Mission Lifecycle Modeling and Simulation

Meemong Lee, Richard J. Weidner, Wenwen Lu
Jet Propulsion Laboratory
4800 Oak Grove, Pasadena, CA 90272
818-254-2228
meemong.lee@jpl.nasa.gov, richard.weidner@jpl.nasa.gov, wenwen.lu@jpl.nasa.gov

Abstract --- mission synthesis and simulation research at the Jet Propulsion Laboratory addresses mission model taxonomy, progressive lifecycle representation, model-based mission design, and simulation-in-the-loop design. The Virtual Mission (VM) project integrates the research activities and implements a virtual mission lifecycle to enable a globally optimal mission. The VM is composed of three interacting modeling and simulation layers, a mission model architecture layer, a mission system simulation layer, and a mission operation simulation layer. The three layers collectively simulate the development, integration, and operation phases of the mission cycle for comprehensive validation of the mission design products. The VM was applied for the MICAS (Miniature Imaging Camera And Spectrometer) payload system of the Deep Space 1 mission (DS1) to validate the integrated system's performance to ensure the desired science return. VM is being applied to design and to validate calibration scenarios and science observation scenarios for the extended science mission of the DS1 project.

TABLE OF CONTENTS

1. INTRODUCTION
2. VIRTUAL MISSION INFRASTRUCTURE
3. MISSION MODEL SPACE
4. VIRTUAL MISSION SYSTEM
5. VIRTUAL MISSION OPERATION
6. SUMMARY

1. INTRODUCTION

The lifecycle of a space mission consists of six phases, concept design, detailed design, development, integration and test, operation, and science analysis. A mission is sequentially carried out by a set of teams, one team per phase with some overlap for inter-team transition. Disadvantages of the sequential lifecycle include a lengthy lifecycle duration, loss of information during the team transition, phase-wise optimization, and disconnect between design and operation. The Mission Simulation and Instrument Modeling Group at JPL has developed various modeling and simulation systems to improve the lifecycle in collaboration with several flight projects.

Virtual Mission is one of the research and development projects of the MSIM Group, which addresses mission lifecycle modeling and simulation. The objectives of the VM are:

1) *Reversible design:* to formulate a mission model space where all phases of the lifecycle can be expressed sharing a common mission model taxonomy, and the design processes can be propagated bi-directionally from components to system to operation, as well as from operation to system to components.
2) *Integrated design*: to develop a virtual mission system that can simulate an integrated system behavior based on the representation of the subsystem designs so that each subsystem design can be progressively refined with system-level perspectives and feedback.
3) *Validated design*: to perform operation-level analyses for comprehensive validation of the mission system design with realistic operation scenarios and mission environments.

VM approaches these objectives by introducing three interacting modeling and simulation layers, a mission model architecture layer, a mission system simulation layer, and a mission operation simulation layer. The overall VM infrastructure and the above three design objectives are discussed in Section 2. The following three sections describe the details of each layer with respect to the technical objectives, challenges, current implementations, and future directions.

Section 3 discusses the mission model architecture layer with respect to a mission model taxonomy composition and a multi-phase, multi-resolution property description. The mission model taxonomy organizes a mission into three categories, mission world, spacecraft system, and payload system. The properties of each category are further organized employing three types of model definitions, structure, performance, and operation. The multi-phase and multi-resolution description constructs a functional abstraction of the development, integration and test, and operation phases of the mission lifecycle.

Section 4 discusses the mission system simulation layer with respect to virtually developing and integrating a spacecraft system, a payload system, and a mission environment based on the mission system properties and phase-dependent property variations described in the mission models. The virtual mission system constructs executable subsystem models from the mission property model scripts via automated model constructors.

Section 5 discusses the mission operation simulation layer with respect to operation scenario composition, time-based and distributed simulation, and monitoring of the simulation process. The operation scenarios are composed in a high-level scenario language that provides mission-generic expressions for experiment conditions and desired science data products. Time-synchronized command handling and data-flow based asynchronous data handling are integrated to support both time-sensitive and fidelity-sensitive aspects of science experiments.

Section 6 summarizes the paper by presenting the past and current roles of the VM as a tool to achieve the three objectives. The VM has been applied as a reversible design tool for driving the mission system design from the science operation perspective, as an integrated design tool for providing the mission system level perspectives to the payload system development team, and as a validation tool for enabling operation-level analysis of MICAS (Miniature Imaging Camera and Spectrometer) project and the Deep Space 1 MICAS observation scenario design for the Deep Space 1 mission.

2. VIRTUAL MISSION INFRASTRUCTURE

The VM approaches mission lifecycle modeling and simulation with emphases on the instrument systems and science experiments. The primary role of the VM is to validate the design products against the ultimate science return objectives by performing virtual development, integration, and operation of the mission system.

The secondary role is to validate the progress of the mission system at each phase of the lifecycle so that the impact of design defects and additional variations can be analyzed and corrected in time. The two roles of the VM are implemented with the three layers, Mission Model Space, Virtual Mission System, and Virtual Mission Operation, each layer with a progressive lifecycle of its own, as shown in Figure 1.

Mission Model Space provides a comprehensive modeling infrastructure for representing the design specifications of the mission system. Virtual Mission System develops a set of subsystems to form a pseudo-mission-system from the subsystem design product representations. Virtual Mission Operation performs science experiments by commanding the pseudo-mission-system and providing it with a realistic mission operation environment.

Each layer is continuously evolved, tracking the mission lifecycle with layer-specific progressive mechanisms: the Mission Model Space layer with multi-phase, multi-resolution mission model methods; the Virtual Mission System layer with fidelity sensitive subsystems; the Virtual Mission Operation layer with high-level operation languages. The fidelity-sensitive subsystems and the high-level operation languages are developed in close alignment with the multi-phase, multi-resolution models so that the mission system simulation can adapt to the phase-specific property characteristics and the languages can be applied for science experiment scenario design and mission operation sequence design.

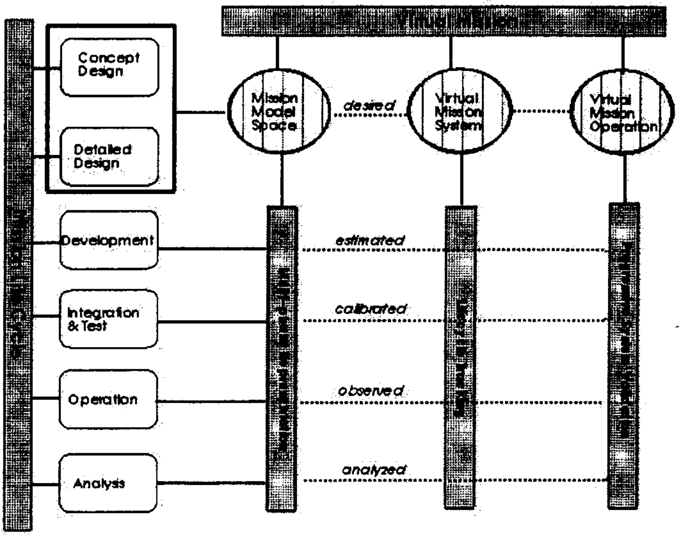


Figure 1 Virtual Mission Infrastructure

Terminology Definitions

The VM pursues three new design paradigms - reversible design, integrated design, and validated design - to achieve lifecycle duration reduction, cross-phase mission model sharing, global optimization, and operation perspective insertion at the design phase.

Reversible design ---The term *reversible* implies bi-directional, where the forward direction refers to the traditional lifecycle process. The reversible design emphasizes the design from the inverse lifecycle direction by bringing the operation as the design driver. The reversible design is approached by a combination of high-level operation languages and automated design space exploration mechanisms. A comprehensive mission model space within which a mission can project the system properties and its lifecycle variations needs to be constructed in order to formulate analytic relationships among them. The mission model architecture layer is a step toward the ultimate mission model space.

Integrated design --- The term *integrated* implies a whole, where a subsystem is a part of the whole system. The integrated design emphasizes each subsystem design to be performed with system-level perspectives and feedback by developing a pseudo-system from the subsystem representations. Virtual Mission System layer is a virtual integration platform that enables integration of the subsystem designs from multiple perspectives. Analysis of the system-level impacts and the inter-subsystem design trade space are performed in this layer.

Validated design --- the term *validated* implies operation-level validation in this paper. In general, an operation-level validation is applied after the system is in integration and test phase of the mission lifecycle since the validation requires an operable system and a realistic operational environment. Virtual Mission Operation layer provides the operation-level validation for science experiment design and payload system design in collaboration with the other two layers.

Mission Interface

The VM supports a mission throughout the lifecycle enabling the above three design paradigms. As shown in Figure 2, it provides a bi-directional path between the system designers and experiment scenario designers during the design phase so that their designs are globally optimal. During the development and integration phases, it provides a mission-level impact of the variations introduced and/or discovered for the experiment scenario optimization. During the operation and analysis phases, it provides a platform to validate experiment scenarios and to analyze the observed behavior against the predicted behavior.

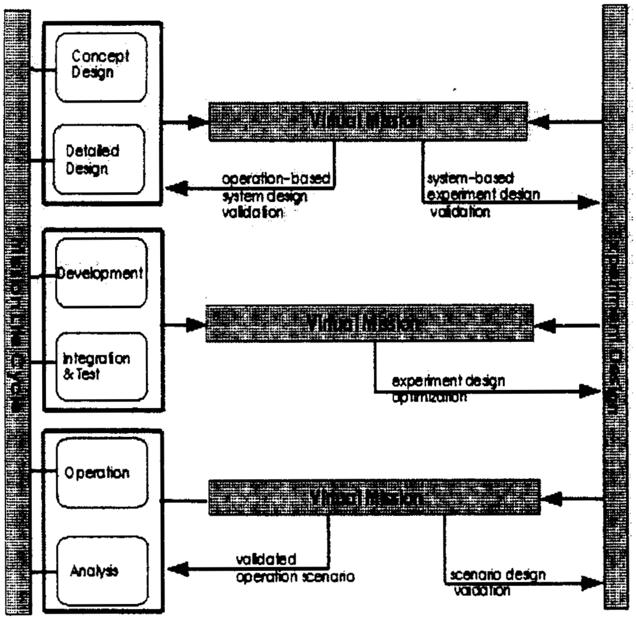


Figure 2 Virtual mission and mission lifecycle

To facilitate communication with the multi-disciplinary mission teams, a set of small languages have been created: a science experiment language to communicate with the mission science team, a mission model language to communicate with the mission system design and development teams, and an operation sequence language, with the mission operation team.

These languages are employed to structurally represent the design objectives, products, and their lifecycle aspects in a domain-specific manner. The domain-oriented representations enable simple expressions for expert knowledge integration while the structured representations enable automatic interpretation mechanisms. The automatic interpretation is approached by employing SAX [1] , an object-oriented parser generator, and LUTHOR[2], an object-oriented lexical analyzer. The SAX/LUTHOR pair generate C++ codes to construct executable mission models from a set of grammar definitions and token definitions of the mission models.

3. MISSION MODEL SPACE

A comprehensive mission model space within which a mission can project the system architecture, system properties, and its lifecycle aspects has been pursued. This section describes two types of mission model spaces employed by the VM's mission system, and mission world. Both model spaces are composed of three axes - system, property, and fidelity - as shown in Figure 3.

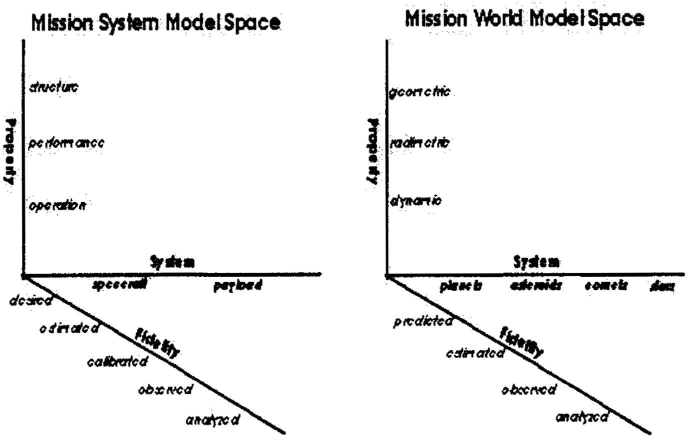


Figure 3 Mission Model Space

System Description

In the mission world model space, the system axis is organized to represent the solar system and the stars. The celestial bodies of the solar system include the sun, the planets and their satellites, asteroids, and comets.

In the mission system model space, the system axis represents the architecture of a spacecraft system and a payload system. A spacecraft system consists of a set of subsystems, including a navigation system, an attitude control system, a telecommunication system, and a data system. A payload system consists of a set of instruments, both active and passive types [3].

Property Description

An abstraction method has been created to model the properties of the mission system and its operation environment in a multi-disciplinary manner.

The mission world is modeled to represent the physical phenomena of the mission environment that impacts the operation of the spacecraft system and payload system during the science experiment. The physical properties of the mission world are divided into dynamic properties, geometric properties, and radiometric properties. The dynamic properties include the orbit dynamics and the rotation dynamics of the celestial bodies in the solar system [4]. The geometric properties include the body shape and size and the surface topography of the celestial bodies. The radiometric properties include visible as well as invisible (ultraviolet and infrared) spectral ranges of the surface reflectance characteristics derived from the surface roughness and material composition information [3].

The three properties of the mission world described above are described employing the SPICE kernels [5] developed at JPL: the SPK (Spacecraft and Planet Kernel) for the dynamic properties, and the PCK (Physical Constant kernel) for the geometric properties. The radiometric properties are described with a set of 2D data files forming a multi-resolution pyramid for surface texture, elevation, and material composition. The resolution of a target body is dynamically set based on the spatial and spectral resolution applicable for the observing distance and the operating instruments.

The Spacecraft System is modeled to represent the system properties that are relevant to the science of a mission in terms of operability and data quality. The system properties are abstracted into three categories: structure, performance, and operation. The structure category includes a solid body representation of the spacecraft system and derived characteristics including the coordinate system definitions of various subsystems and their articulation and geometric constraints. The performance category describes the execution accuracy of the spacecraft system including pointing, tracking, command scheduling, information processing, and data transmission. The operation category specifies the operation modes of each subsystem and their resource usage profiles and constraints.

The Payload system is modeled to represent the observation-specific properties relevant to the quality of the science data products and their generation processes [6]. The observation properties are also categorized in terms of structure, performance, and operation. The structure model of a payload system emphasizes its coordinate alignment with the spacecraft system, instrument's field-of-view geometry relative to the target body, and various protection mechanisms against the mission environment. The performance is described for the signal propagation and distortion of the sensing mechanism and data-processing-related capabilities.

The signal propagation is described for both passive and active instrument types. Each instrument type is modeled with a set of instrument components; passive type with optics, detectors, and electronics components, active type with signal generator and receiver. The operational characteristics are described for command and data handling which involves interaction with the mission data system (on-board and ground data handling) and operation parameter setting.

Fidelity Description

The fidelity axis emphasizes progressive lifecycle representation of the mission models that can adequately track the mission system states at all lifecycle phases. The accuracy of the mission models is critical in mission lifecycle modeling and simulation. Without the accuracy, the modeling and simulation results are as invalid as any noise. The accuracy in this context indicates the accuracy of the information content, whether it is an expected range of values, a probability distribution, or an uncertainty indication.

The understanding of the properties of a real mission system is an evolutionary process that improves continuously throughout the lifecycle of a mission. Various uncertainties and deviations can be introduced to the mission system properties during development, integration, and operation phases. As described in Figure 1, the state of the mission model information can be characterized as *desired, estimated, calibrated, observed,* and *analyzed,* each state indicating the information fidelity and uncertainty available at each phase of the mission lifecycle. Validation of the model information requires the current mission lifecycle to be revised to include the modeling process as an integral part of the lifecycle

4. VIRTUAL MISSION SYSTEM

A mission can be described employing the two mission model spaces discussed above for its science targets, spacecraft system, and payload systems. Virtual Mission System layer is composed of a virtual world, a virtual spacecraft system, and a virtual payload system and they are implemented in the C++ programming language employing the object-oriented paradigm.

Virtual World

A mission environment can be dynamically constructed as a subset of the solar system with desired target bodies. A target body can be of any celestial body type (planet, satellite, asteroid, comet, star) and can be described as a combination of a solid nucleus and an atmospheric layer. The world object employs a set of object-oriented libraries to simulate the target body phenomena.

A dynamics library, OOSPICE [7], is applied to propagate the orbital and rotational dynamics of the target bodies. A solid surface property library, SOLID [8] is applied to provide the geometric and radiometric properties of the surface of a target body. An atmospheric phenomena library, COMA [9] is applied to provide the geometric and radiometric properties of the atmospheric layer of the target body.

Virtual Spacecraft System

A spacecraft system is implemented with an executive object and multiple subsystem objects. The executive object consists of an operation scheduler, a command distributor, and a system state reporter. Each subsystem object is composed of three types of service objects, information service, coordination service, and execution service.

The information service object derives the performance and operation related information from the subsystem's property and resolution models for a specified command time. The information is utilized by the coordination service object for command verification and command execution planning. It is also utilized by the execution service object for the property and resolution simulation of the data product generation. The subsystem information service objects can be automatically constructed from the subsystem models.

The coordination service objects are implemented so that they can interact with the information objects for the performance and operation properties and utilize them appropriately for command verification and planning.

The execution service objects are also implemented to interact with the information service objects, with the additional capabilities to produce the subsystem behavior. The behavior simulation is implemented employing physics-based, as well as statistics-based, algorithms depending on the required spatial and temporal resolution of the behavior.

Virtual Payload System

A payload system is implemented in a similar manner as described above, with an executive object and multiple instrument objects. The role of the executive object includes command validation and distribution, data handling, and integrated instrument state reporting. Each instrument object is composed of an information service object and a scene-generation service object.

The scene-generation service object of each instrument renders science data products simulating the data acquisition process of the instrument. The scene generation process [6] is an integrated process which combines the target body phenomena simulation, spacecraft state simulation, and the instrument signal propagation simulation. The dynamic states of the spacecraft during the observation are supplied by the virtual spacecraft system and the apparent target body properties are supplied by the target objects (solid and atmospheric) in the virtual world.

An instrument object simulates the signal propagation and distortion specific to its type by integrating the geometric and radiometric properties of the instrument components. For example, an optical instrument consists of three basic components: optics, detectors, and electronics; the three components collectively synthesize an image product that would be acquired by observing a target at a specified time during a mission.

Figure 4 illustrates the relationship between Virtual Mission System and Virtual Mission Operation. During the mission operation simulation, the three virtual systems are constructed based on their model descriptions in the mission model space.

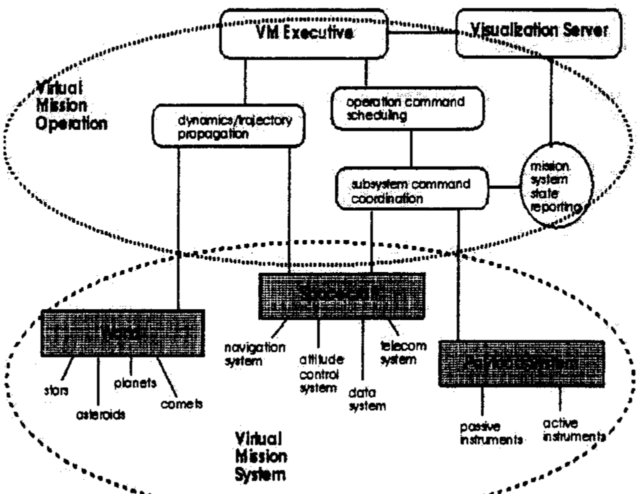


Figure 4 Virtual Mission System and Operation

5. VIRTUAL MISSION OPERATION

The VM enables a globally optimal design of a mission system and its operation by a virtual mission system development that can operate realistic mission scenarios. Operation of science experiments on the virtual mission system, described above, is referred to as virtual mission operation. The virtual mission operation includes dynamic state propagation of the mission world and the spacecraft system, operation sequence scheduling, operation command coordination, and mission system state reporting.

The major challenges in implementing the Virtual Mission operation are:

1) scientifically accurate and operationally feasible observation scenario generation;
2) time-synchronized simulation for physics-based mission operation;
3) concurrent and distributed execution in a variety of heterogeneous distributed environments; and
4) precise monitoring both of the complex relationship between mission subsystems and of their integrated behaviors.

Observation Scenario Generation

The optimization process for the observation scenario design and the hardware and software system designs of spacecraft systems and payload systems can be made bi-directional as illustrated in Figure 5. In the reversible design environment, the observation scenario design starts from the properties of the world and derive the required properties of the mission system, while the mission system design starts from the available system state and derive possible observation strategies. The interdependence of the system design and experiment scenario design requires an iterative validation and optimization process

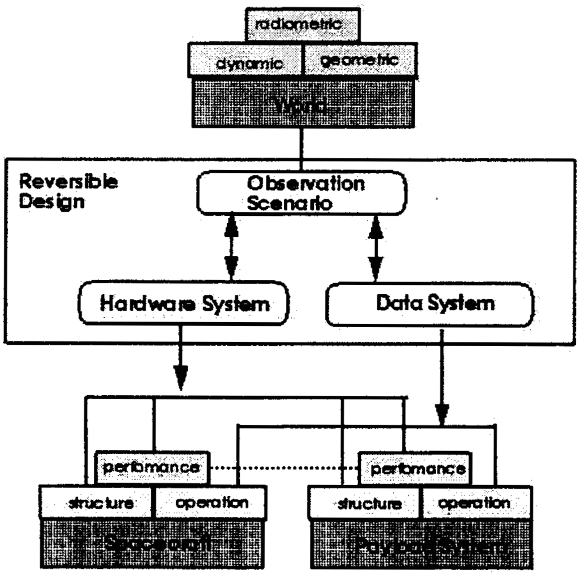


Figure 5 Observation Scenario Design

To streamline the virtual mission operation and the experiment design, an observation scenario language is pursued, which provides high-level observation event expressions, mission model based analysis, automated operation sequence composition.

The event conditions are analyzed by the model-based analysis system to translate the conditions into absolute time for the events to start and to select an optimal target for the observation events. The operation simulation system validates the translated operation sequence by executing it on a virtual mission system.

A high-level experiment scenario language is supported by model-based analysis and the operation simulation systems. An experiment is described as a set of subsystem events with corresponding event conditions. An event condition can be expressed as a combination of time condition, target condition, and other events. The time condition includes absolute time, experiment-relative time, and event-relative time. The target condition includes various physical aspects relevant to the observation, including distance, size, brightness, etc.

Time-Based Simulation

The time plays a critical role in functionally as well as physically accurate mission operation simulation. The time-based operation simulation indicates simulation of the mission system behavior as a function of time, including the absolute time (defined by the solar system inertial reference frame) as well as duration.

The absolute time establishes the state of the mission world which is critical to the science-observation-related operations, such as pointing and tracking a planetary body, target-brightness-sensitive instrument control, etc. The duration-sensitive simulation is important for operation scheduling, resource usage propagation, spacecraft-motion - related data quality analysis, etc.

The time-sensitivity of mission system models must accommodate a wide range of temporal resolutions for various subsystem operation simulations during the entire mission duration with various encounters. The performance and operation properties of a spacecraft system and a payload system must be expressed as a function of time. The temporal resolution of the mission world phenomena properties must be able to support the temporal resolution of the subsystem operations.

The time coordination is performed in a multi-level control distribution hierarchy. The Virtual mission executive, as the master controller, advances the master clock through a fixed interval (min. 1 second), and the operation scheduler distributes the commands whose start time has arrived. The coordination service object of each subsystem verifies the time with its local time and synchronizes the local clock and decomposes the command into a set of subsystem operation steps that are progressed for each time interval. The commands which require physics-based simulation are passed to the execution service object of the subsystem.

Distributed Processing

Distributed simulation requires dynamic process-to-processor mapping and platform-independent process-to-process coordination so that it can be performed in a wide variety of heterogeneous distributed computing environments.

Computational load balancing is performed at two levels of distribution: the system object level, and the command execution level. The system object level distribution employs heavyweight processes and they are mapped to heterogeneous multiple processors. The command execution level distribution is performed in a customized fashion by decomposing a computationally intensive process into multiple lightweight processes sharing data.

Load balancing among the lightweight processes is dynamically made, based on the predicted computational load for a given task. For example, a camera instrument model object is composed of multiple lightweight processes, which collectively compose a scene of a target at a given time for a specified geometric relationship between the camera and the target. The processes compute the predicted target size and divide the target size equally among them and generate the scene together.

The process-processor mapping for both heavyweight and lightweight processes is managed by a JPL software utility, DEM (Distribution Execution Manager) [10] developed by the Mission Simulation and Instrument Modeling Group. DEM provides dynamic process loading, message passing, and graphical monitoring of the processes.

Simulation Process Monitoring

Precise monitoring of the complex relationship between mission subsystems and their integrated behaviors is a challenging task. A distributed visualization server [11] has been implemented for monitoring of the simulation process and presentation of the system states. The visualization server employs three types of viewers for representing a wide range of spatial and temporal resolutions of the mission system dynamics during mission operation

The subsystem viewer type can be applied to a variety of subsystems whose operation is sensitive to its orientation to the mission world, such as an instrument's field-of-view during target observation or an antenna's coverage during uplink/downlink operations. All viewer types are implemented using interactive 3D windows (e.g. OpenInventor's examiner viewer).

For monitoring of the sub-pixel and sub-second states, 2D time-based plots are employed where new states are added to the plot as the old states scroll off the plot. Figure 6 illustrates a composite of nine views of the VM. The dynamic states of the DS1 spacecraft, DS1 high gain antenna, MICAS instrument, and DS1 data system are included in the composite. The 3x3 flat panel screen array represents the micro-Helm that the MSIM Group is developing for more comprehensive visualization of the mission operation simulation.

Figure 6 Virtual Mission Operation & Visualization

6. SUMMARY

In this paper, Virtual Mission has been presented as a tool to enable three types of design paradigms - reversible design, integrated design, and validated design - in support of advancing the mission lifecycle. The effectiveness of the VM has been verified during the last three years while supporting the MICAS project and the DS1 mission.

As a reversible design tool, VM supported derivation of the required performance and operation parameters for the navigation, attitude control, and payload systems during the expected asteroid encounter to achieve the science objectives [12].

As an integrated design tool, VM developed modeling and simulation of the DS1 system and the MICAS payload system so that the MICAS team could develop the instrument from a system-level perspective. The virtual DS1 spacecraft system and the MICAS payload system are currently being applied for design of science experiments for the extended DS1 mission.

As a validation tool, the VM provides a virtual testbed where DS1 operation sequences can be executed prior to submitting to the mission operation team. It also serves as an operation validator after the experiment is completed by visualizing the telemetry data for the mission system states during the observation in comparison with the predicted states.

The current implementation of the VM is biased toward remote sensing oriented science mission applications where the interactions among the three entities (i.e., world, spacecraft, and payload) are loosely coupled. In-situ missions and manned explorations will have more tightly coupled interactions as well as many additional system components, including various mobility mechanisms (such as a rover, an arm, a gimbal system) and multiple spacecraft systems (such as an orbiter and a lander). Also chemical and biological signal detection mechanisms of in-situ missions involve very different kinds of instruments and observations. The VM will be expanded to meet the needs of future in-situ missions.

ACKNOWLEDGEMENTS

The VM research is supported by the Information Systems Branch of the Research Program Management Division and the Advanced Technology and Mission Studies Division of the NASA Space Science Office.

REFERENCES

[1] R. J. Weidner, *"SAX – Object-oriented parser generator,"* http://cicero.jpl.nasa.gov/~richard/sax.html

[2] R. J. Weidner, *"LUTHOR – Object-oriented lexical analyzer,"* http://cicero.jpl.nasa.gov/~richard/luthor.html

[3] Manual of Remote Sensing, Vol. I, American Society of Photogrammetry.

[4] R. R. Bate, D. D. Mueller, J. E. White, *"Fundamentals of Astrodynamics,"* Dover Publications, Inc. New York, 1971.

[5] C. H. Acton, *"Ancillary Data Services of NASA's Navigation and Ancillary Information facility,"* Plantary and Space Science, Vol. 44, No. 1, pp 66-70, 1996.

[6] M. Lee, R. Swartz, and R. Weidner, *"SceneGen,"* NASA Tech Brief, 1997.

[7] R. Weidner, *"OOSPICE – Object oriented SPICE,"* http://cicero.jpl.nasa.gov/~richard/oospice

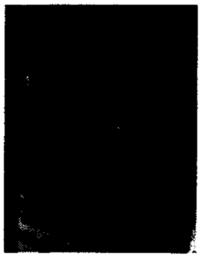
[8] W. Lu, *"Asteroid modeling,"* MSIM, JPL, http://msim.jpl.nasa.gov/~wenwen/asteroid.html

[9] C. Luchini, *"Comet Class Library,"* MSIM, JPL, http://msim.jpl.nasa.gov/~luchini/cometclass

[10] M. Abajian, *"DEM - Distributed Execution Manager,"* http://msim.jpl.nasa.gov/~abajian/DEM-1.0.html

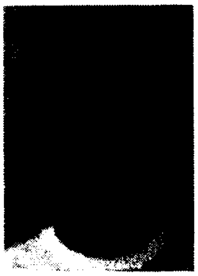
[11] A. Teng, M. Lee, *"Vmlib: an Object-oriented Library for Visualizing Observation Sequences in Spacecraft Missions,"* Military, Government, and Aerospace Simulation, 1998, Advanced Simulation Technology Conference.

[12] M. Lee, R. Swartz, A. Teng, and R. Weidner, *"Encounter Geometry and Science Data Gathering Simulation,"* AIAA Guidance and Control Conference, New Orleans, Aug., 1998.



Meemong Lee *is a principal technologist at the Jet Propulsion Laboratory where she has developed various Earth and planetary information systems and on-board software systems during the last 17 years. She has been leading the Virtual Mission project since 1996 where she develops model-based mission design and the integrated virtual mission operation platform. She is also involved in the Next Generation Infrastructure program and Deep Space 1 mission operation design. She has a bachelor's degree from Sogang University in S. Korea, and a master's degree in computer Science and a doctoral degree in Electrical Engineering from Oklahoma State University.*

Richard J. Weidner *is a principal technologist at Jet Propulsion Laboratory. During the last 18 years, he has developed many advanced information system technologies and mission operation tools . He leads the Mission Simulation and Instrument Modeling Group at JPL and is currently involved in the next generation infrastructure program and the planetary information analysis system program. He has a bachelor's degree, a master's degree, and a doctoral degree in Electrical Engineering from Oklahoma State University.*



Wenwen Lu *is a staff member of the Mission Simulation and Instrument Modeling Group at the Jet Propulsion Laboratory. She has been involved in the Virtual Mission project since she joined the lab in 1997. She develops the distributed visualization system for the VM project and supports various science data analysis tasks for environment modeling and model- based on-board operation planning. She has a bachelor's degree from Shanghai University in China, a doctoral degree in high-energy physics from Caltech.*