MODIS DATA STUDY TEAM PRESENTATION

December 7, 1990

AGENDA

1. Action Items
2. MODIS Data System/Software Validation Plan
3. Anchor Point Accuracies II: Summary and Extensions

ACTION ITEMS:

10/12/90 [Watson Gregg]: Prepare a report on MODIS anchor point requirements. Analyze the utility of alternative parameters to describe MODIS observation and solar geometry. STATUS: Report presented at 11/30/90 meeting. Additional anchor points along-track to be considered. Open.

10/19/90 [John Blaisdell]: Expand introductory material in Earth Model write-up to include broad discussion of MODIS geolocation and need for Earth model. Coordinate with Al Fleig to distribute report. STATUS: Action Item reassigned to Al McKay and Lloyd Carpenter. Open.

10/26/90 [John Blaisdell]: Scope a brief error analysis and impact study on the merits of a geoid model as opposed to an ellipsoid. STATUS: Action Item reassigned to Al McKay and Lloyd Carpenter. Open.

11/16/90 [Doug Hoyt]: Review MODIS Level-1 data flow diagrams and identify data items potentially provided by the MCST. Provide a list of instrument parameters required to Earth locate MODIS pixels (e.g. detector locations, electronic delays, mirror rotations, etc). STATUS: Presently available information insufficient to address item. Report due approximately one year from the assignment date. Open.

11/16/90 [Tom Goff and Al McKay]: Review the preliminary version (28 September 1990) of "Standards and Guidelines for Science Data Processing Software" and provide a list of questions and comments. STATUS: Report presented at 11/30/90 meeting. Closed.

11/30/90 [Al McKay]: Revise comments on "Standards and Guidelines for Science Data Processing Software" and distribute to S. Scott, J. Dozier, B. Conboy, and MODIS.DATA.TEAM. STATUS: Distributed 12/5/90. Closed.

11/30/90 [Team]: Develop a Software Validation Plan for MODIS. STATUS: Delivered to D. Han 12/4/90. Revised 12/5/90.

MODIS DATA SYSTEM/SOFTWARE VALIDATION PLAN

Overview of All Phases/Versions

A. All validation efforts concerning software and the data system will be the responsibility of the MODIS Science Data Support Team (SDST). All geophysical validation efforts concerning the performance of the science algorithms (as opposed to the software) will be the responsibility of the MODIS Science Team (MST) and MODIS Characterization Support Team (MCST). Exceptions are with respect to some Level-1 algorithms (unpacking and navigation algorithms) and Level-3 algorithms (re-sampling and re-mapping), which are written and tested for performance by the SDST.

B. All validation activities will be coordinated with the MST and MCST, as appropriate.

B. Software written by the MCST and MST will be submitted for integration into the data system and will be tested by the SDST.

C. All software will be subject to configuration management, which will be the responsibility of the . This will include establishing a Configuration Management office, which will enforce software standards, maintain and distribute software revisions, and determine and manage a problem-tracking procedure, and maintain records of problems, fixes, and dates.

E. Review meetings with MST and MCST will be established at approximately 6-month intervals.

F. At least four complete, end-to-end versions of the software will be built, tested, and validated, three pre-launch and at least one post-launch.

I. Version 1 -- 1991-1994. Complete data system for Mike King's MODIS aircraft simulator (Wildfire). Performed entirely within Team Leader Computing Facility (TLCF), by the SDST.

A. Validation activities will consist of
   1. Enforcement of software standards/format requirements
   2. Software integration and testing
   3. Operational acceptance and testing
   4. Interface testing
   5. Operational readiness testing

B. Each of these testing phases will apply to all levels of processing
   1. Level-1 (algorithms developed by SDST (except calibration), and applicable to Wildfire; these are likely to be different from MODIS)
      a. Unpacking algorithms
      b. Calibration software
      c. Navigation algorithms
      d. Band registration algorithms
   2. Level-2 (algorithms developed by MST for Wildfire)
      a. Data product software
      b. Ancillary data input/access/integration algorithms
      c. Data products will be validated by research cruises, land and atmosphere field experiments by MST. SDST will incorporate these data sets into the archive and perform testing in cooperation with MST.
   3. Level-3 (algorithms developed by SDST after consultation with MST)

a. Re-sampling algorithms
b. Earth-gridding algorithms

C. Test data will be exchanged between SDST and MST to ensure consistency of output.
D. Wildfire data will be made available to all interested MST members. Distribution software will be built and tested by SDST.
E. Software documentation will be reviewed for conciseness, accuracy, and coherence.
F. A User's Guide will be written, and will contain the results of the validation tests, and subsequent changes made to the code. It will serve as a record of the validation effort.

II. Version 2 -- 1994-1997. First complete end-to-end testing of operational MODIS data system. Performed entirely within TLCF. Validation of ocean science algorithms (some Level-1 and all of Level-2) will be performed by the MODIS Ocean Team. All other validation functions will be performed by the SDST.

A. Validation activities will only consist of
   1. Enforcement of software standards/format requirements
   2. Software integration and testing
   3. Operational acceptance and testing

   (interface control and operational readiness phases are not included for this version)
B. Test data will be simulated (synthetic) MODIS data created by the MST using the following sensor data for input: CZCS, AVHRR, Sea-WIFS, GOES, HIRS, TM, AIS, AVIRIS.
C. Each of these testing phases will apply to all levels of processing
   1. Level-1 -- algorithms developed by SDST (except calibration), and applicable to the simulated MODIS data
      a. Unpacking algorithms
      b. Calibration software
      c. Navigation algorithms
      d. Band registration algorithms
   2. Level-2 -- algorithms developed by MST for the simulated MODIS data
      a. Data product software (as much as is available at this time)
      b. Ancillary data input/access/integration algorithms
   3. Level-3 -- algorithms developed by SDST after consultation with MST
      a. Re-sampling algorithms
      b. Earth-gridding algorithms.
D. Test data will be exchanged between SDST and MST to ensure consistency of output.
E. Software documentation will be reviewed for conciseness, accuracy, and coherence.
F. A User's Guide will be written, and will contain the results of the validation tests, and subsequent changes made to the code. It will serve as a record of the validation effort.

III. Version 3 -- 1997-launch. Launch ready data system. This is a complete end-to-end data system validation effort of the launch-ready algorithms and data system. It will be performed at the Product Generation System (PGS) by the SDST and is designed to test the actual algorithms operating from within their actual operational environment.

A. Validation activities will consist of
   1. Software integration and testing
   2. Operational acceptance and testing
   3. Interface testing
   4. Operational readiness testing

   (the emphasis will be on interface control and operational readiness testing).

B. Final operational test of all Level-1 algorithms.

C. Final operational test of all Level-2 algorithms, including integration and initial testing of last-to-come-in data product algorithms.

D. Final operational test of Level-3 algorithms, including any changes resulting from Version 2 testing.

E. Integration and initial test of Level-4 algorithms, and final operational testing.

F. All utility algorithms will be broken out and given final operational testing as individual, stand-alone (as much as practical) components, delivered to the Data Archive and Distribution System (DADS) and given a final check-out.

G. Software documentation will be reviewed for conciseness, accuracy, and coherence.

H. A final User's Guide will be written, and will contain the results of all the validation tests, and subsequent changes made to the code. It will serve as a record of the validation effort in addition to its function as a help manual.

IV. Post-launch validation activities. These activities will be performed within the first 6 months after launch. They are performed by the SDST, MCST, and MST.

A. Level-1 -- SDST responsibility
   1. Unpacking algorithms will be tested for accuracy and consistency.
   2. Ground control points will be used to validate the navigation algorithms. This will include testing of band ground registration in cooperation with MCST.
   3. Calibration software will be tested against test data from calibration sites provided by the MCST.

B. Level-2 -- primarily MST responsibility
   1. Results of validation experiments involving research cruises, field experiments and the data will be provided by MST to SDST for testing at the PGS.
   2. Test data sets will be exchanged between SDST and MST to ensure consistent output.
   3. Any changes in algorithms or new algorithms will undergo complete testing using MST-supplied test data sets at the TLCF by the SDST before submission to the Algorithm Integration Facility (AIF) of ECS. (This assumes that in the post-launch phase all updated and new algorithms will be sent to the AIF before being incorporated into the DADS.

C. Utility algorithms will be updated as a result of post-launch validation and sent to the DADS.

D. The final User's Guide will be updated to record all of the results of the post-launch validation tests.

ANCHOR POINT ACCURACIES II: SUMMARY AND EXTENSIONS

Extensive simulations of the Earth location accuracies (presented at the MODIS Data Study Team Meeting, November 30, 1990) resulted in several conclusions.

1) The use of anchor points can reduce the computation and storage burden of the system dramatically. For a 2% anchor point density the number of computations was reduced by a factor of 27 and storage by a factor of 52. There also appeared to be a slight reduction in computer run-time, but the simulations were driven by input/output time, and thus did not provide a realistic test.

2) The solar zenith and azimuth angles are slowly varying functions. Even under the worst case examined (0.5% anchor point density, MODIS-T tilted 50°), errors in these angles were small (<< 1°) relative to location knowledge and radiative transfer model accuracies. A case interpolating anchor points across the terminator conformed the conclusion that solar angles need not be the determining factor in the selection of an anchor point strategy.

3) The spacecraft zenith and azimuth angles are also slowly varying functions. The maximum errors occurred near nadir where their accuracies are less important in scattering calculations. These angles also need not be the determining factor in the selection of anchor points.

4) Errors in latitude and longitude can be significant and these parameters should be the determining factor in the selection of anchor point strategy.

5) A non-uniform distribution of anchor points in pixels (uniform distribution in pixel <u>size</u> across and along-track) produced better accuracies than a uniform distribution and came at <u>no cost</u> in computations or storage. Also a cubic spline interpolation method produced better accuracies than a linear interpolation method, and only increased computations slightly (did not increase storage). Thus a non-uniform anchor point distribution with a cubic spline interpolation method is the best strategy.

6) The 2% and 1% anchor point densities produced similar accuracies (≈4 m maximum error in latitude, 23 m in longitude) for MODIS-N. Since the 1% density involves a reduced burden on computations and storage, it is preferred.

7) A 0.5% anchor point density produced acceptable accuracies for an untilted MODIS-T (≈ 2 m maximum error in latitude and longitude), but even a 2% density was unsatisfactory for a 50° tilt (1400 m maximum error in latitude, 2000 m in longitude). This was because the along-track anchor point distributions were chosen to produce uniform spacing in pixel size for an untilted sensor.

There were insufficient anchor points along-track.

The present report focuses on the deficiency described in Conclusion #7, with respect to MODIS-T. We chose non-uniform anchor points along-track based on pixel sizes for a 50° tilt. Only the non-uniform, cubic spline interpolation method is considered here, and only maximum errors in latitude/longitude, based on previous results. We also included error analyses for 20° and 30° tilts, since these may be the operational mode for ocean observations. These results are shown in Table 1.

Even the 0.5% anchor point density produced virtually negligible error in latitude and longitude for tilts $\leq$ 30° (Table 1). However, only a 2% density meets the navigation requirement stated by Justice (125 m). However, the point is that a 2% density can provide satisfactory accuracy.

Finally, we tested the accuracy of a 0.1% anchor point density for MODIS-N. Again we only tested the non-uniform distribution and cubic spline interpolation. Results are also shown in Table 1. This density requires only 6 anchor points along-track. Due to this sparse density, the errors in latitude and longitude increase dramatically. Clearly, there are too few anchor points in a 0.1% density to produce acceptable Earth location accuracy.

MAXIMUM ERROR

Non-Uniform Distribution, Cubic Spline Interpolation

			MODIS-T			
		MODIS-N	0 TILT	20 TILT	30 TILT	50 TILT
2%	Ψ	4	-0.6	-0.6	-0.6	71
	φ	22	-0.6	-0.6	-0.6	106
1%	Ψ	4	-0.6	-0.6	-0.6	372
	φ	24	-0.6	-0.6	-0.7	545
0.5%	Ψ	7	-0.6	-0.7	-0.8	-1245
	φ	51	-0.9	-0.9	-1.0	-1763
0.1%	Ψ	1424				
	φ	10127				

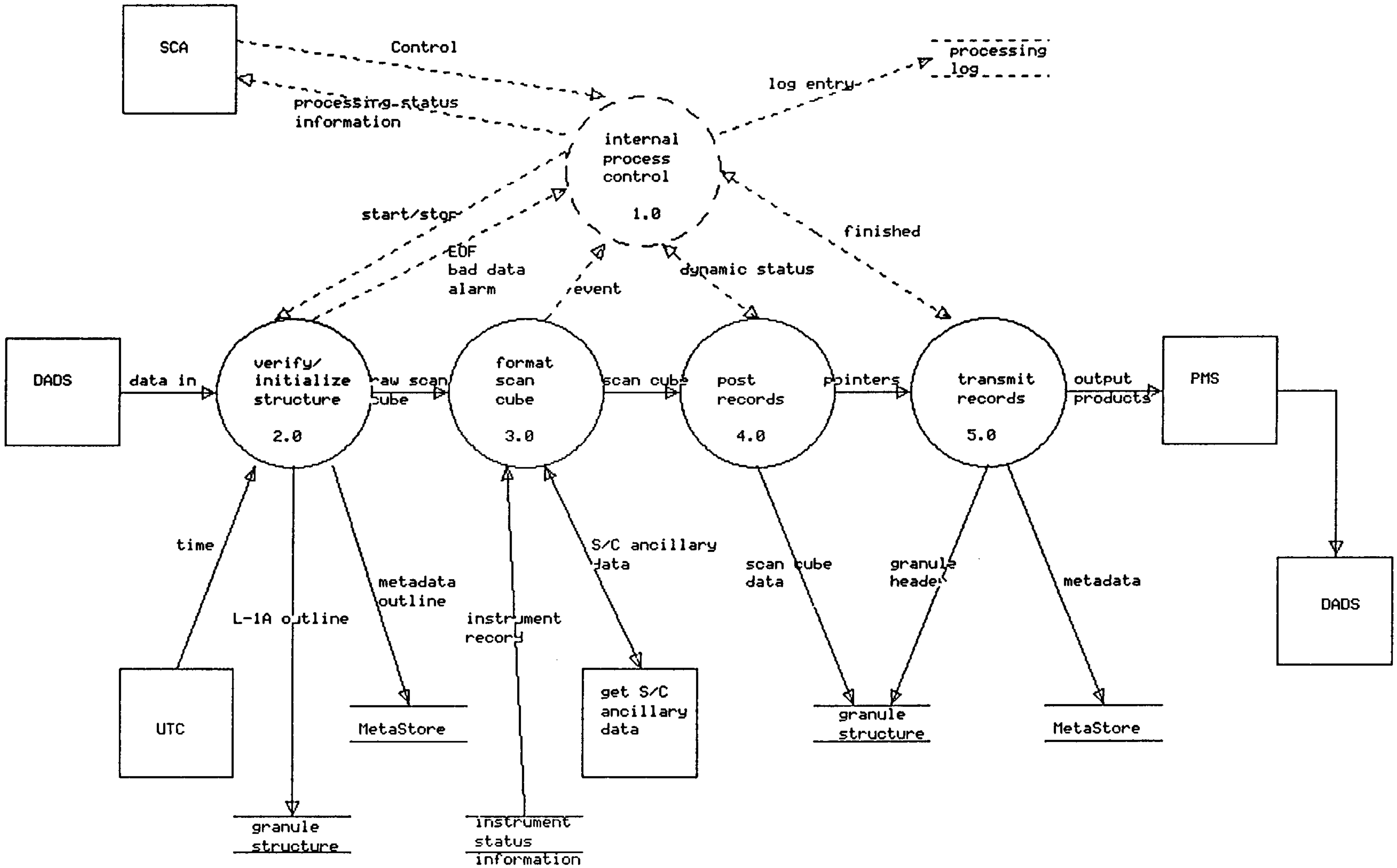
Ψ = latitude (m)
φ = longitude (m)

APPENDIX TO

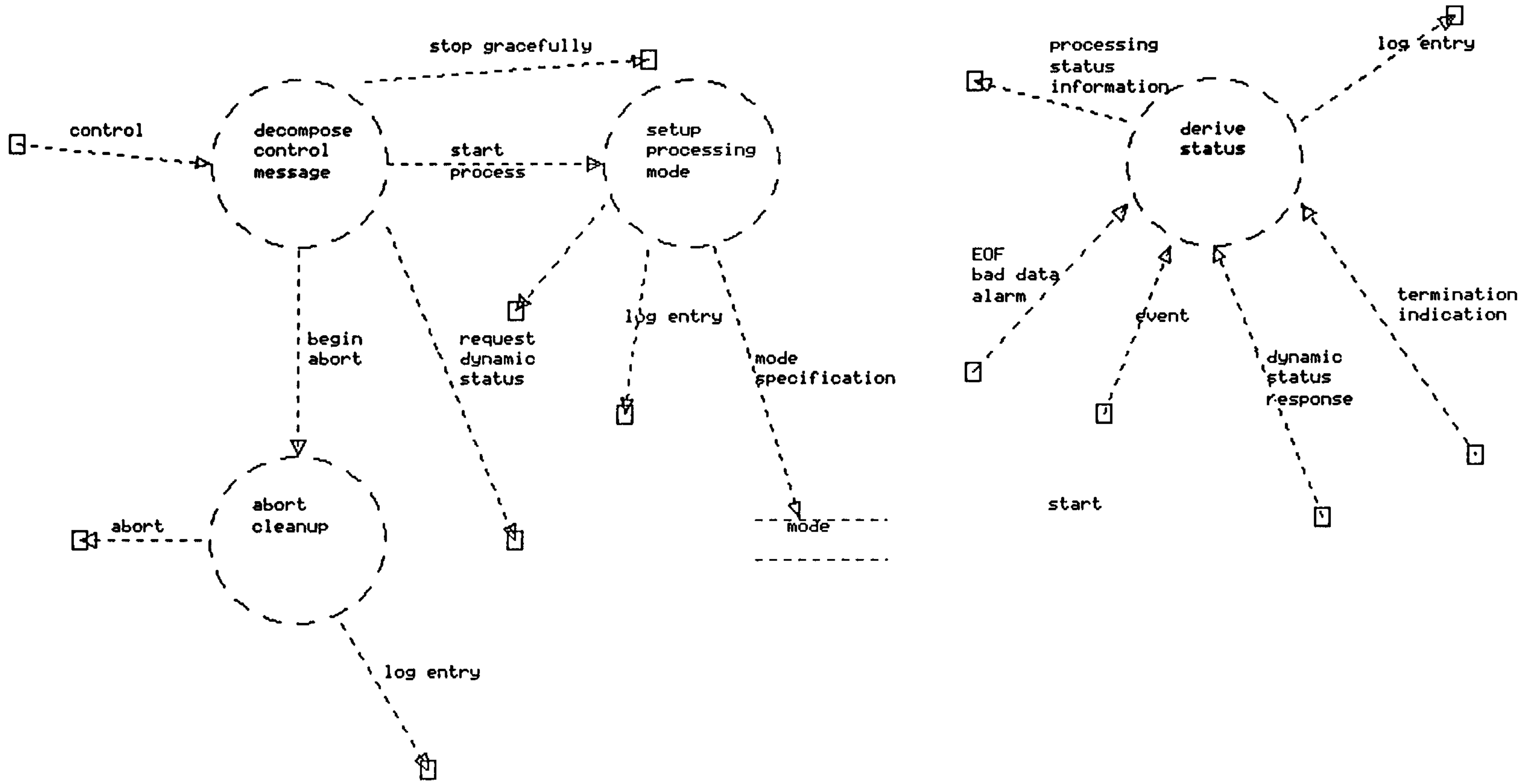
MODIS DATA STUDY TEAM REPORT

DECEMBER 7, 1990

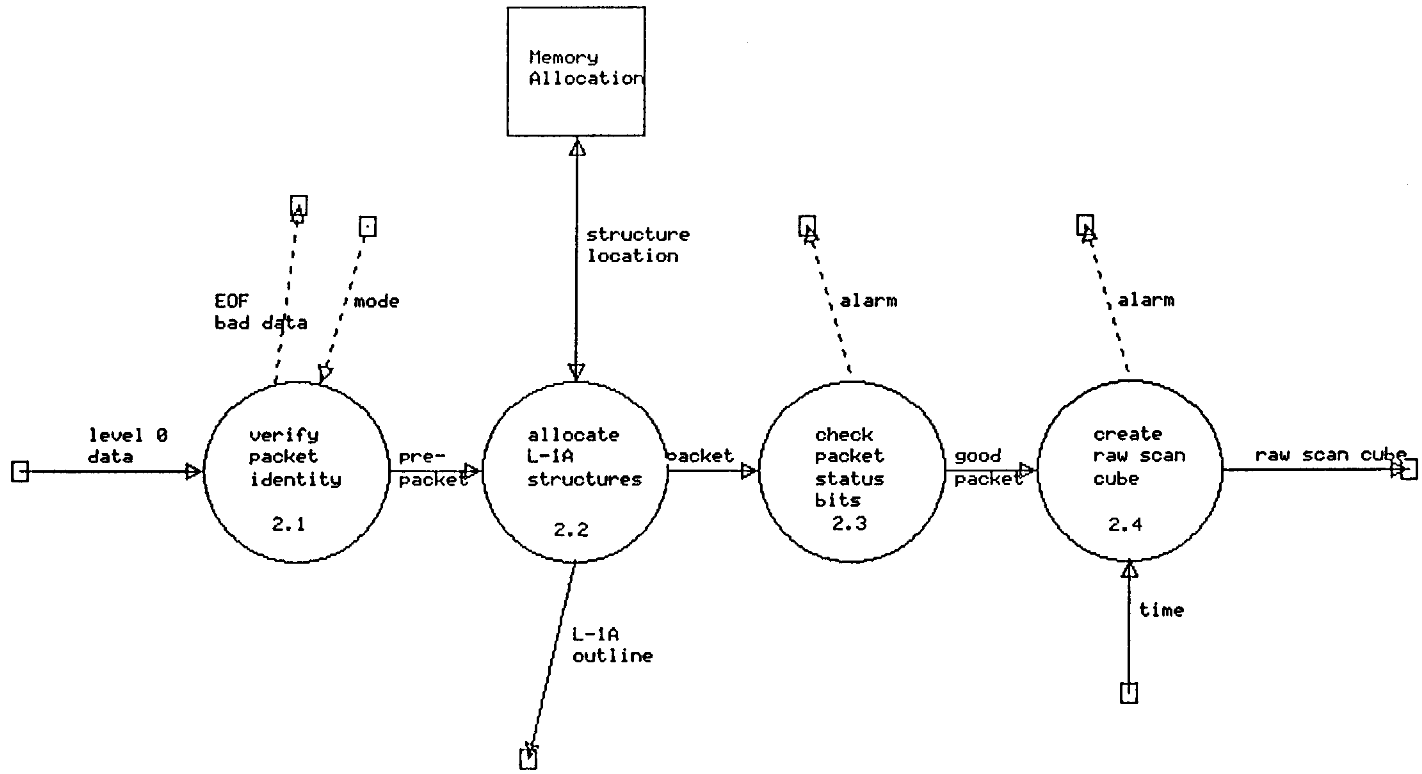
DATA FLOW DIAGRAMS FOR LEVEL-1A



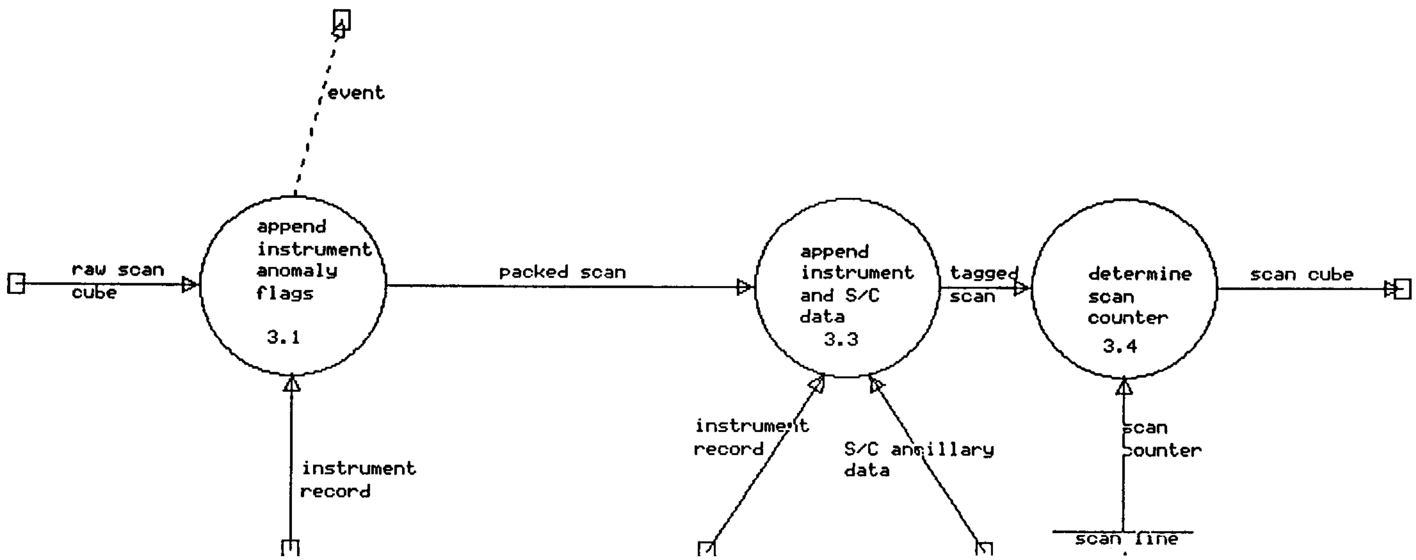
MODIS-1A Design
Level A



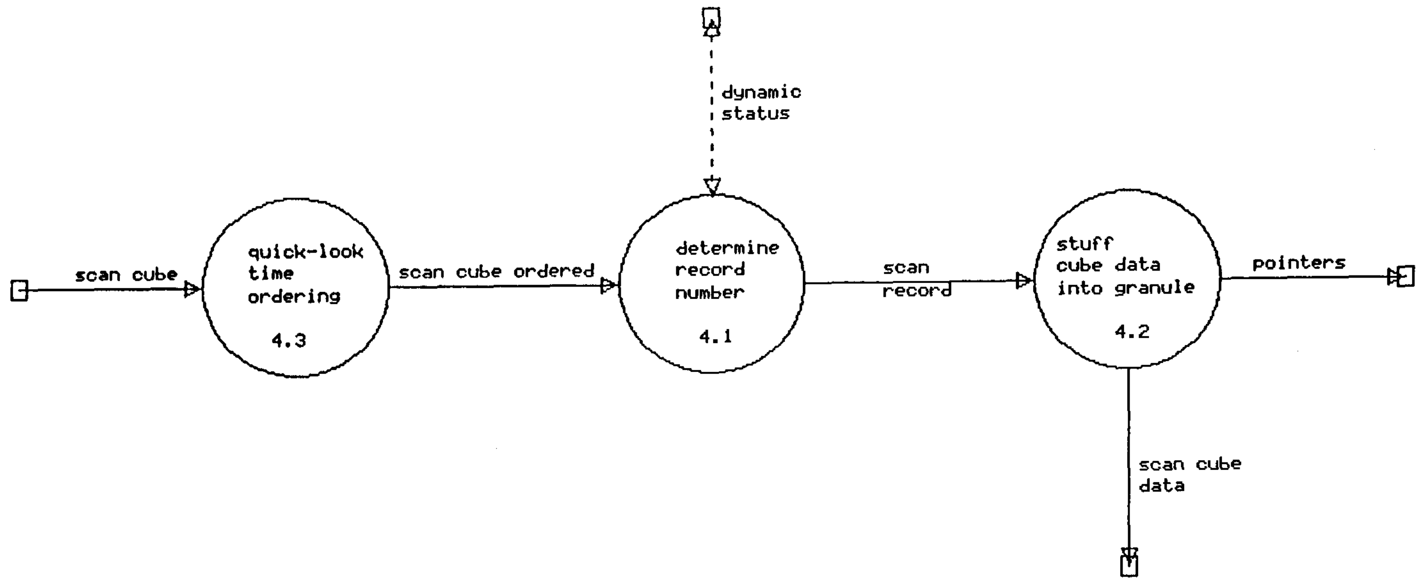
MODIS-1A Design
1.0 Internal
Process Control
Level B



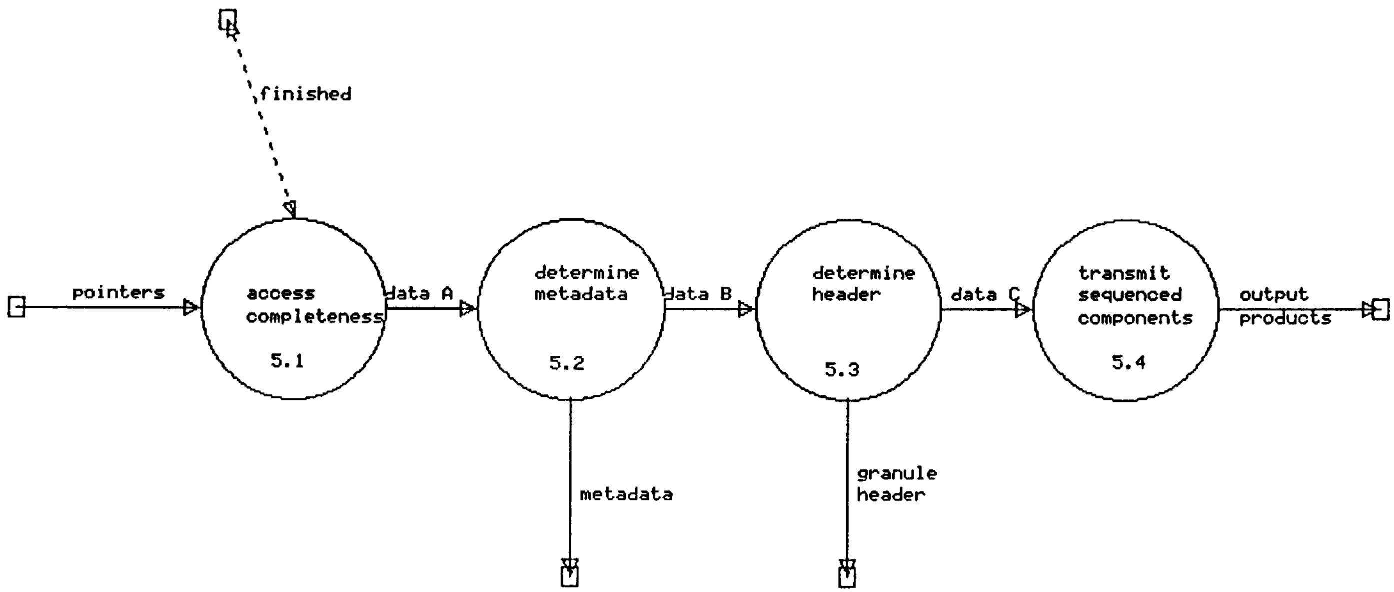
MODIS-1A Design
2.0 Verify/Initialize
Structure
Level B



MODIS-1A Design
3.0 Format Scan Cube
Level B



MODIS-1A Design
4.0 post records
level B



MODIS-1A Design
5.0 transmit records
Level B

<u>Data Dictionary for the Software Engineering</u>
<u>of the MODIS-1A Product Generation Program</u>

This data dictionary document (DDD) contains all items as shown on the various data flow diagrams that accompany this document. All items in the context diagram, data flow diagram, event list, entity relationship diagram, state transition diagram, or other diagrams will be included here. See the appendix at the end of this document for definitions and examples of the content titles.

<u>Name</u> <u>Type</u> <u>From</u> <u>To</u> : <u>Description</u>

Abort ControlFlow AbortCleanup VerifyInitializeStructure : (element of Start/Stop) Flag to perform AbortCleanup function.

AbortCleanup ControlTransform : Deallocate memory, setup terminate message, post entry in the ProcessingLog.

AllocateL-1A_Structure DataProcess : Requests computer memory for the granule, index to the cube within a granule, metadata sizes. Initializes these areas to an invalid data indication.

AppendInstrumentAndS/C_Data DataProcess : Append the instrument record and S/C platform ancillary (ephemeris, attitude, engineering) data to the scan cube.

AppendInstrumentAnomalyFlags DataProcess : Examine the InstrumentStatusInformation and compare this with the telemetered instrument status. Generate an instrument Event if acceptance criteria are not met.

AccessCompleteness DataProcess : Determine if the processing requirements have been met. Send a control message if this criteria is satisfied.

BeginAbort ControlFlow DecomposeControlMessage AbortCleanup : Flag saying begin abort processing.

CheckInstrumentStatusBits DataProcess : Checks and verifies the instrument status and operating conditions.

Control Message SCA InternalProcessControl : Messages that inform the process which mode to operate in, to start, to stop, to suspend, to resume, to return status (dynamically or statically), and to request and verify the staging and/or destaging of data to/from the DADS.

CreateRawScanCube DataProcess : Places a data packet into the proper logical scan cube location.

DADS : Data Archive and Distribution System

DataA ControlFlow AccessCompleteness DetermineMetadata : an indication to update the metadata values.

DataB ControlFlow DetermineMetadata DetermineHeader : An indicator to generate a header record.

DataC ControlFlow DetermineHeader TransmitSequencedRecords : An indicator to transfer the data or pointers to the data to the PMS.

DataIn DataFlow DADS VerifyInitializeStructure : Level-0 data or quick-look data.

DecomposeControlMessage ControlTransformatiom : Determines the control processing and passes control to the resulting process.

DeriveStatus ControlTransform : Formats the message passed to the SCA upon any of the following events: EOF, BadData, Event, DynamicStatusResponse, TerminationIndication.

DetermineHeader DataProcess : Generate the header information.

DetermineMetadata DataProcess : Update the metadata values.

DetermineRecordNumber DataProcess : Answers queries about the status and accounting of the data processing.

DetermineScanCounter DataProcess : Given a completed cube of data, determine where in the granule this cube should be placed. This may be accomplished via an index file or directly. Determine the completeness of each cube of data - are all packets that make up this cube included? The number of scan lines processed is also determined.

DynamicStatus ControlRequest/Message InternalProcessControl PostRecords : Two way dialog asking for current dynamic status and returning this status information.

DynamicStatusResponse ControlFlow PostRecords DeriveStatus : Processing accounting information.

EofBadDataAlarm ControlMessage VerifyInitializeStructure InternalProcessControl : Signals an end of data input, signals bad or inappropriate data, requests an alarm generation.

Event ControlMessage CreateProduct InternalProcessControl : Anomaly in instrument status between instrument record log and telemetered data.

Finished ControlMessage InternalProcessControl TransmitRecords : Request graceful termination (post data) : granule is filled up, terminate.

FormatScanCube DataProcess : Checks instrument status indicators, byte aligns science data, appends S/C platform ephemeris and attitide, and updates packet accounting.

GetS/C_AncillaryData ExternalProcess : An external process (subroutine or separate program) that returns the S/C platform ephemeris and attitude data records in the neighborhood of the given S/C time.

GoodPacket DataFlow CheckInstrumentStatusBits CreateRawScanCube : Packet of data verified for packet sanity and instrument conditions.

GranuleHeader DataStoreRecord TransmitRecords GranuleStructure : Information about the data granule that is necessary for subsequent data processing or for understanding the data in the granule.
An identifier for this data set.

GranuleStructure DataStore : Description of the level-1A granule in processor memory. Initialized with a 'bad data' indicator and filled with 'good data' as it is processed.

VerifyInitializeStructure DataProcess : Verify packet identity, examine level-0 data quality fields, set routing control flags, set up data store areas.

InstrumentRecord StoreRecord InstrumentStatusInformation FormatScanCube : A snapshot of the instrument state at a specified time; integrated from previous instrument commanding.

InstrumentStatusInformation DataStore : Instrument status at a given time as determined by the Instrument Control Center. To be compared with the telemetered status. This is the integrated result of all previous commands.

InternalProcessControl DataProcess : The control functions of the MODIS processor.

L-1A_Outline DataStoreRecord VerifyInitializeStructure GranuleStructure : The definitive parameters of the memory requirements for the elements of the data product granule structure (template); the result of a request for operating system services. The structure data area is initialized to a 'bad data' indication.

Level0Data DataFlow DADS VerifyPacketIdentity : Full level 0 data as generated by the instrument with and CDOS items included and includes instrument engineering and instrument science data.

LogEntry Message InternalProcessControl ProcessingLog : Blow by blow of processing status, time sequential.

MemoryAllocation ExternalEntity : Operating system function that allocates task memory.

Metadata DataStoreRecord TransmitRecord Metastore : Information derived from data sets that provides an understanding of the content or utility of that data set. Updated constantly.

MetadataOutline DataStoreRecord VerifyInitializeStructure Metadata : The definitive parameters of the memory requirements for the elements of the metadata structure; the result of a request for operating system services. The structure metadata is initialized to a 'bad data' indication.

MetaStore DataStore : Processor memory allocated for the placement of the metadata items. Initialized to 'invalid'.

Mode DataStore : Internal store of the mode value (type of processing: standard, reprocessing, quick-look) and other parameters (number of packets, time interval).

ModeSpecification ControlFlow SetupProcessingMode Mode : The type of the processing mode.

OutputProducts DataFlow TransmitRecords PMS : Generated product or pointers to products, where products may consist of: level-1A instrument data with header and data quality information, level-1A metadata, and/or quick-look product.

PackedScan DataFlow AppendInstrumentAnomalyFlags AppendInstrumentAndS/C_Data : A RawScanCube with instrument anomaly appended.

Packet DataFlow AllocateL-1A_Structure CheckInstrumentStatusBits : (Alias for Prepacket) Sanitized packet.

PMS : Product Management Service

Pointers DataFlow PostRecords TransmitRecords : Location and size of the data structures; assess completeness of structure.

PostRecords DataProcess : Place scan cube into the data granule, accounting for time-ordering of non-time-ordered quick-look data.

Prepacket DataFlow VerifyPacketIdentity AllocateL-1A_Structure : (Alias for Packet) Sanitized Packet.

ProcessingLog ControlStore : Log of processing status records, time sequential.

ProcessingStatusInformation Message InternalProcessControl SCA : Information reqarding the fault conditions and processing performance of the data processing system. Status or completion information from the MODIS process to the SCA.

Quick-LookTimeOrdering DataProcess : The translation from physical memory to logical memory and vs that will time order the outputting of the granule while maintaining maximum throughput. Hopefully it can be shown that the localization of data packets in time will allow a tri-buffer or similar scheme to be optimal.

RawScanCube DataFlow VerifyInitializeStructure CreateProduct : Packet data that has been placed into a cube type record containing across track pixels on the x-axis, along track pixels on the y-axis, and wavelength on the z-axis.

RequestDynamicStatus ControlFlow DecomposeControlMessage PostRecords : Interrogate the process to inturn determine the present processing accounting. The process is not stopped.

S/C_AncillaryData GetS/C_AncillaryData FormatScanCube DataFlow : Platform ephemeris and attitude data within the time neighborhood of the requested time. I.e. the ten time-tagged platform ancillary records surrounding the scan cube; asynchronous with the scan cube, not interpolated.

SCA : Scheduling, Control, and Accounting

ScanCounter DataFlow ScanLineCount DetermineScanCounter : An indication of the scan cube completeness.

ScanCube DataFlow FormatScanCube PortRecords : Cube of scan oriented data formatted to the data product specification.

ScanCubeData DataStoreRecord PostRecords GranuleStructure : Cube of MODIS data without header, instrument status, and S/C ancillary appended.

ScanCubeOrdered DataFlow Quick-LookTimeOrdering DetermineRecordNumber : A indication of progress of the processing.

ScanLineCount DataStore : Scan cube accounting values.

ScanRecord DataFlow DetermineRecordNumber StuffDataCubeIntoGranule : Determine the location of this cube of data within the granule structure.

SetupProcessingMode ControlTransformation : Determines the mode of processing (quick-look, reprocessing, normal, etc). Passes the structure parameters to the InitializeStructures process. Sets up and determines the initial contents of the Mode data store. Generates a log entry to the ProcessingLog.

Start ControlFlow SetupProcessingMode VerifyInitializeStructure : Ask the DADS for the first packet and continue accepting packets until told to stop.

StartProcess ControlFlow DecomposeControlMessage SetupProcessingMode : Initialize MODIS processing.

StartStop ControlMessage InternalProcessControl VerifyInitializeStructure : Starts the process with initialization parameters, panic stop executing.

StopGracefully ControlFlow DecomposeControlMessage TransmitRecords

: (element of Finished) Terminate processing, while posting the granule as it is completed at this time, to the PMS.

StructureLocation DataFlow AllocateL-1A_Structure MemoryAllocation : Request for system memory allocation and a return of the specifiers for that allocation.

StuffDataCubeIntoGranule DataProcess : place the data cube logically into a granule position.

TaggedScan DataFlow AppendIstrumentAndS/C_Data DeterminaScanCounter : A scan cube with S/C and instrument data appended.

TerminationIndication TransmitRecords DeriveStatus : The granule has been transmitted and accepted by PMS, now cleanup, write progress report, and go home.

Time DataFlow UTC VerifyInitializeStructure : True universal coordinated time.

TransmitRecords DataProcess : Access product completeness, find location within structure of product components, transmit properly sequenced components to the PMS. (This can be either the actual data or pointers to the data - TBD)

TransmitSequencedRecords DataProcess : Extract the data from the physical memory and transmit those records in a time sequenced manner.

UTC : Universal Time Coordinated

VerifyPacketIdentity DataProcess : Perform packet sanity checks (packet size, CRC, or other), instrument ID check, other sanity checks.

Appendix

NOTES: The form of this DDD is generated to allow computer sorting of the fields for the Name, Type, From, and To specifiers. Each item is to be processed as a record (no hard c/r's) with white space used as delimiters. Field items must be one character string delimited by white space. These fields are case sensitive, beware!

Name:= formal name of the item, e.g., CDOS-input-data

Type:= item type and heirarchy, examples: data-store-element, alias:nnn, schedule-parameterss-composite

From:= flow originating from this segment, example: PGS

To:= flow being passed to this segment.

Description:= full description of the item including: response times, organization, access restrictions and purpose. Is it sequential, repeated, one-of-many, or a combination. Include forward and backward pointers to other hierarchies in a composite item, and cross referenced.

Assumptions and concerns are documented in separate documents that are to be referenced (pointers) by this document where appropriate.

Justification for the existence of structure items will be referenced to the appropriate requirements document or otherwise justified by reference if at all posible.

This file is located on: \EasyCASE\ModIS\DD.wp

ASSUMPTIONS LIST FOR LEVEL 0 OF MODIS LEVEL-1A PROCESSING SYSTEM

1. Data will be broken out and stored as granules with a granule header. These granules are larger than the scan cube but no larger than an orbit.

Justification: Many data processing activities are facilitated by the creation of data granules of reasonable size -- memory and storage can allocated, and processing software is easier to write and handle. Metadata, a required output product, will be in granule format in order to describe a coherent part of the data. So granules must be created at Level-1A anyway. Reasonably-sized granules also facilitate the recovery of data quality information, particularly the important data completeness and existence parameters. Such granules have been used for many satellite sensors, with apparent success. Finally, and perhaps more importantly, reasonably-sized granules are convenient, both in the data system design but also to users, who are adjusted to operating with coherent sets of data.

2. MODIS processing should provide a running comparison between instrument operating status as reflected in the received data stream and expected instrument status as determined from the instrument command history. If a discrepancy is identified, MODIS processing should provide an alert to instrument operations personnel at the ICC and instrument monitoring personnel at the MCST.

Justification: If a discrepancy exists, an investigation should be initiated to determine the source of the apparent error and the factual mode of instrument operation during the disputed period. Data users will need the factual information to interpret data. Instrument operations personnel may be able to use the information to modify control procedures and avoid future recurrences of the problem.

3. MODIS Quick Look data may require time ordering, redundancy elimination, and other quality control measures not required for routine MODIS data.

Justification: Current best thinking is that CDOS may not be able to provide routine processing services that meet Quick Look data timeliness constraints.