The Magnetospheric Multiscale (MMS) Mission Science Data Center: Technologies, Methods, and Experiences in Making Available Large Volumes of In-Situ Particle and Field Data

Chris Pankratz[1], Kim Kokkonen[1], Kristopher Larsen[1], Russell Panneton[1], Brian Putnam[1], Corey Schafer[1], Daniel Baker[1], and James Burch[2]
chris.pankratz@lasp.colorado.edu
1 - Laboratory for Atmospheric and Space Physics (LASP), University of Colorado, Boulder, Colorado, USA
2 - Southwest Research Institute, San Antonio, Texas, USA

http://lasp.colorado.edu/mms/sdc/

EGU2016-10183

Overview

The Magnetospheric MultiScale (MMS) mission consists of four satellites collecting data pertaining to magnetic reconnection and plasma phenomena in an elliptical earth orbit that precesses to probe various regions of physical interest over a period of ~2.5 years following launch in March 2015. Science operations for the mission are conducted at the Science Operations Center (SOC) at the Laboratory for Atmospheric and Space Physics, University of Colorado in Boulder, Colorado, USA. The MMS Science Data Center (SDC) is a component of the SOC responsible for the data production, management, dissemination, archiving, and visualization of the data from the extensive suite of 100 instruments onboard the four spacecraft. This poster provides an informatics-oriented view of the MMS SDC, summarizing its technical aspects, novel technologies and data management practices that are employed, experiences with its design and development, and lessons learned to date.

MMS Science Data Center Key Functions

- Routinely and automatically create science data products from telemetry, calibration data, and ancillary information (e.g. ephemeris)
- Manage all levels of data: version control, metadata, and archiving
- Disseminate data products to the MMS team, science community, and long-term archive
- Enable the science community to discover, access, and use the MMS data products
- Maximize the quality and quantity of data products
- Provide timely data status and release info to users
- Provide support to science community
- Maximize the quality and maintainability of software

MMS Data Volume

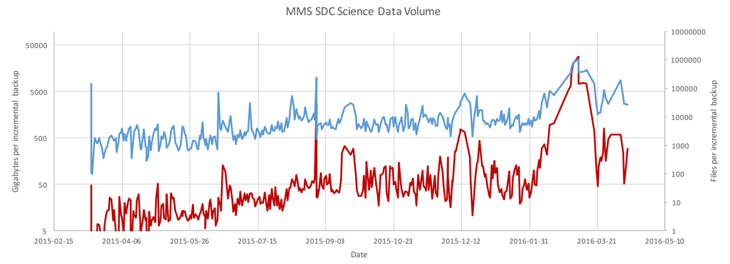
Total Managed Data Volume

- Following 6-months of nominal science operations, the MMS SDC is currently managing 21 terabytes (TB) of MMS science data
- Approximately 100 TB of managed data is expected by the end of the nominal MMS mission in August 2018

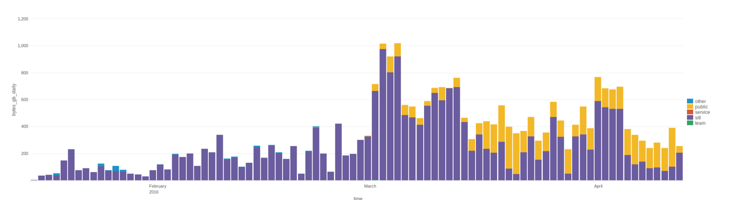
MMS SDC Current Data Volumes

Data Level / Type	Data Volume (GB)
Level 1	7319
Level 2	11735
QuickLook	1599
SITL	55

Data Creation and Dissemination Metrics



This chart shows science data files created by SDC processing or uploaded by team institutions. After routine science operations began on 2015-09-01, about 10,000 CDF files and 50 GB of data were generated per day. In preparation for the public data release on 2016-03-01, significant reprocessing took place and new level 2 products were also generated, which temporarily overwhelmed the backup system. In this ~1 month period about 15 TB of new data was generated. Currently, the daily generation rate is approximately 200 GB/day and 20,000 files/day.



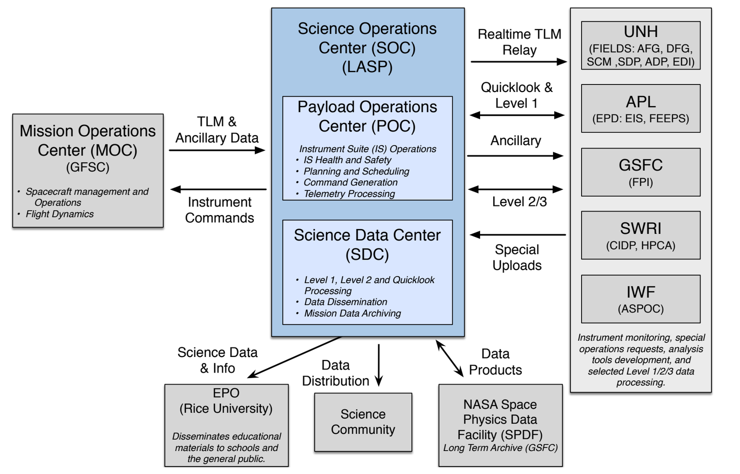
This chart shows gigabytes of science data downloaded daily since the beginning of 2016. Colors represent separate web areas primarily distinguished by mission team access (e.g. "sitl") vs. anonymous access ("public"). Peak downloads reached about 1 TB per day around the time of the first public release on 2016-03-01. Currently, access by the worldwide science community usually exceeds the team data access.

References

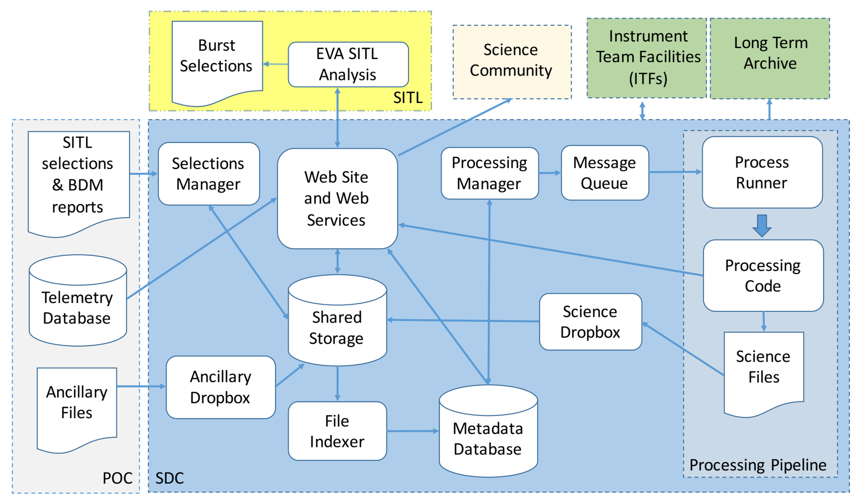
- Poster: EGU2016-10061, The MMS Science Data Center. Operations, Capabilities, and Data Availability. Kristopher Larsen. This meeting.
- D.N. Baker et al., Magnetospheric Multiscale Instrument Suite Operations and Data System. Space Sci. Rev. (2015). doi:10.1007/s11214-014-0128-5
- Magnetospheric Multiscale. Space Sci. Rev. Volume 199, Issue 1-4, March 2016. ISSN: 0038-6308 (Print) 1572-9672 (Online)

MMS SDC Architecture and Design

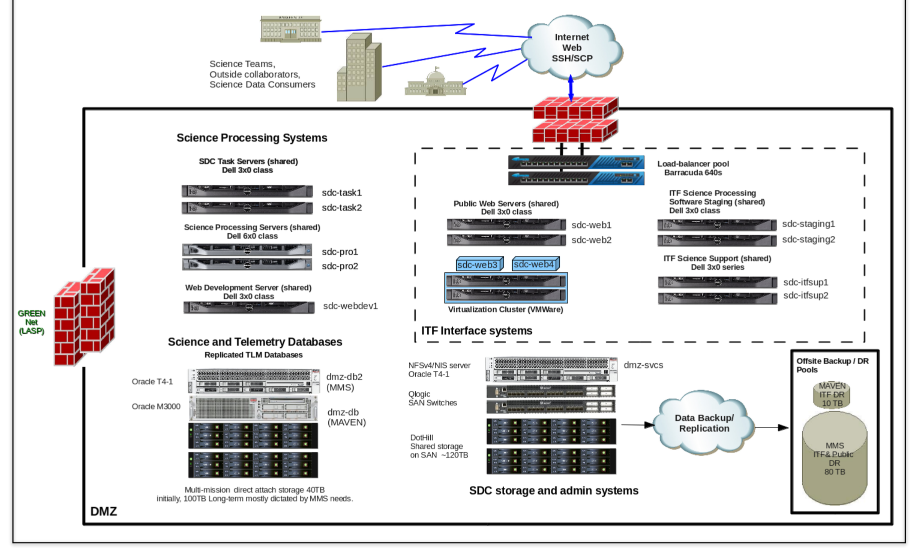
Mission-Level Data Flows



SDC Logical Architecture and Flow



SDC Physical Architecture



Design Drivers

Data production:
- Perform science data processing to produce Level 1, QuickLook, SITL, and Level 2 products
- Generate QuickLook, orbit, and formation plots
- Support large data volumes - typically 10,000 files and 100GB per day, spiking to 1TB per day
- Run science data processing code built with various tools and provided by multiple institutions
- Provide system monitoring and notification capabilities for SDC personnel and outside instrument teams
- Accommodate multiple data versions of every product
- Create off-site backups automatically
- Support periodic full-mission reprocessing

Data access:
- Multi-language access to instrument telemetry, ancillary files, burst management, science data files
- Authenticated web data access for MMS team members to all products
- Automated transfers to NASA SPDF long-term archive plus worldwide data mirrors
- Direct file system access for instrument team data systems personnel from multiple institutions
- Unauthenticated public web access to L2 products following the March 2016 initial public release with team approval and 30 day latency
- Friendly, intuitive web pages plus RESTful APIs

Management:
- Support worldwide science teams in multiple institutions
- Permit team members at other institutions access to computers within the SDC to support code deployments and data access.
- Provide agile code deployments for job triggering, science processing code, etc.

Software Implementation

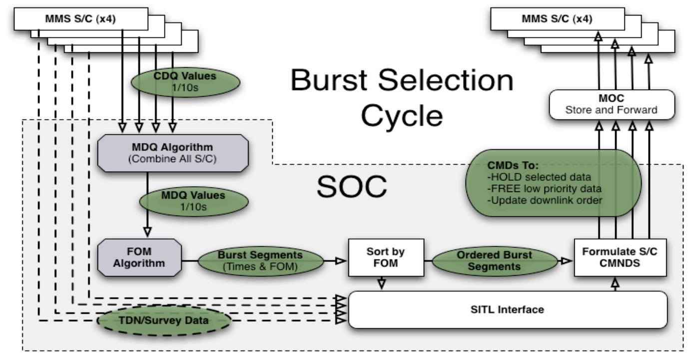
The SDC incorporates two types of software components:

"Core" SDC Functionality Software
- The SDC uses remarkably little code for its core functionality, utilizing python plus open source libraries (see right panel)
- The complete code base is ~20,000 standard lines of code
- Code base includes
  - Software to perform data processing job control, data management, and data handling functions
  - IDL plotting code that generates orbit and formation plots and also QuickLook plots for the entire mission.
  - Custom incremental backup tool that sends all new science data and processing code to Amazon Glacier on a daily basis
  - Unit tests that provide 96% line coverage

Science Data Processing
- Most of the science processing code is developed and maintained by instrument teams at other organizations
- This software is deployed into the SDC and executes as part of a distinct processing pipeline for each instrument
- Both staging and production environments are used. New deployments are first tested in the staging environment, then promoted to production
- Multiple languages are supported, including Java, IDL, Matlab, and Python
- Software libraries are also provided, including SpacePy, SPICE, and TPLOT

MMS Burst System & Scientist-in-the-Loop (SITL)

At their highest resolution, the MMS instruments produce data at rates many times the orbit-averaged telemetry rate. These Burst Data are key to understanding magnetic reconnection, but are essential for only a fraction of the orbit.



Rather than rely on triggers to turn burst data collection on or off, the instruments always collect burst data throughout Region of Interest (ROI) encounters (approx 50% of the orbit period). Onboard flash memory (96 GB of data storage organized as 24,576 buffers of 4MB each) stores the entirety of ROI data while ground-based software use telemetered trigger and survey data to determine, with input from the Scientist-In-The-Loop (SITL), which onboard data buffers are most worthy to be downlinked in the limited telemetry stream.

Using this approach, the most interesting concurrent data from all instruments on the four spacecraft can be downlinked for analysis, thus maximizing the science return in the expected regions of magnetic reconnection.

The MMS SDC services facilitate the SITL process in several key ways:
- Metadata about onboard stored Burst Data are downlinked by the POC and forwarded to the SDC, where they are made available to the SITL
- Downlinked Survey science data are processed quickly at the SDC into SITL-specific data products for use in the SITL selection process
- The on-duty Scientist-In-The-Loop (SITL) runs an IDL graphical tool called EVA that calls SDC web services to obtain the latest automated burst selections and processed science data for the most recent Region of Interest.
- The SITL selects burst time intervals of greatest scientific interest to downlink and submits them to the SDC, which relays them to the POC for uplink to the spacecraft. Selections are translated into commands that control the onboard data downlink queue.
- The available time window for SITL review and submission can be short, sometimes just a few hours.

Technologies In Use

- Common Data Format (CDF) for released data products
- Red Hat Enterprise Linux: reliable with strong tools for automated operations
- NFS over scalable disk arrays: expandable storage that can be mounted from multiple servers
- Python: high-level language with expansive libraries and system access. Script language eases fast deployments of code minor code changes
- PostgreSQL/SQLAlchemy: powerful, reliable open source database (and python access) used for storing indexed file metadata, processing records, mission events, etc.
- RabbitMQ/Pika: reliable message queuing server for management of automated processing jobs
- Apache/mod_wsgi/Flask: web server plus RESTful web services using python
- Amazon AWS S3/Glacier/Boto: inexpensive offsite data archiving
- Python utilities: pytz
- IDL, SPEDAS: plotting and analysis tools with a heritage from space physics
- Software process: git, JIRA, Confluence, Crucible, Jenkins, pylint, unittest, nosetests, symilar, mock

Experiences and Lessons Learned

- Supporting reliable and automated processing in a distributed multi-institution fashion:
  - The original MMS concept was for teams to process data at their home institutions and deliver products to the SDC for dissemination and archiving.
  - All processing systems need to operate with high reliability to meet overall mission data production and dissemination timelines.
  - The SDC has the advantage of being designed with redundancy and reliability in mind, which led to some teams transitioning their automated processing systems into the SDC to avoid the extra burden of "high availability".
- Managing Operational Expectations
  - Users are always eager to receive the latest data. User *expectations* for system availability, reliability, and staff reachability - especially after hours - can exceed *requirements* and *plans*. This must be managed through regular communication and appropriate staff budgeting. A support model should also be made.
- Accommodation of science team provided code - An SDC must be accommodating of language and library choices needed by different teams, as well as their own institutional processes.
- Reprocessing surges involving large time ranges and all instruments are inevitable and can be costly to support. Design for scalability and surge support.
- Network bandwidth limitations are a major constraint for modern space missions with large data volumes. However, estimating data interest and download needs is challenging and good capacity planning is not always possible. Offsite backups that are transferred over the internet can further exacerbate the issue.
- Overall system complexity is quite high given that it depends on asynchronous components in the MOC, SOC, SDC, numerous remote teams and facilities, etc.
- Unavoidable problems like missed contacts and institution's hardware outages require the inclusion of *regularly scheduled* reprocessing to minimize the need for manual intervention.
- The scripting language python was an excellent choice due to its ease of deployment and patching
- It can be easy to under budget for needed support staff and computer hardware needs.
- Virtual Machine and Cloud technology provide a strong option for resource optimization.
- Oversight of third-party code bases can require considerable effort, and should be distributed to cognizant teams wherever possible.

Future Direction

- Apply architecture to the next big mission: EMM going to Mars in 2020
- Make the code less mission-specific by completely separating configuration and custom data adapters from the core code
- Move to virtual machines rather than dedicated hardware, enabling deployment to VMWare, Hyper-V, or Amazon AWS
- Transition to "infrastructure as code", using tools such as Ansible to spin up highly scalable SDC instances in little time

Acronyms

- POC - Payload Operations Center
- SOC - Science Operations Center
- SDC - Science Data Center
- SITL - Scientist-In-The-Loop
- ITF - Instrument Team Facility
- MOC - Mission Operations Center