LRO MPS

Mission Planning and Scheduling System for NASA’s Lunar Reconnaissance Mission

GSAW 2009

GMV
- Gonzalo Garcia: LRO MPS Project Controller
- Assaf Barnoy: LRO MPS Project Manager
- Theresa Beech: VP of Business Development

Honeywell Technology Solutions Inc
- Rick Saylor: LRO Ground Systems & Ops Lead
- Jennifer Sager Cosgrove: LRO MOMS MPS Project Manager

NASA Goddard Space Flight Center
- Sheila Ritter: NASA Code 583 MPS Product Development Manager
LRO MPS OVERVIEW
OVERVIEW: LRO Mission

- The Lunar Reconnaissance Orbiter (LRO) is the first mission in NASA's planned return to the moon.
- LRO will launch in Q2, 2009
- Objectives
  - find safe landing sites
  - locate potential resources
  - characterize the radiation environment
  - test new technology
OVERVIEW: LRO Mission Planning & Scheduling (MPS): Functions

**MISSION CRITICAL FUNCTIONS:**
- **Produce an integrated schedule** of non-conflicting, coordinated ground and space segment operations
- **Build Stored Command Loads** (Relative and Absolute Time Sequences)
- **Generate Ground Pass Scripts** for Automation
- **Build Ephemeris Load Files**

**MISSION SUPPORT FUNCTIONS:**
- Slew Maneuver Planning
- Onboard Memory Modeling
- Execution Verification Feedback
- Generation of Activity Reports
OVERVIEW: LRO MPS Heritage

- LRO MPS is based on **flexplan**, also selected for:

  | **Metop** – European Organization for the Exploitation of Meteorological Satellites (EUMETSAT): Joint mission with NOAA |
  | - Launched October 19, 2006 |
  | - Currently operational. |

  | **SMOS** (Soil Moisture and Ocean Salinity) – European Space Agency (ESA): |
  | - Final release accepted in 2006 |
  | - Expected launch in mid-2009 |

  | **LDCM** (Landsat Data Continuity Mission - Landsat 8) – NASA Goddard Space Flight Center (GSFC) / US Geological Survey (USGS) |
  | - Expected launch in 2012. |
OVERVIEW: Interfaces

- MPS interfaces with various elements using a file based transfer.
OVERVIEW: Architecture

MPS

Key:
EI – External Interface
MEP – Mission Environment Preparation
TEG – Tailored Event Generation
PIC – Product Input Customization
SG – Schedule Generation
CR – Conflict Resolution
RQT – Report Query Tool
SE – Schedule Execution

GN – Ground Network
FDF – Flight Dynamics Facility
T&C – Telemetry and Command

MOC

File Exchange (XML, ASCII, etc.)

GNSO Schedule  Activity Requests  Orbital-events, Maneuvers  Events/Requests  Command/Procedure DB  MPS Reports  flexplan Log  Execution Status  Command/Procedure Schedules  Command/Procedure Loads

GN Station  Science Centers  FDF  …  T&C

Load Builder
CCSDS Packetizer

Adapters
XML Adapter

Oracle Central Database

RTS Export  MEP  CR  SG  TEG  Log  SE  RQT

flexplan module
Optional module

flexplan

WWW
SCHEDULE GENERATION PROCESS

1. Ingest products into MPS
   - Add, modify, or delete an event

2. Generate PIC
   - Yes
     - Initiate Schedule Generation
   - No
     - MOT modifies, deletes, or adds sequences
     - No
       - Ready to conflict check
       - Yes
         - Initiate Conflict Resolution
       - No
         - Review schedule

3. Review schedule for conflicts
   - Conflicts
     - Yes
       - Resolve conflicts
     - No

4. Generate Executable Schedule
   - Generate ATS and pass scripts
   - Approve and sign products
   - Products ready for operations
INPUTS: Processing

- LRO MPS receives and processes over 100 different **input events** belonging to more than 15 categories from various internal and external elements of the MOC.

- **Inputs** include:
  - Space or ground events identifying periods of time in which mission activities must or must not take place
  - Events of possible interest and relevance to some or all LRO scheduling elements
  - Specific request to add activities with certain characteristics to the schedule at a specific time or during a particular event

- All the inputs are not required to generate a daily schedule.
**INPUTS: Generic Input XML Schema**

- **flexplan** implements a single open XML schema for all planning inputs, of any type.

- The schema structure provides a flexible XML message that easily maps to any information of the planning inputs.

```
<FILE_NAME>DFD9_2008302_2008309_B05.FDP</FILE_NAME>
<CREATION_DATE>04-apr-2008 09:50:57.000</CREATION_DATE>
<START>20-oct-2008 00:00:00.000</START>
<END>4-nov-2008 00:00:00.000</END>
<STATE_VECTOR_EPOCH/></STATE_VECTOR_EPOCH>
<STATE_VECTOR/></STATE_VECTOR>
<Event>
<UTC_Start_Time>28-oct-2008 12:28:42.000</UTC_Start_Time>
<Duration>12078</Duration>
<Event_Description>DFD9_OMNI_DS24</Event_Description>
<Sat>GRD</Sat>
<Entity>Omni</Entity>
<List_of_Event_Parameters>
  <Event_Parameter>
    <Event_Par_Name>ORBIT</Event_Par_Name>
    <Event_Par_Value>1</Event_Par_Value>
  </Event_Parameter>
  <Event_Parameter>
    <Event_Par_Name>MAX_ELEV</Event_Par_Name>
    <Event_Par_Value>6.494</Event_Par_Value>
  </Event_Parameter>
  <Event_Parameter>
    <Event_Par_Name>TIME_MAX_ELEV</Event_Par_Name>
    <Event_Par_Value>2008302.122842</Event_Par_Value>
  </Event_Parameter>
</List_of_Event_Parameters>
<Event>
...</Event>
</Event>
</FDF_to_FP>
```

XML: eXtensible Markup Language
MISSION DEFINITION: Operational Issues

- Off-line process performed during the definition phase of the mission.
- Create and define all the data structures that will be used routinely to generate schedules.
- These data structures reside in the MPS Oracle database.
- Master Schedules with all scheduling rules reside in configuration controlled repositories.
- The data in the MEP implements the set of operational requirements for the LRO Orbiter.
- The Mission Definition can be updated during the operation phase as required.
MISSION DEFINITION: Resources and Events

- **Resources:**
  - Configured to keep track of the resource usage and avoid scheduling of conflicting tasks.
  - Allowed to create Analytical modeling of physical elements (e.g. solid state recorder) based on schedule activities.
  - Can represent logical elements (e.g. availability of personnel).

- **Events:**
  - Planning inputs automatically ingested by *flexplan*
  - Defined by category and source
  - Can have input parameters and predefined attributes
MISSION DEFINITION: Scheduling Rules

- Information from scheduling inputs and resources are used in user defined scheduling rules to add tasks to the schedule.
- Rules are saved in files and are placed under Configuration Management.

Scheduling Input Event

```
<Event>
  <UTC_Start_Time>15-Jan-2009 01:22:16.000</UTC_Start_Time>
  <Duration>4386000</Duration>
  <Event_Description>GNS01_S-BAND</Event_Description>
  <Sat>GRD</Sat>
  <Entity>LRI5</Entity>
  <List_of_Event_Parameters>
    <Event_Parameter>
      <Event_Par_Name>Automation</Event_Par_Name>
      <Event_Par_Value>Automated</Event_Par_Value>
    </Event_Parameter>
    <Event_Parameter>
      <Event_Par_Name>LRO_Antenna</Event_Par_Name>
      <Event_Par_Value>LRO_Antenna</Event_Par_Value>
    </Event_Parameter>
    <Event_Parameter>
      <Event_Par_Name>MiAC</Event_Par_Name>
      <Event_Par_Value>MiAC</Event_Par_Value>
    </Event_Parameter>
    <Event_Parameter>
      <Event_Par_Name>S_Antenna</Event_Par_Name>
      <Event_Par_Value>S_Antenna</Event_Par_Value>
    </Event_Parameter>
    <Event_Parameter>
      <Event_Par_Name>KA_START</Event_Par_Name>
      <Event_Par_Value>2009-11-01:32:18</Event_Par_Value>
    </Event_Parameter>
  </List_of_Event_Parameters>
</Event>
```

Mission Scheduling Rule

```
WHEN
  there is a Event [ ] called ?Event
  [where]
    such that name equals("GNS01_S-BAND") [...] %

THEN
  assert [ ] Task [ ]
    so that parentEvent = ?Event.ID
    and name = "START_CONTACT"
    and offset = -600 * 1000

  assert [ ] Task [ ]
    so that parentEvent = ?Event.ID
    and name = "START_CONTACT"

  addIntParameter(true,"AOS_YEAR",getYear(?event.startUTC)) [...] 
  addIntParameter(true,"AOS_DY",getIntDOY(?event.startUTC)) [...] 
  addIntParameter(true,"AOS_HOU",getIntHour(?event.startUTC)) [...] 
  addIntParameter(true,"AOS_MIN",getIntMinute(?event.startUTC)) [...] 
  addStringParameter(true,"STATION",?event.entity) [...] 
```

GSAW 2009 - LROMPS 2009/03/24 13 © GMV, 2009
SCHEDULE GENERATION: Operational Issues

- Involves populating a working schedule with instances of Sequences.
- The majority are inserted automatically during the execution of rules, triggers are the scheduling inputs.
- User selects set of rules to use for a given schedule.
SCHEDULE GENERATION: Orbiter and Ground Schedule

- The LRO MPS schedules Orbiter and Ground activities simultaneously on a single time line.

- Orbiter Activities are exported in the Absolute Time Command Sequence Loads (ATS).

- Ground Activities are exported in a series of Pass Scripts.
CONFLICT RESOLUTION: Constraint Checks

- All schedules generated by the MPS are checked for:
  - Timing relationship constraints
  - Resource consumptions violations
  - All command parameter values must be within DB limits

In addition, user defined constraint rules are supported:
- Maximum Orbiter commands per schedule and per second
- Maximum slews per orbit and per day
- Maximum slew angle and duration
CONFLICT RESOLUTION: Display Notification

Timing Constraint

OOL Constraint

Resource Constraint

Over consumption

Nominal resource usage

Resource availability
AUTOMATION: Ground Pass Scripts

- Automation of nominal supports is driven with **pass scripts** generated by the MPS.
- Pass scripts conform to formats from the Satellite Test and Operations Language (STOL) used by the LRO Telemetry and Command (T&C) system.
- The T&C system reads the pass scripts using a STOL procedure developed by the Mission Operations Team.
- Once the pass script is read successfully, the T&C system will queue each of the scheduled activities as defined in the pass script.
Activity Plan: Overview

- LROMPS *Activity Plan Manager* is:
  - A web based application
  - Protected by secure access for multiple user levels
- LROMPS *Activity Plan Manager* allows the user to:
  - Access mission planning reports
  - View the activity map
  - Share comments to the activity map and reports
Activity Plan: User’s Interface

- Displays past, current and future LRO ground and Orbiter events and activities and associated reports.
CONCLUSIONS:
UNIQUE FEATURES OF THE LRO MPS

- Compared to other missions, the LRO MPS presents some unique features that have been very valuable for the mission preparation and soon for operations.
  - The **operator has full control of the evolution of the MPS**, no need for software changes. Why?:
    - **Fully table-driven**: All the characteristics of the mission (resources, event types, command parameters, limits, etc) are stored in a database and can be modified easily by the operators.
    - **Soft rules**: MPS logic for schedule generation is stored in rules that can be edited by the operator.
  - Extensive use of **XML interfaces** for exchange of information with external systems. Easier integration, automatic validation of inputs.
  - **Web interface** to provide external access to MPS information.
  - **CCSDS load builder** generator fully integrated within MPS.
Thank you

GMV LROMPS Team

www.gmvflexplan.com

www.gmv.com
ACRONYMS LIST

- ATS: Absolute Time Sequence
- CR: Conflict Resolution
- DMS: Data Management System
- DB: Database
- EI: External Interface
- ESA: European Space Agency
- EUMETSAT: European Organization for the Exploitation of Meteorological Satellites
- FDF: Flight Dynamics Facility
- GN: Ground Network
- GSFC: Goddard Space Flight Center
- LDCM: Landsat Data Continuity Mission
- LRO: Lunar Reconnaissance Orbiter
- MEP: Mission Environment Preparation
- MOC: Mission Operations Center
- MPS: Mission Planning System
- NASA: National Aeronautics and Space Administration
- NOAA: National Oceanic and Atmospheric Administration
- OOL: Out Of Limits
- PIC: Product Input Customization
- RQT: Report Query Tool
- RTS: Relative Time Sequence
- SE: Schedule Execution
- SG: Schedule Generation
- SMOS: Soil Moisture and Ocean Salinity
- STOL: Satellite Test and Operations Language
- T&C: Telemetry and Command
- TEG: Tailored Event Generation
- USGS: United States Geological Survey
- XML: eXtensible Markup Language
- WWW: World Wide Web