Using Quality Attributes to Bridge Systems Engineering Gaps: A Juno Ground Data System Case Study

Lydia P. Dubon[1], Maddalena M. Jackson[2] and Marla S. Thornton[3]
Jet Propulsion Laboratory, California Institute of Technology,
Pasadena, California 91109

The Juno Mission to Jupiter is the second mission selected by the NASA New Frontiers Program. Juno launched August 2011 and will reach Jupiter July 2016. Juno's payload system is composed of nine instruments plus a gravity science experiment. One of the primary functions of the Juno Ground Data System (GDS) is the assembly and distribution of the CFDP (CCSDS File Delivery Protocol) product telemetry, also referred to as raw science data, for eight out of the nine instruments. The GDS accomplishes this with the Instrument Data Pipeline (IDP). During payload integration, the first attempt to exercise the IDP in a flight like manner revealed that although the functional requirements were well understood, the system was unable to meet latency requirements with the as-is heritage design. A systems engineering gap emerged between Juno instrument data delivery requirements and the assumptions behind the heritage flight-ground interactions. This paper describes the use of quality attributes to measure and overcome this gap by introducing a new systems engineering activity, and a new monitoring service architecture that successfully delivered the performance metrics needed to validate Juno IDP.

I. Introduction

The Juno quest is to further decipher the origins and early formation of our solar system through the study of Jupiter. Juno will probe Jupiter's interior structure and properties by mapping the gravitational and magnetic fields; will map variations in atmospheric composition, temperature and cloud opacity; will explore the three-dimensional structure of Jupiter's polar magnetosphere; and will determine water abundance on Jupiter to gain insight into its origins. To accomplish this, a comprehensive suite of instruments is on its way to Jupiter on-board the Juno spacecraft. Execution of the Juno mission hinges on successful collaboration among institutional partners.

The Jet Propulsion Laboratory (JPL) provides mission operations management, and systems engineering leadership throughout the mission lifecycle. Lockheed Martin Space Systems (LM-SS) in Denver, Colorado, built and now operates the Juno spacecraft. The Southwest Research Institute (SwRI) in San Antonio, Texas, leads the overall scientific investigation and is responsible for Juno science operations. Goddard Space Flight Center (GSFC) co-leads the scientific investigation. Instrument providers include SwRI, Applied Physics Lab (APL), University of Iowa, Malin Space Science Systems (MSSS), JPL, GSFC, Agenzia Spaziale Italiana (ASI), and the Danish Technical University (DTU). Fig. 1 captures the instrument contribution and

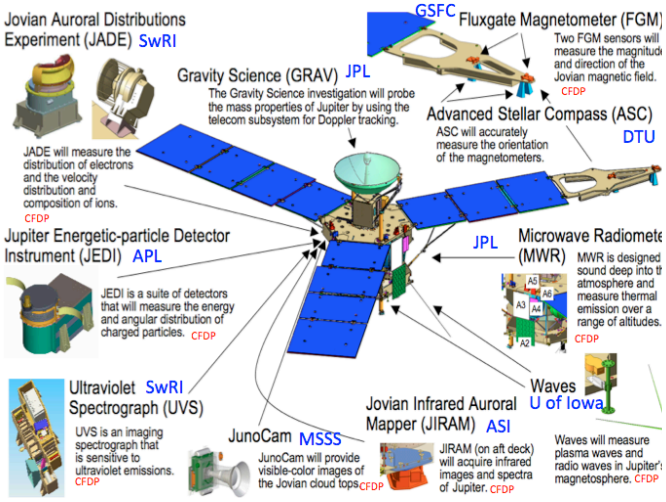


Figure 1. Juno Payload System Overview

[1] Deputy Manager, Ground Systems Engineering Section, 4800 Oak Grove Dr /MS 301-285, AIAA Senior Member.
[2] Ground System Architect, Ground Systems Engineering Section, 4800 Oak Grove Dr /MS 301-285.
[3] Juno Ground Data System Engineer, Ground Systems Engineering Section, 4800 Oak Grove Dr /MS 264-767.

investigations provided by each of these organizations.

The Juno Ground Data System (GDS) spans the Mission Operations System (MOS), and the Science Operations System (SOS) as seen in Fig. 2. The GDS provides the software, hardware, networks, and information services required to conduct mission operations. The GDS enables the distributed nature of Juno operations. The GDS delivers the instrument engineering data and raw science data to the instrument home institutions and the Juno Science Operations Center (JSOC). The capture, processing and delivery of the raw science data are the focus of this case study.

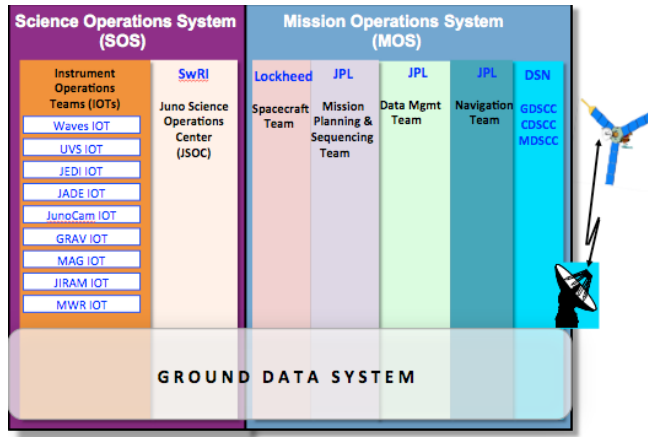


Figure 2. Scope of Juno Ground Data System

The Juno flight system heritage is based on the Mars Reconnaissance Orbiter (MRO) also built by LM-SS. The Juno ground system heritage is based on JPL's Advance Multi-Mission Operations System (AMMOS), which was also used by MRO. A fundamental Juno system engineering assumption was that flight-ground interactions would be similar to those of MRO based on heritage flight and ground systems. Both Juno and MRO use CFDP (CCSDS File Delivery Protocol) to capture and deliver instrument product telemetry. In addition, based on the expected Juno data volumes and downlink data rates, the ground system capacity and delivery performance was assessed to meet Juno data delivery requirements. However during flight system test and payload integration, instrument-ground interactions had an unexpected impact on the GDS, which revealed that the heritage design behind the Instrument Data Pipeline (IDP) was unable to meet data delivery expectations from the instrument teams. This case study describes the systems engineering gap and the approach to bridge it by learning from the behavior of the heritage ground components and feeding the information back into the systems engineering process.

The case study is organized into four sections. Section II contains a detailed description of the problem. Section III describes the modified systems engineering approach. Section IV specifies the architecture of the monitoring service developed to answer the question did we build the right system, and explores the concept of ground truth in the context of the ground data system. Section V contains the analysis of the gathered data and conclusions drawn from the data. Section VI summarizes lessons learned that can be carried forward to the next mission as part of the conclusion.

II. Systems Engineering Problem Statement

The key and driving data processing requirement allocated to the GDS is specified in terms of total raw science data volume per 11-day orbit:

> ***L3-MS-1507****: The MOS shall be able to process a maximum of 19 gigabits of downlink and gravity science data per orbit. (This requirement was further qualified by estimating a possible maximum downlink science data rate of 120 Kbps during an 8-hour 70-meter pass, which results in a potential 3.46 gigabits per pass.)*

Given MRO flight heritage and the fact that AMMOS and the DSN had already demonstrated the ability to process MRO's 6 Mbps science data rate and a possible maximum of 173 gigabits per downlink pass, the MRO ground heritage was assessed capable of meeting the above requirement. The latency requirements allocated to the GDS were also specified in the context of raw science data:

> ***L3-MS-1218****: The MOS shall make edited science data products from initial instrument turn-on and calibration activities available for access to authorized users within 15 minutes after collection of the corresponding data at the DSCC.*
>
> ***L3-MS-1579****: The MOS shall make quick look science data products available for access to authorized users during science orbital operations within 15 minutes of the end of the corresponding DSN pass.*

Drawing on actual MRO CFDP-based product generation statistics, it was determined that the heritage raw science data pipeline would meet the above requirements. Table 2 provides a high order level of MRO instrument product generation.

Table 1. MRO Yearly CFDP Product Generation

Year	2008	2009	2010
Num CFDP Products	~124,000	~88,000	~129,000

The problem surfaced during ATLO (Assembly, Test and Launch Operations) payload-related tests and the use of the GDS in "Test As You Fly" (TAYF) mode. The first two months of this activity resulted in the generation of approximately 525,000 instrument products. This volume of product telemetry surpassed the MRO ALTO numbers event though MRO also uses CFDP for 8 of its instruments. The above Juno MOS requirements still hold true as specified. However, ground impact from individual instrument behavior is not captured in the requirements. Specifically, the size of JIRAM and JEDI products were initially on the order of 2 kilobytes rather than the expected minimum size of 12 kilobytes. The problem was compounded by the ATLO downlink rate of 1.6 Mbps rather than the expected 120 Kbps rate during science operations, and from the multiplier effect on the ground. The GDS generates three files per CFDP product. These files serve as input for higher-level science data processing and to meet the instrument quick look requirements. The Juno instrument data pipeline faltered when more than 100,000 files were simultaneously in process inside the pipeline. Each node of the pipeline encountered system limitations associated with the default configuration of the software and data repositories.

It should be noted that both the JIRAM and JEDI teams had valid reasons for wanting to generate large number of smaller products. The engineering disconnect was due to lack of GDS penetration into the instrument's software architecture and behavior. As the systems engineering gap was bridged, it was revealed that JIRAM favored small products to minimize need for instrument memory; to reduce the loss of data due to telemetry packet corruption; and to maximize benefit from compression algorithm. In the JEDI case, three separate sensors make up the JEDI instrument, each generating a stream of instrument data. JEDI generates each type of science data in separate packet telemetry, which contributes to the smaller sized packet telemetry. In addition, JEDI also wanted to reduce the amount of memory used for its data structures.

A. Juno Instrument Data Pipeline Description

The Juno IDP consists of four nodes. The TLM Node extracts CFDP packets from telemetry frames and delivers these to the File Delivery Manager (FDM) Node. This node builds CFDP product files and CFDP transaction log files and publishes them to the Distributed Object Manager (DOM) Node. The DOM Node is used for short-term CFDP product telemetry storage. This node generates a metadata header file per CFDP product file. The DOM node then publishes the CFDP product file, transaction log file and metadata header file to the Front End Interface (FEI) Node. The FEI Node archives these files and makes them available to any active Instrument Operations Team (IOT) subscriber. Fig. 3 depicts the Juno ATLO GDS IDP data flow.

The first ATLO test that exercised the IDP in the TAYF mode was the November 2011 EMI-EMC test. The instrument data did not emerge from the pipeline. The first order of GDS business was to address ground software anomalies and revisit system configuration. Table 2 summarizes the Juno IDP design remediation actions.

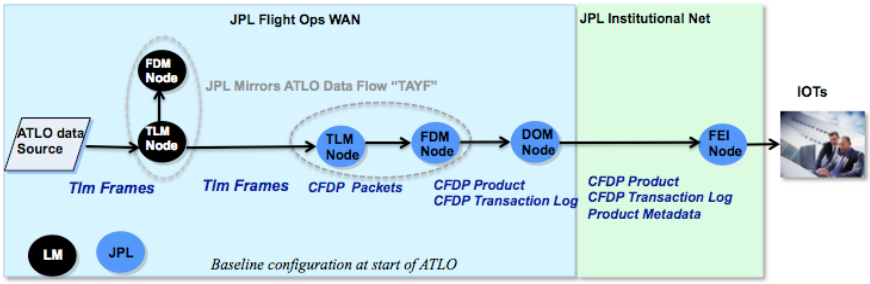


Figure 3. Juno Instrument Data Pipeline TAYF Configuration

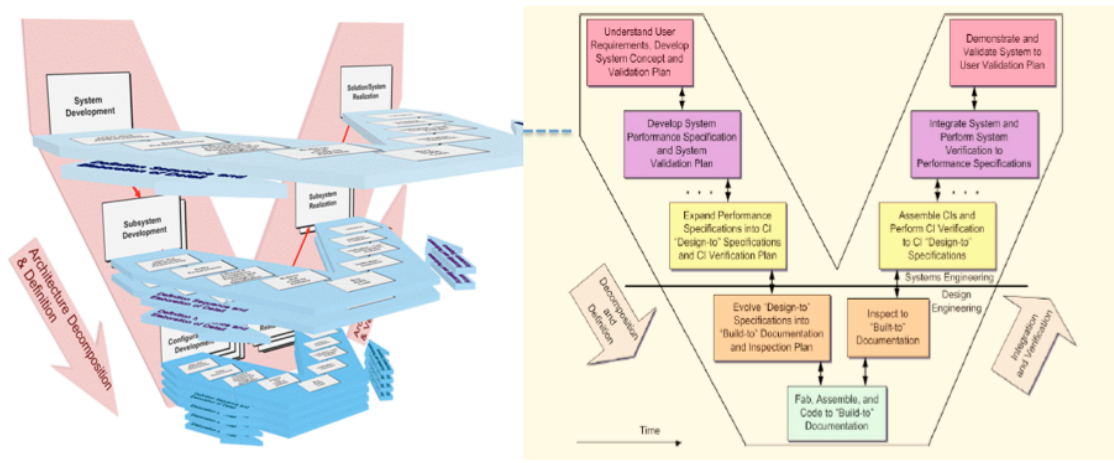
Table 2. Juno IDP Design Remediation Actions

IDP Node	Design Remediation Actions
FDM (CFDP Product builder)	1. Multiple threads (6) of product building per FDM instantiation. 2. Three FDM nodes: JIRAM FDM, JEDI FDM, and all other FDM 3. CFDP Timer adjustments
DOM (Mission Data Repository)	1. Network Appliance storage system parameters adjusted 2. Unix system max file descriptors increased from multi-mission default
FEI (Front-End Science Data Processing)	1. Science orbit-based storage structure for raw science data archive (Deferred post-launch)

The above remediation actions were implemented immediately and the GDS team re-focused on the systems engineering process. It was necessary to methodically bridge the systems engineering gap and provide a framework by which the GDS behavior could be characterized quantitatively in response to the flight system behavior.

III. Systems Engineering Approach to Bridging the Gap

The Juno project systems engineering process was very robust and adhered to the *NASA System Engineering Processes and Requirements, NPR 7123.1.* The Juno GDS systems engineering approach was consistent with the 2006 Forsberg-Mooz Dual Vee Architecture and Entity model of systems engineering seen in Fig. 4 [1,2,3]. The identification of systems engineering gaps is part of the systems engineering process represented by this model. The model in Fig. 4(a) ties verification planning to requirements analysis, emphasizes entity validation as input into system validation, and very importantly provides for early problem identification and resolution. This approach was in family with the Juno project's overall TAYF approach adopted early on during payload integration.



(a) (b)

Figure 4. a) Multi-Dimensional Forsberg-Mooz Dual Vee Architecture and Entity Systems Engineering Model. b) System Lifecycle Development Model

To maintain systems engineering integrity, the GDS team focused on the *Develop System Performance Specification and System Validation Plan* box in Fig. 4(b). Innovation was required to validate the system built against the performance requirements that surfaced during initial payload integration. Innovation manifested itself on two fronts. The first was the introduction of an auxiliary systems engineering activity, and the second was to define a monitoring service architecture, which was needed to provide the necessary data to validate the GDS IDP capabilities and behavior. The new systems engineering activity encompassed the following steps: Identification of quality attributes needed to validate the revised architecture in response to the systems engineering gap; identification of software performance metrics to measure the quality attributes; mine ground truth and extract performance metrics; and use quality attributes to verify the system architecture against requirements and to validate the deployed system. Fig. 5 ties the new GDS systems engineering activity to the Vee model.

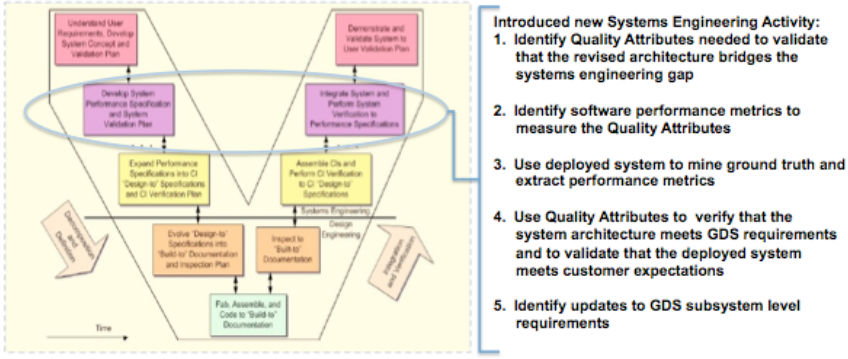


Figure 5. Juno GDS Bridging the Gap Systems Engineering Activity

A. Identification of Quality Attributes

The GDS system level goal was to identify an approach to the assessment of the operational behavior of the IDP. The systems level approach was to leverage from the practice of identifying key quality attributes of a system to the underlying software architecture [4], in order to assess a system's intended behavior. ATLO and Mission Operations stakeholder expectations in combination with the theme of "follow the instrument product" were used to determine the key quality attributes:

1. ***Predictability***
   Based on known science operations scenarios, instrument users need to predict when their products will arrive at their home institution and the GDS users need to predict where the product is in the pipeline.
2. ***Reliability***
   Both instrument and GDS users rely on continuity of IDP service. A key function of the monitoring service is to provide visibility into the operational status for each of the pipeline nodes.
3. ***Recoverability***
   Telemetry frame retransmission is not an option to meet the quick look requirement. Thus, instrument product reconstruction at each of the IDP nodes is a high priority in the case of any pipeline node failure. Recovery of the pipeline requires resumption of product processing. This proved essential during ALTO Thermal Vacuum testing of the perijove* science scenario.
4. ***Traceability***
   In order to assess the above quality attributes, instrument product processing status at each pipeline node is necessary. GDS users verify the processing history and location of the instrument products by "following" the product along the pipeline.

B. Identification of Performance Metrics and The Use of Ground Truth To Obtain Metrics

A set of metrics was defined based on GDS engineering experience with manual trouble-shooting of the pipeline. Knowledge engineering of the logs available at each node was required to determine the optimum approach to compute or derived the metrics. Key metrics are: product size, number of products completed, incomplete and invalid at each node; elapsed product build time at the FDM node; elapsed time between product build time and product arrival at final node; elapsed product arrival time between consecutive nodes; elapsed time between completion of first product and the last product; and elapsed time between arrival of first product and last product at each node. The heritage IDP system components did not advertise the availability of these metrics. However, it was quickly recognized that the IDP was generating useful information and it was a matter of listening to the system. Each component of the as-is system was already generating logs that contained information about the timeliness and the completeness of the data received. In addition, the intermediate data repositories also provided valuable information. The available information from the as-is system components quickly became the source of ground truth for the system level behavior of the pipeline. A parallel can be drawn with earth science use of ground truth for remote sensing. The GDS team developed the IDP monitoring service as a means to survey the movement of products along the pipeline, thus relating product-level data to the overall performance of the IDP as a system. As ATLO and operations readiness marched through the systems engineering lifecycle, the need to understand the behavior of all instances of the IDP in the test bed, ATLO and operations venues became paramount. The monitoring service became a GDS verification cornerstone. It enabled the GDS team to confidently verify the raw science data processing requirements allocated to the GDS, and validate that the pipeline was the right one for Juno.

IV. Architecture Framework for Instrument Data Pipeline Monitoring Service

The first failure of the pipeline occurred November 2010 during a practice run of the electromagnetic interference (EMI)/ electromagnetic compatibility (EMC) test in ATLO. The pipeline needed to be up and running for the run for record of the EMI/EMC test. Thus, implementation of the remediation actions in Table 1 was given highest priority. However, in parallel, the software systems engineering of the monitoring service went into a rapid system development mode. The GDS needed to meet the quick look requirements in a TAYF configuration by March 2011 in support of Thermal Vacuum testing. Key science scenarios were to be exercised during Thermal Vacuum testing. In addition, instrument team expectations put the spotlight on the GDS. The instrument teams expected the same level of immediate visibility into their data from the TAYF GDS, as they had in the local Lockheed flight system test environment.

* The nearest point of the Juno spacecraft in its orbit about Jupiter.

The rapid system development mode included parallel testing of both the monitoring service and the IDP; closed loop feedback with ATLO, instrument and user customers; and analysis of GDS requirements. The analysis of requirements resulted in refinement of software requirements for the monitoring service, and updates to GDS IDP subsystem level requirements. Figure 6 depicts how the monitoring service development cycle fed back into the GDS systems engineering Vee development model. In effect, the monitoring service development effort implemented Step 5 of the engineering activity captured in Fig. 5 *identify updates to GDS subsystem level requirements*.

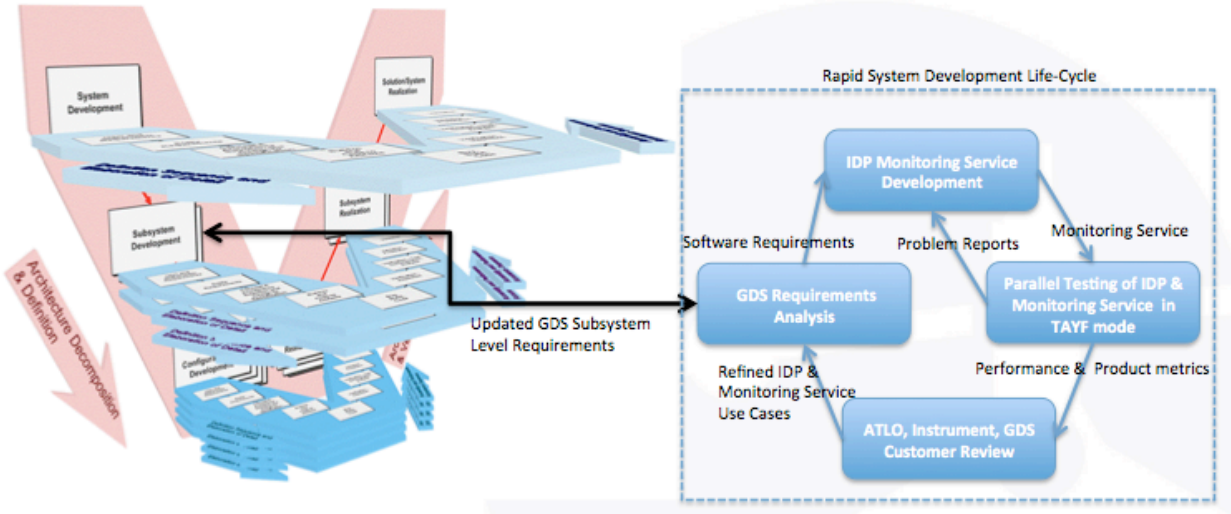


Figure 6. IDP Monitoring Service Rapid Development Life-Cycle

A. Monitoring Service Architecture

The driving requirements for the IDP Monitoring Service were triggered by the project requirement to have the GDS operate in ATLO in a TAYF manner:

- Allow multiple instances of the service to run in parallel
- Compare products built locally in the test bed and ATLO to those built in the operations pipeline at JPL
- Monitor pipeline nodes in real-time
- Not interfere with pipeline performance (read-only mode)
- Auto-detect pipeline product processing errors
- Auto-compute IDP metrics
- Provide first level of metrics analysis
- Display data and analysis
- Provide web-based remote access
- Run continuously and auto-detect pipeline activity (reliability)
- Smooth transition from ATLO operations to post-launch mission operations

Model-based systems engineering was used to capture the architecture. Figure 7 contains a logical view of the pipeline system to be monitored per the driving requirements.

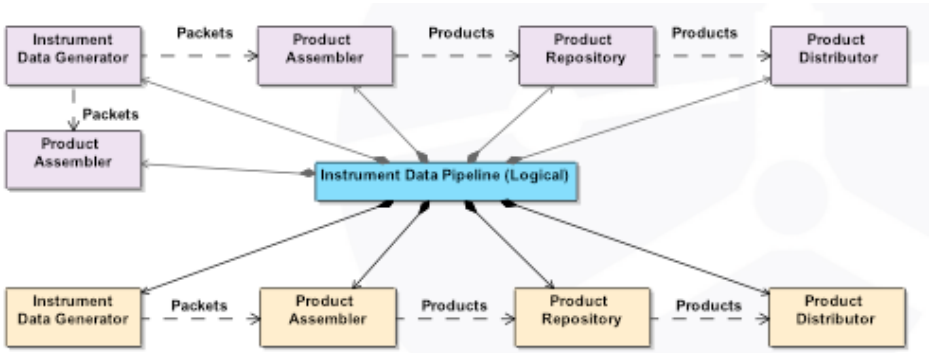


Figure 7. Instrument Data Pipeline logical architecture view

The monitoring service architecture is characterized by plug-and-play monitoring agents; a standard HTTP interface; and open source web server and database framework. This architecture was implementable within the rapid software system development approach. The monitoring services architecture is captured in Figure 8. Figure 9 provides an integrated architecture view, which highlights the "remote sensing" nature of the monitoring service.

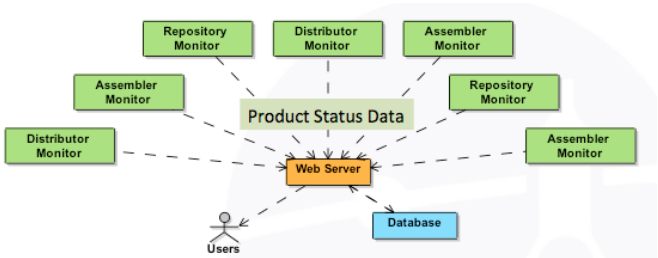


Figure 8. Plug-and-Play Monitoring Agent Architecture View

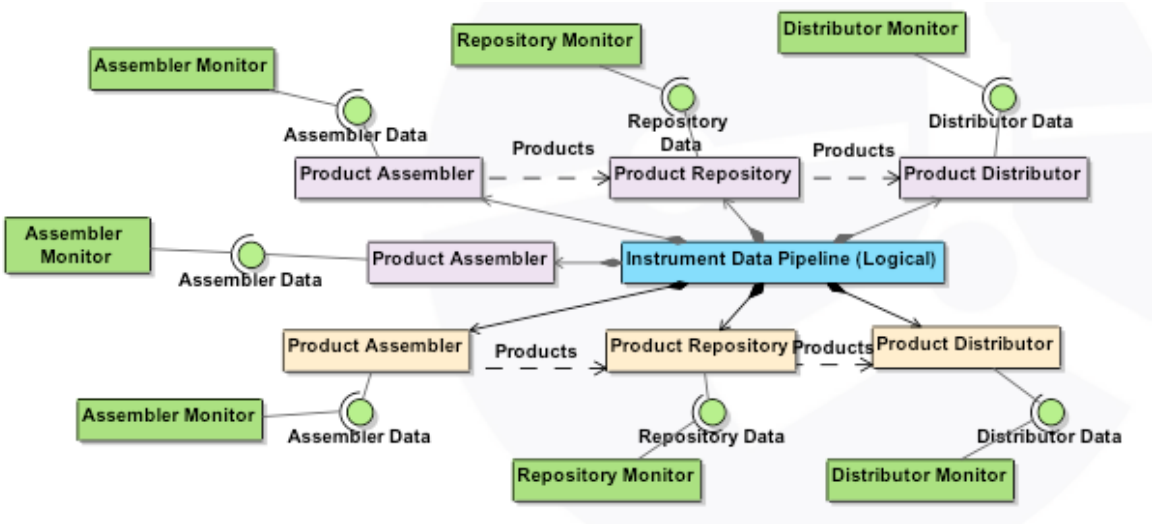


Figure 9. Integrated Pipeline and Distributed Monitoring Service Architecture View

The emphasis on software architecture during the rapid development cycle was a cornerstone in the ability to close the GDS systems engineering gap with integrity in requirements and design. Furthermore, the GDS successfully met the two critical milestones of ATLO Thermal Vacuum test readiness and immediate post-launch instrument checkout readiness, with the existing GDS team due to the automation achieved through the monitoring service.

V. Closing the Systems Engineering Loop From Data Analysis to Successful V&V

In the spring of 2010, as the ATLO campaign was in full swing with a series of flight system science verification tests, the Mission Operations Verification and Validation (V&V) campaign was also in full throttle. The GDS team was at the point where it needed to complete V&V of the Level 3 requirements allocated to the GDS, which included those listed in section II of this paper. Verification analysis was performed with the data collected by the monitoring service. The ATLO science verification tests were leveraged by the GDS to validate that the right system had been built for operations. The discovery of additional flight/ground interoperability behaviors continued through the MOS V&V campaign, which re-calibrated GDS assumptions. For example, the quick look requirement had been initially interpreted in the context of individual instrument products. The science operations scenarios, however, quickly revealed that quick look was expected at two levels: for real-time individual products but also for the set of products associated with the playback of science orbit data. Achieving ground quick look during real-time on-board product generation has different implications than achieving quick look when the spacecraft transmits ~10,000 JIRAM products and ~3,000 JEDI products in playback mode towards the end of the spacecraft tracking pass. Thus, a calibration of quick look expectations also took place. Analysis showed that in the playback mode, ~90% of JIRAM and JEDI products met the quick look requirement. However, the ~10% of products that did not achieve the quick look requirement did not allow the complete data set to be viewed in quick look mode. The results captured in

Table 3 were those presented at the Juno Operations Readiness Review (ORR) June 2011. The results demonstrated a solid understanding on the GDS Instrument Data Pipeline behavior for science operations scenarios, and the performance profile satisfied stakeholder expectations at the ORR. Through the adjustment of CFDP timers, the adjustment of product sizes by the JEDI and JIRAM teams, and the design remediation actions in Table 2, the quick look requirement was met for individual products generated and transmitted in real-time by the spacecraft. The results in Table 3 are for the playback mode at the ATLO downlink rate of 1.6 Mbps.

Table 3. Juno GDS IDP Metrics Presented at Juno ORR June 2011

Data Type	01/20 Science SVT		03/02 TVAC Func Perijove		03/04 TVAC Func Perijove		05/14 Science SVT	
	Num Products	Max Latency[1] (minute)	Num Products	Max Latency[1] (minute)	Num Products	Max Latency[1] (minute)	Num Products	Max Latency[1] (minute)
FGM Lo Rate	28	16 sec	39	2.5	43	3.8	30	0.42
JADE Hi Rate	17	25	14	9 sec	15	9 sec	0	N/A
JADE Lo Rate	617	27	581	5.7	692	3.8	37	1
JEDI 090 Lo Rate	0	N/A	0	N/A	0	N/A	1223	118
JEDI 180 Lo Rate	0	N/A	0	N/A	0	N/A	657	114
JEDI 270 Lo Rate	0	N/A	0	N/A	0	N/A	1223	118
JIRAM Hi Rate	6336	57	7466	108 (1.8 hr)	10404	138 (2.3 hr)[1]	6336	243 (3.9 hr)[1]
JunoCAM Hi Rate	56	16 sec	46	5.7	69	3.6	58	13
MWR Lo Rate	555	1	1232	5.7	1939	6.8	604	30
UVS Hi Rate	867	1	757	5.7	807	5.5	868	64
WAVES Hi Rate	0	N/A	7	12.7	20	39 sec	9	64
WAVES Lo Rate	450	14.4	655	5.5	663	6.9	357	63
Total CFDP Products built	8926		10797		14652		11402	
Total # Files Delivered by FEI to IOT subscriptions	26778		32391		43956		34206	

1 Max latency is worst case relative to an individual product but also represents elapsed time between first product received on ground and last product received by the instrument team for that orbit data set
2 During TVAC demonstrated that could meet Quick Look Data requirement (15 min) by intercepting pipeline at DOM Node
3 First time JEDI and JIRAM exercised in parallel during ATLO Science SVT. CFDP parameters are being revisited

Step 1 of the new Systems Engineering activity captured in Fig. 5(b), addressed the selection of the IDP quality attributes for the purpose of characterizing, and eventually validating, the expected behavior of the IDP system. The goal to measure the identified quality attributes shaped the implementation of the monitoring service and led to the results in Table 3. The outcome of Step 4 in Fig. 5(b) is summarized as follows:

1. ***Predictability***: 100%, assuming nominal flight and ground operations
   During ATLO, the Jupiter polar orbit with close perijove science scenario was tested repeatedly. The IDP metrics gathered for each test were consistent in terms of product statistics and matched the expected product generation by the instrument teams. Thus, through the monitoring service the GDS and instrument users can confidently predict the arrival of the raw science data in terms of product names, arrival sequence in addition to the timing.
2. ***Reliability***: Meets MTBF requirement of at least 1 month and recovery from failure within 24 hours
   The 24x7 continuous and autonomous operations of the monitoring service demonstrated the continuity of operations of the IDP service during the -3, +3 hours (6 hours) perijove science during ATLO tests.
3. ***Recoverability***: 100%
   Resumption of product processing was demonstrated consistently when any of the nodes encountered a problem. Persistence of both product data and processing information was demonstrated at each node. The monitoring service is able to gather metrics after a test or operations pass has occurred by mining the logs generated during the tests and operations passes.
4. ***Traceability***: 100%

Both in real-time and in batch mode, the monitoring service demonstrated the availability and immediate access to product metadata at each IDP node. In real-time, through the monitoring service, a user may follow his products along the pipeline. The monitoring service has a very effective search engine that allows users to query IDP product processing information and CFDP transaction information. Figure 10 contains a snapshot of the monitoring service. The displayed metrics are updated every 5 minutes during real-time data processing. Users can download reports in real-time during a test or during an actual tracking pass, or for both if they are occurring in parallel.

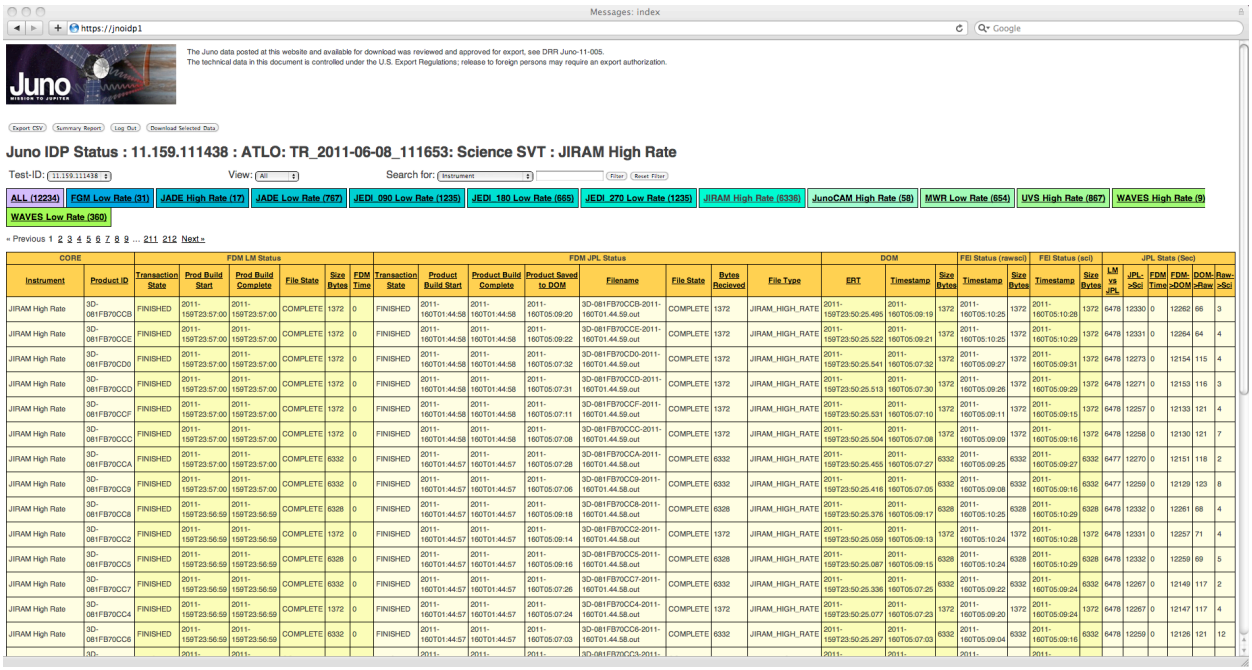


Figure 10. Juno Instrument Data Pipeline Monitoring Service Web Front-End

The bridging of the systems engineering gap was not confined to the GDS systems engineering effort. The metrics computed and generated by the monitoring service are downloadable in spread sheet format both in real-time as data products are received by the ground, and after the fact, when all data has been processed and orbit data set level metrics have been computed. The Juno End-to-End Information System (EEIS) Engineer took advantage of the IDP metrics to perform his own analysis of the payload-ground interactions relative to the expected behavior he had captured in the Flight-Ground ICD.

VI. Conclusion

This paper describes the Juno GDS team systems engineering journey to understand and rectify with a methodical approach the systems engineering gap stemming from heritage assumptions. The primary GDS lessons learned are: the GDS need to probe and understand the impact of payload/spacecraft bus interactions on flight/ground interactions, in order to assess if the heritage assumptions still hold; the benefits to be gained from infusing system level quality attributes definition and measurement into the GDS systems engineering life-cycle process, and the use of quality attributes as a cornerstone of the V&V GDS strategy due to the software-intensive nature of the GDS; and the effectiveness of model-based systems engineering to capture and iterate system architecture in a rapid system development environment. This paper also describes two by-products that future JPL mission GDS will leverage from: a quantified performance characterization of multi-mission instrument data pipeline components; and a monitoring service architecture that is extensible to all nodes of GDS data processing and very importantly, has been captured in model for re-use by the next mission. This paper also extended the concept of ground truth remote sensing to the harvesting of metric data by treating the GDS as a network of ground processing nodes, in order to learn the behavior of the ground system as it evolves during development and once it is deployed as an operations-worthy system.

Acknowledgments

This task was managed out of the Jet Propulsion Laboratory, a division of the California Institute of Technology, under a contract with the National Aeronautics and Space Administration. The work was funded and performed under the project leadership of the Juno Principal Investigator, Dr. Scott Bolton from SwRI, Juno Project Manager, Ms. Jan Chodas from JPL; and Juno Mission System Manager, Chuck Scott, and Deputy manager, Ed Hirst. The authors thank and acknowledge the following team members for their key contributions to the work described in this paper: Vance Heron, monitoring service senior software developer; Violet Torossian, IDP Monitoring Service Integration & Test; Dipak Achhnani, Juno Lead GDS Integration & Test Engineer; Karen Liao, Juno GDS Engineer; Payam Zamani, FEI Lead Subsystem Engineer; Lavin Zhang and Anna Romero, DOM Software Engineers; Curtis Eaton and Esker Davis, Deep Space Operations Integration & Test; Michela Munoz-Fernandez, Instrument Operations Engineer; Tim Kaufman, Juno EEIS Engineer; and Luis Morales, Juno MOS Engineer. In addition, the authors thank and acknowledge the technical support for this effort received from the Lockheed Martin GDS and ATLO Teams, and the Juno Instrument Operations Teams.

References

[1] Forsberg, K., Mooz, H., and Cotterman, H., *Visualizing Project Management: Models and Frameworks for Mastering Complex Systems*, 3rd ed., John Wiley & Sons, Inc., Hoboken, New Jersey, 2005, pp. 341-360.

[2] Estefan, J. A., "Survey of Model-Based Systems Engineering (MBSE) Methodologies," INCOSE TD-2007-003-001, 2007.

[3] Haskins, C. (ed.), "Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities," INCOSE TP-2003-002-03.2.2, 2011.

[4] Barbacci, M., Klein, T. A., Longstaff, C. B., and Weinstock, "Quality Attributes," CMU/SEI-95-TR-021, 1995.

[5] Garlan, D., and Shaw, M., "An Introduction to Software Architecture," CMU/SEI-94-TR-21, 1994.