MODIS
SCIENCE DATA SUPPORT TEAM
PRESENTATION

November 22, 1991

AGENDA

1. Action Items
2. MODIS Airborne Simulator (MAS)
3. MODIS SDST Project Plan Outline
4. SDST-Generated Source Code
5. MODIS Team Leader Computing Facility Plan

ACTION ITEMS:

08/30/91 [Lloyd Carpenter and Team]: Draft a schedule of work for the next 12 months. Include primary events and milestones, documents to be produced, software development, MAS support, etc. (Awaiting review) STATUS: Open. Due date 09/27/91.

10/04/91 [Phil Ardanuy and Team]: Prepare questions for the project to characterize the spacecraft position and attitude knowledge and the MODIS pointing knowledge in a way that will facilitate the evaluation of methods such as image registration to meet the science team requirements for earth location. (The letter to the project was prepared, 10/28/91.) STATUS: Open. Due date 10/18/91.

11/08/91 [Tom Goff]: Meet with Angel Li (currently at GSFC) again before he leaves GSFC for more information regarding the DSP. Status: Open. Due date 12/06/91.

Progress on MAS Level-1B processing system development

Progress up to 21 November 1991

(1) MAS FIRE deployment

I was in Houston from November 12th to November 15th in support of the MAS deployment for FIRE (First ISSCP Regional Experiment Phase 2).

The NASA ER-2 with the MAS onboard arrived at Ellington Field, Houston on November 12th. The personnel involved in the MAS project were Pat Grant (Ames), Ken Brown, Mike King, Tom Arnold (GSFC), Chris Moeller (Wisconsin) and myself. Several other instrument teams were also at the deployment site.

To date, three MAS science flights over the FIRE field site in Coffeyville Kansas have occurred. These were on the 14th, 18th and 21st of November. Data from the flights on the 12th, 14th and 18th have been examined in Houston with the Ames Quick View System (QVS). The QVS is a PC system which reads the MAS "Level-0" aircraft tapes and converts the data to PC-MCIDAS format for analysis. A brief summary follows.

Date : 12 November 1991
Flight : Ferry flight from Ames to Houston
Data : All channels recorded, channels 2-8 @ 8 bits, channels 9-12 @ 10 bits
Comments
All channels showed good image data. Some coherent noise was apparent in channels 7, 8, 9. This was believed to be due to electronic pickup from the MAS pod heater. Channels 10, 11 and 12 all had low noise levels.

Date : 14 November 1991
Flight : Science flight over Coffeyville
Data : All channels recorded, channels 2-8 @ 8 bits, channels 9-12 @ 10 bits
Comments
All channels showed good image data. The coherent noise in channels 7, 8 and 9 was significantly smaller than in the previous flight. This was as a result of shielding improvements to the instrument supervised by Pat Grant and Ken Brown. The flight track over Coffeyville was a racetrack pattern.

Date : 18 November 1991 (information from Ken Brown)
Flight : Science flight over Coffeyville
Data : All channels recorded, channels 2-8 @ 8 bits, channels 9-12 @ 10 bits
Comments
All channels showed good image data. The coherent noise in channels 7, 8 and 9 was no longer visible in the QVS imagery. Examination of the black body counts in channels 7-12 showed variations of less than $\pm$ 3 counts. The black body temperatures were likewise stable at cruising altitude. Calibration of the visible/near-IR channels (2-6) was done prior to this flight using the GSFC integrating sphere. Post flight calibration remains a problem since the MAS is typically covered with condensed moisture when it descends from altitude, and remains so for a significant time. Currently only pre-flight visible/near-IR calibration is planned for the rest of FIRE. The flight track over Coffeyville was a racetrack pattern.

Date : 21 November 1991 (information from Ken Brown)
Flight : Science flight over Coffeyville
Data : All channels recorded, channels 2-8 @ 8 bits, channels 9-12 @ 10 bits
Comments
The ER-2 was still in the air when this information was compiled.
Ken Brown stated that the instrument appeared to be running as expected, and all present were pleased with it's performance. It is still planned to fly about 10 flights in total during the FIRE field phase, which ends on December 7th.

Ames Research Center plans to turn the MAS data around from Houston in 4-5 days. They are almost finished with the data from the 14th and will be sending it tomorrow (22nd).

(2) MAS software development

This week effort was devoted to porting the MAS processing code to the Silicon Graphics Iris in LTPCF. This was done based on experience running comparable processing tasks on the VAX and Iris, which showed that the Iris handled disk I/O intensive tasks up to an order of magnitude quicker than the VAX.

The MAS processing code has now been ported to the Iris, and a small flight track in netCDF produced. All results from the processing code agree with the output from the VAX. The optimal strategy for processing data operationally is currently being determined. This strategy will no doubt evolve once some FIRE data is processed.

DRAFT

MODIS Science Data Support Team (SDST)

Project Plan

Version 1

DRAFT

Table of Contents

DRAFT

Table of Contents

Sample Listing of a "C" Program

Thomas E. Goff

This program is included as a sample source listing to generate comments from the user community concerning the readability of SDST generated source listings. The intent is to gather a list of ideas that can be used to generate a recommended style guide across all the code that is generated by both the MODIS SDST and the MODIS Team Members. We are especially interested in comments and preferences concerning items such as:

Variable naming conventions: Case sensitivity, under bars, length

Comments: amount and form of the in line comments

Function or procedure size: number of lines per module

Embedded documentation: using a script to automatically generate documentation from comments in the source code

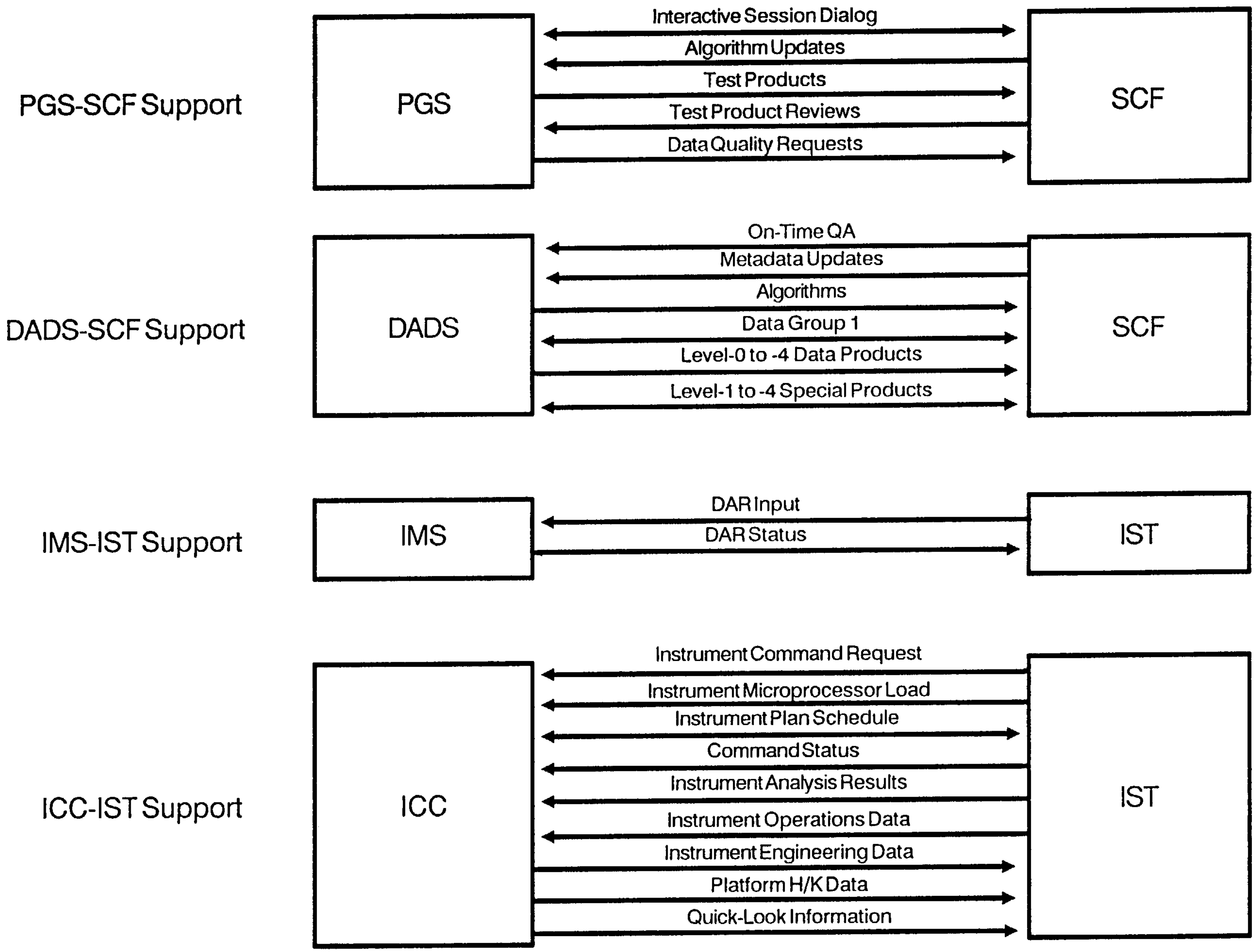
Include files: use and misuse thereof

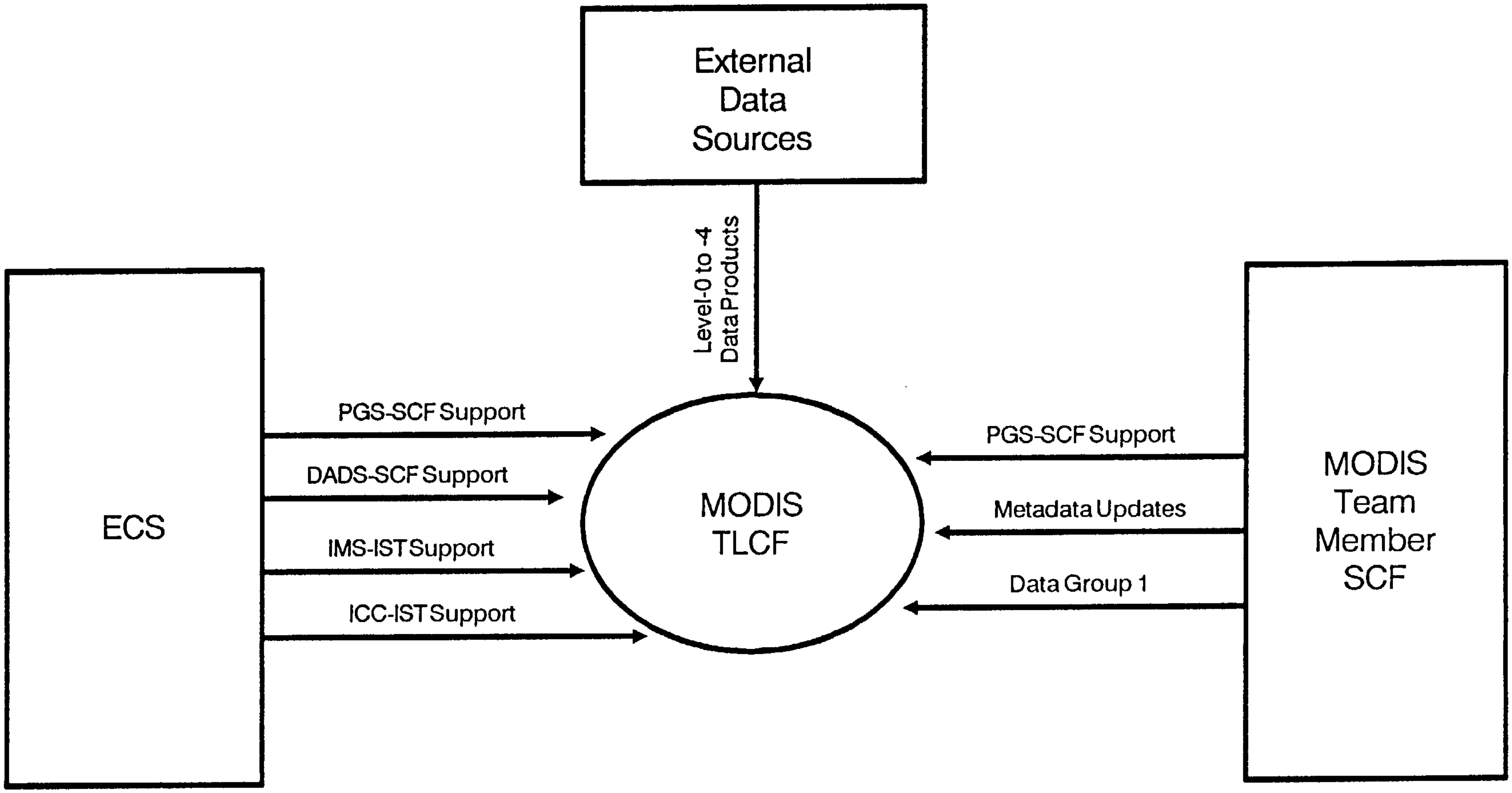
We realize that "C" programming styles are significantly less formal than FORTRAN styles and are only interested in a minimum specification for allow an easier understanding and reuse of group code.

```
/* This program retrieves in NetCDF format and performs a "hyperslab" get
   of the radiance values for one channel from a NetCDF file.  This slab and
   the remaining slabs (channels) are then placed sequentially into a PCI
   EASI/PACE flat data file */

/* Written by Tom Goff, RCD, (301) 982-3704, teg@ltpiris2.gsfc.nasa.gov  */

/* To compile: cc -o netcdf2pci netcdf2pci.c -L. -lnetcdf
   This requires that the NetCDF library be in your current directory tree. */

/* It is assumed that the EASI/PACE software requires that a line of data have
   a multiple of 8 pixels per line and and 8 lines per band.  Also note that
   the size of a band of data must be a multiple of 512 Bytes */

/* All pixel data is assumed to be 16 bit for now !! */

#include <stdio.h>
#include "netcdf-1.17/netcdf/src/netcdf.h"

main()
{
   int i,j,fp;                                  /* indices, file pointer */
   int size;                                /* generic buffer size in Bytes*/
   int LineSize;           /* size of the output line in Bytes, mod(8) */
   long BandSize;         /* number of lines in an output band, mod(8) */
   int NumOfLines;                /* number of input lines per channel */
   int NumOfPixels;                /* number of input pixels per line */
   int NumOfBands;                     /* number of channels of data */
   int Bpp = 2;                         /* number of Bytes per pixel */
   char infile[80],outfile[80];                          /* file names */
   int BandNum,LineNum;                               /* array indices */
   int cdfid,Tid,Cid,Pid,Hid,Hdrid,Datid;           /* NetCDF data IDs */
   char *Header;                /* pointer to the header data buffer */
   int start[3], count[3];               /* hyperslab start and size */
   char *datain;           /* one input channel of data (hyperslab) */
   char *dataout;                        /* one line of output data */

   ncopts = NC_VERBOSE;               /* enable NetCDF error reporting */

   printf("\nEnter the input file name (must have .cdf extend): ");
   scanf("%s",infile);
   cdfid = ncopen(infile, NC_NOWRITE);
   if (cdfid < 0) {
      printf("\nThis file does not exist\n");
      exit(0);
      }
   printf("\nEnter the output file name: ");
   scanf("%s",outfile);
   if ((fp = creat(outfile, -1)) == -1)  {
      printf("\nCannot create this file\n");
      exit(0);
      }
                                 /* get the dimension and variable IDs */
```

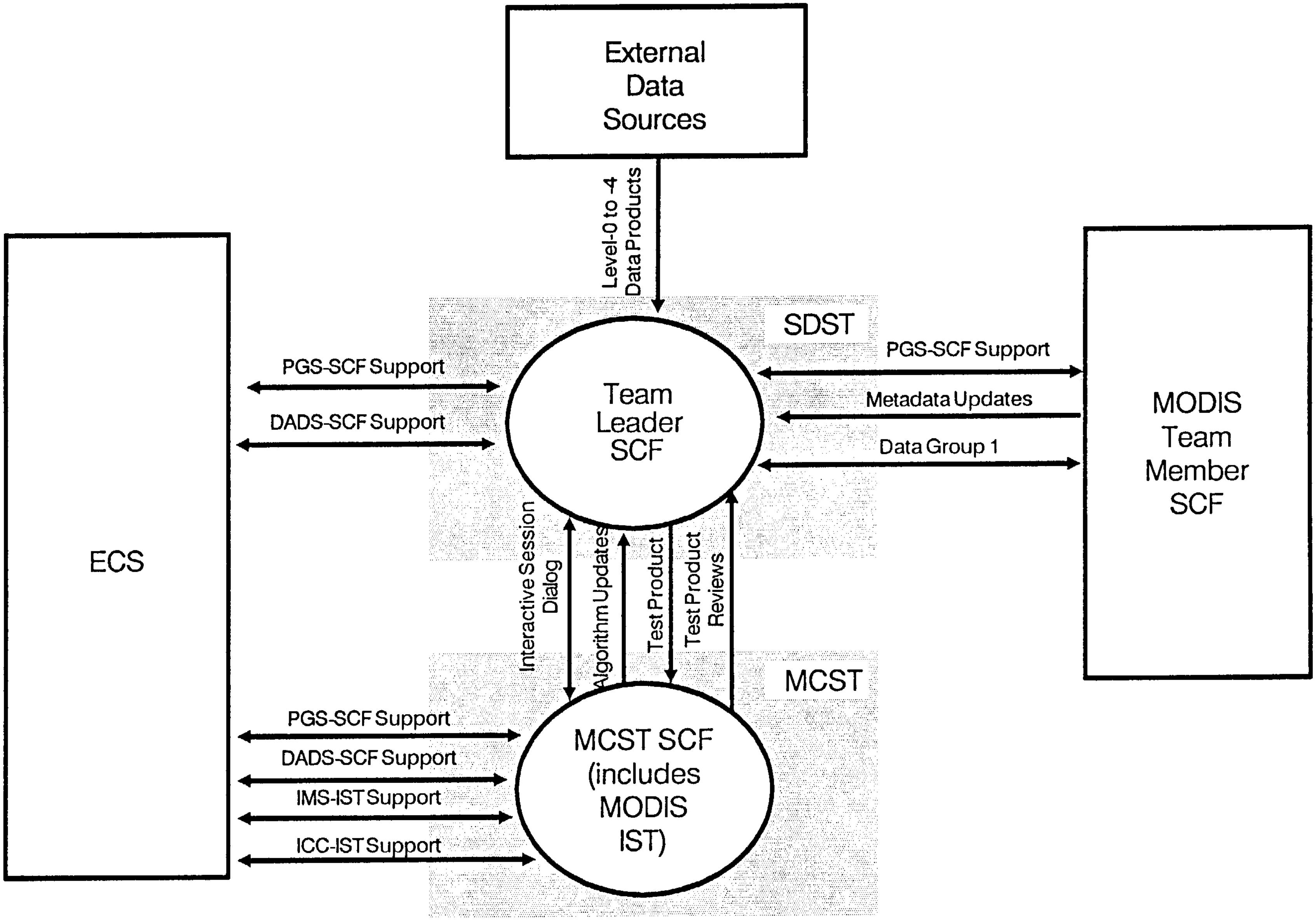
```
Tid = ncdimid(cdfid, "Time");
Cid = ncdimid(cdfid, "NumberOfChannels");
Pid = ncdimid(cdfid, "NumberOfPixels");
Hid = ncdimid(cdfid, "HeaderLength");

Hdrid = ncvarid(cdfid, "DataSetHeader");
Datid = ncvarid(cdfid, "CalibratedData");

                                /* get the data buffer sizes */
ncdiminq(cdfid, Hid, (char *) 0, &size);
if ((Header = (char *)malloc(size)) == NULL)  {
   printf("\nUnable to allocate memory for the header record: %d Bytes",
                 size);
   exit(0);
   }
ncvarget(cdfid, Hdrid, start, &size, Header);
printf("\nThe data set Header is: %s\n",Header);

ncdiminq(cdfid, Tid, (char *) 0, &NumOfLines);
printf("\nThe data set has: %d lines",NumOfLines);
ncdiminq(cdfid, Pid, (char *) 0, &NumOfPixels);
printf("\n  with %d pixels per line",NumOfPixels);
ncdiminq(cdfid, Cid, (char *) 0, &NumOfBands);
printf("\n  and %d channels",NumOfBands);

if ((datain = (char *)malloc(NumOfLines*NumOfPixels*Bpp)) == NULL)  {
   printf("\nUnable to allocate memory for an image band");
   exit (0);
   }
            /* line size must be a multiple of 8 */
LineSize = (((NumOfPixels-1)/8)+1)*8*Bpp;                /* (in Bytes) */
if  ((dataout = (char *)malloc(LineSize)) == NULL)   {
   printf("\nUnable to allocate memory for an output line");
   exit (0);
   }
                          /* clear the output buffer for padding reasons */
for (i=0; i<LineSize; ++i) dataout[i] = 0;

            /* band size must be a multiple of 8 */
BandSize = ((NumOfLines+7)/8)*8;
printf("\nBandSize, LineSize is: %d, %d\n", BandSize, LineSize);

for (BandNum=0; BandNum<NumOfBands; ++BandNum)
{

/*       line #                  Band #              Pixel #          */
  start[0] = 0;            start[1] = BandNum;  start[2] = 0;
  count[0] = NumOfLines;   count[1] = 1;        count[2] = NumOfPixels;

                              /* get the hyperslab of data (one band) */
     ncvarget(cdfid, Datid, start, count, datain);
     for (LineNum=0; LineNum<NumOfLines; ++LineNum)
        {
        for (i=0; i<NumOfPixels*Bpp; ++i) dataout[i] =
                                 datain[LineNum*NumOfPixels*Bpp+i];
```

```
        if ((size = write(fp, (char *) dataout, LineSize)) <= 0 )
          {
           printf("\nfile write error: %d", size);
           exit (2);
           }  ;
        if (LineNum/100*100 == LineNum) printf("\rLine: %d", LineNum);
        } ;
     for (i=0; i<LineSize; ++i) dataout[i] = 0;
     for (LineNum=NumOfLines; LineNum<BandSize; ++LineNum)
        {
        if ((size = write(fp, (char *) dataout, LineSize)) <= 0 )
           {
           printf("\nFile pad write error: %d", size);
           exit (2);
           } ;
        } ;
     printf("\nWrote the data for band number %d\n",BandNum+1);
     printf("  of size %d pixels by %d lines\n",LineSize,LineNum+1);
     } ;
  close(fp);
  printf("\nWrote a %d Byte per line, by %d line file\n",i,LineNum);
  printf("      from a %d pixel by %d line NetCDF file\n",
                                               NumOfPixels,NumOfLines);
  ncclose(cdfid);
  exit(0);
}
```

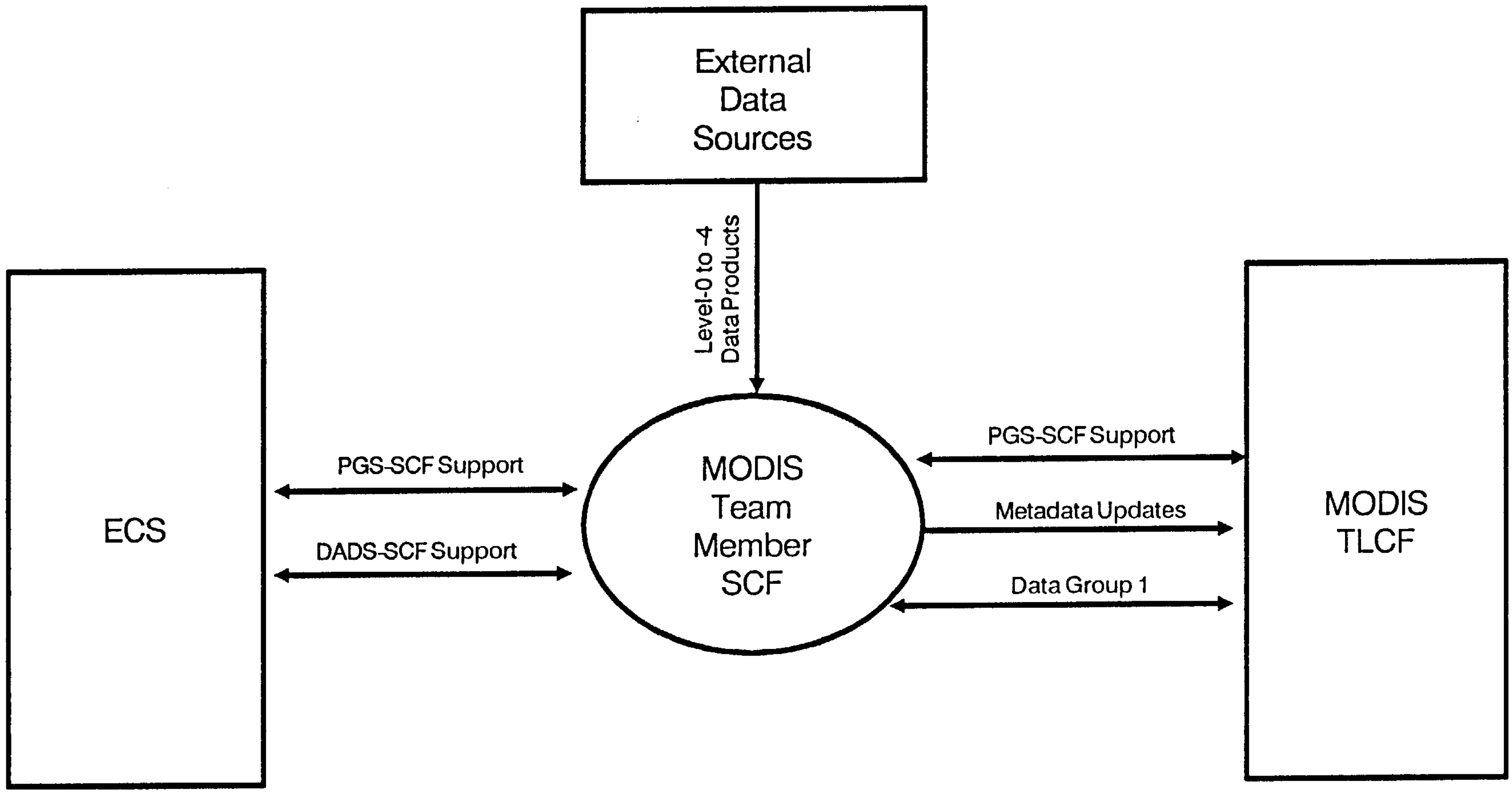

PGS-SCF Support
PGS
Interactive Session Dialog
Algorithm Updates
Test Products
Test Product Reviews
Data Quality Requests
SCF
DADS-SCF Support
DADS
On-Time QA
Metadata Updates
Algorithms
Data Group 1
Level-0 to -4 Data Products
Level-1 to -4 Special Products
SCF
IMS-IST Support
IMS
DAR Input
DAR Status
IST
ICC-IST Support
ICC
Instrument Command Request
Instrument Microprocessor Load
Instrument Plan Schedule
Command Status
Instrument Analysis Results
Instrument Operations Data
Instrument Engineering Data
Platform H/K Data
Quick-Look Information
IST



MODIS TLCF Context Diagram



TLCF Level-1 Data Flow Diagram



MODIS Team Member SCF Context Diagram

Data Dictionary for SCF Context

ALGORITHMS consist of the executable programs for science product generation, source code of these executable programs, job control scripts, and algorithm documentation. Algorithms are the result of new or updated science algorithms passing through the integration and test process, involving the scientist and the PGS's algorithm integration and test staff. After formal approval, algorithms are delivered by the PGS to the DADS for storage, and are retrieved as needed to support product production. The DADS shall also archive algorithms contributed as EOSDIS resources by other data centers. Algorithms shall be orderable and distributed to authorized users. Some frequently used algorithms may also be kept on line in the PGS.

ALGORITHM UPDATES are delivered to the PGS's integration and test environment by scientists at an SCF. They represent changes to existing production algorithms, or a new algorithm to produce a new Standard Product. Algorithm updates include the source code for the candidate algorithm, its associated documentation, and a job step control skeleton. The source code will be compiled to form an executable program suite as part of the integration and test process. The job step control skeleton contains instructions that control the sequence of execution of, and the interchange of data between programs from the executable program suite. Test data sets and calibration data should also be included.

PGS-0610 The PGS shall accept from the SCFs new or modified calibration coefficients to be validated in the test environment. Calibration coefficients shall contain the following information at a minimum:

a. Identification of coefficient data set
b. Calibration coefficients values
c. Author and version number
d. Identification of related processing algorithm
e. Start and stop date/time of applicability
f. Documentation (e.g., author, date and time, SCF identification, reasons for update, etc.)

PGS-0640 The PGS shall accept from the SCF new or modified Standard Product algorithms to be tested at the processing facility. This software shall be received into the test environment and shall contain the following information at a minimum:

a. Algorithm identification
b. Algorithm source code
c. List of required inputs
d. Processing dependencies
e. Test data and procedures
f. Algorithm documentation

ANCILLARY DATA refers to any data, other than Standard Products, that are required as input in the generation of a Standard Product. This may include selected engineering data from the EOS platform, ephemeris data, as well as non-EOS ancillary data. All ancillary data is received by the PGS from the DADS.

CALIBRATION is the collection of data required to perform calibration of the instrument science data, instrument engineering data, and the spacecraft or platform engineering data. It includes pre-flight calibration measurements, in-flight calibrator measurements, calibration equation coefficients derived from calibration software routines, and ground truth data that are to be used in the data calibration processing routine.

CORRELATIVE data are scientific data needed to evaluate and validate EOS data products.

DATA QUALITY REQUEST is a request issued by the PGS to a scientist at an SCF to perform QA of a particular product before future processing or distribution. A time window is applied to the request in keeping with the production schedule.

DOCUMENTS are the hardcopy or digitized references or records about an instrument or the products generated from its data. These shall be archived at the DADS.

INTERACTIVE SESSION DIALOG consists of messages that flow between a scientist at an SCF and the PGS that support general communication with the Integration and Test Service. This includes logins, mail messages, etc.

L0-L4 DATA PRODUCTS consist of L0 Data Products from the IPs, the ADCs and ODCs, and L1-L4 Standard Products produced in the PGS.

PGS-1260 The PGS shall send the DADS quick-look products for routing to the appropriate destination (e.g., ICC, SCF).

L1-L4 SPECIAL PRODUCTS are special science data products consisting of L1A, L1B, L2, L3, and L4 which are produced at the SCFs. These shall be archived at the DADS and distributed to authorized requestors.

METADATA is data which describes the content, format, and utility of a Standard Product. It includes standard metadata (i.e., algorithm and calibration numbers, size of product, date created, etc.), algorithm-derived metadata, QA information from the PI's, summary statistics and an audit trail. Metadata is received by each DADS with the corresponding data sets. DADS validates it physically, updates it with inventory information, enters it into a distributed database (to which the IMS has access), and archives it. Metadata about special products produced at SCF shall be sent to DADS along with their associated data products.

METADATA UPDATES are additional or changed items in the metadata for a previously delivered product.

ON TIME QA is a response to a data quality request that is received within the established production time window. It is received from a scientist at an SCF. It consists of data which will be used to complete the QA fields of the metadata. Overdue QA responses are sent directly to the DADS.

PGS-1130 The PGS shall receive product QA from the SCF which shall describe the results of the scientist's product quality review at an SCF. Product QA shall contain the following information at a minimum:

a. Identification of product
b. QA results
c. Product storage and processing instructions

TEST PRODUCTS are science products generated by new or updated algorithms during the integration and test period. Test products are delivered to scientists at an SCF.

PGS-0900 The PGS shall send test products to the SCF for analysis. These shall contain the results of algorithm testing and shall contain the following information at a minimum:

a. Algorithm identification
b. Test time(s)
c. Processor identification
d. Test results

TEST PRODUCT REVIEWS are evaluations of test products that are used to determine how to proceed in the integration and test process for a new or updated algorithm. A review may indicate the need for further algorithm refinement, or it may indicate that a candidate algorithm is ready for formal adoption into the production environment. Test product reviews are received by the PGS from scientists at an SCF.

IST References from the ECS SOW and Specification

1.4.1 ECS Functional Elements

....

3) Instrument Support Terminals (ISTs) - Investigator-site ECS software toolkits to connect a Principal Investigator (PI) or Team Leader (TL) to an ICC in support of remote instrument planning and monitoring. (Investigator facilities are shown outside the FOS in Figure 1.4.1-1; they are connected to the FOS via the EOSDIS Science Network (ESN)). IST toolkits are hosted on user-provided systems.

1.4.2.1 FOS Operations

....

The ECS Contractor shall provide software (toolkits) for ISTs. ISTs will allow a remote instrument scientist to participate in the planning and scheduling function, review engineering data, develop command requests, and assist in the resolution of anomalies.

4.1 ECS ARCHITECTURE

....

Instrument Support Terminals (ISTs) - investigator-site ECS software to connect a Principal Investigator (PI) or Team Leader (TL) to an ICC in support of remote instrument control and monitoring. (Investigator facilities are shown in Figure 4.1-1 outside the FOS, but connected to it via the ESN.)

4.5 OPERATIONAL VIEW

....

The ICC performs several functions, some of them in conjunction with the ISTs, which may be located remotely or adjacent to the ICC. Among these functions are the monitoring of on-board instrument health and safety, interactive planning and scheduling with the EOC, and instrument command generation.

6.2.2 Physical View

....

It is planned that there will be one IST per instrument, with the locations of each IST at the PI/TL site. However, the design of the FOS shall not preclude the provision of additional ISTs if desired by a PI/TL to accommodate Co^^Investigators and Team Members, who may be at geographically separate locations, but are required to support the operations of the instrument. The capabilities available at the IST will be provided as a software toolkit that can run at a user-provided terminal or workstation.

6.3 OPERATIONAL VIEW

....

The ICCs are responsible for the health and safety monitoring of their instruments. An ICC is capable of "safing" its instrument after identifying an emergency condition. An ICC, supported by the PIs/TLs at the IST, will also perform anomaly investigations and trend analyses to determine the performance of its instrument. The ICC will provide general instrument status information to the EOC.

....

The EOC shall operate 24 hours a day, with some functions, such as planning, not supported by all shifts. The ICCs are also expected to operate 24 hours a day, with similar reductions in support during non-normal hours. ISTs will operate as desired by the appropriate PI/TL, and therefore may not support around the clock operations.

6.5.2.1.5 Telemetry Processing Service

....

Monitoring information will be provided by the ICCs to their respective ISTs, allowing the PIs/TLs to participate actively in the instrument monitoring activities, when desired or necessary for expert analysis.

6.5.2.1.10 IST

The IST will support the input of DARs into the ECS system. Tools that are co-resident with the IST will be provided to aid in the construction of the DAR and account checking with SMC prior to the submission of the DAR and subsequent transmission to the IMS (via the ESN). The IST will provide access to DARs that are being initially processed by the ICC, so that the PI/TL may provide input into their processing.

The IST will provide access to the plans and schedules being developed by the ICC and EOC, so that the PI/TL may review the planned instrument use. In addition, the IST will permit the PI/TL to have interactive scheduling capabilities to aid in the decision making of the schedule generation. The interactive scheduling at the IST will permit only the generation of "what-if" versions of the schedule and not the final schedule.

The IST will permit the PI/TL to transmit to the ICC instrument microprocessor memory loads and updates to the ICC Instrument Data Base (IDB). The IST will also provide the capability for the PI/TL to issue command requests to the ICC, if urgent commands are deemed necessary by the PI/TL or designee. The PI/TL will have access to command uplink status via the IST, as received from the ICC.
To support anomaly and long-term analysis, in addition to the calibration of the instrument, the IST will receive and display engineering telemetry from the ICC. The IST will also provide the capability to display quick-look products after receipt from the ICC. Instrument status, as determined by the ICC, will be made available to the IST, upon request.

6.5.2.3.6 Instrument Analysis Service

ICC-4600 The ICC shall accept from the IST at a minimum the following:
a. Instrument anomaly notifications and instructions
b. TL/PI analysis results
c. Calibration information
d. Performance data

ICC-4610 Upon request from the TL or instrument PI, the ICC shall provide the IST with at a minimum the following:
a. Instrument performance assessment data
b. Quick-look data

6.5.2.3.10 IST

ICC-7010 The IST shall have the capability to accept from the ICC and display, in parallel with any current activities in the IST, a notification regarding at a minimum the following:
a. Conflicts found in planning and scheduling
b. Arrival of instrument engineering data
c. Arrival of quick-look data
d. Instrument anomalies found during instrument monitoring

ICC-7030 The IST shall have the capability to accept the requested data from the ICC in parallel with any current activities in the IST.

ICC-7050 The IST shall have the capability to provide the ICC with updates to the IDB.

ICC-7060 The IST shall have the capability to ingest data from the IST that supports PI/TL instrument operations, which include at a minimum:
a. Microprocessor memory loads
b. Changes in the instrument parameters

ICC-7070 The IST shall have the capability to provide data to the ISTs that support PI/TL instrument operations, which include at a minimum:

a. Microprocessor memory dumps
b. Instrument analysis results

6.5.2.3.10.1 DAR Processing Support

ICC-7110 The IST shall provide the capability for the TL or instrument PI to communicate with a DAR originator(s) for modification and for clarification of DARs.

ICC-7150 The IST shall provide the capability for the TL or instrument PI to generate and submit DARs to the IMS.

ICC-7170 The IST shall receive from the IMS status of the DARs generated at the IST.

ICC-7180 The IST shall have the capability to accept from the ICC a DAR evaluation request along with the corresponding DAR.

ICC-7190 The IST shall provide the capability for the TL or instrument PI to view and modify the DARs received from the ICC.

ICC-7200 The IST shall provide the ICC with either DAR acceptance notification with updates, if any, or DAR rejection notification along with the reasons for rejection.

6.5.2.3.10.2 Planning and Scheduling Support

ICC-7210 The IST shall provide the capability for the TL or instrument PI to generate an ISAR.

ICC-7212 The IST shall interface with the ICC to submit an ISAR.

ICC-7214 The IST shall interface with the ICC to receive notification of ISAR receipt.

ICC-7220 The IST shall have the capability to request and accept from the ICC planning and scheduling information, which includes at a minimum the following:

a. LTSP goals and priorities
b. Current resource availability information
c. Current predicted orbit data and related information
d. Plans and schedules

ICC-7230 The IST shall have the capability to access planning and scheduling functions in the ICC or EOC (via ICC).

ICC-7240 The IST shall accept from the ICC a request for resolving conflicts identified at the ICC while building or updating plans or schedules for its instrument.

ICC-7250 The IST shall provide the capability for the TL or instrument PI to view and evaluate the requests for resolving problems of the instrument plans and schedules.

ICC-7270 The IST shall provide the ICC with the results of evaluating the plans and schedules (in response to the conflict resolution requests from the ICC).

6.5.2.3.10.3 Instrument Commanding Support

ICC-7290 The IST shall have the capability to request and accept from the ICC at a minimum the following:

a. Current CFS
b. Instrument commands and memory loads
c. Instrument command status

ICC-7330 The IST shall provide the capability for the TL or instrument PI to review instrument commands.

ICC-7350 The IST shall provide the capability for the TL or instrument PI to generate and display a command request.

ICC-7360 The IST shall have the capability to provide a command request to the ICC.

ICC-7370 The IST shall have the capability to accept from the ICC command request status, which contains at a minimum the following:

a. Receipt of the command request at the ICC
b. Receipt of the commands at the EOC
c. Status of command transmission to the platform
d. Receipt of the commands at the platform
e. Execution of the commands at the instrument

ICC-7390 The IST shall have the capability to send instrument microprocessor memory loads to the ICC.

ICC-7400 In support of a TOO observation, the IST shall be capable of retrieving and completing the corresponding command request and forwarding it to the ICC within 10 minutes of receiving initiation notification from the TL or instrument PI.

ICC-7410 In support of a critical instrument support activity, the IST shall be capable of retrieving and completing the corresponding command request and forwarding it to the ICC within 10 minutes of receiving initiation notification from the TL or instrument PI.

6.5.2.3.10.4 Instrument Monitoring Support

ICC-7430 The IST shall have the capability to request and accept from the ICC at a minimum the following:

a. Instrument housekeeping and engineering data
b. Platform housekeeping data
c. Derived parameters for its instrument
d. Quick-look data
e. Instrument performance assessment data

ICC-7460 The IST shall provide the capability for the TL or instrument PI to extract any subset of the raw or engineering unit converted instrument engineering data for display and subsequent processing.

ICC-7500 The IST shall have the capability to analyze instrument housekeeping and engineering data to determine the instrument trend.

ICC-7510 The IST shall have the capability to access any data in the ICC history log.

ICC-7530 The IST shall have the capability to display quick-look data.

ICC-7550 The IST shall provide the capability for the TL or instrument PI to notify the ICC of any instrument anomalies detected.

Approximate Projections of Near-Ten[illegible] ODIS Team Leader SCF Utilization

	1991				1992				1993				1994				1995			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
External Milestones								▼[1]				▼[2]			▼[3]					▼[4]
Prototype Data Processing																				
MAS[5] Algorithm Development and Maintenance																				
MAS[5] Operational Processing																				
Data Format Implementation and Testing																				
Image Registration Trials																				
DEM Correction Trials																				
Team-Member-Defined Support Processing																				
MODIS Data Simulation																				
Preliminary Data Cataloging and Distribution System																				
Integration with Version-0 DAAC																				
Integration and Testing of Version-1 Software													6			7		8		
MODIS Level-1 Software Implementation																				
Level-1A Algorithm Development																				4
Level-1B Algorithm Development																				4
Software Implementation Support																				
CASE Tools (Evaluate, Select, and Run)																				
Project Management Tools																				
Software Guidelines and Standards Validation																				
Software Configuration Management																				
ECS Toolkit Evaluation (Beta Testing)																				

1ECS Contract Award
2PDR-PGS Architecture Chosen (approx.)
3PGS-compatible machine delivered (approx.)
4Version 1 software due
5MAS or a successor instrument
6Team Member Version 0 code at SDST for integration and testing (with selected TMs, not contractually required), simulated data need[illegible]
7Review progress, make changes
8Team Member Version 1 code due at SDST for integration and testing