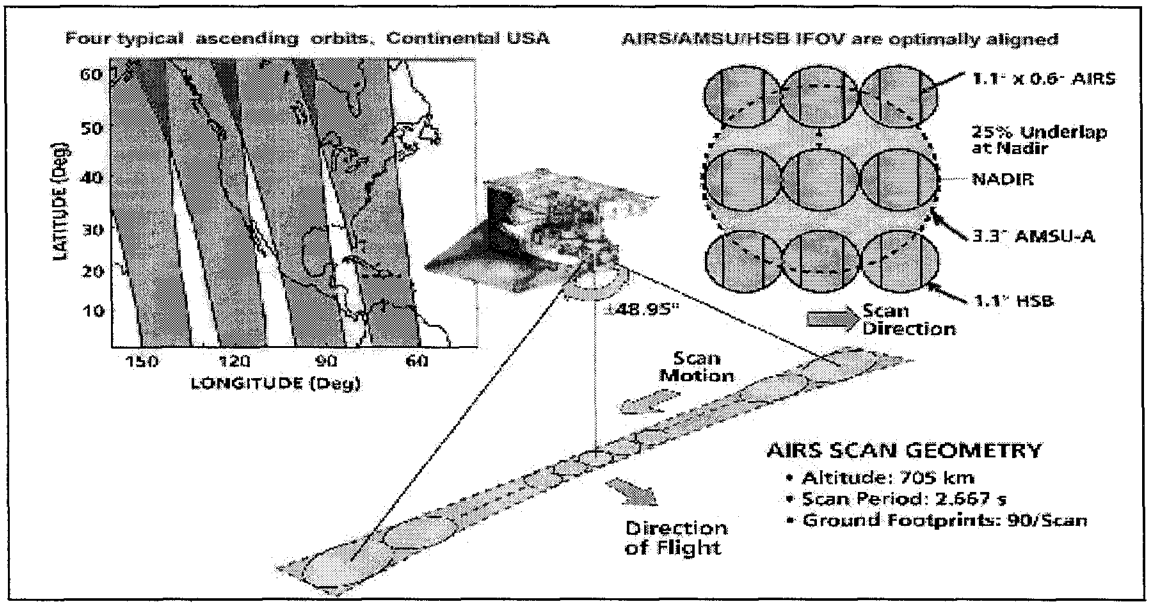

Figure 1. AIRS/AMSU-A/HSB Scan geometry

Aqua spacecraft data is processed to level 0 products by Earth Observing System (EOS) Data and Operations System (EDOS) at the GSFC DAAC. The result of this processing is Production Data Set (PDS) files of instrument science and engineering packets. EDOS processing removes redundancy and time orders the data packets. Each AIRS PDS file contains two hours of instrument data and is processed by the AIRS SPS at the GSFC DAAC, and also by the TDS at JPL, to produce AIRS science products.

PDS files are processed by the SPS and TDS to generate 6-minute granules of level 1A and level 1B data for each instrument. Level 1A and level 1B AMSU granules contain 45 scans of AMSU data, while level 1A and level 1B granules of AIRS and HSB data, because of their higher scan rates (3 times the AMSU rate), contain 135 scans of AIRS and HSB data respectively. Each 6-minute granule covers an area of approximately 45X30 by 45X45 square kilometers of the surface of the Earth. Level 2 processing uses 8 seconds of data (one scan of AMSU data and 3 scans of HSB and AIRS data) to retrieve the required geophysical parameters as specified in table 1.

Product	Unit	Absolute Accuracy
Temperature profile	Kelvin	1.0K
Humidity Profile	gm/kgm	20%
Total Perceptible Water	mm	5%
Fractional Cloud Cover	None	0.05
Cloud Top Height	Km	0.5km
Cloud Top Temperature	Kelvin	1.0K
Land Skin Surface Temperature	Kelvin	1.0K
Land Surface Spectral Emissivity	None	0.05
Ocean Skin Surface Temperature	Kelvin	0.5K
IR Spectral Cloud Emissivity (3-16μm)	None	0.05
Total Ozone Burden	Dobson	15%
Cloud-Cleared Brightness Tem.	Kelvin	1K

Table 1. AIRS Standard Science Products

AIRS Science Processing System (SPS) Architecture

The AIRS SPS is developed at JPL and delivered to GSFC DAAC for generation, archive, and distribution of AIRS standard products. After validation, users of the AIRS science products can access AIRS products through the EOS Data Gateway (EDG) interface.

Design of the AIRS SPS takes advantage of the definition of standard product size (6-minute granule) in order to develop a system that is easily distributable over a heterogeneous operational environment and is also capable of taking advantage of parallel processing of science data packets.

The AIRS major science data products are generated in three distinct phases;

Phase 1 - Level 1A processing is performed by three Product Generation Executables (PGEs) for AIRS, AMSU, and HSB instruments. Each PGE performs the following for each instrument:

- Synchronizes level 0 data packets
- Unpacks and reformats level 0 data packets
- Converts engineering parameters to engineering units
- Performs geolocation (i.e. calculates observed location, altitude, etc.)
- Merges and archives engineering and calibration data
- Puts science ground footprints from instruments into swath format and generates level 1A swath products.

Phase 2 - Level 1B processing is performed by four PGEs for AIRS (Infrared, and visible), AMSU, and HSB. Each level 1B PGE reads the level 1A science and calibration data from appropriate instrument and calculates the radiances. Corrections for polarization, non-linearity, and drift is handled in this phase.

Phase 3 - level 2 processing is performed by a single PGE. This PGE reads level 1B data from all instruments and performs a mathematical process called "*retrieval*" that calculates temperature and humidity profiles as functions of atmospheric pressure.

Processing at each of the three phases described above is performed by Product Generation Executables (PGEs). Each PGE is a standalone Unix process activated to perform a task. In the Unix environment, multiple instances of the same PGE (process) can be activated concurrently to perform the same processing on different granules of science data. The AIRS SPS architecture takes advantage of this capability as well as the design of the AIRS standard science granule size to provide parallel processing of its data. The architecture of the major AIRS SPS PGEs is provided in figure 2.

As level 0 data from EDOS becomes available, appropriate level 1A PGEs are activated by the production system based on production rules at the GSFC DAAC. Each level 1A PGE receives level 0 data from the GSFC DAAC archive and processes it into 6-minute granules of level 1A data. Each level 1A PGE can process data independent of other level 1A PGEs and therefore, level 0 data from all instruments can be processed concurrently. As shown in figure 2, the level 1A PGE for AIRS also splits the level 1A products into *infrared* and *visible* level 1A data.

Upon availability of level 1A products, level 1B PGEs are activated based on production rules that govern the processing of level 1B PGEs. Level 1B PGEs process each 6-minute granule of level 1A data and generate 6-minute granules of level 1B (calibrated) products. Each level 1B PGE performs its processing independently of other level 1B PGEs and therefore, multiple level 1A granules can be processed concurrently. In addition, as in the case of level 1A PGEs, multiple instances of each level 1B PGE for an instrument (i.e. Level 1B AIRS PGE) can be active at the same time, allowing the processing of different level 1A granules for the same instrument data concurrently. Level 2 processing requires availability of data from all instruments (AIRS, VIS, HSB, and AMSU) and therefore, it is activated when all appropriate granules of instrument data are available. For each 6-minute set of instrument granules (AIRS, VIS, HSB, and AMSU), the level 2 PGE retrieves geophysical parameters and produces standard level 2 products as specified in table 1. Multiple granule sets of level 1B products can be processed at the same time by activating multiple instances of level 2 PGE.

Each of the PGEs described above can be distributed over a networked heterogeneous CPUs. Although the operational environment at the GSFC DAAC is centered exclusively on the SGI computers, the JPL TDS operational environment as described below uses a variety of Unix and Linux based CPUs to perform the same function.

The AIRS SPS is developed at JPL and delivered to the GSFC DAAC for operations. The AIRS SPS is integrated to the scheduling (production planning system, PDPS) and the archive and catalog system at the GSFC DAAC by a JPL/GSFC integration team. The resulting level 2 products along with the intermediate level 1A and level 1B products are archived at the GSFC DAAC for access by the science community through the standard EDG interface.

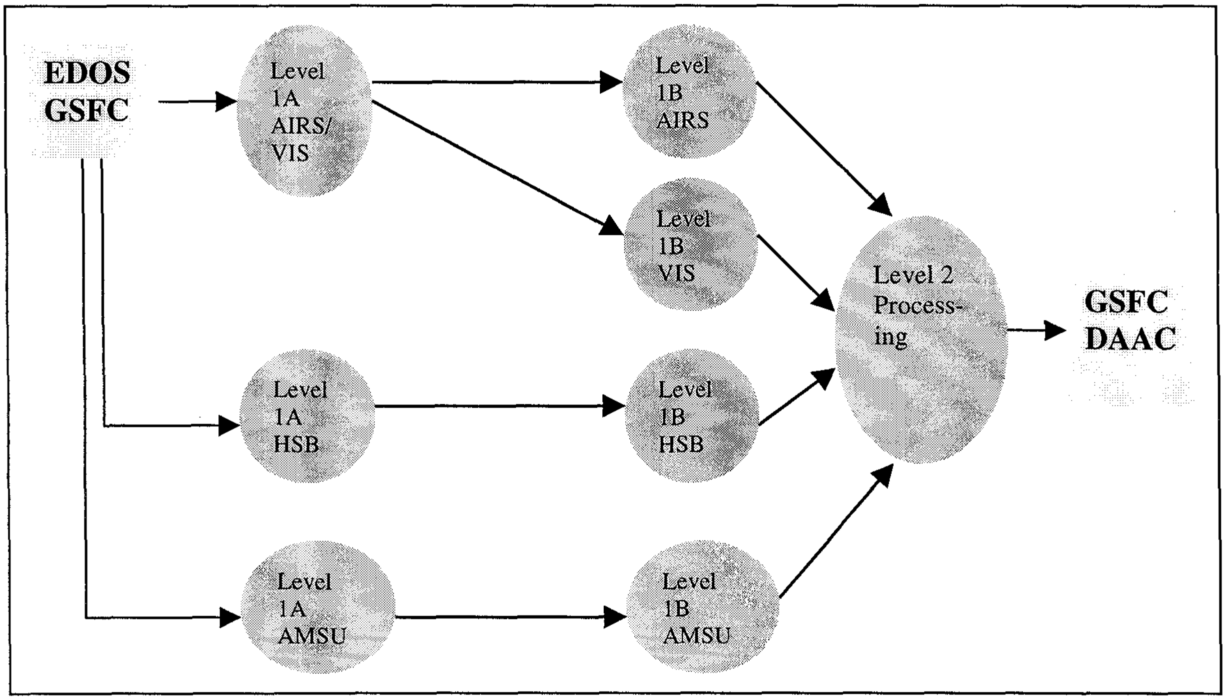

Figure 2.0 AIRS Science Processing System Architecture of major PGEs

TLSCF Data System (TDS) Architecture

The AIRS Team leader Science Computing Facility (TLSCF) Data System (TDS) provides an automated operation environment to produce AIRS science data products that are needed to support the AIRS post-launch science data calibration, verification, and validation activities scheduled to take place during the first twelve-month of mission operation. Figure 3 shows the architecture of the TDS.

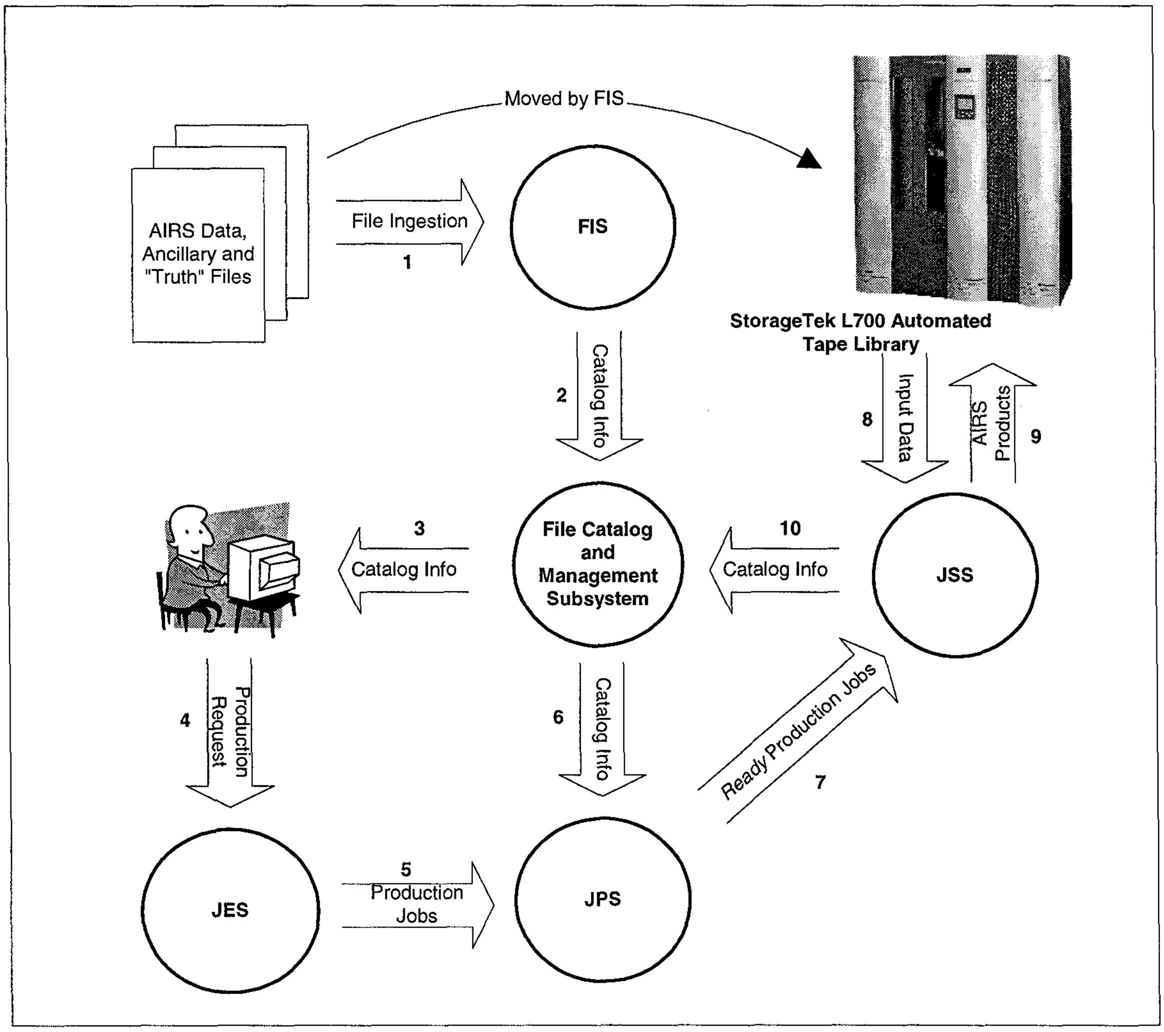

Figure 3. TDS Architecture

The TDS is a complete end-to-end software solution that starts with a File Ingestion Subsystem (FIS) at the front end to ingest AIRS Level 0 and other input data. The Job Planning Subsystem (JPS), that implements the production rules (i.e. prerequisites), gets into action upon receiving data production request created by the operator via the Job Entry Subsystem (JES). Once the production prerequisites are satisfied, the production job is marked *ready*. The Job Scheduling Subsystem (JSS) then takes over the *ready* production job to create and archive the data products. And, of course, the system is not complete without a centralized File Catalog and Management Subsystem. These elements are described below:

File Ingestion Subsystem (FIS)
Upon receiving email notification from the data provider, the File Ingestion Subsystem retrieves the external data product files that were deposited by the data provider in the external data file spooling area. For each data product file, it then creates a metadata file based on the predefined metadata policy for a specific product file type and data source.

Both the external data file and the TDS-created metadata file are submitted to the File Catalog and Management Subsystem. The File Catalog and Management Subsystem subsequently parses the metadata file to create catalog information and stores the data file in the TLSCF's StorageTek L700 automated tape library. FIS is mostly implemented in Perl programming language and include some C utility programs that utilize the time conversion library provided by the EOS Core System (ECS).

Job Entry Subsystem (JES)
The Job Entry Subsystem enables the SCF operator to enter a production request. It then decomposes the production request into one or more Product Generation Executable (PGE) jobs.

For each PGE job request, the Job Entry Subsystem creates a Job Description File (JDF) that is "submitted" to the requested jobs spooling area for processing by the Job Planning Subsystem.

The JDF is implemented using the Extensible Markup Language (XML). The Job Entry Subsystem's graphical user interface utilizes the existing web browser technology that is available on most of the modern computer desktop environment. All the Job Entry Subsystem's back end processing tasks take place in Java Servlets operating under the servlet-enable web server.

Job Planning Subsystem (JPS)
The Job Planning Subsystem periodically retrieves JDF for the newly requested jobs from the requested jobs spooling area. Once a requested job is successfully inserted into the internal pending queue, the Job Planning Subsystem removes the corresponding JDF from the spooling area.

The Job Planning Subsystem uses the predefined PGE production rules to create a dependency list regarding the required and optional input files for each input PGE job that it received. Periodically, the Job Planning Subsystem checks the latest available file catalog information to determine if the input dependencies for a given PGE job request have been satisfied. When all the dependencies for a given PGE job are satisfied, the Job Planning Subsystem creates a JDF for the *ready* job that contains information needed to initialize the PGE job execution. The Job Planning Subsystem then submits the *ready* job to the Job Scheduling Subsystem for execution.

The Job Planning Subsystem utilizes the object-oriented design methodology and is implemented in Java programming language. We hope that this will improve re-usability when there is a need to adapt the software to support other projects that may have varying production rules.

Job Scheduling Subsystem (JSS)
The Job Scheduling Subsystem periodically retrieves new *ready* JDF from the ready jobs spooling area. Once a ready PGE job is successfully inserted into the internal working queue, the Job Scheduling Subsystem deletes the *ready* JDF from the spooling area.

The Job Scheduling Subsystem allocates a work area (i.e., computer disk space) and creates a Processing Control File (PCF) for each *ready* PGE job with information provided in the corresponding JDF. The PCF is an ASCII text file that contains all the runtime parameters required to execute the applicable PGE. As the system resource becomes available, the Job Scheduling Subsystem executes the PGE job and archives the generated products in the StorageTek L700 automated tape library.

The core of the Job Scheduling Subsystem is implemented with Java programming language and utilizes the IBM XML parser for reading the *ready* JDF written in XML format. It also includes some Perl utility programs for setting up the scheduling job queues and creating the PCF.

File Catalog and Management Subsystem

The File Catalog and Management Subsystem is built on top of the Distributed Object Manager (DOM) developed at the Jet Propulsion Laboratory and has been in operation since 1997 to support various Deep Space Missions.

The Distributed Object Manager includes a homegrown on-line file catalog system. In the current configuration, the file catalog information is being kept on-line with 6 GByte of physical memory on a Sun Enterprise 3500 Server running four (4) 360 MHz CPUs. For the TDS application, DOM uses the StorageTek L700 automated tape library for file repository. It is estimated that the TDS will archive up to twenty-five (25) Tera-byte of data products during the first twelve months of operation.

The Distributed Object Manager provides command-line programs that allow user to:

- Store and catalog a new data file
- Search for an archived file
- Retrieve an archived file

In addition, DOM also provides application level interface libraries (in C++ and Java) for other software applications to store, search and retrieve files.

Summary

The AIRS SPS is a distributed, parallel Processing capability that is designed and developed at JPL. for processing the AIRS science data at the GSFC DAAC and at the JPL Science Computing Facility. The TDS capability uses modern tools and development techniques to provide the automated processing, archiving, and distribution of AIRS science products to the AIRS science team members. Use of object oriented technology provides a set of general and reusable modules that we are hoping to use for future projects with similar needs.

Acknowledgement

The research described in this paper was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement by the United Stated Government or the Jet Propulsion Laboratory, California Institute of Technology.