Planetary Data System

Report Service

Software Requirements and Design Document (SRD/SDD)

Sean Hardman

June 12, 2011
Version 1.0

Jet Propulsion Laboratory
Pasadena, California

CHANGE LOG

Revision	Date	Description	Author
0.1	2009-12-09	Initial draft.	S. Hardman
0.2	2010-03-08	Filled out the document with use cases, requirements and architecture.	S. Hardman
0.3	2010-03-15	Added level 4 requirements for static metrics.	S. Hardman
0.4	2010-05-24	Updated the requirements according to discussions with the SDWG. Other updates to references and diagrams for consistency with other components.	S. Hardman
0.5	2010-06-02	Updated the requirements according to discussions with the SDWG.	S. Hardman
0.6	2010-06-21	Updated the interfaces to detail the common logfile format.	S. Hardman
0.7	2010-09-28	Updated the controlling document reference and cleaned up requirements derivation.	S. Hardman
1.0	2011-06-12	Updated the controlling document references, added the combined log format and included the package evaluation.	S. Hardman

TABLE OF CONTENTS

1.0   INTRODUCTION

The PDS 2010 effort will overhaul the PDS data architecture (e.g., data model, data structures, data dictionary, etc) and deploy a software system (online data services, distributed data catalog, etc) that fully embraces the PDS federation as an integrated system while leveraging modern information technology.

This service provides functionality for capturing and reporting metrics. One or more Commercial Off-The-Shelf (COTS) or Open Source products would most likely satisfy the requirements of this service.

1.1   Document Scope and Purpose

This document addresses the use cases, requirements and software design of the Report service within the PDS 2010 data system. This document is intended for the reviewer of the service as well as the developer and tester of the service.

1.2   Method

This combined Software Requirements and Software Design Document (SRD/SDD) represents the software by defining use cases and requirements and by using architecture diagrams, functional descriptions, context diagrams and data flow diagrams for the high-level design. UML diagrams will illustrate the detailed design.

1.3   Notation

The numbering of the requirements in this document will be formatted as **LX.RPT.AA.X**, where:
- **LX** represents the requirements level where X is a number.
- **RPT** is an abbreviation representing the report requirements section for the specified level.
- **AA** is a two-letter abbreviation representing the requirement sub-category (optional).
- **X** is a unique number within the section and optional sub-category for the requirement.

Following the text of a requirement may be a reference to the requirement or use case from which it was derived. The reference will be in parenthesis. A paragraph following a requirement, which is indented and has a reduced font size, represents a comment providing additional insight for the requirement that it follows. This comment is not part of the requirement for development or testing purposes.

1.4 Controlling Documents

[1] Planetary Data System (PDS) Level 1, 2 and 3 Requirements, March 26, 2006.

[2] Planetary Data System (PDS) 2010 Project Plan, February 2010.

[3] Planetary Data System (PDS) 2010 System Architecture Specification, Version 1.2, May 25, 2011.

[4] Planetary Data System (PDS) 2010 Operations Concept, February 2010.

[5] Planetary Data System (PDS) General System Software Requirements Document (SRD), Version 1.0, June 11, 2011.

1.5 Applicable Documents

[6] PDS User Metrics Collection and Reporting Requirements, PDS Mission Interface Working Group (MIWG), December 16, 2009.

[7] ESDIS Metrics System Summary, Mike Martin, April 30, 2010.

1.6 Document Maintenance

The component design will evolve over time and this document should reflect that evolution. This document is limited to design content because the specification content will be captured in separate documentation (e.g., Installation Guide, Operation Guide, etc.). This document is under configuration control.

2.0   COMPONENT DESCRIPTION

The Report service provides functionality for capturing and reporting metrics. Although each new service will have functional requirements to track metrics, those metrics should be submitted to this service via a common interface or captured in a common format so that they can be harvested this service. The service is not limited to metrics generated by PDS 2010 services, but should also include metrics from the FTP and web logs from each of the nodes as well as any other commonly generated metric. The following diagram details the context of the Report service within the system:
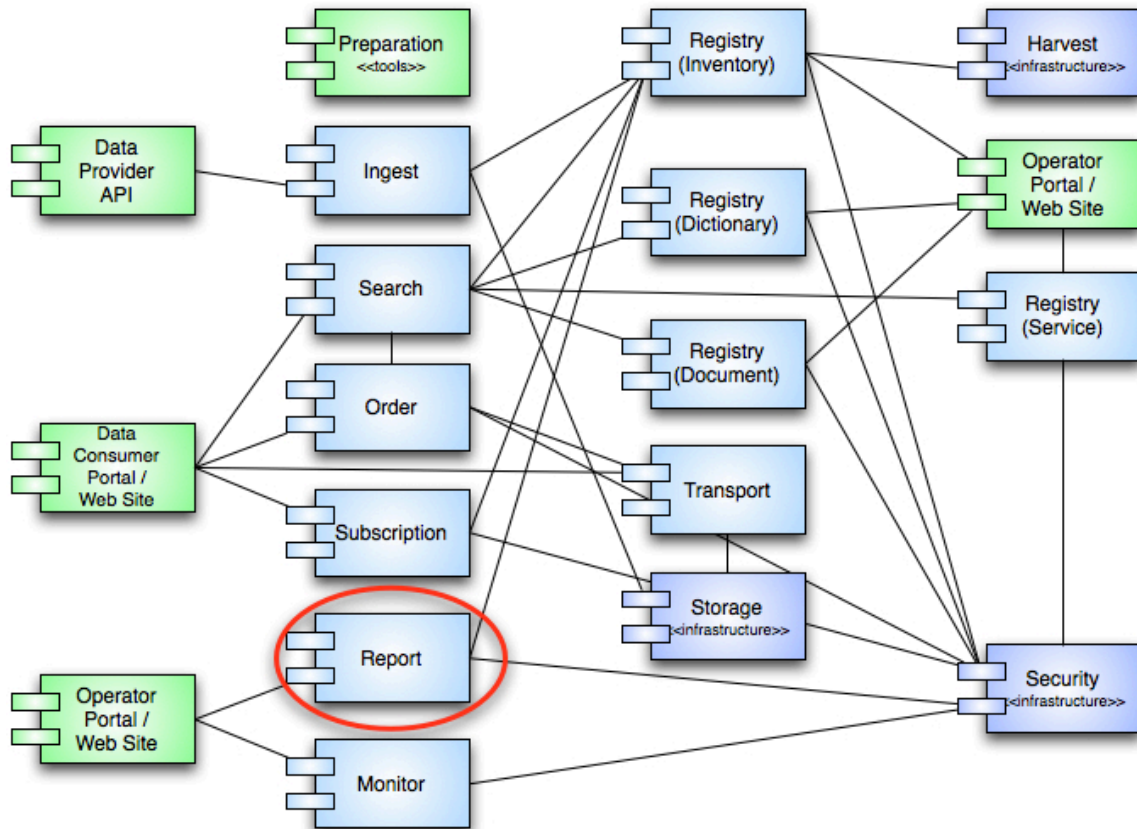
Figure 1: Report Service Context

Although the Report service is a service within the system, it will have very little direct interaction with the other services. This is mainly due to the desire to satisfy these requirements with one or more COTS / Open Source packages. Minimizing the direct interfaces with the service will make it easier in the future to replace it with another package if the need arises.

Metrics information within the PDS focuses mainly on file access requests. They fall into two categories:

Product Transfers

This metric focuses on the distribution of PDS data to data consumers. This information is tracked in FTP logs, web logs and logs generated by specialized transport services. This service should associate transferred files with product information from the Registry service to enhance the meaning of this metric.

Web Page Accesses

This metric focuses on the Node web sites detailing which pages are accessed. This information is available from the web logs but can be further enhanced by utilizing page tagging. The following metrics are possible with page tagging:

- Visits (all sources)
- Unique visitors (by people/session)
- Visits by country/region
- Visits by domain
- Page Views (raw access page counts)
- Pages per Visit
- Bounce Rate (percentage of single-page visits)
- Time per visit (session duration)
- New visits (No previous visits)
- Traffic sources (where did the visitor come from: direct, other site, search engine)
- Entry page
- Exit page
- File type
- File size

Web page tagging is not considered a requirement of this service at this point in time but will be considered for a possible future enhancement.

Another area of metric interest is with the information contained within the Registry service. This would include information on the inventory of products residing at each of the Nodes. Once the Registry service is populated, these metrics are obtainable by querying the aggregate Registry service instance.

Although the current PDS system does not have a common Report service, every Node in the PDS has the responsibility to report their metrics. The service defined in this document will provide the PDS 2010 system with a single service for capturing and reporting metrics within the system.

3.0 USE CASES

A use case represents a capability of the component and why the user (actor) interacts with the component. It should be at a high enough level so as not to reveal or imply the internal structure of the system. An actor is an object (e.g., person, application, etc.) outside the scope of the component but interacts with the component. This section captures the use cases for the Report service based on the description of the component from the previous section. These use cases will be used in the derivation of requirements for the component. The following diagram details the use cases:
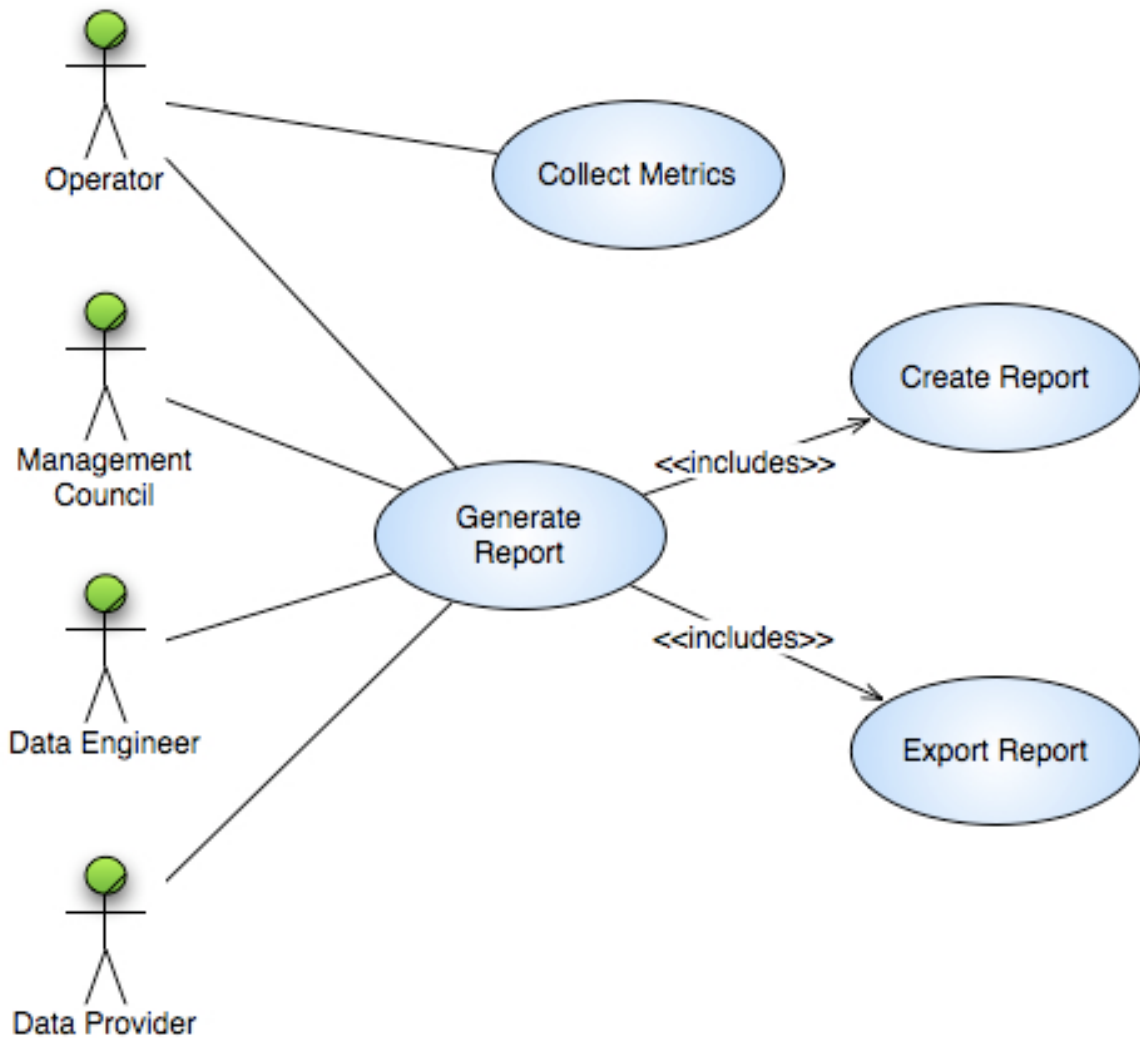
Figure 2: Report Service Use Cases

The above diagram identifies the following actors (represented as stick figures):

Data Engineer
This actor represents a portion of the PDS Technical group that curates the data before and after it enters the PDS system.

Data Provider
This actor represents the mission, instrument team and NASA-funded researcher who are involved with delivering data to the PDS.

Management Council
This actor represents the management level of the PDS.

Operator
This actor represents a portion of the PDS Technical group that is responsible for configuring and monitoring the system.

The following sections detail the use cases identified in the above diagram.

3.1 Collect Metrics

The service collects metrics from various sources across the Engineering and Discipline Nodes including FTP/web logs and service logs. This use case pertains to the Operator actor.

1. Operator configures the Report service to retrieve/receive one or more log files of a defined type.
2. Report Service periodically pulls the specified log files to the service staging area.
3. Report service processes the log file(s) and captures the metrics in the service data store.

Alternative: Report Service Receives Log File(s)
At step 2, the Operator pushes the log file(s) to the service.
    a. Operator periodically pushes the log file(s) to the service staging area.
    b. Return to scenario at step 3.

3.2 Generate Report

Users of the system may generate a metrics report according to their selected criteria. This use case pertains to all actors defined above referred to as "User" in the use case.

1. User authenticates for access to the Report service interface (include Security service Authenticate User use case).
2. User creates a metrics report (include Create Report use case).
3. Report service generates the specified report from the service data store.

4. User exports the generated report (include Export Report use case).

Alternative: User Selects Existing Report
At step 2, the User selects an existing report.
    a. User selects a metrics report from their list of saved reports or the service-wide list of saved reports.
    b. Return to scenario at step 3.

3.3   **Create Report**

Users may create a metrics report or report template according to their own selected criteria. This use case is included as part of the Generate Report use case.

1. User specifies criteria for a metrics report including content, representation, filter and scope.
2. User saves the report as a template for personal reuse or service-wide reuse.
3. Report service saves the specified metrics report template criteria.

Alternative: Report Not Saved
At step 2, the user chooses not to save the report.
    a. User does not save the report.

3.4   **Export Report**

Users may export a generated report in various formats including but not limited to a spreadsheet or PDF file. This use case is included as part of the Generate Report use case.

1. User specifies the export format for the report.
2. Report service transforms the report to the selected export format.

4.0   REQUIREMENTS

The architecture definition phase of the PDS 2010 project resulted in the decomposition of the system into several elements [3]. The Report service does not derive directly from any of those elements but is derived from requirements 2.2.2 and 2.6.3 of the PDS Level 1, 2, and 3 Requirements document [1]. The PDS Mission Interface Working Group (MIWG) also prepared a set of requirements for collection and reporting of metrics [6]. In addition, the PDS Program Manager (PM) also provided requirements for collection of metrics. The level 4 requirements derive from these inputs. The following level 3 requirement is relevant to this service:

2.2.2   PDS will track the status of data deliveries from data providers through the PDS to the deep archive

2.6.3   PDS will integrate the catalog with the system for tracking data throughout the PDS

In addition to the level 4 and 5 requirements specified below, the Report service must also comply with the general service-based requirements found in the General System SRD document [5].

4.1   Level 4 Requirements

The level four requirements in PDS represent subsystem or component requirements at a high level. The following requirements pertain to the Report service:

L4.RPT.1 - The system shall maintain a repository for collection and storage of PDS-wide metrics. (2.2.2, MIWG)

L4.RPT.2 - The system shall collect the following metrics for file access requests at each PDS Node: (2.2.2, MIWG)

   a. Host requesting access (e.g., IP address or DNS host name)
   b. User requesting access (e.g., user name), optional
   c. Date and time of request
   d. Request including file specification
   e. Status of request
   f. Bytes transferred

> This requirement focuses on distribution of data (via HTTP, FTP, etc.) and web content (via HTTP) to users.

11

L4.RPT.3 - The system shall associate a file specification with a registered product in the archive. (2.2.2, 2.6.3, MIWG, PM)

> Not all file access requests will be for files that reside in the archive (e.g., web pages). In these cases, no association will be available.

L4.RPT.4 - The system shall associate a registered product in the archive with the following information: (2.6.3, MIWG, PM)

a. Number of files
b. Number of bytes
c. Time of ingestion
d. Archive state
e. Archive disposition (i.e., delivered to Deep Archive)
f. Curating node
g. Investigation (Mission)
h. Observing system (Instrument Host / Instrument)
i. Collection (Data Set)
j. Release

> This information is available from the Registry Service as registered associations to the Product registration.

L4.RPT.5 - The system shall allow report generation from collected metrics and their associated information. (2.2.2, MIWG, PM)

4.2   Level 5 Requirements

The level five requirements in PDS represent subsystem or component requirements at a detailed level. The following requirements pertain to the Report service:

L5.RPT.1 - The service shall support periodic submission of metrics. (L4.RPT.1, UC 3.1)

L5.RPT.2 - The service shall allow the submission of metrics in the form of a log file. (L4.RPT.2, UC 3.1)

L5.RPT.3 - The service shall utilize a secure transfer protocol for transferring log files across the Internet. (L4.RPT.2, L4.GEN.6, UC 3.1)

> The L4.GEN.6 requirement resides in the General Service SRD [5].

L5.RPT.4 - The service shall support log files from the following sources: (L4.RPT.2, UC 3.1)

a. Web server (e.g., Apache HTTP and Tomcat servers)

b. FTP server
c. PDS 2010 service (e.g., Search, Transport, etc.)
d. Node-specific service (e.g., Imaging's Atlas, Geosciences' ODE, etc.)

L5.RPT.5 - The service shall discover product-related information by querying the Registry service. (L4.RPT.3, L4.RPT.4, UC 3.1)

L5.RPT.6 - The service shall aggregate and store the metrics in a repository. (L4.RPT.1, UC 3.1)

L5.RPT.7 - The service shall control access to the user interface and metrics repository. (L4.GEN.6, UC 3.2)

> The L4.GEN.6 requirement resides in the General Service SRD [5].

L5.RPT.8 - The service shall allow users to tailor reports and report templates as follows: (L4.RPT.5, UC 3.2, UC 3.3)

a. Content (attribute selection based on desired metric)
b. Representation (layout of the attributes and/or chart generation)
c. Filter (exclusion of bots, internal or other request sources)
d. Scope (curating node, investigation, observing system, collection, date/time range)

L5.RPT.9 - The service shall allow users to save report templates for reuse. (L4.RPT.5, UC 3.2, UC 3.3)

L5.RPT.10 - The service shall allow periodic generation of reports from saved templates. (L4.RPT.5, UC 3.2, UC 3.3)

L5.RPT.11 - The service shall export reports in the following formats: (L4.RPT.5, UC 3.2, UC 3.4)

a. Spreadsheet
b. HTML
c. PDF

5.0 DESIGN PHILOSOPHY, ASSUMPTIONS, AND CONSTRAINTS

The intent of the Report service is to provide a PDS-wide capability to capture and report on metrics. There are several commercial and open source packages on the market that will satisfy these requirements. The development team will evaluate a couple of these packages against the requirements with the selected package or packages integrated into the PDS 2010 system. See Appendix B – Package Evaluation for details on this evaluation.

6.0   ARCHITECTURAL DESIGN

The architectural design covers the component breakdown within the service, external/internal interfaces and the associated data model.

6.1   Component Architecture

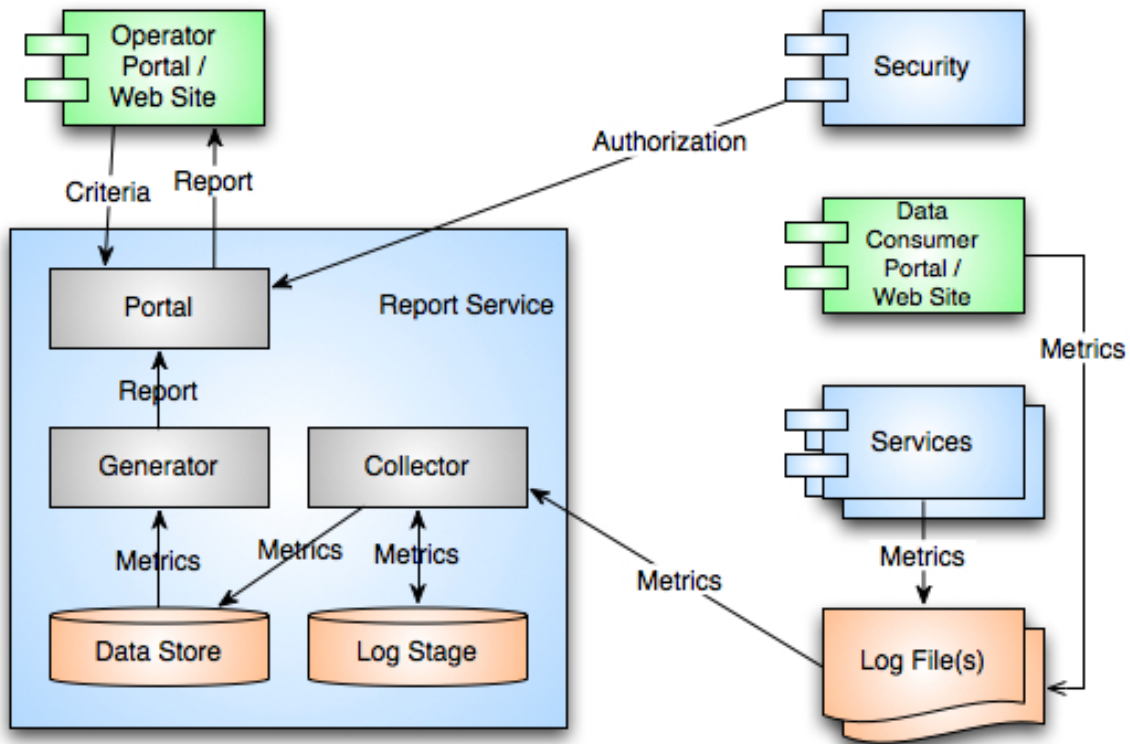The following diagram details the architecture of the Report service:

Figure 3: Report Service Architecture

The Report service provides two main functions. The first is to collect and aggregate metrics from multiple sources (services that produce log files and tagged web pages). The second is to provide an interface for tailoring and generating reports from the aggregated metrics.

6.2   Interface Design

The external and internal interfaces are subject to the selected packages but all of the packages under consideration support the Apache/NCSA Combined Log format for input of metrics. The format is as follows:

```
remotehost rfc931 authuser [date] "request" status bytes
referrer user_agent cookies
```

- remotehost
  Remote hostname (or IP number if DNS hostname is not available, or if DNSLookup is Off.
- rfc931
  The remote logname of the user.
- authuser
  The username as which the user has authenticated himself.
- [date]
  Date and time of the request.
- "request"
  The request line exactly as it came from the client.
- status
  The HTTP status code returned to the client.
- bytes
  The content-length of the document transferred.
- referrer
  The URL which linked the user to your site.
- user_agent
  The web browser and platform used by the visitor to your site.
- cookies
  Cookies take the form KEY = VALUE. Multiple cookie key value pairs are delineated by semicolons(;).

6.3   Data Model

The Report service does not have an associated data model at this time.

7.0   ANALYSIS

The decision to look strictly at commercial and open source packages centers on previous experience by the Engineering Node staff with the Physical Oceanography Distributed Active Archive (PO.DAAC) project. The managing entity for that project, the Earth Observing System Data and Information System (EOSDIS), uses a commercial package for capturing and reporting on metrics from the multiple DAACs. An analysis of their system is captured in the ESDIS Metrics System Summary [7] report. Their approach and the software packages that they utilize and similar packages were evaluated against the requirements in section 4.0. See Appendix B – Package Evaluation for details on this evaluation.

8.0   IMPLEMENTATION

The PDS 2010 system is a phased implementation with increasing capabilities delivered in three planned builds. The builds are as follows:

- **Build 1** – This build consists of the Ingestion subsystem including the Security, Harvest, Registry (Inventory, Dictionary, Document, Service) and Report components along with the Data Provider tool suite.
- **Build 2** – This build consists of the Distribution subsystem including the Search and Monitor components along with a revised web site and general portal applications.
- **Build 3** – This build consists of enhanced user capabilities include the Order and Subscription components along with integration of Discipline Node applications and science services.

The Report service is planned for delivery in Build 1. This initial delivery will support candidate product testing. Additional capabilities are planned for follow-on deliveries as testing progresses and desired features are identified.

The scenario for deployment is to run a single instance of the Report service at the Engineering Node. This will allow each Node to submit their metrics to a centralized location. The following diagram depicts this deployment scenario:
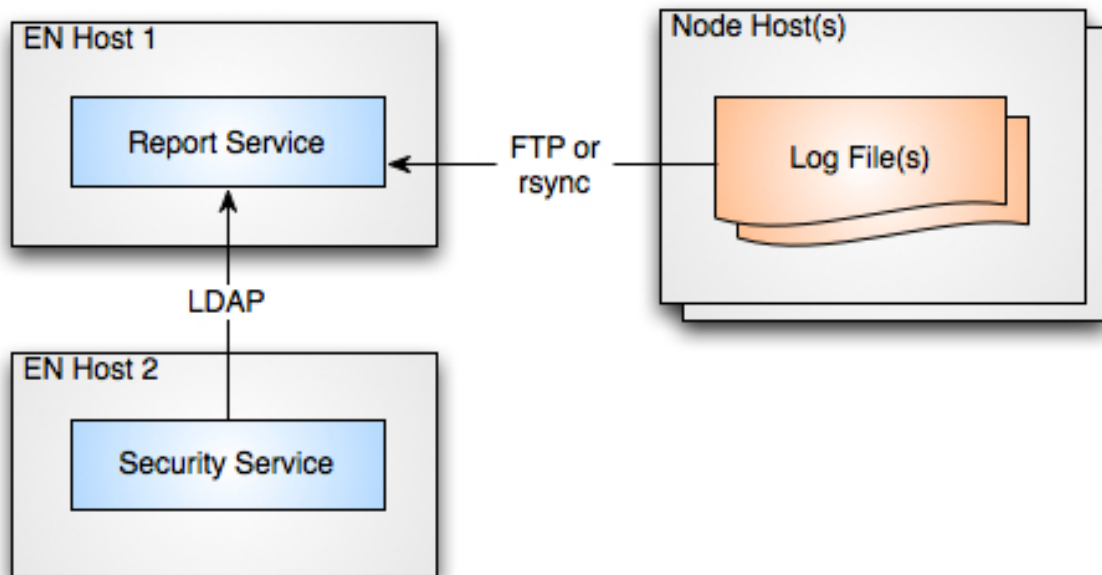
Figure 4: Report Service Deployment

Nodes will submit their log files via a File Transfer Protocol (FTP) interface or via rsync where supported. The Security service provides authentication via a

Lightweight Directory Access Protocol (LDAP) interface for the portal portion of the Report service.

9.0   DETAILED DESIGN

Since we are integrating an off-the-shelf software package, there is no detailed software design for this component.

APPENDIX A   ACRONYMS

The following acronyms pertain to this document:

API	Application Programming Interface
COTS	Commercial Off-The-Shelf
FTP	File Transfer Protocol
EOSDIS	Earth Observing System Data and Information System
ESDIS	Earth Science Data and Information System
JPL	Jet Propulsion Laboratory
LDAP	Lightweight Directory Access Protocol
NASA	National Aeronautics and Space Administration
MIWG	Mission Interface Working Group
PDS	Planetary Data System
PM	Program Manager
PO.DAAC	Physical Oceanography Distributed Active Archive Center
RFC	Request for Comments
SDD	Software Design Document
SRD	Software Requirements Document
UC	Use Case
W3C	World Wide Web Consortium

APPENDIX B   PACKAGE EVALUATION

The following spreadsheet compares the top three software packages against the Report service requirements:

Level 4 Requirement	Level 5 Requirement	AWStats	Google Analytics	Sawmill
L4.RPT.1 - The system shall maintain a repository for collection and storage of PDS-wide metrics.	**L5.RPT.1** - The service shall support periodic submission of metrics. **L5.RPT.6** - The service shall aggregate and store the metrics in a repository.	- Software uses files as acting "repository" for log data. No database is used. - In-house development may be required to submit metrics.	- Google will provide the necessary repository. - Historical metrics cannot be submitted. - Current metrics are extracted dynamically using Javascript tags	- Enterprise Version - Uses MySQL DB to store log information. - Scheduler can be used to periodically download (FTP) log files from nodes. - Import functionality is built in to allow user to submit metrics.
L4.RPT.2 - The system shall collect the following metrics for file access requests at each PDS Node…	**L5.RPT.2** - The service shall allow the submission of metrics in the form of a log file. **L5.RPT.4** - The service shall support log files from the following sources…	- Metrics are collected by server logs, so most information provided in those logs can be used in a report, both HTTP and FTP. - As long as format of log is similar to common log formats, they can be included in reports.	- Metrics are only collected from HTTP servers and require tagging of pages in Javascript. - For FTP downloads, specific links that lead to a file to be downloaded (i.e. href="ftp://…") can be tagged and tracked. - Log files submissions are NOT supported.	- Any information contained within server logs can be reported by Sawmill. - Can process over 700 log file formats.
L4.RPT.3 - The system shall associate a file specification with a registered product in the archive. **L4.RPT.4** - The system shall associate a registered product in the archive with the following information…	**L5.RPT.5** - The service shall discover product-related information by querying the Registry service.	- if the registered product is referenced in the query string (GET request) of the file being requested, otherwise, it would have to be developed in-house. - Any further associations would require further development.	- File access can only be track if it is requested in a web page and can be tagged separately.	- If registered product is referenced in query string. - Can use configuration to develop report relationships between products and files (would require some development to create this configuration)
L4.RPT.5 - The system shall allow report generation from collected metrics and their associated information.	**L5.RPT.8** - The service shall allow users to tailor reports and report templates as follows… **L5.RPT.9** - The service shall allow users to save report templates for reuse. **L5.RPT.10** - The service shall allow periodic generation of reports from saved templates. **L5.RPT.11** - The service shall export reports in the following formats…	- Report content can be fairly easily modified. - Code would be required to modify config files and reports would have to be re-generated. - Layout of reports is not modifiable other than content. - Bot filters are applied automatically. Other filters would require re-generating reports. - Initial scope is based on date/time, but user interaction with reports (links to secondary, detailed reports) allows scope to be narrowed. - Main concern is with re-generating reports. Process is time consuming, so any user prefs should be specified ahead of time. - The "template" information would be included in the config files provided by Awstats. An actual template cannot be viewed unless dummy data is provided. - Export in spreadsheet would require in-house software to reformat - Export in HTML or PDF are available through the sofware or a plugin.	- Google provides very malleable dashboard and reports that can be modified and changed to provide personalized reports based on user needs. - Plug-ins are available to allow for export of reports in various formats.	- Admin interface provided to view and modify reports and report groups. - Export to Excel and PDF available.
L4.RPT.6 - The system shall secure Personally Identifiable Information (PII).	**L5.RPT.3** - The service shall utilize a secure transfer protocol for transferring log files across the Internet. **L5.RPT.7** - The service shall control access to the user interface and metrics repository.	- This part of the report generation can be removed, but the IP addresses will still be present in the server logs. - Access to the service will be completed by Security Service.	- It is possible to mask the last number in an IP address so that the values stored cannot be tracked. - Storage and access control will be completed by Google.	- Login interface is provided. -SFTP and SSH can be used for secure file transfers. - Database or File repository will be housed on secure machine.
Additional Features		- Sections can be created to compare any type of product information as long as it is contained in the URL query		
In-House Development		- Back-end SW to modify config files for templates - Back-end DB to associate users to logs submitted - Templates creation interface - Batch script to run reports on a weekly/monthly basis - Include in SW possibility to combine several machines logs if same information is requested - Login screen - Reports front page to combine the reports for a specific user from several machines - To allow for batch processing, may use email of reports, rather than creating an interface.	- Option 1: Develop own interface to reports information using Analytics Data API. - Option 2: No development, besides documentation. Point users to Google and allow them to personalize their interface to their liking.	Develop configuration to map registry service data.