Producing Global Science Products for the Moderate Resolution Imaging Spectroradiometer (MODIS) in MODAPS

Ed Masuoka/NASA[1], Curt Tilmes/NASA, Dr. Gang Ye SAIC/GSC and Dr. Neal Devine SAIC/GSC

[1]Goddard Space Flight Center, Code 922, Greenbelt, MD 20771
(301) 614-5515 voice, (301) 614-5269 fax, Email: emasuoka@pop900.gsfc.nasa.gov

INTRODUCTION

The MODerate resolution Imaging Spectroradiometer (MODIS) was launched on NASA's EOS-Terra spacecraft in December 1999. With 36 spectral bands covering the visible, near wave and short wave infrared, MODIS produces over 40 global science data products, including sea surface temperature, ocean color, cloud properties, vegetation indices land surface temperature and land cover change[1].

The MODIS Data Processing System (MODAPS) produces 400GB/day of global MODIS science products from calibrated radiances generated in the Earth Observing System Data and Information System (EOSDIS.) The science products are shipped to the EOSDIS for archiving and distribution to the public. An additional 200GB of products are shipped each day to MODIS team members for quality assurance and validation of their products. In the sections that follow, we will describe the architecture of the MODAPS, identify processing bottlenecks encountered in scaling MODAPS from a 50GB/day backup system to a 400GB/day production system and discuss how these were handled.

MODAPS SOFTWARE

MODAPS processing software is made up of 5 subsystems, SIPS Ingest/Export, Product Generator, Archiver, MODIS Data Ordering System (MODDOS) and Reports, which interact via messages and three data stores (Product Storage, Archive Request Queue and Product Catalog) [2] Fig. 1.

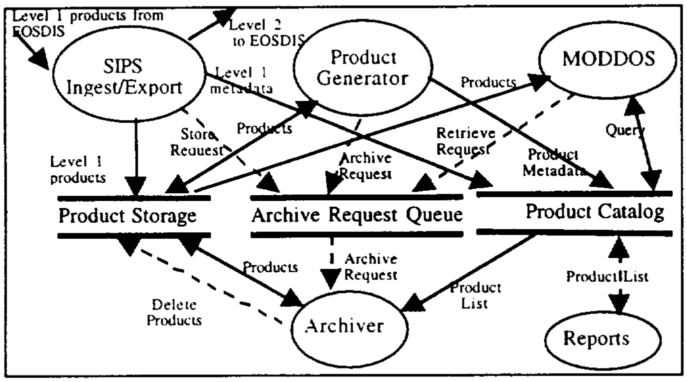

Fig. 1 MODAPS Data Flow Diagram

Production of MODIS science products begins with the receipt of delivery records from EOSDIS for MODIS Level 1B (calibrated radiances), Geolocation (earth location for the Level 1B) and ancillary data products. The SIPS Ingest component of MODAPS parses each delivery record and pulls the files listed in it from a file server used to buffer data transfers between MODSPS and the EOSDIS onto the MODAPS production disk. SIPS Ingest then extracts metadata fields needed for production from the products and stores the metadata in the Product Catalog.

All processing on MODAPS is controlled by a scheduler [3], which monitors and controls science processing, archiving and distribution jobs running on the production system. The scheduler has a component, PROSTAT, that monitors processes that are run at specific times or on a fixed interval, such as SIPS Ingest/Export, Loaders, Archive Controller and ftp pusher processes. Science data processing jobs which run only once for any given instance of a recipe are controlled by the Visual Database Cookbook (VDC) component of the scheduler, which monitors the execution of individual science processing streams in the system.

At regularly intervals Loaders run under PROSTAT control to initiate production of a suite of related science products such as land products in instrument swath format (Level 2) which require a common set of inputs. The loaders create production recipes for each job, load runtime parameters for specific production profiles (night versus day processing for example), submit requests to stage the input files for the job and insert the recipes in the processing database. A single loader generates tens to hundreds of processing recipes that wait in a queue for CPU resources to become available.

The makevdc process runs at intervals to determine which processing recipes have all their inputs online. If all inputs are present, a recipe's status in the database is changed from TBD to RECEIVED and the job is moved to the entrance directory. The Entrance daemon monitors this directory and when a CPU is available it will assign it to the recipe and insert the recipe in a processing streams table. The Master daemon monitors the streams table and initiates steps in a recipe by launching the perl script and executable pointed to by recipe steps. Master also allows operators to control the execution of recipes and if necessary to halt or disable steps in a recipe. When the last step in a recipe complete, the Exit daemon frees up the production resources allocated to it.

Each recipe consists of perl scripts and C or Fortran executables, which produce MODIS products. MODIS science software was originally designed to run in the EOSDIS where applications can only access system services through a suite of science data processing toolkit routines. These toolkit routines, which are used by all MODIS software are useful for writing products in HDF-EOS format and provide other key services, such reading records from a global DEM. Rather than removing the toolkit routines, we have used perl scripts to set-up environmental variables and generate the control structures needed to recreate the processing environment of the EOSDIS.

When a science job in the processing recipe completes, metadata fields for the products, which are to be archived in the EOSDIS are extracted from the product file and inserted in the MODAPS Product Catalog. Finally, the product files are pushed to the EOSDIS by SIPS Export.

The Archiver manages disk space and file migration from nearline tape silos in the MODAPS system. It compares the available on-line disk space with the lowest value of free space allowed. If free space falls below this value, then files, which have been stored on tape and are not required for production or product orders, are deleted from disk on a last in first out basis. The Archiver also monitors an archive request queue and handles the transfer of products between disk and the tape libraries. Legato Networker, a commercial disk backup package, handles the actual file stores and retrieves from the tape libraries.

Products are ordered from MODAPS through the Web-based MODIS Data Ordering System (MODDOS). A map of the world and selection box are implemented as JAVA applets to enable the user to select data products by dragging and resizing a selection box as well as by typing in the latitude and longitude of corner points of a bounding box. If the scientist ordering data through MODDOS has asked for files on DLT tape or via an ftp push to their computer system, a message confirming shipment is sent to them via email. If the scientist wishes to ftp the data to their system, then they are notified by email when the data sets are online and have 4 days to retrieve the files. In addition to orders entered through MODDOS, science team members have registered standing orders for ftp of files to their computing facilities. These standing orders total over 200GB/day of products.

Production status is available on the Web via "tic-tac-toe " charts, which show status of each file produced as a colored square (where green indicates successful completion and red job failure.) The charts are updated by querying a Product Catalog table in the MODAPS production database for the time period of interest.

MODAPS HARDWARE

In the current MODAPS, processing and distribution is handled by a central SGI Origin 2000 with 80 250mhz MIPS R10000 processors and 40GB of memory. The Origin is connected to local systems used for testing, quality assurance and generation of special products by Fibre Channel, switched 100Mbps and Gigabit Ethernet and FDDI networks. An Origin 2000 file server with 2TB of storage is connected to the MODAPS production system by 2 HiPPI channels and serves as a buffer for data transfers between MODAPS and the EOSDIS data centers. The configuration of the MODAPS production system and supporting systems use in science software development and testing is shown in Fig. 2.

Storage for data processing and distribution is provided by 10TB of Fibre Channel RAID. The RAID array is configured as a single RAID 3 file system with 240 50GB disk drives attached to 8 controllers each on a dedicated Fibre Channel loop connected to the Origin 2000. In theory each Fibre Channel loop could deliver data to the Origin at the rate of 100MB/sec.

Three Ampex 812 tape libraries, which together store up to 36TB of data products, hold the input files required for monthly, quarterly and yearly products and provide short term storage (12 days beyond a file's last use in production) to facilitate quality assurance activities. Each library has 3 DST tape drives which have a sustained transfer rate of 15MB/sec per drive.

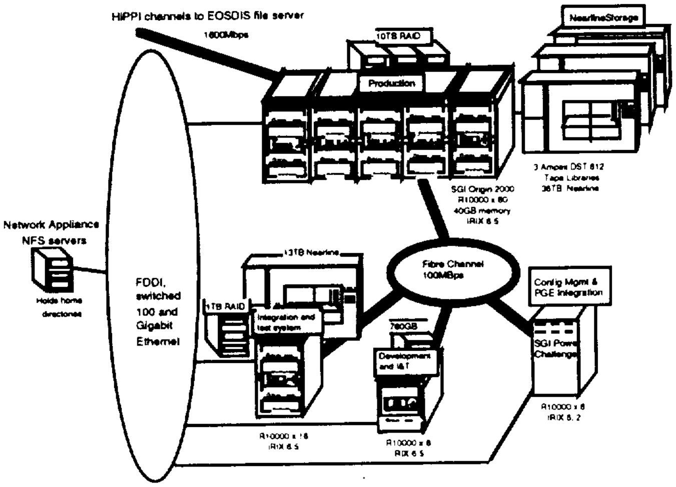

Fig. 2 MODAPS development, test and production systems

SYSTEM PERFORMANCE

In 1998, we ran benchmarks on MODIS science code on a Silicon Graphics Power Challenge with synthetic test data to estimate the I/O rates for each program. I/O rates ranged from 4MB-12MB/sec when single instances of each MODIS program were run on a system with no other processes competing for resources. Multiplying these rates by 79 (the number of processors used in production) yields an estimate of the I/O rates of science processes, 300MB-900MB/sec, which is independent of data staging and ftp processes.

During production with real MODIS data, I/O rates on MODAPS range from 100MB-1,500MB/sec. There are several reasons for the range of aggregate I/O performance observed in production. First, there are significant differences in I/O rates of the different MODIS science processes. Second, while science processes use HDF 4.5 to read and write records, which forces the use of buffered I/O, data transfer processes, which move files between disk and tape, use direct I/O to make better use of the striped file system.

For some science jobs the 40GB of system memory, masks the effect of the file system's upper bounds of 100MB/sec and 2,500IOPS (I/Os per second.) To take maximum advantage of the large system memory, we delay writing file system buffers to disk for a relatively long time. Nonetheless, during the execution of many jobs, like those that bin data for Level 3 products, the system spends up to half the time waiting for disk I/O to complete. On the current system, we've tried to mitigate this I/O bottleneck by mixing in jobs that do a lot of I/O in the kernel buffers with jobs that require more I/O from disk. In the latest version of MODAPS, the single large file system has been replaced by many smaller file systems (1 per controller.) This should remove the performance bottleneck of one file system being unable to service a larger number of concurrent I/Os and will also have the added benefit of limiting the amount of data lost in instances where there are file system failures.

The current production system uses a single table with a field for each of the required metadata fields. The database fields are many fewer than are actually in the products--- we extract only those fields that are needed in evaluating which files to stage in production and to support data search and ordering. Every product is represented by a single record in this table. In general, the simple approach has worked fine, but it is difficult to optimize the database for the wide variety of searches we need to support and the use of a single table has led to database deadlocks that can slow or halt production.

In the next delivery of the system, rather than a single table for all the metadata, the files are split into several tables based on the type of data: Geolocation, products in instrument swath format (Levels 1 and 2), gridded data sets (Levels 3 and 4), and ancillary products. Each of these tables has different fields based on the type of product that will be stored. Splitting the records among separate tables keeps the tables much smaller, both in total number of records per table, and also in number of bytes per record since only the fields relevant for the specific type of products are needed. Each record maps the metadata back to the unique FileId for the record in the File table. The smaller record size and table size makes searches faster, and we can also optimize the indices for each table based on the types of searches we anticipate for each type of table. For example, the table for "Tiled" data has a Tile ID index not found in other tables.

Other performance bottlenecks we encountered were in the GUI used to control processes and the production status charts we provide on the WWW. In the production monitoring GUI, each job step in every recipe has a separate widget, which allows operators to view its processing logs. In the original SeaWiFS system and in our first limited production tests, there was little impact on performance from tens of jobs completed per hour was relatively small. However, as hundreds of multi-step recipes completed per hour in MODIS production, the GUI took minutes to refresh, making it difficult to control or initiate processing. As a workaround, completed jobs are deleted from the database table that the GUI displays via a routinely run query. We also ran into difficulty with our production status page on the WWW. When a user selected a period of interest, the Product Catalog table was queried to update the production chart for that period. When many users ran production reports, they slowed down other queries in the Product Catalog, needed by production. This was avoided by creating status charts once a day and posting them on the Web as TIFF files.

REFERENCES

[1] Masuoka, E., A. Fleig, R.Wolfe and F. Patt, July 1998, "Key characteristics of MODIS data products", IEEE Transactions on Geoscience and Remote Sensing, Vol. 36, number 4, pp. 1313-1323.

[2] Read, S. and N. Devine, April 1999, MODIS Data Processing System (MODAPS) V0 System Description, document SDST-118 on the WWW at URL: http://ltpwww.gsfc.nasa.gov/MODIS/SDST/docs.html

[3] SeaWiFS home page is located at on the WWW at URL: http://seawifs.gsfc.nasa.gov/SEAWIFS.html