MODIS
SCIENCE DATA SUPPORT TEAM
PRESENTATION

May 15, 1992

AGENDA

ACTION ITEMS:

04/24/92 [Lloyd Carpenter] Prepare the Team Leader's Software and Data Management Plan for review. (An updated draft version was distributed at the May 8th meeting.) STATUS: Open. Due Date: May 10, 1992.

04/24/92 [Lloyd Carpenter] Prepare the Team Leader's Science Computing Facility Plan for review. (The latest version version has been discussed with the Project. Further revisions are being made.) STATUS: Open. Due Date: May 10, 1992.

04/24/92 [Tom Goff] Develop a detailed schedule through to the delivery of Version 1 to the DAAC for Level-1A and -1B software design and development, identification of risk areas in Level-1A and -1B design, and prototyping of risks. STATUS: Open. Due Date: 05/22/92

04/24/92 [J. J. Pan] Develop a detailed schedule for the Level-2 Processing Shell design and development, identification of risk areas in the Level-2 Processing Shell design and development, and prototyping of risks, through to the delivery of Version 1 to the DAAC. STATUS: Open. Due Date: 05/22/92

04/24/92 [J. J. Pan] Develop a detailed schedule for a typical algorithm integration into the Level-2 processing shell. STATUS: Open. Due Date: 06/05/92

04/24/92 [Lloyd Carpenter & Team] Develop a staffing plan for the accomplishment of the tasks shown on the schedule. STATUS: Open. Due Date: 06/12/92

MODIS Airborne Simulator status (Liam Gumley)

Progress up to 14 May 1992

(1) MAS Field Data System

The components of the MAS Field Data System (FDS) are being assembled and tested at RDC. Components tested so far include

- 80486 33Mhz PC compatible, 200MB hard disk, SuperVGA card, 9600 bps modem
- 15" SuperVGA flat screen NEC monitor

All of the FDS software has been transferred to and compiled on the new PC. As of 14 May, all Level-1 processing code appears to be working correctly, including those programs which utilize the NetCDF library. This means that a functionally equivalent set of MAS Level-1 software now executes on the PC, VAX and Iris.

A menu driven shell for the processing programs has been developed (in DOS batch language) to make the user interface a little more friendly.

Current plans call for the FDS to be shipped to Ames for the MAS test flight on 22 May. Shipping to the Azores will take place around 26 May. The FDS will be used at Ames to test and characterize the MAS performance, as the usual Quick View System (QVS) is not available. Chris Moeller from Wisconsin will bring some of his MCIDAS code to Ames for integration into the FDS.

The Exabyte tape drive supplier was selected earlier this week, and the unit should be arriving at GSFC on 15 May or 18 May. Testing of this unit will necessarily be rapid since it must be shipped to Ames for the test flight on 22 May.

MODIS SDST
Processing Master Plan
(Very Early Draft)
Thomas E. Goff
14 May, 1992

Purpose and Background.

This plan is the master outline for the software to be developed by the MODIS SDST in support of the MODIS-N instrument to be launched in 1998 as part of the EOS project. It follows the IEEE Standard for Software Quality Assurance Plans (SQAP) as outlined in the ANSI/IEEE Std 730-1984 document with items added from the FIPS-38 documented. This plan is limited to the development of the MODIS software from the Level-0 product, through to the Level-2 product, and encompasses the intermediate software processing Levels. Levels-3 and above are currently expected to be developed by other authorities. The specific software products to be developed are:

1). The Level-1A Data Product Generator. This program reformats (reversibly) the telemetered packetized MODIS instrument data into a form that represents the original raw data as generated at the MODIS instrument. This raw data will be in the form of a scan cube with ground location coordinates, ancillary data, engineering data, and auxiliary data appended.

2). The Level-1B Data Product Generator. This program accepts the MODIS Level-1A data and generates a Level-1B data product consisting of a MODIS scan cube that has been calibrated to at-satellite radiances. A second set of ground coordinates that incorporates an Earth DTM correction will be appended.

3). The Level-2 Data Product Generator. This suite of programs will generate Science Products at the scan cube geometry coordinates and Team Member (TM) specified coarser (subsampled) resolutions, still based on the instrument scan geometry. The output data will be atmospherically corrected to at-ground radiances on a per product basis.

Reference Documents. (list of references)

Management.

Organizational Environment. (Org Chart -> delegated responsibilities: MCST, SDST, MAST, contractors, etc)

Tasks. (list of tasks to be performed)

Responsibilities. (by task, who is responsible:)

Organizational Elements. (Supervisory Functions)

Manpower Staffing Levels. (Personnel)

Hardware Equipment. (Computers & who administrates them.)

Documentation.

Purpose. (Identify the documents, especially for IV&V. How are documents checked for accuracy - review, audit, adequacy.)

Minimum Documentation Requirements.

Software Requirements Specification. (Functions, performance, design constraints, attributes - objectively verifiable; anticipitated operational changes.)

Data Format, Volume, Rate, and Timeliness. (required equipment.)

Accuracy. (Mathematical, Logical, Legal, Transmission.)

Flexibility.

Operating Environment. (Portability, tool kit contents, Languages.)

Software Operational Modes. (Normal processing and failure backup modes.)

Data Retention and Retrieval Proceedures. (shared memory/DADS.)

Reporting Methods. (Logging and Error Alerts.)

System Failure consequences and Recovery Proceedures. (data reversability, abnormal termination.)

Planned Changes and Improvements. (Best guess at the future.)

External Interfaces. (By breakdown of external entity.)

Requirements Traceability Matrix. (Forwards from all reference documents.)

Software Design Description.

Assumptions. (All items that outside parties need to be aware of: deficiencies, limitations.)

Features. (items not directly traceable to requirements.)

Internal Interfaces. (Intercommunication between processing elements.)

Design Traceability. (Backwards to requirements documents.)

Software Verification and Validation Plan. (Methods - automatic QA programs, hand analysis, test data, battlemaps, adherence to standards, adherence to requirements.)

Software Verification and Validation Report. (Results of SVVP execution.)

User Documentation. (Installation guides.)

Other.

Standards, Practices, and Conventions.

Purpose. (Identify standards.)

Content.

Source Code Standards.

Embedded Documentation Standards. (auto documentation criteria, man page generation, key word database, auto revision codes.)

Code Language Standards. (High level - not enforced style.)

Code Commenting Standards. (Location of comments, amount.)

Software Builds. (Make files, JCL)

Data Dictionary. (Data flows, processors, data structure definitions.)

Review and Audit. (for beta, V1, V2 releases.)

Purpose. (Technical and Managerial reviews.)

Minimum Requirements. (to be correlated with milestones.)

Software Requirements Review.

Preliminary Design Review.

Critical Design Review. (for each build.)

Software Validation and Verification Review. (adequacy and completeness.)

Functional Audit.

Physical Audit. (internal consistency.)

In-Process Audits. (periodic audits. - How often?)

Managerial Reviews.

Software Configuration Management. (automated)

Problem Reporting and Corrective Action. (the process!)

Tools, Techniques, and Methodologies. (CASE, software tools, supplied tool kits, science and engineering data simulators.)

Code Control. (Who has check-in and -out capabilities, build authority.)

Media Control. (Redundant backup w/ locations.)

Records Collection, Maintenance and Retention. (Librarian office.)

DRAFT

PRODUCT GENERATION SYSTEM TOOLKIT STUDY REPORT

DRAFT

1 INTRODUCTION

This is the first status report on a study that will define the functionality and usage of the Earth Observing System (EOS) Data and Information System (EOSDIS) Product Generation System (PGS) Toolkit. The study of the PGS Toolkit is part of an overall EOS Investigator-to-EOSDIS Interface Study to investigate and define the relationships between the EOS investigators, their software, software development efforts and systems, and the EOSDIS. The goal of the PGS Toolkit study is to define the science product software interface to the Toolkit as completely as possible. Early definition allows EOS science programmers and managers to make design decisions early in the software development life cycle. This paper is the first version of an evolving and expanding report that is based on the needs of the EOS science software developers.

The audience for this study is members of the EOS instrument and interdisciplinary investigator teams associated with the development of software and the staffs of the Distributed Active Archive Centers (DAACs) that will host PGSs. In its final form this study will be provided to the EOSDIS Core System (ECS) contractor to provide a basis for direction in the realization of the Toolkit in the PGSs and the Science Computing Facilities (SCFs).

1.1 EOS Investigator-to-EOSDIS Interface Study

The EOS Investigator-to-EOSDIS Interface Study is intended to define the interfaces required by science investigators to successfully integrate science software into EOSDIS. The goal of the study is to define the

- EOSDIS services required by investigators;
- Tools to emulate EOSDIS in investigator SCF environments;
- Common libraries, languages, utilities, and Commercial-Off-the-Shelf (COTS) Software required;
- Science investigator approaches to developing portable, maintainable software;
- Approaches to data management; and

(ADC), and Other Data Centers (ODC). The NASA Communications network (NASCOM), EOS Communications network (ECOM), Tracking and Data Relay Satellite (TDRS), Deep Space Network (DSN), Ground Network (GN), and Wallops Tracking Station (WTS) are also shown. Shaded areas in the IST, SCF, and FST boxes in Figure 1 represent ECS software toolkits that will reside on compatible external hardware.

The first of the three ECS segments is the Flight Operations Segment (FOS). The FOS manages and controls the EOS spacecraft and instruments. The FOS elements include the EOS Operations Center (EOC), Instrument Control Centers (ICCs), and Instrument Support Terminals (ISTs). Several ICCs constitute an Instrument Control Facility (ICF). ICCs will be functionally similar and will interface with the EOC but will be used to manage different instruments. The EOC provides mission planning and scheduling and the control and monitoring of mission operations of the EOS spacecraft and instruments. ICCs schedule, command, and operate the science instruments and monitor instrument performance. ISTs are defined to be software to connect science team representatives to an ICC in support of remote instrument control and monitoring.

DRAFT

The Communications and System Management Segment (CSMS) provides overall ECS management and operations of the ground system resources, provides facilities and communications and networking services for the science data communications network, and manages the interfaces to NASA networks and other communications networks. The CSMS elements include the System Management Center (SMC) and the EOSDIS Science Network (ESN). The SMC provides system management services for EOSDIS ground system resources. ESN provides an internal network for ECS communications, a network interface to the science user network, network services at the application layer and a network management and help facility.

The Science Data Processing Segment (SDPS) provides a set of processing and distribution elements for science data and a data information system for the entire EOSDIS. The SDPS elements include Distributed Active Archive Centers (DAACs) consisting of ECS and institutional facilities including Product Generation System (PGS), Data Archive and Distribution System (DADS), and Information Management System (IMS). PGS and DADS facilities process data from the EOS instruments to standard Level 1 through 4 data products; provide short- and long-term storage for EOS, non-EOS earth science, and other related data, software, and results; and distribute the data to EOSDIS users. IMS provides a distributed data and information management service for the ECS including a catalog system in support of user data selection and ordering. IMS also provides Field Support Terminals (FSTs). These are portable terminals that can be employed at field investigation sites to provide access to one or more DAACs.

SDPS toolkits include those for IMS, DADS, and PGS. The configuration of toolkits for a particular installation is tailored to the user's site-specific requirements and responsibilities for the EOS mission. The PGS toolkit is the subject of this study report and is discussed in the following sections. Figure 2 illustrates the relationships between the PGS, IMS, and DADS Toolkits on the PGSs and the SCFs.

other functions and interfaces, for purposes of the PGS Toolkit study, only the interfaces available to the science data production software will be considered.

The ECS contractor will develop the PGS and SCF implementations of the Toolkit. A SCF will reside with each EOS investigator and be the host computer for science software development, science data validation, and processing of special data products. The SCFs will be procured by the science investigators based on the investigators' requirements and preferences.

1.3.1 Purpose of the Toolkit

The EOS Science Software Toolkit will define the software environment in which the science data production algorithms will be developed and reside, both at the investigators' SCFs and the EOSDIS PGSs. Science software developers will be expected to use only the Toolkit as a standard library in designing and writing software. From the perspective of the science software implementer, the Toolkit will replace parts of other standard libraries, such as the ANSI C library or the IMSL math tools, for most uses. The Toolkit will serve to isolate the science software from hardware, system, and language interface library dependencies.

The Toolkit will be a library in the programmer's sense of the word, and will be implemented to support science software programmers and programs on the SCFs and PGSs.

The Toolkit will be production oriented. It will be designed for the purpose of developing software to generate standard science products. Therefore, it will not have an extensive set of routines for presentation graphics since, in a production system, it is unlikely that such graphics would be viewed by scientists or users.

The Toolkit will
- Be a library of callable functions and subroutines ;
- Be production oriented;
- Serve to isolate the science software from system dependencies;
- Include routines for input/output;
- Include routines for math and statistics support;

to implement their software. All program input and output will be routed through these tools to ensure correct execution and implementation in the operational environment.

The life span of the programs generated by EOS scientists may be decades. In that time frame, and even before EOSDIS is actually implemented, we can be certain that data processing technology will make major and sometimes unexpected advances. It is hoped, that whatever the advancements, the interfaces to the Toolkit will remain a constant in order to continue to use decades-old code with minimal software maintenance. The Toolkit itself will be designed to be portable and will be upgraded to make use of advancements in technology such as high speed mass storage systems, high performance processing hardware, and improved communications technology. By using the Toolkit science developers will create software that may be able to benefit from these advances.

Another goal of the Toolkit is to separate the maintenance of the science algorithm and code from the maintenance of the support software. Maintenance of the science code is likely to be a difficult problem over the EOS time frame. The higher the percentage of the code that can be placed into the institutionally maintained Toolkit facilities, the less imposition and reliance on the science maintenance resources.

1.3.1.2 Product Standardization

The Toolkit will provide investigator teams access to common procedures, data models, reference frame models, physical parameters, platform position information, and data structures. Usage of these parameters will aid in the generation of standard products which will improve interoperability, simplify the sharing of data sets, and allow more accurate and concise interdisciplinary correlation of data.

1.3.1.3 Science Resources

With hundreds of investigators developing software intended to be used in production of standard EOS science products, it is almost certain that many will be developing or obtaining software with similar or same functionality.

development of prototype tools by science groups and the EOS Project. Other similar projects, such as the Upper Atmosphere Research System (UARS) Project, are also sources for input to this study. A final version of this study report, based on this iterative feedback, will be delivered to the ECS contractor once the contractor is selected and begins work.

The ECS contractor will provide an interface tools specification, based, in part, on this study. At that time, full science software development can begin using the provided specification as the basis for interfacing with the operational environment.

During the period from now until the ECS contractor provides the specification, science software developers should, as much as possible, design and prototype science software in a modular fashion based on the understanding of the Toolkit interfaces that this study report provides. It is important to realize, though, that this first version of the report is tentative and subject to change based on science community feedback.

2.1 PGS Toolkit Study Activities

The study is being undertaken through discussions, interviews, working group meetings, and correspondence with the EOS investigators, DAACs, and other science software experts and developers.

EOS Ground System and Operations Project (GSOP) personnel have met with the instrument teams listed in Table 1 and have presented an initial overview of the Toolkit. Information culled from the discussions and concerns presented at these meetings as well as the EOS Mission Operations Working Group (EMOWG) and EOSDIS Version 0 System Engineering Meetings over the last year are considered part of the Toolkit study and have been incorporated in this status report.

Another source of information used in this report was the ECS RFP. The RFP delineates, at a high level, aspects of a PGS Toolkit the ECS contractor will provide.

will be distributed and again feedback from the investigator community will be requested. A final version of this specification is planned for delivery in the last quarter of 1993. EOS science software developers will use this specification as the basis for designing production code.

GSOP plans for a beta-test version of the PGS Toolkit library to be made available to the SCFs in the first quarter of 1995. On delivery of this library, science software developers can be implementing their production code on their SCFs in a PGS emulation mode. A beta-test DAAC is expected to be ready in the third quarter of 1995. At that time, science software developers will have the opportunity to test software portability in a preliminary PGS environment. EOS AM flight instrument investigators will be encouraged to begin testing early versions of their software in the beta-test DAAC.

2.3 Science Community Feedback

It is hoped that the Toolkit study and this status report will motivate feedback from the investigators. There is strong interest in receiving specific proposals and requirements for specific tools, definitions of science data processing conventions and standards, calling arguments, numeric methodology, and operations concepts from the EOS science data processing community. It is desired that interested scientists will take a proactive role in the Toolkit design. Section 3 and Appendix A will be expanded to incorporate specific information pertaining to tools as it becomes definitive during this study.

Figure 3 portrays the importance of investigator participation in the Toolkit Study. Limited science software developer participation will result in an incomplete Toolkit specification and likely portability problems. By taking an active role in developing this interfaee specification, developers can help to minimize the problems in porting the operational code from the SCF to the PGS and from a PGS machine to a replacement PGS machine.

3.2 Language Considerations

The ANSI C library provides access to system capabilities. The C Library, like the Toolkit, provides a common interface to varying hardware and system platforms. In some cases the science software programmer will not directly access standard ANSI C Library functions. Instead, the Toolkit will act as an interface to the C functions. For example, C file creation and open functions, such as fopen, will be mapped to PGS Toolkit functions with options to implement high speed file access, or temporary or production stream file access. A related situation exists when using FORTRAN, although FORTRAN's I/O functions are intrinsic in the language and are not part of an external library. In using FORTRAN, science software programmers will avoid certain functions, like the OPEN statement, and will instead call Toolkit functions. The Toolkit "open" functions will provide information to the PGS Scheduler pertaining to the type of file, the ID of the program or production stream, and the criteria for file closing. Standard ANSI C fopen and FORTRAN OPEN statements will not be used by programmers.

This will be true for much of the C library such as memory allocation functions, I/O functions, and bit manipulation functions. The C library functions or FORTRAN intrinsic functions that are not replaced by Toolkit calls are those that are system independent and, therefore, preserve the Toolkit's purpose to isolate the science software from system dependencies. Figure 4 illustrates the relationship between the Toolkit and the ANSI C Library and FORTRAN.

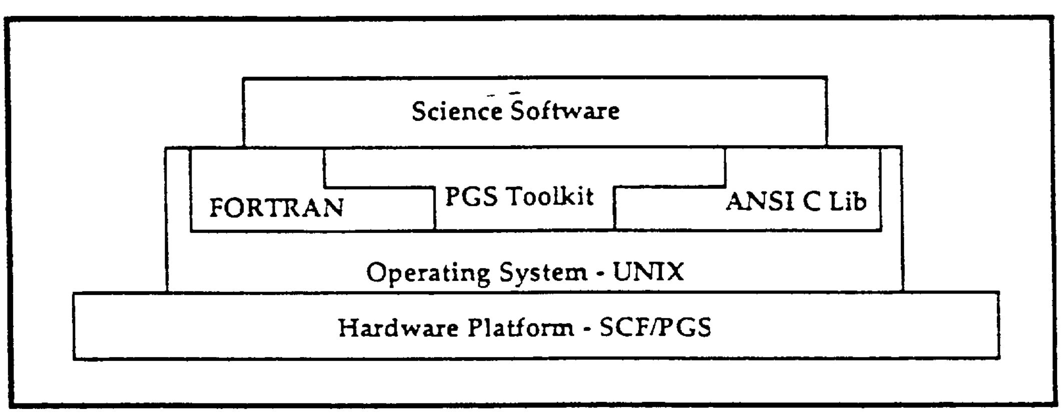

Figure 4. PGS Toolkit, ANSI C Library, and FORTRAN

stream which will determine the sequence in which runs are executed and what resources may be allotted to a production stream.

The PGS Scheduler will have to handle partial execution and be able to restart production streams. Science programs and production streams will need to be designed with breakpoints or restart points and flags so that if interruptions occur, for example due to non-availability of data, previous processing will not have been wasted.

Significant reprocessing of data is being planned in sizing the PGS. As science understanding changes or bugs are detected, improvements and fixes to production software will be made. The Scheduler will have to manage the reprocessing decisions along with on-going processing requirements.

A production stream may require data from remote DAACs in order to start or continue processing. Two options, not necessarily exclusive, may be available for acquiring these data: staging these data in preparation for the processing, or accessing the data across network links during processing. In the first case the PGS Scheduler will have to come into play to query for the availability of data before starting production streams. For the second case the science software may query for data availability and then inform the Scheduler of the need to request the data and complete processing, or suspend and restart processing when data become available.

3.3.1.1 Data Availability Tools

Although the specifics of the scheduling system are unknown, it is obvious that certain tools will be required. Tools to query for availability and validity of data will be required by science production runs. Toolkit interfaces to the Scheduler at the start of a production run may provide metadata about the dimensions of input data sets which would be used to refine processing methods.

Higher level data products may require partial processing of data before science algorithms can define what additional data may be needed to complete products. In this case mid-production run data availability queries

3.3.2 I/O Tools

There are three broad categories of I/O Tools described here: temporary file I/O, production stream I/O, and product and auxiliary data access. Temporary file I/O refers to using files that exist for the duration of a single program (i.e., process or execution). Production stream I/O refers to using files that exist for the duration of a complete flow of programs and program control. Product and Auxiliary data access refers to interfacing with permanent files, data, and processes such as science data products and archives, metadata and catalog information, and orbit, attitude, and digital terrain elevation data.

Tools to access temporary and production stream files will allow various methods of access. Direct (or random access) and sequential (or serial) are two common types of access that will supported by the toolkit. Another option, high speed, has been suggested as a method of creating and accessing temporary data files. High speed file access might be implemented in a variety of ways including RAM cache buffers for mass storage media, RAM based mass storage emulation similar to the FORTRAN internal file type, or other new technology. High speed access would allow scientists to vastly improve performance of algorithms that might normally have I/O bottlenecks or be otherwise I/O intensive.

3.3.2.1 Temporary File I/O Tools

Temporary file I/O refers to files that exist for the duration of a single process or program execution. These files can be considered to be equivalent to FORTRAN's SCRATCH file type. The purposes of these files include providing temporary storage for high volume data structures and staging data in preparation of creating the final data product.

The content and organization of a temporary file is left entirely to the science software designer. However, tools to create, read, write, and otherwise manipulate standard data structures will be available and compatible with data structure tools for production stream files and standard data products. Access to a temporary file will be direct or sequential access and can be high speed or not depending on the priority of the production stream and the requirements of the science software.

Production stream files may be made accessible by foreign production streams allowing a method for multiple programs and investigator teams to share access to current or related science data.

3.3.2.2.1 Production Stream File Housekeeping Tools

These tools are similar to standard file interface functions for creating, opening, closing, and defining the parameters of files. For production stream files though, the tools will only allow the files to exist as long as the programmer defined criteria are met. Tools that would allow the definition of criteria for file deletions, foreign program access, and passing of file ownership from one stream to another would be included in this group.

Access methods for a production stream file (direct or sequential, high speed, or normal) will be determined by the parameters passed to these tools.

In FORTRAN, examples of similar functions include INQUIRE, OPEN, REWIND, BACKSPACE, CLOSE, and ENDFILE. In the standard ANSI C library some of the related functions include feof, fclose, fflush, ferror, fopen, and fseek.

3.3.2.2.2 Production Stream File Access Tools

As with temporary file tools, production stream file access tools will include functions to write, read, or append data to a production stream file. Examples of similar FORTRAN functions include READ, WRITE, and PRINT. In C similar functions include fgets, fgetc, fprintf, fread, and fputs.

Common function tools to read and write standard science and administrative data structures will be available for production stream file access. Examples of these tools might include write_image, read_image, read_palette, write_registered_image, read_browse_header.

3.3.2.3 Product and Auxiliary Data Access Tools

Product and Auxiliary data access refers to using the permanent files and data that exist outside of the execution of an individual production stream, such as science data products and archives, metadata and catalog information, and

3.3.2.3.3 Metadata and IMS Access Tools

This tool set will include tools to pass metadata and catalog information in the appropriate formats and data structures to the IMS on completion of data products. Tools for querying the IMS or DADS for the availability and content of data products may also be provided.

3.3.2.3.4 Telemetry & Command Access Tools

Relatively small adjustments to the spacecraft or other instruments during observation periods may affect science data. These tools will allow access to instrument and spacecraft command history files to provide investigators data to adjust processing parameters based on the estimated impacts of such actions.

3.3.2.3.5 Spacecraft Ephemeris & Attitude Data Access Tools

This tool set will provide spacecraft orbit and attitude data. Options will allow programmers to request data in various reference frames such as earth centered, earth fixed, or inertial. Tools will allow the user to request and report availability of predicted or precise ephemeris and attitude data. These tools will provide interpolation to given times or provide spans of time tagged ephemeris.

3.3.2.3.6 Lunar/Solar/Major Body Position Access Tools

This tool set will provide access to algorithms that compute the positions of major celestial bodies such as the Moon and Sun. Options will allow programmers to request positioning data in various reference frames such as spacecraft centered, earth centered or inertial. This information will provide programmers data to compute solar angle and determine if light from celestial bodies may be impacting instrument observations.

3.3.2.3.7 Instrument Calibration Data Access Tools

These tools provide access to calibration parameters that are derived by the instrument developers or measured during instrument testing and

3.3.2.3.12 Status Tools

The status tools will be used to report status of production streams and product availability to the PGS system managers and investigator teams.

3.3.2.3.13 Data Validation Graphic Output Tools

Visual inspection of data products by trained personnel is still one of the most accurate methods for recognizing and reporting data problems. However, the volume of data expected to be processed by EOSDIS precludes visual inspection of all data products. Science production software should be designed to automate data validation as much as possible. This tool set is included in order to provide production software the ability to present validation graphics to PGS system managers or investigators for data that has failed an automated inspection.

3.3.2.3.14 Geographic Standards Tools

This set of tools will provide science software developer teams a standard data base and set of methodologies for integrating geographic information into their products. It is hoped that the availability and use of these tools and database access functions will provide commonality and interoperability between the data products of various EOS disciplines and simplify many aspects of intra- and interdisciplinary data correlation.

3.3.2.3.14.1 Digital Terrain Elevation Data Access Tools

These tools will provide access to a database of a worldwide grid surface elevation data. Functions to interpolate for any given geographic coordinate, estimate elevation accuracy and precision, generate contour data, and produce mean sea-level, geodetic, or ellipsoid elevations are possible candidates for this tool group.

3.3.2.3.14.2 Geographic Information Access Tools

Geographic Information tools will provide access to various data bases including geo-political features, geoid height, geo-magnetics data, gravity data structures to incorporate into data products will be included in the Toolkit. Such tools might include pattern search, pattern match, image overlay, palette definition, image statistics, pixel selection, area computation, and area selection functions. Some uses of these tools might include advanced automated data validation by pattern matching known geographic surface or related atmospheric features from another instrument's data sets, generating appropriate color palettes for a browse product, and using image statistics to measure the surface area of a particular surface or data feature.

3.3.3.3 High Performance Processing Tools

The Toolkit will be designed to be optimized for specialized PGS architectures. To as large a degree as possible, the Toolkit will be implemented using the vector processing or parallel processing capability the PGSs have to offer. Some investigators in special circumstances, however, may require direct access to architecture specific functions to achieve the performance levels necessary to process data in a timely manner.

Currently, these specialized processors come from a variety of vendors and do not have standard interfaces. However, in order to develop science software which makes use of such architectures on an SCF, special tools that emulate the PGS architecture-specific functions may be required.

3.3.4 Data Structure Manipulation Tools

This tool set will include functions to efficiently manipulate common data structures. Some examples include extracting subsets of data structures such as arrays, searching memory or data structures for specific data patterns, and sorting of data for product generation or for improving processing efficiency.

3.3.5 Memory Management Tools

Many processors have differing implementations for byte and word ordering, memory page and bank edges, alignments, and sizes that have to be taken into account during a porting process. The memory management tool set will be designed to normalize these differences in the SCF and PGS environments.

3.3.6.2 DAAC Specific Tools

The architecture and capabilities of the DAACs are not expected to be uniform. Each DAAC will likely have some value added features which will be specific to the needs of the science community being served by that DAAC. If required, the PGS Toolkit may incorporate tools which will provide access to the features which are not common to all DAACs.

3.3.6.3 Debugging Tools

This tool group would consist of a set of debugging support functions that would provide science programmers a common set of tools during code development and integration to the PGS. Examples of such tools might include functions to present variables and source code line pointers during execution; set debugging flags; and set breakpoints with variable, register, or memory dumps.

DRAFT

APPENDIX A TOOL TABLES

Appendix A will be used to present preliminary proposals for calling syntax, and input and output argument requirements and will be expanded to incorporate specific information pertaining to tools as it becomes definitive during this study. The following are the current proposed tool categories.

- Production Control & Scheduling Tools
  - Data Availability Tools
  - Initialization Tools
  - Termination Tools
- I/O Tools
  - Temporary File I/O Tools
    - Temporary File Housekeeping Tools
    - Temporary File Access Tools
  - Production Stream File I/O Tools
    - Production Stream File Housekeeping Tools
    - Production Stream File Access Tools
  - Product and Auxiliary Data Access Tools
    - Level 0 Data Access Tools
    - Level 1 - 4 Product Access Tools
    - Metadata and IMS Access Tools
    - Telemetry & Command Access Tools
    - Spacecraft Ephemeris & Attitude Data Access Tools
    - Lunar/Solar/Major Body Position Access Tools
    - Instrument Calibration Data Access Tools
    - Time and Date Access Tools
    - Browse Output Tools
    - Quicklook Product Tools
    - Error Output Tools
    - Status Tools
    - Data Validation Graphic Output Tools
    - Geographic Standards Tools
      - Digital Terrain Elevation Data Access Tools
      - Geographic Information Access Tools
      - Coordinate Transformation Tools
    - Tools to Access Common Physical Constants

DRAFT

APPENDIX B TOOLKIT USAGE SCENARIOS

In later versions of this paper Appendix B will be used to present science software implementation scenarios as examples of how tools or tool sets might be used.