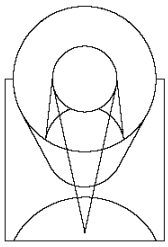

# TECHNICAL REPORT



SPACE TELESCOPE SCIENCE INSTITUTE
Operated for NASA by AURA

| Title: Functional Allocation of Processing Responsibilities between the Flight Software and the On-board Scripts | Doc #: JWST-STScI-000876, SM-12<br>Date: May 15, 2006<br>Rev: A |
|---|---|
| Authors: V. Balzano Phone: 4736 | Release Date: |

## 1.0 Abstract

This technical report summarizes the functional allocation guidelines between the flight software and on-board scripts for the processing tasks necessary to accomplish JWST event-driven operations. These allocation assignments originate in the extensive command concept definition work done in a collaborative effort by the JWST chief software system engineer and the lead STScI flight software support engineer during the period from early 2001 through the fall of 2004. Although the guidelines have appeared in several viewgraph presentations by STScI staff and by the science instrument teams, this is the first textual documentation.

## 2.0 Introduction

To accomplish JWST event-driven operations and achieve the required scientific efficiency, the flight software and the on-board scripts must function in a coordinated manner. The JWST command concept specifies a ground-to-flight interface of text files containing high-level constructs that outline the required operations. These files, known as the Observation Plan segment file and the Visit files, are uploaded into an ISIM file store on a periodic basis for interrogation by the on-board scripts. Expansion of the ground-supplied constructs into individual flight software commands takes place on-board within scripts that encode the operational rules of the observatory subsystems. Commands execute one after another based upon the prior command's execution status. The exact content of the command and telemetry interface is defined in cooperation with each of the flight software teams, their supporting system engineers, the Project system engineers, and the STScI commanding team.

The functional allocation guidelines summarized below were discussed with the science instrument and spacecraft teams early during their respective flight software requirements specification phase. Several important attributes were finalized by the "Event Driven Operations Trade Summary" written by Ken Rehm in August 2004 and by the JWST Project's selection of the trade's 'Option C' design. The general allocation guidelines were also outlined for the wider JWST community at the November 2004 Quarterly in a

viewgraph presentation entitled "JWST Science Instrument Operations Implementation Strategies."

The functional allocation guidelines have been reflected in all Project-approved flight software requirement and design documentation to date. The ongoing script requirement definitions by the science instrument operations working groups and the script-to-flight software interface definition work by the command working groups also have been consistent with the allocation guidelines. It is our collaborative work within these working groups and our participation at all flight software requirement and design reviews that enables us to verify compliance to the allocation guidelines. The JWST chief software system engineer is primarily responsible for monitoring the nature of the script-to-flight software interface.

A request to modify the allocation guidelines would originate within the working groups. Once a consensus is reached within the group, a member would be assigned to pursue the change through the established official Project channels. The same is true for any higher-level requirement or design modification identified by the groups. The official change request procedure would be used to recommend the change. Project approval and requirement flow-down would be obtained through the established change processes.

## 3.0 Flight Software Allocation

The JWST flight software is responsible for all direct hardware interfaces and for the important tasks of observatory health and safety monitoring and fault management. The flight software exclusively encodes the knowledge required for hardware communications. Within the ISIM, this knowledge is distributed among the IC&DH services, the ISIM applications and the science instrument applications according to the "CCDH FSW to Application FSW Interface Specification" (JWST-SPEC-002100).

The flight software also supports user commands to execute basic modular functions such as move a mechanism, start an exposure, turn on a lamp, and perform a vehicle maneuver. These modular functions are sequenced together by the on-board scripts to complete a single observatory capability, such as a NIRCam direct image, a guide star acquisition, or a NIRSpec target acquisition. The granularity and nature of the modular functions are defined to provide the maximum operational flexibility. Of course, when two functions are required to be executed together due to a health and safety consideration, the flight software is responsible for providing the necessary sequence from a single user command. The detailed definitions of the command and telemetry interface between the flight software and the scripts is discussed and agreed to by the command working groups. The agreements are then translated into PRD XML format and displayed on documentation sheets by the STScI commanding team. These sheets are delivered to the flight software teams for inclusion in their respective users' guide. The XML can also be provided (as is done, currently, for the ISIM and MIRI flight software).

In the area of on-board image data processing, such as would be needed for target acquisition, the flight software is responsible for calculations that are done on a pixel-by-pixel basis as these are large, repetitive procedures. Examples for the target acquisition

process are image flat fielding and target centroid determinations. These algorithms are unlikely to require modification during flight.

## 4.0 On-board Script Allocation

The JWST on-board scripts provide three major areas of functionality to enable event-driven operations.

1. The JWST on-board scripts are responsible for Observation Plan execution. This script set has become known as the Observation Plan Executive (OPE). The OPE provides the ability to skip activities when the associated constraints are not met, and manages parallel operations. The allocation of the OPE to the on-board scripts was made in 2004 when the JWST Project adopted 'Option C' from the Event Driven Operations trade study.
2. The JWST on-board scripts are responsible for using the on-board data dictionaries to create commands and telemetry requests. This script set is known as the Dictionary Interface scripts. The dictionary interface management was allocated to the on-board scripts as a result of the JWST Project adoption of 'Option C' from the Event Driven Operations trade study.
3. The JWST on-board scripts are responsible for sequencing the basic modular flight software functions to execute observatory operations. These scripts are known as Activity Description scripts (ADs). They chain together flight software commands and monitor associated telemetry items to execute the details of the Observation Plan. These scripts encode the operational rules of the science instruments and the spacecraft.

   The command and telemetry interface between the flight software and the AD scripts has been defined so that the AD scripts are allocated the tasks that depend on operational factors. This enables the flight software implementation to maintain independence from primarily operational concerns.

   The flight software commands are being defined with enough granularity to provide operational flexibility. If the exact sequencing of a science instrument capability needs to change once in-flight, then only the on-board scripts will need to be modified. The operational changes have been isolated to enable cost effective in-flight modifications. Note that the on-board scripts are under formal configuration control, and are certified on ground simulators prior to in-flight use.

   In the area of on-board image data processing, such as would be needed for target acquisition, the on-board scripts are responsible for calculations that are not done on a pixel-by-pixel basis, again to provide the maximum operations flexibility. This also centralizes the calculations that are required for multiple science instruments. Examples for the target acquisition process are reference star pattern matching and vehicle offset determinations. Target acquisitions are among the most difficult procedures to design and to test pre-launch. Our STScI experience with similar science instruments predicts that adjustments will be needed once in flight. These modifications have been isolated in the on-board scripts.

## 5.0 Conclusions

The JWST flight software/on-board script functional allocation guidelines have been described in this report. These guidelines have been followed for all current JWST flight software and on-board script development work. They have been reflected in all Project-approved flight software requirement and design documentation. The JWST chief software system engineer along with the operations and command working groups monitor adherence to the functional allocation guidelines.