

MODIS

SCIENCE DATA SUPPORT TEAM

PRESENTATION

June 19, 1992

AGENDA

	<i>Page</i>
1. Action Items	1
2. MODIS Level-2 Design Status	2
3. MODIS Level-1 Project Tasking	10
4. SDST Time Management	16

ACTION ITEMS:

06/05/92 [Lloyd Carpenter] Update the Team Leader's Software and Data Management Plan. (An updated version was included in the handout and discussed at the meeting on 06/12/92.) STATUS: Open. Due Date: 07/10/92

06/05/92 [Lloyd Carpenter] Update the Team Leader's Science Computing Facility Plan. STATUS: Open. Due Date: 07/10/92

04/24/92 [J. J. Pan] Develop a detailed schedule for a typical algorithm integration into the Level-2 processing shell. (A draft task list and schedule were presented on 06/05/92.) 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

06/12/92 [Tom Goff] Develop separate detailed schedules using Microsoft Project for Level-1A and -1B software design and development. (Preliminary results are included in the handout.) STATUS: Open. Due Date: 07/10/92

04/24/92 [Lloyd Carpenter] Develop a system for collecting time management data for the SDST effort. (A strawman format is included in the handout.) STATUS: Open. Due Date: 06/26/92

DRAFT

The Log of the MODIS Level-2 Processing Shell Design

J. J. Pan
Research and Data Systems Corp.
(301) 982-3700

Date: June 12 - June 18, 1992

- 6/12 1. Discussed the preliminary atmospheric data processing diagrams in the SDST meeting. Had a meeting to discuss the Local Network Requirements.
 2. Checked the updated algorithms information which provided by STX.
- 6/15 3. Called Hyo-Duck Chang (STX) to request the detailed information of all 52 algorithms. He said it should be ready in two days.
 4. Started to design a "toolkit" which can be used to reduce the burden of manual check of the dependency of Input/Algorithms/Output. Also, the mistakes resulted from manual check are very difficult to detect. Here is a rough estimation of time used in updating the diagram each time:

Time (manually)	Time (use "Toolkit")
5 - 8 hours	Less than 1 hour

Some additional time is required to draw the diagram using Microsoft Word.

Basically, the toolkit will have the following functions:

- (1). Check that if one data product is generated by more than one algorithm.
- (2). Check the discrepancy between input and output data.

Example:

Algorithm ID	Input Data ID	Output Data ID
100	10	20
101	20	10

- (3). Specify the dependency of algorithms.

Example:

Data ID	Generated from Algorithm	Input to Algorithm
20	100	101
30	101	102

- (4). List the input data which will be used by more than one algorithm.

- 6/16 5. Went to STX to get the following updated lists:

- (1). A list of data products (sorted by data product ID).
- (2). A list of algorithms (sorted by algorithm ID).

DRAFT

A list of input data, which will include the data ID, data name, data type, time frame, source instrument or platform, DAAC, and other information, is not available at this moment. But Dr. Brain Krupp (STX) would like to check how to generate a list of input data from the database.

6. Started to develop the "toolkit". It is written in FORTRAN because I have not installed the C compiler on my PC yet.

- 6/17-6/18
7. Finished the "toolkit" development (except the 4th function) and tested its functions. The output provided a lot of useful information for the Level-2 processing shell design. The source code will be modified again to improve its "qualities" if this source code will be distributed to other users in the future.
 8. The "Toolkit" detected the discrepancy of Data Product 2126 (Cloud Emissivity) which appeared in both input and output data descriptions in Algorithm 533 (Cloud Effective Emissivity). Informed Dr. Krupp this question, he checked the original documents and found the Data 2126 is the output only.
 9. More tests are required to make sure that output information is correct.
 10. Currently the correlative data is not included in the test.
 11. Started to learn Mac Project.

```

1 c=====
2 c Program : ALGOCHK.FOR
3 c Purpose : Used to manipulate the algorithms, input data,
4 c and output data.
5 c
6 c Programmer: J. J. Pan
7 c Research and Data Systems Corp.
8 c (301) 982-3700
9 c
10 c Date : 6/15/92
11 c
12 c Subroutines Called:
13 c     1. CHKOUT : Check if one data comes from more
14 c             than one algorithm.
15 c     2. DISCREP: Check the discrepancy between input
16 c             and output data.
17 c     3. DEPEND : Specify the dependency of algorithms
18 c     4. STORAGE: List the input data which will be used
19 c             by more than one algorithm.
20 c
21 c Variables:
22 c     Maxalgo : Maximum no. of algorithms
23 c     Maxin  : Maximum no. of input data for each algorithm
24 c     Maxout : Maximum no. of output data for each algorithm
25 c     Algor  : Array used to store the algorithm's ID
26 c     Intab  : Table used to store Input Data ID of each algorithm
27 c     Outtab : Table used to store Output Data ID of each algorithm
28 c     Innum  : Array used to store No. of input data of each
29 c             algorithm
30 c     Outnum : Array used to store No. of output data of each
31 c             algorithm
32 c     Nalgo   : No. of Algorithms
33 c
34 c Input/Output Files:
35 c     Input File Name: ALGOCHK.DAT
36 c     Output File Name: ALGOCHK.OUT
37 c
38 c =====
39 c
40 c
41 c Parameter (Maxalgo=80, Maxin=40, Maxout=20)
42 c Integer Algor(Maxalgo)
43 c Integer Intab(Maxalgo,Maxin)
44 c Integer Outtab(Maxalgo,Maxout)
45 c Integer Innum(Maxalgo), Outnum(Maxalgo)
46 c
47 c Open(unit=5,file='ALGOCHK.DAT',status='old')
48 c Open(unit=6,file='ALGOCHK.OUT',status='unknown')
49 c
50 c
51 c     write(*,10)
52 c     10 format(' Enter the no. of algorithms:')
53 c     read(*,*) Nalgo
54 c
55 c     do 20 i=1,Nalgo
56 c     read(5,*), Algor(i), Innum(i), Outnum(i)
57 c
58 c     if (Innum(i) .eq. 0) then
59 c         write(6,50) Algor(i)
60 c         50 format(' ATTENTION: No required (EOS) input data for'
61 c                 '$ ' Algorithm ',14')
62 c     else
63 c         read(5,*), (Intab(i,j),j=1,Innum(i))
64 c     end if
65 c
66 c     if (Outnum(i) .eq. 0) then

```

```

67 c         write(6,55) Algor(i)
68 c         55 format(' ATTENTION: No data product from Algorithm ',14)
69 c     else
70 c         read(5,*), (Outtab(i,j),j=1,Outnum(i))
71 c     end if
72 c
73 c         write(*,*), Algor(i), Innum(i), Outnum(i)
74 c         write(*,*), (Intab(i,j),j=1,Innum(i))
75 c         write(*,*), (Outtab(i,j),j=1,Outnum(i))
76 c     20 continue
77 c
78 c
79 c-----Function 1: CHKOUT
80 c     Check if one data product will be generated by more than
81 c     one algorithm.
82 c     Call Chkout(Nalgo,Algor,Outnum,Outtab)
83 c
84 c
85 c-----Function 2: DISCREP
86 c     Check the discrepancy between input and output data
87 c     Call Discrep(Nalgo,Algor,Innum,Outnum,Intab,Outtab)
88 c
89 c-----Function 3: DEPEND
90 c     Specify the dependency of algorithms
91 c     Call Depend(Nalgo,Algor,Innum,Outnum,Intab,Outtab)
92 c
93 c-----Function 4: STORAGE
94 c     Specify the input data which will be stored in a "common"
95 c     block
96 c     Call Storage(Nalgo,Algor,Innum,Outnum,Intab,Outtab)
97 c
98 c
99 c
100 c Stop
101 c End
102 c
103 c
104 c
105 c
106 c
107 c
108 c
109 c=====Subroutine Chkout(Nalgo,Algor,Outnum,Outtab)
110 c
111 c Parameter (Maxalgo=80, Maxin=40, Maxout=20)
112 c Integer Algor(Maxalgo)
113 c Integer Outtab(Maxalgo,Maxout)
114 c Integer Outnum(Maxalgo)
115 c
116 c
117 c     write(6,90)
118 c     90 format(ix)
119 c
120 c     do 10 i=1,Nalgo
121 c
122 c         if (Outnum(i) .gt. 1) then
123 c             do 20 j=1,Outnum(i)
124 c                 ID=Outtab(i,j)
125 c
126 c-----compare within the same algorithm
127 c                 do 25 jj=j+1,Outnum(i)
128 c                     if (ID .eq. Outtab(i,jj))
129 c                         $           Write(6,95) ID, Algor(i)
130 c                         $           format(' ERROR (Duplication): Data product ',14,
131 c                               '$ ' is duplicated in Algor.',14)
132 c                         25 continue

```

```

133 c-----compare with other algorithms
134      do 30 ii=1,Nalgo
135         if (i .ne. ii .and. Outnum(ii) .ne. 0) then
136             do 40 jj=1,Outnum(ii)
137                if (ID .eq. Outtab(ii,jj))
138                   Write(6,99) ID, Algor(ii), Algor(i)
139                   99          format(' ERROR (Duplication): Data product ',14,
140                   $           ' is generated from Algor.: ',214)
141                   40          continue
142                   end if
143             30          continue
144             20          continue
145             end if
146         10          continue
147
148         Return
149
150     End
151
152
153 c=====
154 Subroutine Discrep(Nalgo,Algor,Innum,Outnum,Intab,Outtab)
155
156 Parameter (Maxalgo=80, Maxin=40, Maxout=20)
157 Integer Algor(Maxalgo)
158 Integer Intab(Maxalgo,Maxin)
159 Integer Outtab(Maxalgo,Maxout)
160 Integer Innum(Maxalgo), Outnum(Maxalgo)
161
162 write(6,90)
163 90 format(1x)
164
165      do 10 i=1,Nalgo
166
167 c-----compare input and output in the same algorithm
168      if (Innum(i) .ne. 0) then
169          do 20 j=1,innum(i)
170             ID=Intab(i,j)
171             if (Outnum(i) .ne. 0) then
172                 do 25 k=1,Outnum(i)
173                    if (ID .eq. Outtab(i,k)) Write(6,95) ID, Algor(i)
174                    95          format(' ERROR (Discrepancy): Data ',14,
175                    $           ' is input and output in Algor.: ',14)
176                    25          continue
177             end if
178             20          continue
179         end if
180
181         10          continue
182
183         Return
184
185     End
186
187
188 c=====
189 Subroutine Depend(Nalgo,Algor,Innum,Outnum,Intab,Outtab)
190
191 Parameter (Maxalgo=80, Maxin=40, Maxout=20)
192 Integer Algor(Maxalgo)
193 Integer Intab(Maxalgo,Maxin)
194 Integer Outtab(Maxalgo,Maxout)
195 Integer Innum(Maxalgo), Outnum(Maxalgo)
196
197 write(6,90)

```

```

199      90 format(1x)
200
201      write(6,93)
202      93 format(' Dependency:----',/
203      $           ' Data ID Generated from      Input_to', /
204      $           ' Algorithm      Algorithm', /
205      $           ' -----')
206
207      do 10 i=1,Nalgo
208
209 c-----compare input and output in the different algorithms
210 c-----and find the order of processing
211
212 c-----iflag used to control output format
213      Iflag=0
214      if (Innum(i) .ne. 0) then
215          do 20 j=1,Innum(i)
216             ID=Intab(i,j)
217             do 25 ii=1,Nalgo
218                if (Outnum(ii) .ne. 0) then
219                  do 30 jj=1,Outnum(ii)
220                     if (ID .eq. Outtab(ii,jj)) then
221                        Iflag=1
222                        Write(6,95) ID, Algor(ii), Algor(i)
223                        95          format(4x,14,9x,14,4x,' ----> ',14)
224                     end if
225             30          continue
226             end if
227             25          continue
228         20          continue
229     end if
230     if (Iflag .eq. 1) write(6,90)
231
232     10          continue
233
234     Return
235
236
237
238 c=====
239 Subroutine Storage(Nalgo,Algor,Innum,Outnum,Intab,Outtab)
240
241 Parameter (Maxalgo=80, Maxin=40, Maxout=20)
242 Integer Algor(Maxalgo)
243 Integer Intab(Maxalgo,Maxin)
244 Integer Outtab(Maxalgo,Maxout)
245 Integer Innum(Maxalgo), Outnum(Maxalgo)
246
247 write(6,90)
248 90 format(1x)
249
250
251
252     Return
253

```

```
*****
*          *
*   The input file ALGOCHK.DAT          *
*          *
*****  
511 3 2          (Algo. ID, # of input, # of output)  
2339 2392 51          (ID of each input data)  
1688 2254          (ID of each output data)  
512 7 8  
2339 2416 2417 18 32 42 101  
2555 2556 2557 2558 2559 2560 3216 3217  
513 9 3  
1588 2282 2283 2284 2339 2481 2523 44 75  
1874 3321 3322  
514 6 1  
1529 1781 2116 2312 2339 2466  
2094  
515 11 4  
1332 1333 1680 1735 2338 2339 2392 3594 23  
50 51  
2580 2581 3662 3663  
516 1 8  
2339  
2573 2574 2593 2594 3317 3318 3319 3320  
517 9 2  
2429 2430 2431 2434 112 113 114 115 847  
2424 2425  
518 5 4  
2088 2338 2339 2392 2828  
2429 2430 2431 2434  
.....
```

```
*****
*                                     *
* The output file ALGOCHK.OUT.      *
* -- This output is for demo. purpose only. *
*****
```

ATTENTION: No data product from Algorithm 522

ATTENTION: No required (EOS) input data for Algorithm 550

ERROR (Duplication): Data product 2127 is generated from Algor.: 534 533

ERROR (Discrepancy): Data 2126 is input and output in Algor.: 533

Dependency:----

Data ID	Generated from Algorithm	Input to Algorithm
2416	545	----> 512
2417	545	----> 512
1529	534	----> 514
1781	536	----> 514
2312	536	----> 514
2466	534	----> 514
1333	523	----> 515
2429	518	----> 517
2430	518	----> 517
2431	518	----> 517
2434	518	----> 517
2379	560	----> 520
2429	518	----> 520
2484	538	----> 520
2751	537	----> 520
3021	541	----> 520
1334	523	----> 524
1874	513	----> 524
2003	527	----> 524
2750	537	----> 524
2293	524	----> 525
2003	527	----> 526
2293	524	----> 526
2001	806	----> 527
1529	534	----> 533
2126	533	----> 533
2126	533	----> 534
1764	535	----> 536
1765	535	----> 536
2015	540	----> 536
2016	540	----> 536
1333	523	----> 540
1875	530	----> 540
2750	537	----> 540

2254	511	----> 545
2416	545	----> 547
2417	545	----> 547
2416	545	----> 548
2417	545	----> 548
2416	545	----> 549
2417	545	----> 549
2416	545	----> 551
2417	545	----> 551
2416	545	----> 553
2417	545	----> 553
1333	523	----> 557
2295	554	----> 558
2424	517	----> 806
2425	517	----> 806
2429	518	----> 806
2430	518	----> 806
2431	518	----> 806
2434	518	----> 806
2424	517	----> 807
2425	517	----> 807
2571	547	----> 914
2572	547	----> 914
2575	546	----> 914
2576	546	----> 914
2266	558	----> 916
2267	558	----> 916

```
algo_id algo_name
511    Glint Field
512    Backscattering Coefficients
513    Water Vapor
514    Cloud Joint Probability Density Function
515    Dissolved Organic Matter
516    Chlorophyll Fluorescence Line Curvature
517    Bidirectional Reflectance (BRDF)
518    Directional Reflectance
519    Photosynthesis: MODIS and EOSP Polarization
520    Land Cover and Land Cover Change
521    Canopy Carbon and Water Fluxes
522    Cloud Detection
523    Total Column Ozone
524    Aerosol Optical Depth
525    Aerosol Size Distribution
526    Aerosol Mass Loading
527    Aerosol Single Scattering Albedo
528    Lifted Index
530    Total Precipitable Water
531    Cloud Fractional Area
532    Cloud Area and Perimeter
533    Cloud Effective Emissivity
534    Cloud Top Pressure and Temperature
535    Cloud Water Thermodynamic Phase
536    Cloud Droplet Effective Radius & Optical Thickness
537    Vegetation and Soil Indices
538    Surface Temperature
539    Thermal Anomalies
540    Spectral Surface Reflectance
541    Snow Cover
543    Sea Surface Temperatures
544    Sea Ice
545    Water Leaving Radiance
546    Chlorophyll Fluorescence Line Height
547    Chlorophyll-A Pigment Concentration
548    Case-II Waters Chlorophyll-A Concentration
49    Detached Coccolith Concentration
50    Photosynthetically Active Radiation
551    Diffuse Attenuation at 490 nm
552    Attenuation of PAR
553    Surface Primary Productivity via Fluorescence
554    Angstrom Exponent
555    Single Scattering Aerosol Radiation
557    Pigment Concentration
558    Incident PAR
559    Dissolved Organic Matter Concentration
560    Land Leaving Radiance
561    Organic Matter & Suspended Solids Concentration
806    Hemispheric Spectral Albedo
807    Surface Roughness
808    Georadiance Corrections
914    Chlorophyll Fluorescence Efficiency
916    Chlorophyll Concentration via Fluorescence
```

Table 3. Data Products from Level-2 Processing

<u>Prod #</u>	<u>Prod Name</u>	<u>Investigator</u>	<u>Instrument</u>	<u>Platform</u>	<u>DAAC</u>
1017	Acrosol Mass.	Kaufman	MODIS*	AM,FM	GSFC
1333	O3 Total Burd.	Menzel	MODIS*	AM,FM	GSFC

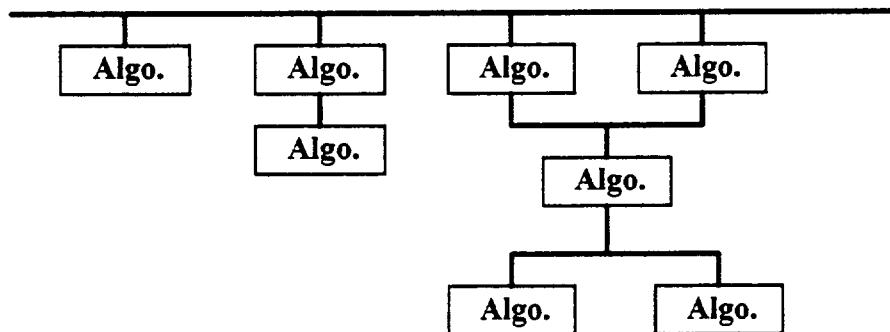
Table 2. Input Data for Level-2 Processing

<u>Input Prod #</u>	<u>Input Prod Name</u>	<u>Input Type</u>	<u>Time Frame</u>	<u>Source Inst</u>	<u>DAAC</u>
101	Integ. Ozone	Anc	AL	AIRS/AMSU	GSFC
/102	Aerosol Op. Dep.	Dev	BL	AVHRR	NESDIS

Table1. Algorithms in Level-2 Processing

<u>Algo. ID</u>	<u>Algo. Name</u>	<u>Investigator</u>	<u>Input Data #</u>	<u>Output Data #</u>
511	Glint Field	Gordon	2339,2392,51	1688,2254
512	Backscat. Coe.	Gordon	2339,2416,2417	2553,2556,.....

Sorted by
Number

Algorithms Dependency in Level-2 Processing

MODIS Level-1 Project Tasking
Thomas E. Goff
18 June, 1992

teg@cheshire.gsfc.nasa.gov,
(301) 982-3704
tgoff on GSFC mail

- **Microsoft Project on the Mac** - I have been entering the MODIS Level-1 task assignments into the Microsoft Project software on the Apple Macintosh computer and learning this software as I go along. This package has filter functions and sub-project capabilities that can be put to good use in the management of resources (staff) if the entire MODIS effort (by the SDST) is placed into one master file. This would allow the future adjustment of staff among various portions of the project. The power of this project management system can only be best utilized under a single project file.

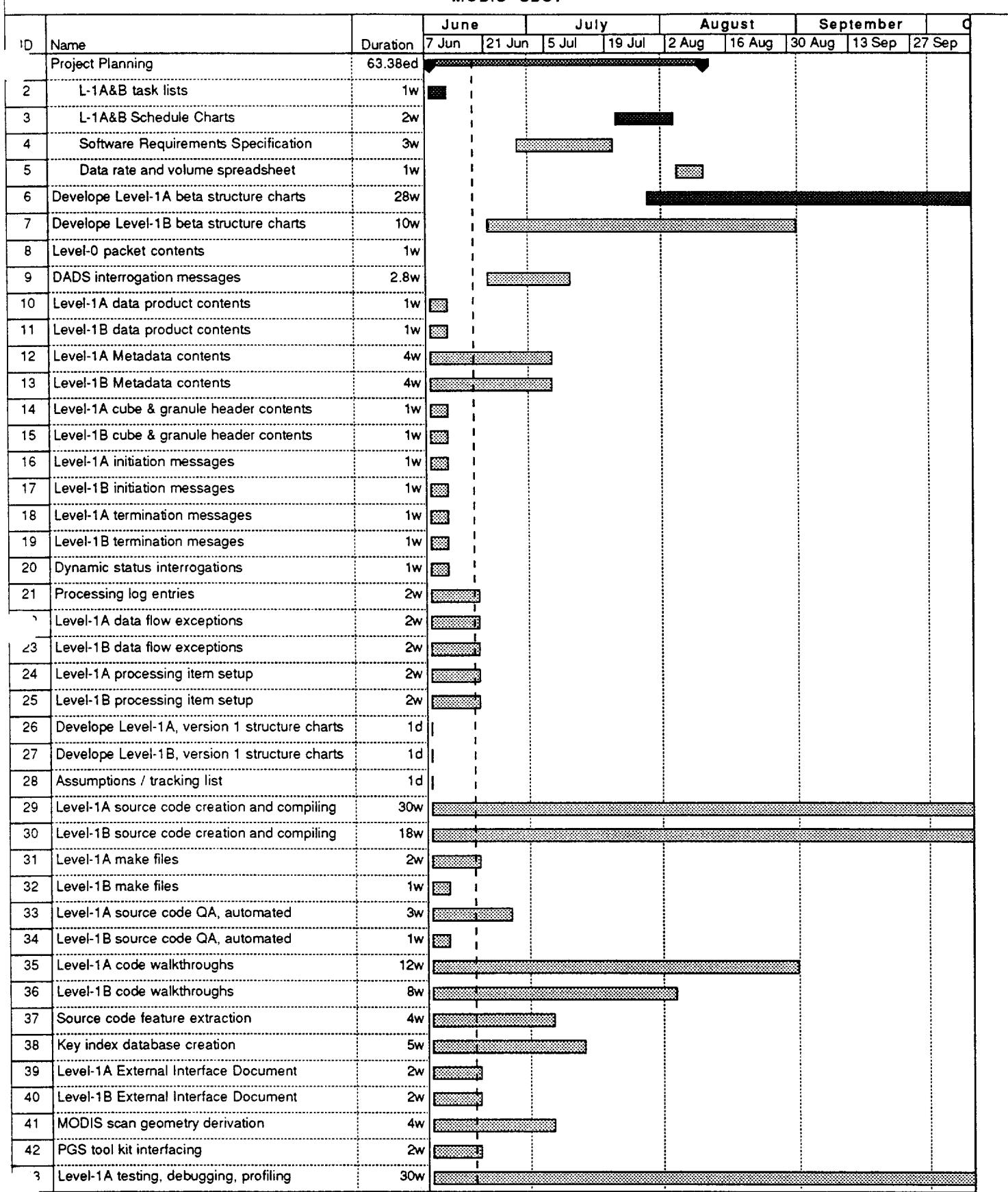
-- Miscellaneous --

- **X-Windows Connections** - Quarterdeck has just announced the release of their DesqView/X software for PCs. This allows a PC to be a pier in an X-Window based network. This means that a PC can be used as an X-terminal server to a UNIX based X-client, and conversly, that a UNIX GUI can be used as a window onto a real PC. Apple Macs can be included in this scheme as X-window servers or possible as X-Window clients. This capability would be highly desirable for the mixture of machines that the SDST is currently utilizing.
- **Word for Windows** - My PC has been purged of all files and is now running Microsoft Word for Windows 2.0 (WinWord) under Microsoft Windows 3.1.

Level-1A and -1B

ID	Name	Duration	Scheduled Start	Scheduled Finish	Predecessors	Resource Names
1	Project Planning	63.38ed	Mon 8 Jun, '92	Mon 10 Aug, '92		
2	L-1A&B task lists	1w	Mon 8 Jun, '92	Fri 12 Jun, '92		TEG,MS-WinWord
3	L-1A&B Schedule Charts	2w	Tue 21 Jul, '92	Mon 3 Aug, '92	2	TEG
4	Software Requirements Specification	3w	Sun 28 Jun, '92	Mon 20 Jul, '92	3	TEG
5	Data rate and volume spreadsheet	1w	Tue 4 Aug, '92	Mon 10 Aug, '92	4	TEG
6	Develop Level-1A beta structure charts	28w	Tue 28 Jul, '92	Fri 12 Feb, '93		
7	Develop Level-1B beta structure charts	10w	Sun 21 Jun, '92	Mon 31 Aug, '92		
8	Level-0 packet contents	1w	Mon 15 Feb, '93	Fri 19 Feb, '93	6	
9	DADS interrogation messages	2.8w	Sun 21 Jun, '92	Fri 10 Jul, '92		
10	Level-1A data product contents	1w	Mon 8 Jun, '92	Fri 12 Jun, '92		
11	Level-1B data product contents	1w	Mon 8 Jun, '92	Fri 12 Jun, '92		
12	Level-1A Metadata contents	4w	Mon 8 Jun, '92	Mon 6 Jul, '92		
13	Level-1B Metadata contents	4w	Mon 8 Jun, '92	Mon 6 Jul, '92		
14	Level-1A cube & granule header contents	1w	Mon 8 Jun, '92	Fri 12 Jun, '92		
15	Level-1B cube & granule header contents	1w	Mon 8 Jun, '92	Fri 12 Jun, '92		
16	Level-1A initiation messages	1w	Mon 8 Jun, '92	Fri 12 Jun, '92		
17	Level-1B initiation messages	1w	Mon 8 Jun, '92	Fri 12 Jun, '92		
18	Level-1A termination messages	1w	Mon 8 Jun, '92	Fri 12 Jun, '92		
19	Level-1B termination messages	1w	Mon 8 Jun, '92	Fri 12 Jun, '92		
20	Dynamic status interrogations	1w	Mon 8 Jun, '92	Fri 12 Jun, '92		
21	Processing log entries	2w	Mon 8 Jun, '92	Fri 19 Jun, '92		
22	Level-1A data flow exceptions	2w	Mon 8 Jun, '92	Fri 19 Jun, '92		
23	Level-1B data flow exceptions	2w	Mon 8 Jun, '92	Fri 19 Jun, '92		
24	Level-1A processing item setup	2w	Mon 8 Jun, '92	Fri 19 Jun, '92		
25	Level-1B processing item setup	2w	Mon 8 Jun, '92	Fri 19 Jun, '92		
26	Develop Level-1A, version 1 structure charts	1d	Mon 8 Jun, '92	Mon 8 Jun, '92		
27	Develop Level-1B, version 1 structure charts	1d	Mon 8 Jun, '92	Mon 8 Jun, '92		
28	Assumptions / tracking list	1d	Mon 8 Jun, '92	Mon 8 Jun, '92		
29	Level-1A source code creation and compiling	30w	Mon 8 Jun, '92	Fri 8 Jan, '93		
30	Level-1B source code creation and compiling	18w	Mon 8 Jun, '92	Tue 13 Oct, '92		
31	Level-1A make files	2w	Mon 8 Jun, '92	Fri 19 Jun, '92		
32	Level-1B make files	1w	Mon 8 Jun, '92	Fri 12 Jun, '92		
33	Level-1A source code QA, automated	3w	Mon 8 Jun, '92	Fri 26 Jun, '92		
34	Level-1B source code QA, automated	1w	Mon 8 Jun, '92	Fri 12 Jun, '92		
35	Level-1A code walkthroughs	12w	Mon 8 Jun, '92	Mon 31 Aug, '92		
36	Level-1B code walkthroughs	8w	Mon 8 Jun, '92	Mon 3 Aug, '92		
37	Source code feature extraction	4w	Mon 8 Jun, '92	Mon 6 Jul, '92		
38	Key index database creation	5w	Mon 8 Jun, '92	Mon 13 Jul, '92		
39	Level-1A External Interface Document	2w	Mon 8 Jun, '92	Fri 19 Jun, '92		
40	Level-1B External Interface Document	2w	Mon 8 Jun, '92	Fri 19 Jun, '92		
41	MODIS scan geometry derivation	4w	Mon 8 Jun, '92	Mon 6 Jul, '92		
42	PGS tool kit interfacing	2w	Mon 8 Jun, '92	Fri 19 Jun, '92		
43	Level-1A testing, debugging, profiling	30w	Mon 8 Jun, '92	Fri 8 Jan, '93		
44	Level-1B testing, debugging, profiling	24w	Mon 8 Jun, '92	Wed 25 Nov, '92		
45	Packet telemetry simulator	8w	Mon 8 Jun, '92	Mon 3 Aug, '92		
46	L-1A Data Product validator	8w	Mon 8 Jun, '92	Mon 3 Aug, '92		
47	L-1B Data Product validator	8w	Mon 8 Jun, '92	Mon 3 Aug, '92		
48	L-1A performance on TLCF	4w	Mon 8 Jun, '92	Mon 6 Jul, '92		
49	L-1B performance on TLCF	4w	Mon 8 Jun, '92	Mon 6 Jul, '92		
50	MODIS Executive Information Summary	3w	Mon 8 Jun, '92	Fri 26 Jun, '92		
51	L-1A Software Design Description	4w	Mon 8 Jun, '92	Mon 6 Jul, '92		
52	L-1B Software Design Description	3w	Mon 8 Jun, '92	Fri 26 Jun, '92		
53	L-1A Software V&V Plan	8w	Mon 8 Jun, '92	Mon 3 Aug, '92		
54	L-1B Software V&V Plan	6w	Mon 8 Jun, '92	Mon 20 Jul, '92		
55	L-1A Software Installation Guide	2w	Mon 8 Jun, '92	Fri 19 Jun, '92		
56	L-1B Software Installation Guide	2w	Mon 8 Jun, '92	Fri 19 Jun, '92		
57	L-1A Operations Guide	1w	Mon 8 Jun, '92	Fri 12 Jun, '92		
58	L-1B Operations Guide	1d	Mon 8 Jun, '92	Mon 8 Jun, '92		
59	L-1A Configuration management overhead	2w	Mon 8 Jun, '92	Fri 19 Jun, '92		
60	L-1B configuration management overhead	2w	Mon 8 Jun, '92	Fri 19 Jun, '92		
61	TLCF computer class	4w	Sun 21 Jun, '92	Mon 20 Jul, '92		
62	Cadre's Teamwork class	2w	Mon 8 Jun, '92	Fri 19 Jun, '92		
63	SoftBench usage class	2w	Mon 8 Jun, '92	Fri 19 Jun, '92		
64	Configuration management class	2w	Mon 8 Jun, '92	Fri 19 Jun, '92		
65	Programming QA class	2w	Mon 8 Jun, '92	Fri 19 Jun, '92		

MODIS Level-1A and -1B design
MODIS SDST



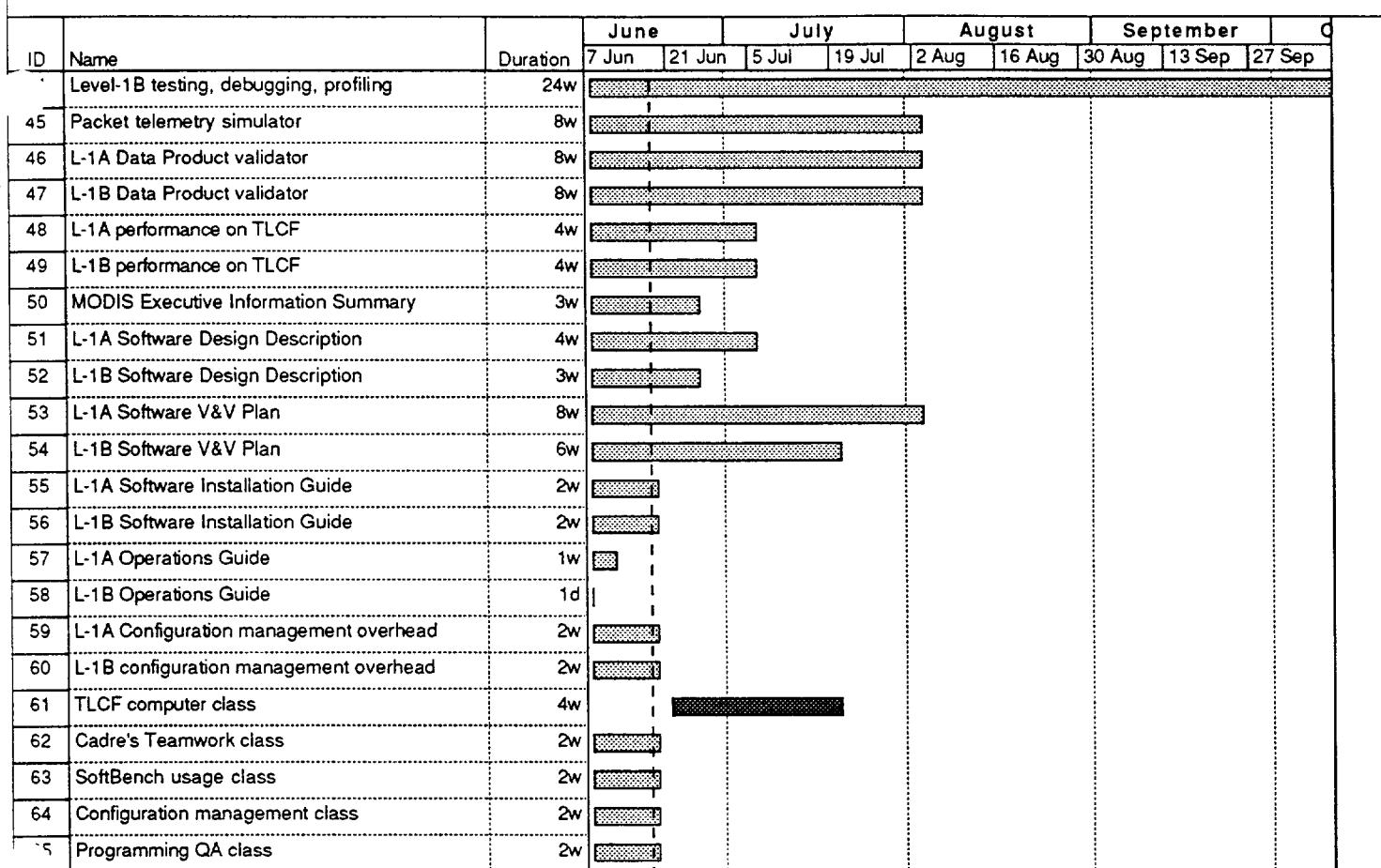
Project: MODIS Level-1A and -1B
Date: Thu 18 Jun, 92

Critical [■]
Noncritical [■■■■■]

Progress [■■■■■]
Milestone ▲

Summary [■■■■■]

MODIS Level-1A and -1B design
MODIS SDST



Project: MODIS Level-1A and -1B
Date: Thu 18 Jun, '92

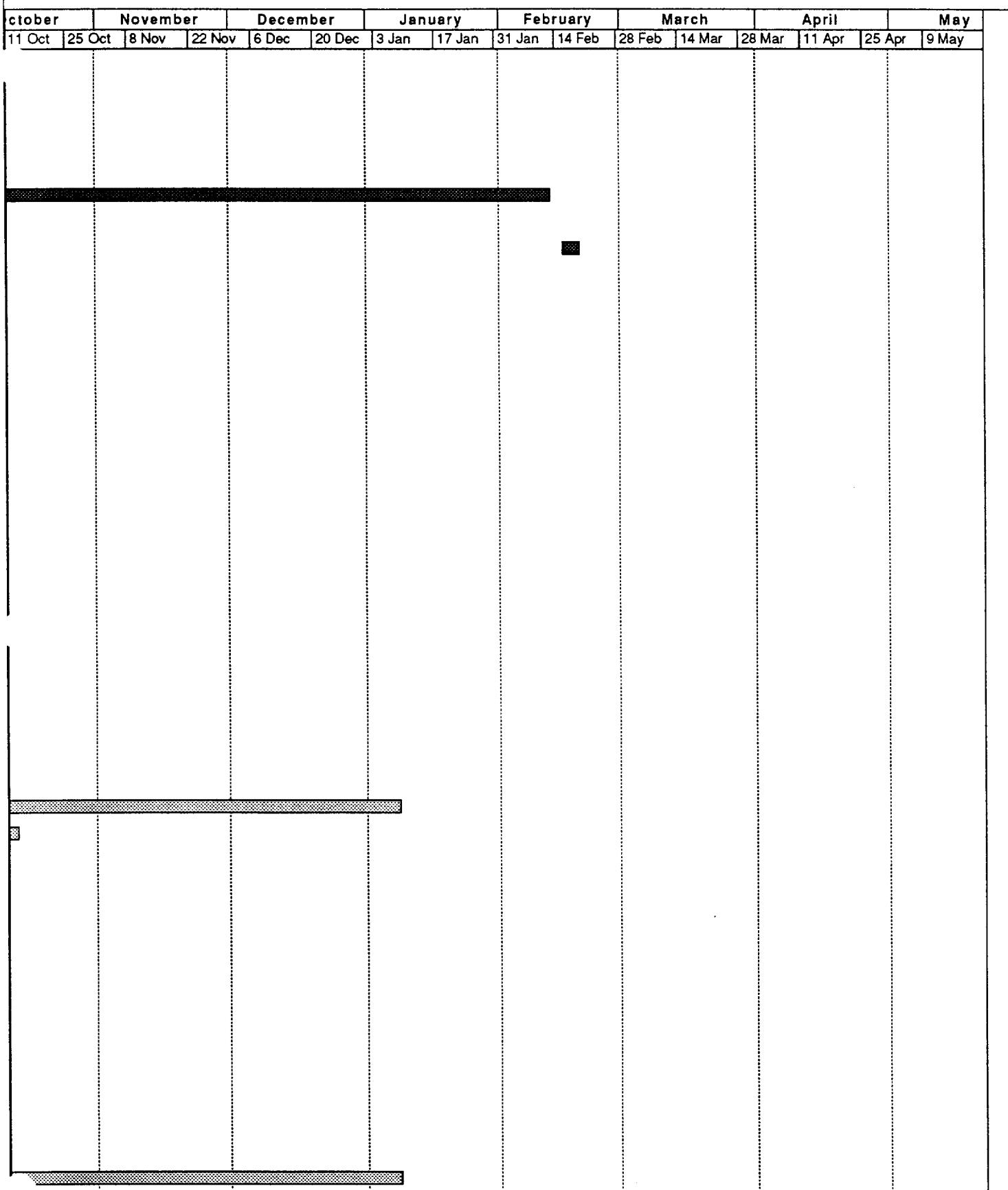
Critical
Noncritical

Progress

Milestone

Summary

**MODIS Level-1A and -1B design
MODIS SDST**



Project: MODIS Level-1A and -1B
Date: Thu 18 Jun, 92

Critical



Progress



Summary



Noncritical



Milestone



**MODIS Level-1A and -1B design
MODIS SDST**

October	November		December			January		February		March			April			May
11 Oct	25 Oct	8 Nov	22 Nov	6 Dec	20 Dec	3 Jan	17 Jan	31 Jan	14 Feb	28 Feb	14 Mar	28 Mar	11 Apr	25 Apr	9 May	

Project: MODIS Level-1A and -1B
Date: Thu 18 Jun, '92

Critical



Progress



Summary



Noncritical



Milestone

MODIS SDST Time Record

Name:	June 1992										July 1992													
	Mo 15	Tu 16	We 17	Th 18	Fr 19	Mo 22	Tu 23	We 24	Th 25	Fr 26	Mo 29	Tu 30	We 1	Th 2	Fr H	Mo 6	Tu 7	We 8	Th 9	Fr 10	Mo 13	Tu 14	We 15	Th 16
Level-1																								
Develop Requirements																								
Requirements Review																								
Develop Design																								
Design Review																								
Packet Simulation																								
Earth Location																								
Implement																								
Implementation Review																								
Iterate																								
Image Registration																								
Identify Approaches																								
Study Extensions For MODIS Case																								
Develop Ground Control Points																								
Prototype Implementation																								
Test With Real (AVHRR LAC?) Data																								
Review with Team																								
Accurate Ephem Data; NOAA,FDS,NORAD																								
Implement for EOSDIS																								
Level-2 Superstructure																								
Develop Requirements																								
Requirements Review																								
Develop Design																								
Design Review																								
Partial Implementation																								
Implementation Review																								
Iterate																								

DRAFT

MODIS SDST Time Record

Name:	June 1992										July 1992													
	Mo 15	Tu 16	We 17	Th 18	Fr 19	Mo 22	Tu 23	We 24	Th 25	Fr 26	Mo 29	Tu 30	We 1	Th 2	Fr H	Mo 6	Tu 7	We 8	Th 9	Fr 10	Mo 13	Tu 14	We 15	Th 16
Level-2, 3 and 4 Algorithms																								
Identify What We Will Get																								
Collect Existing Code																								
Examine Code, Provide Feedback																								
Convert Data Sets																								
Implement into MAS Prototype																								
Collect Version 1 PI Code																								
Start Integration into Deliverable																								
Quality Control and Validation Tools																								
Identify																								
Design																								
Review																								
Implement																								
Iterate with Science Team																								
Processing																								
Incorporate Improved Calibration																								
Develop a Metadata/Catalog																								
Extend to 50-Channel Version																								
Documentation																								
Produce Level-1B Data Sets																								
Field Support																								
Prototype MODIS with MAS																								
Identify Prototype Goals																								
Determine Processing Modules																								
Develop Initial Level-2 Structure																								
Collect Existing Software																								
Convert to Meet MODIS/EOS Standards																								
Integrate into Structure																								
Produce MAS Level-2 Products																								

DRAFT

MODIS SDST Time Record

Name:	June 1992										July 1992													
	Mo 15	Tu 16	We 17	Th 18	Fr 19	Mo 22	Tu 23	We 24	Th 25	Fr 26	Mo 29	Tu 30	We 1	Th 2	Fr H	Mo 6	Tu 7	We 8	Th 9	Fr 10	Mo 13	Tu 14	We 15	Th 16
Prototype MODIS with MAS (continued)																								
QC/Validate MAS Level-2, 3 Products																								
Archive, Catalog, and Distribute																								
MAS Data to Team																								
Iterate with Additional Functions																								
Simulation and System Tests																								
Utilities for Use with MAS or MODIS																								
Establish Utility/Subroutine Database																								
File Dump																								
Access																								
Selection																								
Browse																								
Display/Visualization/Graphing																								
Location																								
ta Conversion																								
Subsampling/Resampling																								
Lossy Compression																								
MetaData																								
TLCF																								
Requirements Definition																								
Develop Evolution Strategy																								
Team Leader Computing Facility Plan																								
Design																								
Procurement																								
Utilities																								
Input Data																								
From Other Instruments																								
From MODIS Tests																								
From Simulations																								
For Development																								

MODIS SDST Time Record

Name:	June 1992										July 1992													
	Mo 15	Tu 16	We 17	Th 18	Fr 19	Mo 22	Tu 23	We 24	Th 25	Fr 26	Mo 29	Tu 30	We 1	Th 2	Fr H	Mo 6	Tu 7	We 8	Th 9	Fr 10	Mo 13	Tu 14	We 15	Th 16
Input Data (continued)																								
For Development																								
Valid for Algorithm Examination																								
Defective for Algorithm Testing																								
For Validation/QC																								
Auxiliary for Processing																								
Output Formats																								
Establish Structures																								
Conversion Routines																								
Conventions																								
Tools																								
CASE																								
Configuration Management																								
Software Metrics																								
Inverse Engineering																								
Optimization																								
SDST/Science Team Guidelines																								
Data Format																								
Programming																								
Documentation																								
Structure																								
Naming Conventions																								
Glossary																								
Computerized Acronym List																								
DEM																								
Identify Approach																								
Calling Routines																								
Implement																								
Populate																								
Mapping Projections/Conversions																								

DRAFT

MODIS SDST Time Record

Name:	June 1992												July 1992												
	Mo	Tu	We	Th	Fr	Mo	Tu	We	Th	Fr	Mo	Tu	We	Th	Fr	Mo	Tu	We	Th	Fr	Mo	Tu	We	Th	Fr
15	16	17	18	19		22	23	24	25	26	29	30	1	2	H	6	7	8	9	10	13	14	15	16	17
Project Interaction																									
Software and Data Plan																									
EOS Data Working Group																									
Toolkit Evaluation																									
Operations Concepts																									
Identify Problems																									
Develop Proposed Solutions																									
EOS Standards Review																									
ECS Contract Reviews																									
Software Metrics																									
IMC Coordination																									
Cataloging Requirements																									
Metadata																									
Browse Products																									
DIS Processing Requirements																									
Systems Requirements for EOSDIS																									
Iterate Size Requirements																									
Documentation																									
Management																									
Project Plan																									
Tools (Scheduling/Task Leveling/EV)																									
Monitoring and Reporting																									
Documentation Control																									
Tracking List																									
Action Items																									
Total Quality Management																									
Identify Approaches																									
Select an Approach																									
Implement																									

DRAFT

