JWST: MAXIMIZING EFFICIENCY AND MINIMIZING GROUND SYSTEMS

Ilana Dashevsky[1], Dr. Vicki Balzano[1]

[1] *Space Telescope Science Institute, 3700 San Martin Dr., Baltimore, MD 21218, USA, Email: idash@stsci.edu*

ABSTRACT

The James Webb Space Telescope (JWST) will use an innovative event-driven architecture, which will maximize the flexibility of telescope operations. The autonomy of the event-driven system provides commanding of the spacecraft and science instruments based on the telemetry response. In the event of a failure, the telescope will continue with the portions of the science observation plan unaffected by the event, maximizing the efficiency of the observatory. Furthermore, lessons learned from the successful Hubble Space Telescope (HST) mission result in several lifecycle reduction measures, including a high-level ground to flight interface to minimize ground systems, since the detailed planning traditionally performed by ground software systems will be accomplished on-board the spacecraft. After reviewing the HST and JWST space telescope operations, the JWST event-driven design will be discussed and how it minimizes ground systems. Also, other cost-effective approaches employed by JWST will be presented.

1. BACKGROUND

The James Webb Space Telescope (JWST) will be a large infrared telescope, planned for launch in 2013 (see Figure 1). The JWST was formerly known as the Next Generation Space Telescope (NGST). It was renamed after James Webb, NASA's second administrator during the Apollo era.

The JWST uses innovative technologies, including ultra-lightweight beryllium optics; a folding 6.5 meter (20 foot) mirror made up of 18 individual segments, adjustable by cryogenic actuators; a deployable multilayer sunshade; programmable micro-shutters to allow object selection for the spectrograph; a mechanical cryogenic cooler; and state-of-art near and mid-IR detectors to detect very weak signals. Also, the JWST builds on the technical heritage and scientific discoveries of the Hubble and Spitzer space telescopes.

The JWST is an international collaboration between NASA, European Space Agency (ESA) and Canadian Space Agency (CSA). The NASA Goddard Space Flight Center (GSFC) is managing the development effort. Northrop Grumman Space Technologies is the prime contractor.

Four instruments will be provided: (1) the European Consortium with ESA and NASA (Jet Propulsion Laboratory) will build the Mid-Infrared Instrument; (2) the Near-Infrared Camera will be provided by the University of Arizona; (3) ESA and NASA (GSFC) will develop the Near-Infrared Spectrograph; (4) and the Fine Guidance Sensor, which contains a Guider and a Tuneable Filter Camera, will be provided by the CSA.

The Integrated Science Instrument Module (ISIM), which will house the four main instruments, will be provided by GSFC. Also, GSFC will develop the ISIM Control and Data Handling Subsystem and Flight Software. The Space Telescope Science Institute is developing the on-board scripts that will direct JWST operations after launch.

Figure 1. Full scale model of JWST on temporary display in Washington, DC.

1.1. JWST Orbit and Thermal Environment

Infrared is an inherently difficult observing regime. The heat from the instruments and telescope drowns out faint astronomical targets. On the Earth, the thermal environment requires complex cooling systems and the instability of the atmosphere limits infrared observing. In space, heat and light from the Earth, Moon and Sun interfere with the target. For these reasons, infrared observing is well suited to a space telescope, distant from Earth.

The JWST is destined for a solar orbit 940,000 miles or 1.5 million kilometres in space, called the second Lagrange Point or L2 (see Figure 2). The L2 is a semi-stable point in the gravitational potential around the Sun and Earth. The JWST will require relatively little rocket thrust to maintain this orbit. Also, at this point in space, the Sun, Earth and Moon are in the same relative line of sight, reducing JWST operational constraints. Furthermore, the sunshield will protect the JWST instruments from the solar heat and light.

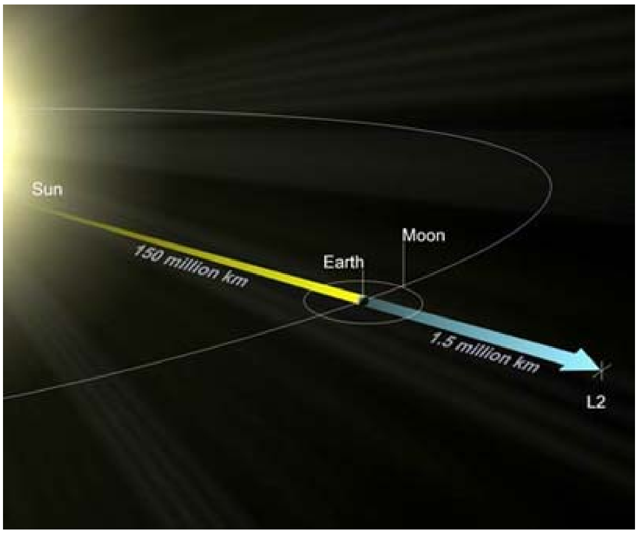

Figure 2. JWST will operate in an L2 orbit. (Credit: STScI)

The cold environment at the L2 orbit reduces the need for complex refrigeration systems. The JWST sun-shield will reduce the telescope operating temperature to under 50 Kelvin (-223 deg C). The near-infrared detectors will work at about 39 Kelvin (-234 deg C) through passive cooling. The mid-infrared instrument will operate at a temperature of 7 Kelvin (-266 deg C), using a cryogenic cooling system.

1.2. JWST Science Goals

The JWST is sensitive to wavelengths of light from 0.6 to 28 micrometers (HST infrared range stops at 0.8 micrometers). The key science goals that drive the JWST design include observations of the early universe, galaxy formation, intra-galaxy and interstellar composition, stellar birth, planetary system formation and composition.

The early universe is well suited for the infrared since light following the Big Bang was in the ultra-violet and visible wavelengths but has been stretched by the expansion of the universe into the infrared (see Figure 3). Galaxies are crowded with stars, surrounded by dust and the glow overwhelms most wavelengths of light, hiding the structure within. However, infrared will penetrate dust and distinctly pick out any source that emits heat such as the stars inside galaxies. The space between stars consists of heavy elements ejected from old stars, which are heated by young stars and collisions with other materials. Infrared is sensitive to the resulting re-emitted energy and may be used to study the composition of interstellar space, as well as the evolution of galaxies. Also, infrared pierces through dusty proto-planetary blobs revealing the structure, temperature and other physical properties of the disk.

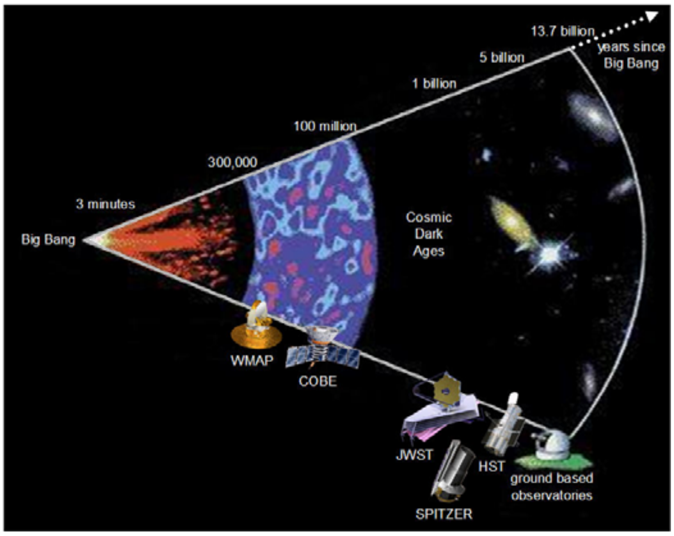

Figure 3. JWST is designed to study the first stellar formation in the universe, 12.7 to 12.9 billion years ago. (Credit: NASA/GSFC, updated by author)

2. HST SCIENCE OPERATIONS

The heritage of JWST is based on HST operational experience. What follows is a summary of the lifecycle for an HST observation, which starts from a science proposal and ends as archival data.

The HST observing efficiency is limited by physical constraints, including light from the Sun, Earth and Moon, and the South Atlantic Anomaly zones (higher particle flux from non-uniformities in Earth's magnetic field). As a result, the HST observing efficiency, measured as the total time on the target, is about 40%. For HST science instruments with electronics not affected by the South Atlantic Anomaly, the HST observing efficiency is over 80%.

2.1. Background

The HST is a very productive observatory covering wavelength regimes from the ultra-violet to the infrared, which provide a plethora of discovery opportunities. As of June 2007, over 3700 peer-reviewed papers have been published about HST science data, spanning 17 years of operation. Also, the HST is uniquely designed for upgrade during an astronaut servicing mission. The servicing missions renew the detector technology and replace aging or broken hardware, extending HST scientific usefulness and operational life. The fifth and officially last servicing mission is scheduled for the fall of 2008.

2.2. HST Ground System

Once a science proposal is chosen, following an extensive review process, the Principal Investigator is provided a template for describing the observations that he intends to do. The completed template is submitted to the Space Telescope Science Institute where it is syntax checked and ingested into a database that stores basic information for each proposal. Next the proposal is processed by software to model the time for the tasks needed to perform the observations and to break each observation into a "Scheduling Unit" (SU).

These SUs are used to plan HST's overall observing in the short and long terms. Each SU has observing time windows associated with it, which defines when the observation can be executed on-board the HST.

Hubble schedules are developed on the ground for 7 days of automated, uninterrupted operations. Complex software is used to generate the most efficient schedule for a given list of SUs. This high-level schedule includes many activities needed to operate a space telescope, including guiding and slewing, uplink and downlink events, and instrument power transitions. The high-level schedule will then be expanded to include lower-level commanding issued for each activity. Following verification of the expanded schedule, the schedule is converted into binary command blocks. Finally, individual command blocks for the spacecraft and the science payload are uplinked to the HST, approximately once every 12 and 24 hours respectively.

2.3. Absolute Time Based Operations

The scheduled activities for the Hubble spacecraft and instruments are executed using absolute time commanding. There are several varieties of computers and microprocessors that operate the spacecraft and the science payload. Each computer has its own flavor of commands and associated timing, which must be accounted for in the schedule.

The HST spacecraft computer will protect the instruments against health and safety related events by shutting down the detector and/or computer. However, frequent non-threatening events, such as a guide star acquisitions failure, will cause the associated observation to also fail. The on-board flight software will not progress forward in the plan until the observation time has elapsed. The telescope will be idle until the start of the next planned activity.

The on-board schedule may be changed for special observations called a “target-of-opportunity”, such as a super-nova or gamma ray burst, which occur infrequently. For such an observation, the scheduling process may be turned around within 48 hours by HST's dedicated scheduling team. Included in these 48 hours is the development of a new 7 day schedule for the same weekly boundaries as the executing schedule, except with the inclusion of the target-of-opportunity observation.

3. JWST SCIENCE OPERATIONS

An advantage of the JWST orbit is that there will be fewer physical constraints. In the case of the HST, a weekly ephemeris is needed to define the positions of the Sun, Moon and Earth. For the JWST, the Sun, Moon and Earth will always be relatively in the same line of sight. Thus, the target visibility windows for observations are much larger than for HST and benefit an event-driven design. However, JWST real-time communication is limited to about 4 hours per day, unlike HST that has essentially continuous communications.

Furthermore, JWST does not have South Atlantic Anomaly impacted zones. As a result, the predicted observing efficiency, measured as the total time on the target, for JWST is about 90%, with the remaining 10% required for spacecraft maneuvers and maintenance.

3.1. Event-driven Operations

The JWST science operations will be driven by ASCII (instead of binary command blocks) on-board scripts, written in a customized version of JavaScript. The script interpreter is run by the flight software, which is written in C++. The flight software operates the spacecraft and the science instruments.

The on-board scripts will autonomously construct and issue commands, as well as telemetry requests, in real-time to the flight software, to direct the Observatory Subsystems (e.g., Science Instruments, Attitude Control, etc.). The flight software will execute the command sent by the calling on-board script and return telemetry, which will be evaluated in real-time by that on-board script. The calling script will then send status information to a higher-level on-board script, which contains the logic to skip forward in the observing plan in response to certain events (see Section 4.1).

3.2. JWST Ground System

Observing with JWST is intended to start from a proposal process, similar to the HST. The JWST proposal will be ingested into a database and broken up into scheduling units called Visits. These Visits will be used to build a long range plan (about a year in duration) and a short range plan (about 22 days in duration).

Each Visit has a corresponding observing window used by the short and long range plans. The observing window specifies the start and end time windows for the Visit. If the start time window has elapsed the Visit will be skipped by the on-board scripts.

A major difference between the HST and JWST ground systems, is that the JWST science operations does not need detailed task modelling of the scheduled observations and spacecraft operations, nor computer memory management mapping. There will not be an absolute time driven schedule, since the JWST on-board scripts will construct the commands and telemetry requests on-board (see Figure 5). The products of the JWST ground system include an ASCII Observation Plan for 7 to 10 days of operation, the ASCII Visits files that specify the activity parameters, and other ancillary files such as the slit arrangement needed for the spectroscopy instrument (see Figure 4).

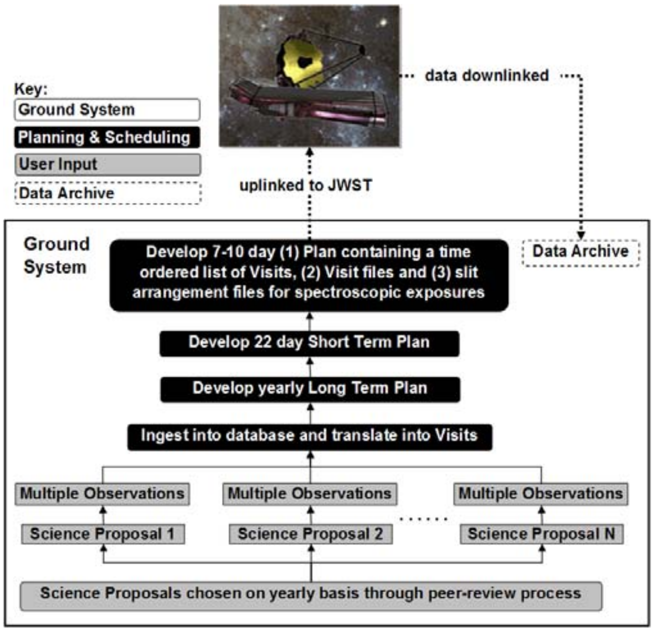

Figure 4. JWST science observation lifecycle. The JWST ground system translates observations into an ASCII Observation Plan and Visit files, which are uplinked to the JWST.

3.3. Architecture

The highest-level on-board scripts, which are called the Observation Plan Executive (OPE) process the Observation Plan that contains a time ordered list of activities called Visits (see Figure 5). The OPE will pass the activity parameters from the Visits to lower-level on-board scripts. These scripts construct the commands and telemetry requests in real-time, on-board the spacecraft. After a command is issued, these lower-level scripts evaluate the telemetry response and pass the script status up to the OPE. Based on the script status, the OPE can skip Visits, however, the OPE cannot reorder the list of Visits in the Observation Plan (see Section 4.1).

The OPE capabilities include removing Visits to redefine the end of the Plan, adding Visits to the end of the Plan, stopping the Plan after a specified Visit or portions of a Visit, as well as stopping the entire Plan. Also, the Observation Plan can be managed from the ground using the same high-level on-board scripts.

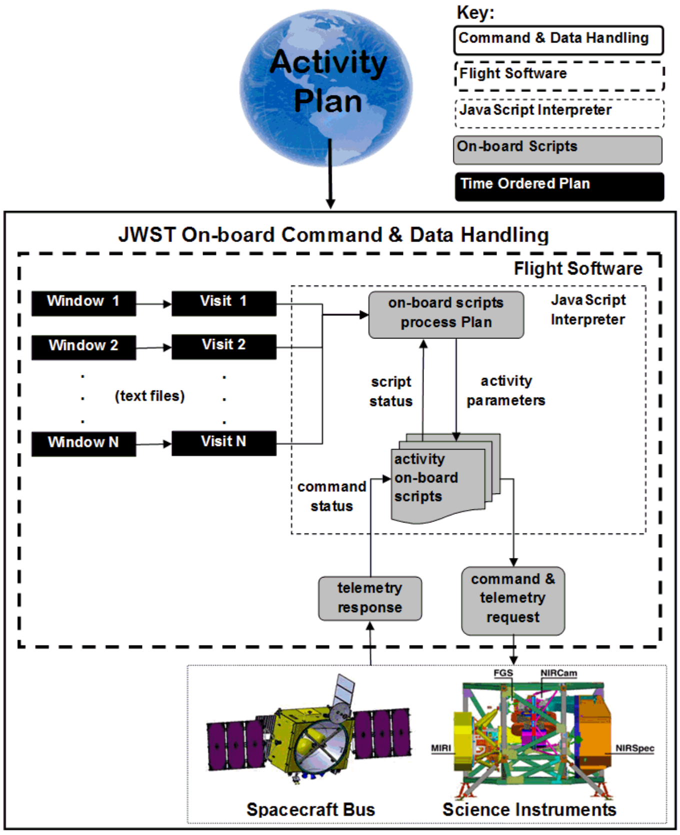

Figure 5. JWST simplified on-board architecture. The plan, consisting of Visits, is uplinked from the ground and processed by high-level on-board JavaScripts. Each Visit contains observatory activities (detector configuration, slew requests, etc.), which are processed by lower-level on-board scripts. These on-board scripts construct the commands and telemetry requests to operate the Observatory Subsystem (e.g., Science Instruments, the Attitude Control Subsystem in the Spacecraft Bus, etc.).

The activity requests within each Visit specify the science detector configuration, including the spectral element positions, exposure time, exposure patterns, and other information needed for an observation. The Visit also includes slews and guide star acquisition requests, as well as parallel detector configurations.

The activity on-board scripts are organized by the Observatory Subsystem (includes subsystems for Science Instruments, Attitude Control, etc.) into categories, which includes a category for each science instrument. Each category has a high-level or main script. Also, there are generic utility scripts used for all subsystem operations.

Each category is unique, since each subsystem has different capabilities and is optimized for different observing regimes. The basic functions performed by the science instruments include: retrieving parameters from the Visit file, sending commands to the flight software to configure the detector, position optical mechanisms, expose, read the data, and store it in an available area of the data buffer. Each calling script will determine whether a command has completed by polling the corresponding Observatory Subsystem telemetry, which removes the need to model an observation on the ground.

3.4. Parallel Processing

The JWST architecture supports up to ten threads of simultaneous processing. Examples of parallel scenarios include multiple instrument processing commands at the same time and internal calibration images during spacecraft maneuvers. To run the on-board scripts the OPE uses separate threads of execution. Also, the OPE may be commanded from the ground to alter the observing plan without interrupting ongoing science instrument or spacecraft operations.

3.5. Scripting Language

JWST uses an extended version of JavaScript, which was developed as a COTS product called Nombas ScriptEase 5.00e. ScriptEase provides functionality common to many modern software languages and follows the ECMAScript standard. It is highly customizable and portable to a wide variety of operating systems. ScriptEase JavaScript allows for a modular design flow, where on-board scripts call lower-level scripts that are defined as functions.

4. COST COMPARISON

During the development phase, cost is largely driven by hardware and technology innovation. Once JWST is launched, lifecycle costs may be saved over the long term by reducing the size and complexity of the ground systems.

Improvements on the absolute time driven design can be made in these areas:

1. on-board flight software command sequences interface;
2. translate user input into absolute/relative time command sequences;
3. generate the binary command loads, including detailed flight software memory mapping;
4. independent real-time script development and verification;
5. model mechanism and hardware functionality.

The JWST event-driven design replaces and simplifies the above with:

1. on-board flight software command and telemetry interface (replaces 1);
2. software needed to translate user input into activity descriptions (replaces 2 and 3);
3. common real-time and on-board scripting interface (replaces 4);
4. event-driven on-board scripts (replaces 5)

4.1. Event-driven Constraints

Both designs require intense verification using custom software and hardware simulators. A disadvantage of the JWST event-driven design is increased risk due to the uncertainty of all the possible failure scenarios.

However, this risk is mitigated by (1) not permitting on-board re-ordering of activities and (2) limiting the type of events that will allow the on-board scripts to skip activities. These permitted events are:

1. time window violation,
2. failed guide star acquisition,
3. failed slew execution,
4. failed small angle maneuver,
5. failed target acquisition preceding science observation,
6. no space on Solid State Recorder for science data,
7. science instrument is offline,
8. science detector is offline.

Similar to the HST, JWST still relies upon real-time operations to perform Solid State Recorder data dumps and spacecraft orbit maintenance, such as for momentum unloading.

4.2. Event-driven Advantages

Advantages of the JWST event-driven design include:

1. human-readable on-board scripts and observation plan, which simplifies implementation and lowers risk of interpretation mistakes;
2. no bit translation or binary command load verification is necessary;
3. minimal transition from Integration and Testing to Operations phases;
4. increase observing efficiency by allowing on-board scripts to skip activities in response to some real-time events;
5. minimal time between software response to instrument motion and state changes since scripts check telemetry for instrument status;
6. flexibility during operations to alter the observing plan without the need to uplink an entire new plan (OPE can be commanded from

the ground to add or remove individual Visits on-the-fly).

The above advantages will improve the operational efficiency over the long term during the JWST mission.

5. CONCLUSION

The JWST event-driven commanding concept takes advantage of the virtually constraint-free L2 orbit. Also, limited contact opportunities (about 4 hours/day) provide a special challenge for observatory efficiency. An autonomous, absolute time driven schedule, similar to what is used for HST, cannot recover lost time following unavoidable events, such as guide star acquisition failures or instrument down-time due to anomalous events. The JWST on-board JavaScripts will have the logic to respond to these events autonomously in real-time by moving forward to unaffected activities specified in an uplinked ASCII activity plan. To mitigate risk, the on-board scripts will not re-order activities in the plan.

The predicted increase in the observing efficiency compared to the HST operations is about 10%, measured as the total time on the target. Also, an increase in efficiency during operations is expected as a result of the flexibility to alter the observing plan on-the-fly and the simplicity of modifying the human-readable on-board scripts after launch.

The JWST event-driven design reduces the need for complex ground systems. Event-driven operations do not require modelling on the ground of hardware and software functions, since the on-board scripts include logic to autonomously make decisions based on the status from telemetry items. Development and maintenance of multiple interfaces for binary and human-readable forms of the activities are not needed. Also, verification of binary command loads is not required, since the on-board scripts and the listing of observations are text (ASCII) files. The common interface for on-board and ground commanding reduces development, maintenance and verification. It also reduced the transition between development and operations, as well as the preparation time to trouble-shoot flight anomalies.

6. ACKNOWLEDGEMENTS

The authors would like to acknowledge the dedicated efforts of the Operations Scripts Subsystem (a.k.a. Commanding) team at the Space Telescope Science Institute. We would also like to thank the JWST ISIM and payload flight software teams at the GSFC for their support of the on-board JavaScript interpreter. Special thanks to Merle Reinhart and George Chapman (Space Telescope Science Institute) for explaining subtle details related to the HST scheduling system.

7. REFERENCES

1. Balzano, V. and Zak, D., "Event-driven James Webb Space Telescope Operations using on-board JavaScripts", *SPIE Conference Proceedings*, Vol. 6274, 2006.
2. Balzano, V. and Isaacs, C., "Event-driven James Webb Space Telescope Operations", *SpaceOps Conference Proceedings*, AIAA-2006-5747, 2006.
3. Balzano, V. and Rehm, K., "Command system architecture for NGST", *SPIE Conference Proceedings*, Vol. 4848, 2002.