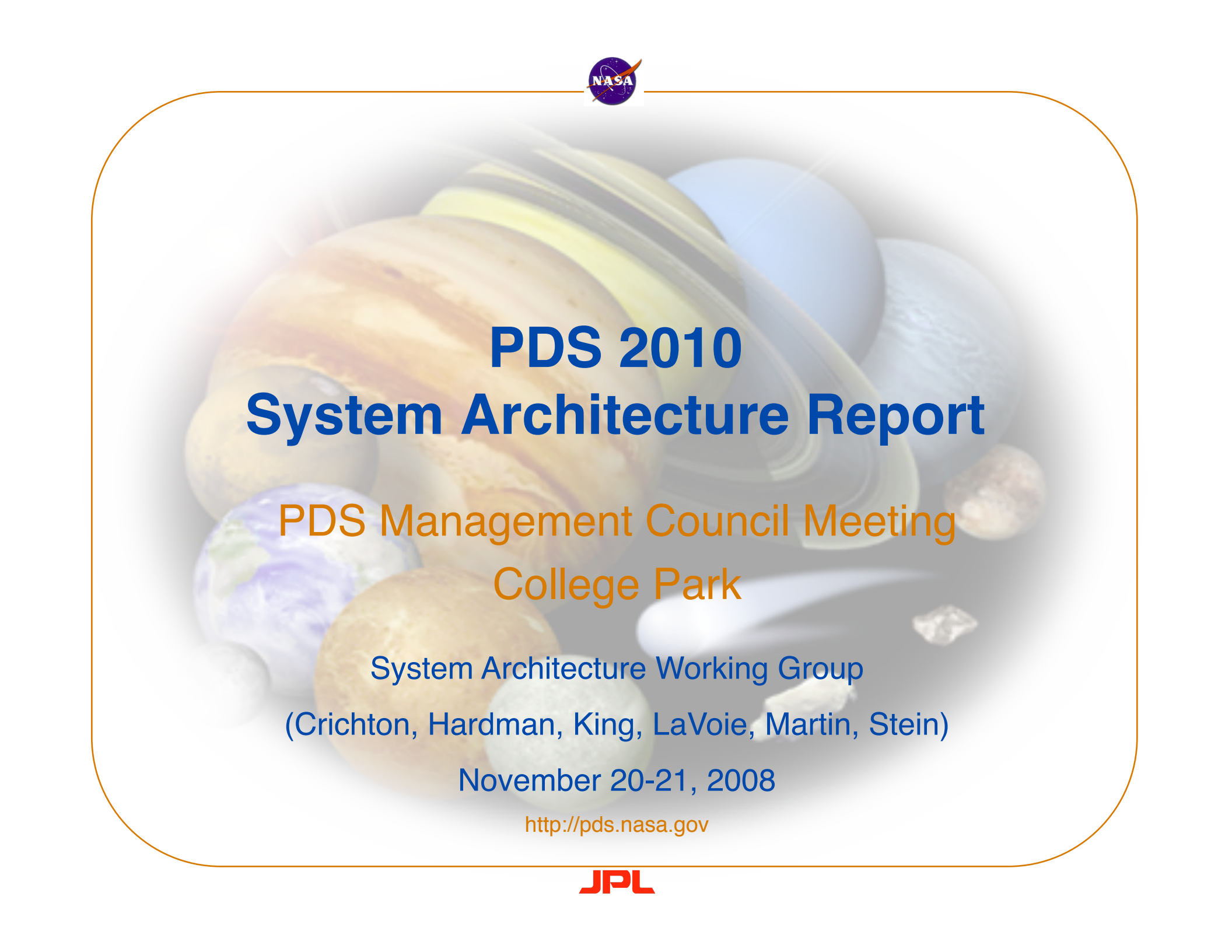


PDS 2010
System Architecture Report

PDS Management Council Meeting
College Park

System Architecture Working Group

(Crichton, Hardman, King, LaVoie, Martin, Stein)

November 20-21, 2008

http://pds.nasa.gov

Topics

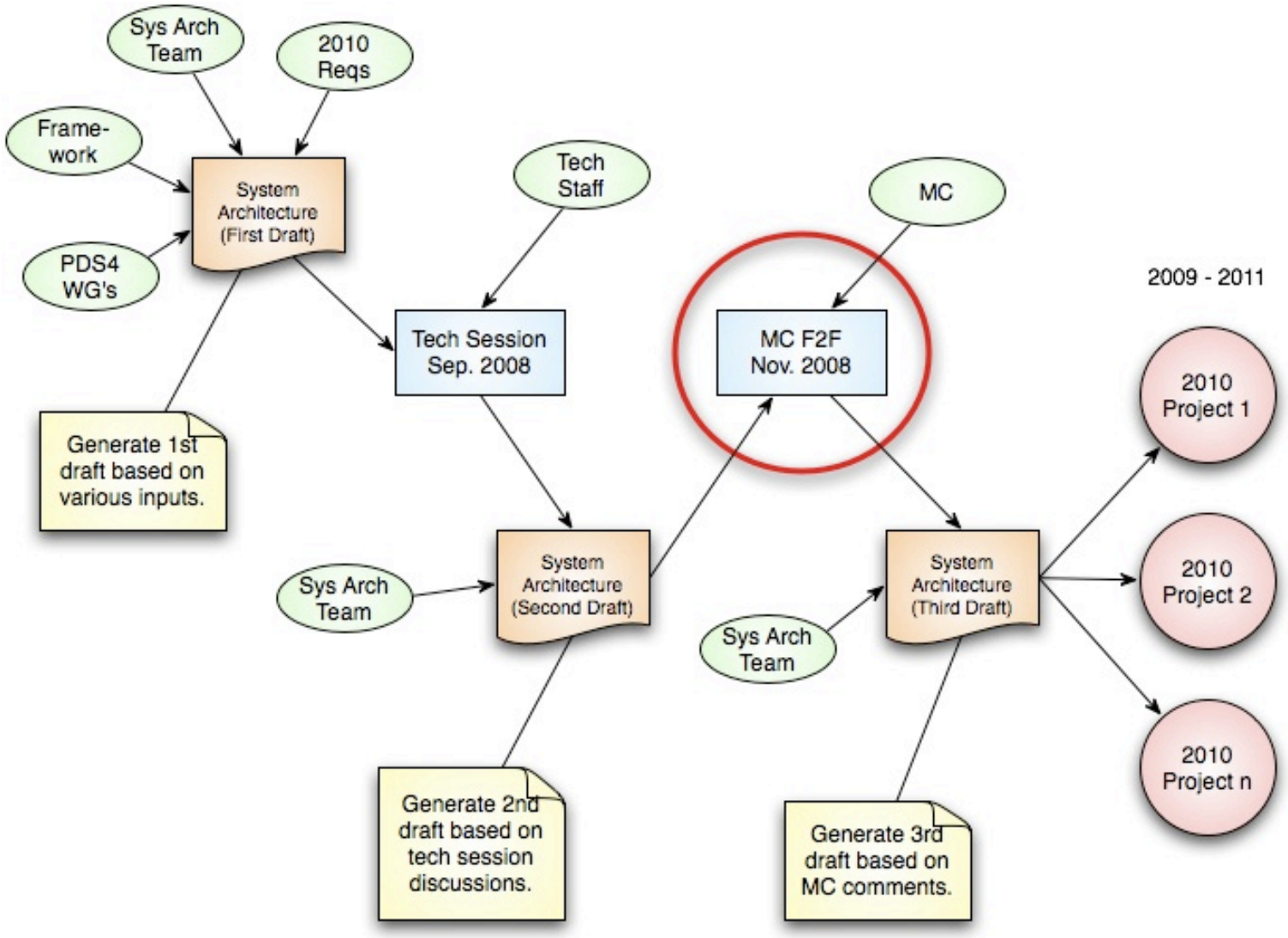
- Objectives
- Roadmap
- Scope
- Progress To Date
- Architecture
  - Service-Oriented Architecture
  - Service Identification
  - Product Lifecycle
  - Service Provisioning
- Addressing Architectural Drivers
- Plans

Objectives

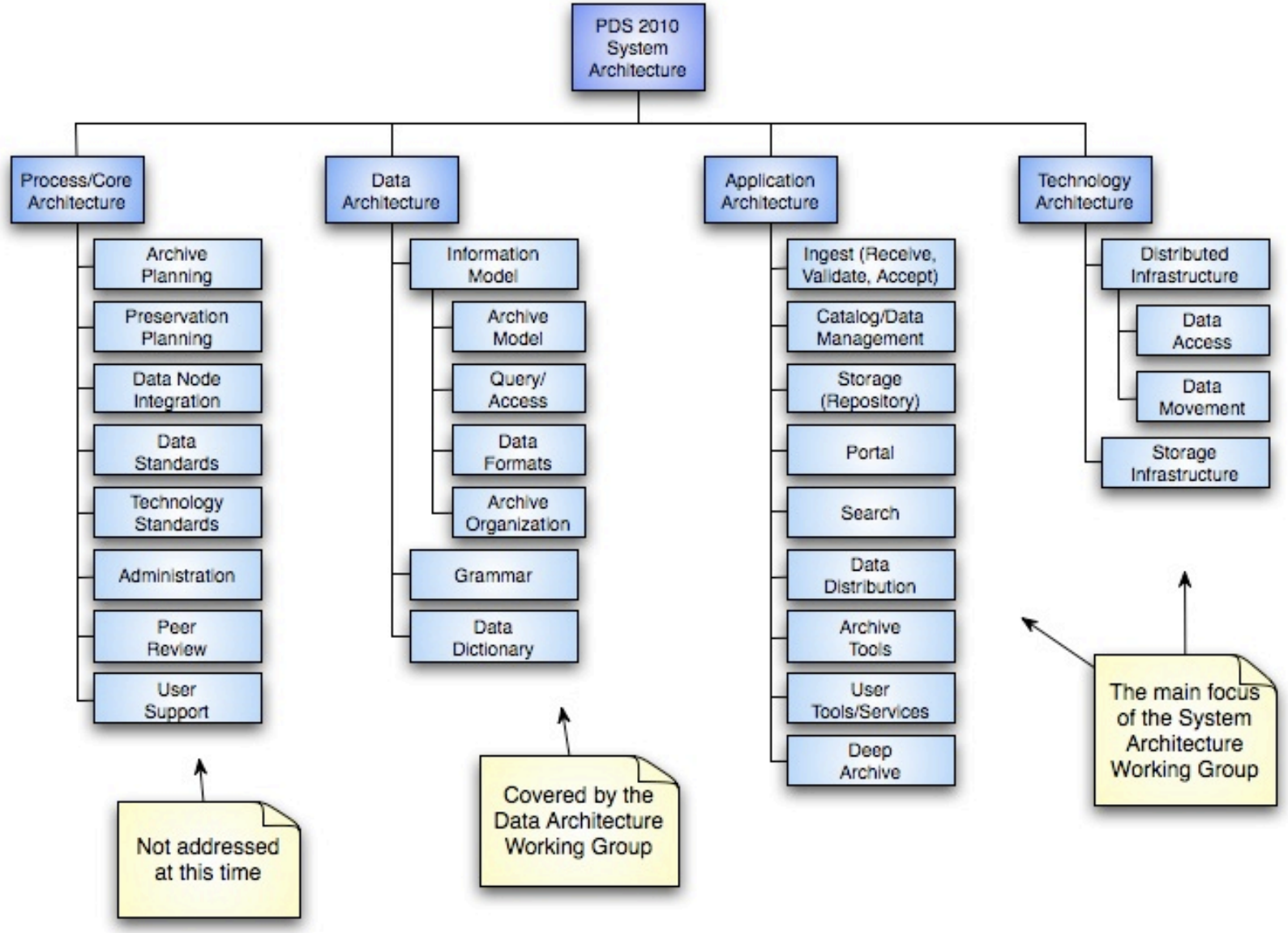
- The objective of this working group is to define a preliminary System Architecture for PDS, which will encompass PDS 2010 and future projects.
  - This includes projects developed at the Engineering Node,
  - As well as projects within the scope of the System Architecture at the Discipline Nodes.

- The objective of this presentation is to demonstrate to the MC that the work to define a System Architecture is progressing and will be a solid basis for design and development of PDS 2010.

Roadmap


Sys Arch Team
2010 Reqs
Frame-work
PDS4 WG's
System Architecture (First Draft)
Generate 1st draft based on various inputs.
Tech Staff
Tech Session Sep. 2008
Sys Arch Team
System Architecture (Second Draft)
Generate 2nd draft based on tech session discussions.
MC
MC F2F Nov. 2008
Sys Arch Team
System Architecture (Third Draft)
Generate 3rd draft based on MC comments.
2009 - 2011
2010 Project 1
2010 Project 2
2010 Project n

Scope


PDS 2010 System Architecture
Process/Core Architecture
Archive Planning
Preservation Planning
Data Node Integration
Data Standards
Technology Standards
Administration
Peer Review
User Support
Data Architecture
Information Model
Archive Model
Query/ Access
Data Formats
Archive Organization
Grammar
Data Dictionary
Application Architecture
Ingest (Receive, Validate, Accept)
Catalog/Data Management
Storage (Repository)
Portal
Search
Data Distribution
Archive Tools
User Tools/Services
Deep Archive
Technology Architecture
Distributed Infrastructure
Data Access
Data Movement
Storage Infrastructure
Not addressed at this time
Covered by the Data Architecture Working Group
The main focus of the System Architecture Working Group

Tech Session Findings

- The System Architecture will not address administrative functions.
  - Beyond providing functionality for capturing and reporting metrics, the architecture will not address functions like budgeting, scheduling calendars and status reporting.

- The Data Architecture will focus on modeling observational and associated ancillary data.
  - Follow-on efforts can incorporate data related to deliveries, reporting, tracking, etc.

- Sub-nodes and data nodes are subject to the System Architecture.
  - Service interfaces or wrappers will be provided to aide in interfacing or integration.

Tech Session Findings (cont)

- Existing mission project interfaces are subject to the System Architecture.
  - The general consensus was that most existing projects did not need to be grandfathered, but each project should be considered on a case-by-case basis.
  - The main issue here is whether a project is migrated or bridged to the new system.

- The general consensus was that the PDS 2010 system should have a rigid core while allowing local extensions.

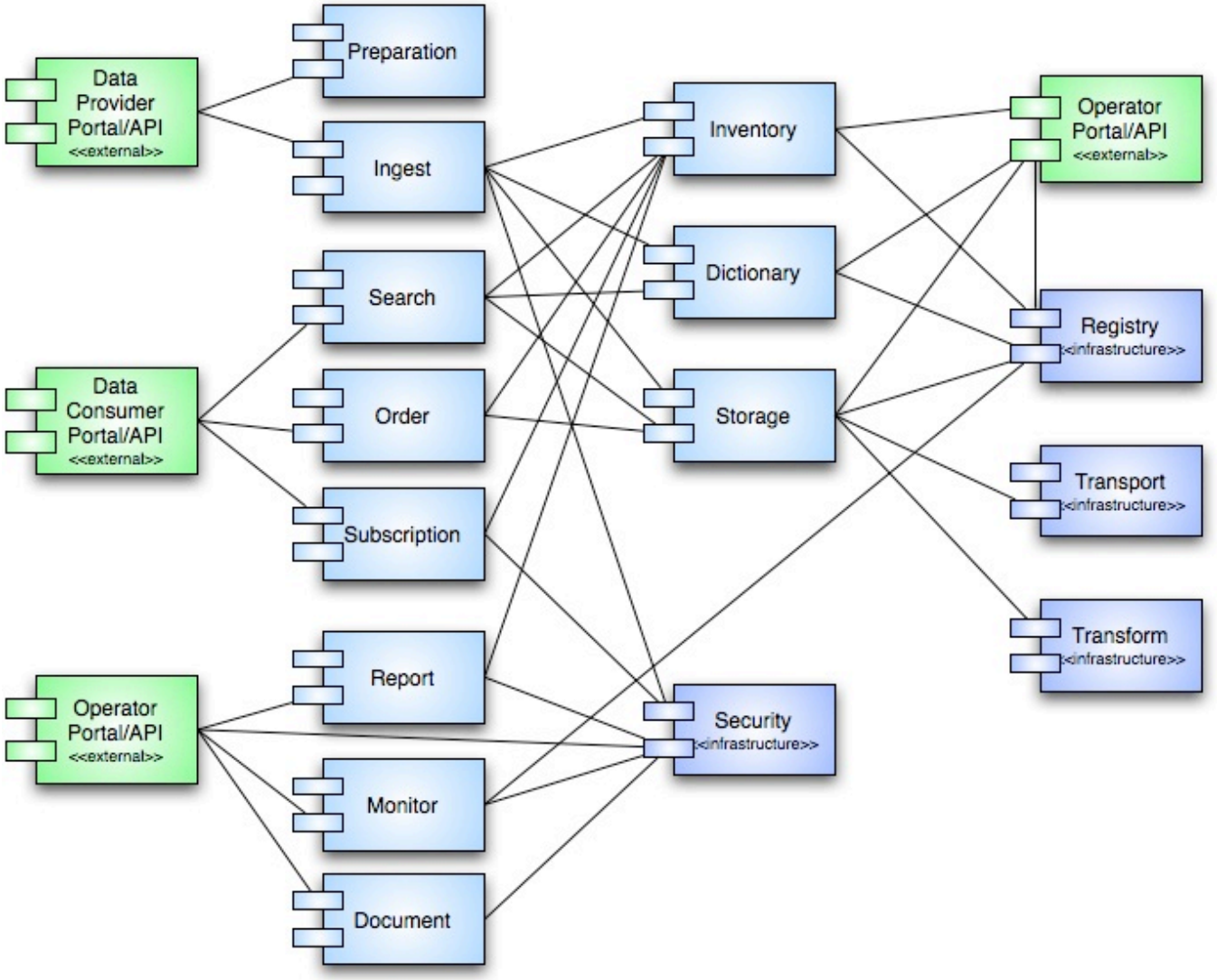
Progress To Date

- Architecture Framework
  - The Open Group Architecture Framework (TOGAF) was selected to help guide the system architecture development effort.
- Architectural Drivers
  - Inherited from the PDS4 Architecture Study Team.
- Architectural Principles
  - Based on the initial list from the PDS4 Architecture Study Team.
  - Modified and organized based on the TOGAF process.
  - Refined and added to based on input from the Tech Session.
- Viewpoints and Views
  - Identified and defined to better determine the artifacts to be developed for communicating the system architecture.
- Services
  - An initial set of services have been identified (derived from the architectural elements).
  - Refined and added to based on input from the Tech Session.
  - An initial cut at provisioning the services to reflect their distributed versus centralized nature.

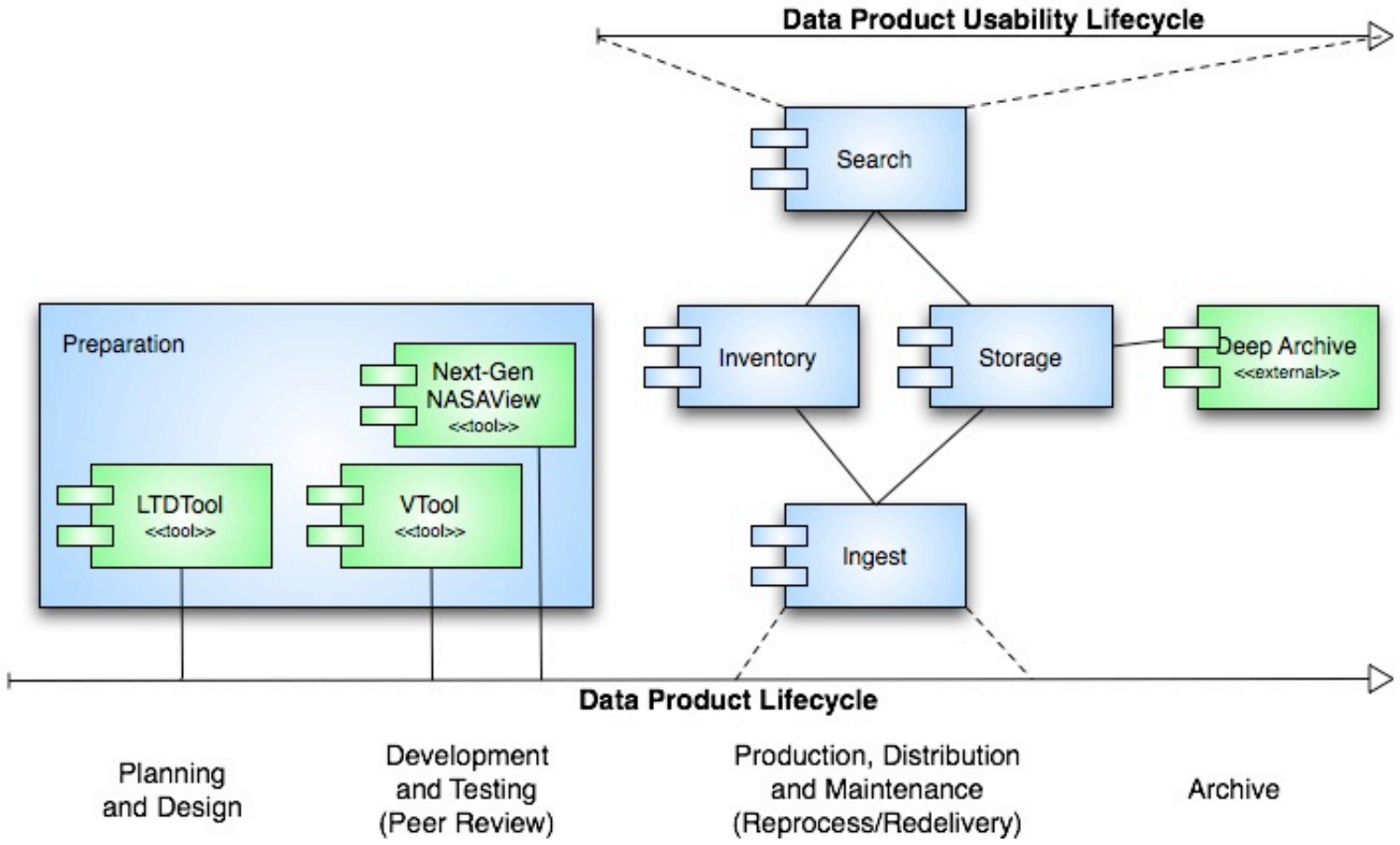
Service-Oriented Architecture (SOA)

- An SOA was deemed most appropriate for satisfying requirement 2.8 ("… maintain a distributed architecture …").
- SOA provides methods for systems development and integration where systems group functionality around organization processes and package these as interoperable services on the network. (Adapted from Wikipedia)
- There are several advantages to adopting SOA:
  - Captures many of the best practices of previous architectures.
  - Well suited for a distributed system.
  - Promotes "loose coupling", "software reuse", "encapsulation" along with other hot buzz phrases in software development today.
  - A service-based architecture provides currency and timeliness for the system.
- The work isn't complete though, there are still several options to consider for deployment.
  - This architecture can operate under several protocols (e.g., SOAP, REST, Web Services, etc.).
  - This decision and others like it will be made during the design phase.

Service Identification



Product Lifecycle / System Interaction



- The lifecycle phases are based on the APG.
- Planning through testing phases will be supported via a tool suite.
- Interaction with the “system” begins with the production phase.

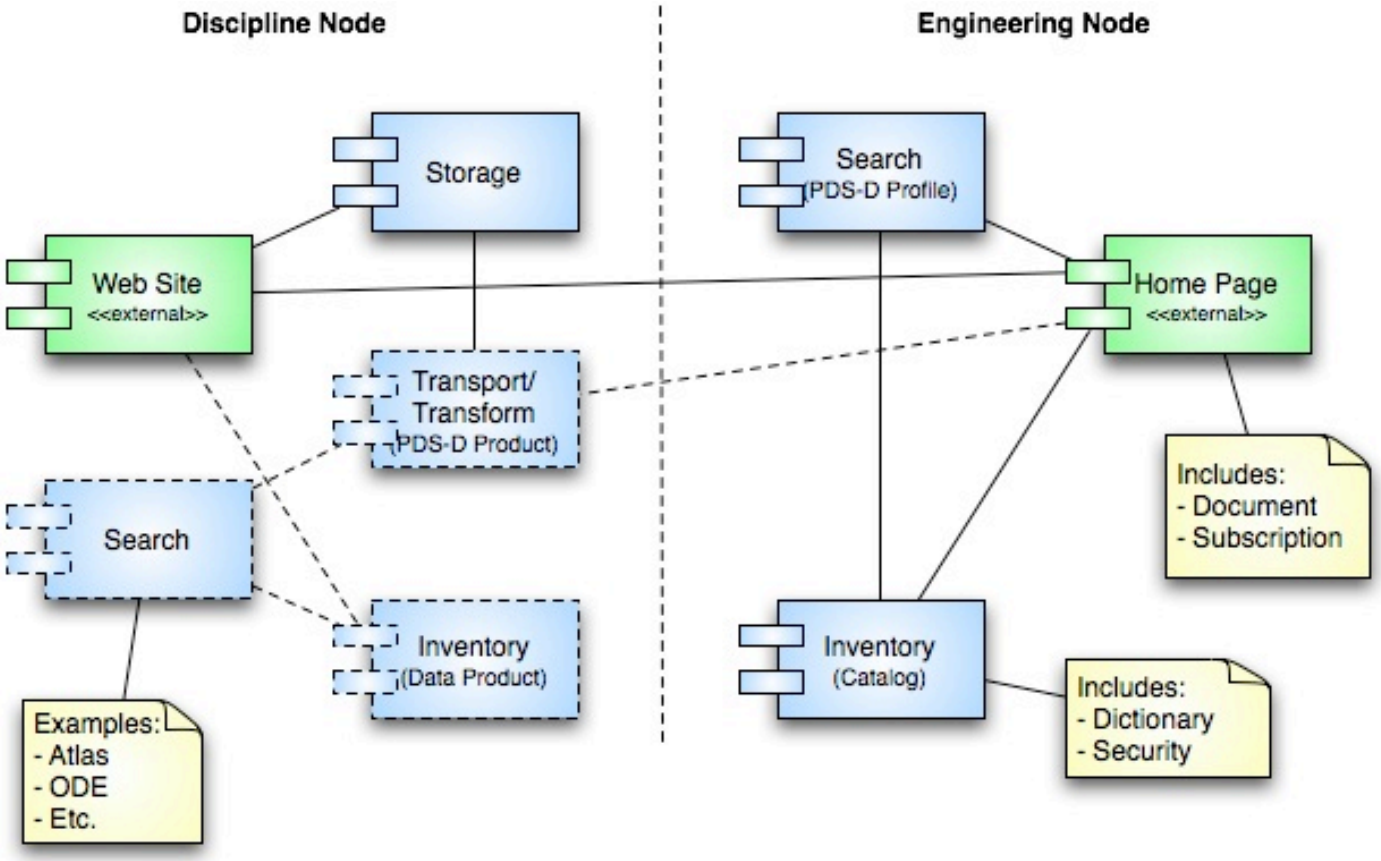
Service Provisioning

- Provisioning details how services can be deployed centrally versus distributed at the nodes or whether they will have common or node-specific implementations.

- First we offer a comparison of the current “service” provisioning with what is being proposed.

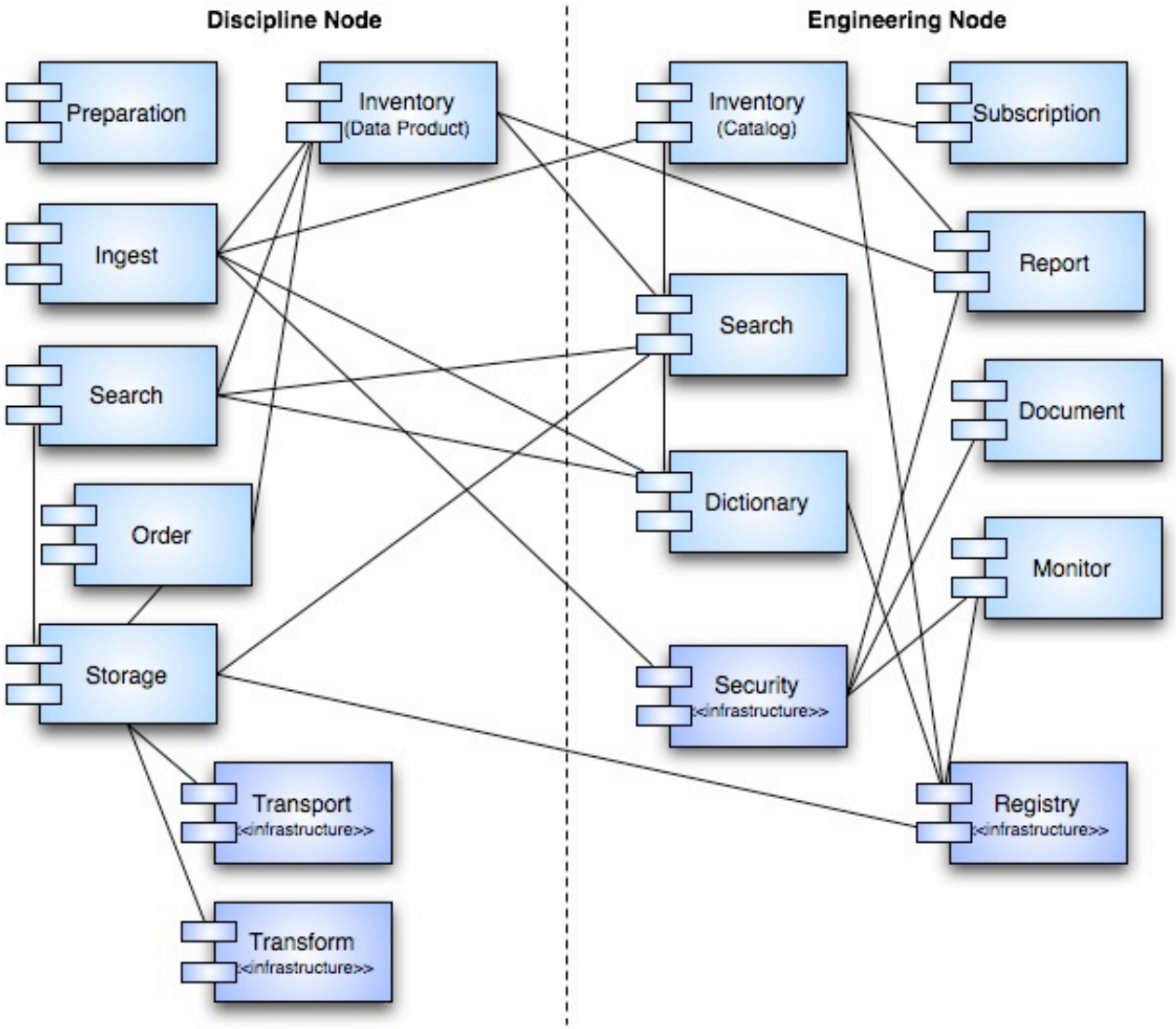
- The diagrams that follow depict provisioning based on functional scenarios:
  - Ingestion
  - Search (originated from the Engineering Node)
  - Search (originated from the Discipline Node)
  - Deep Archive
  - Miscellaneous Operations

Service Provisioning
Current "Services"

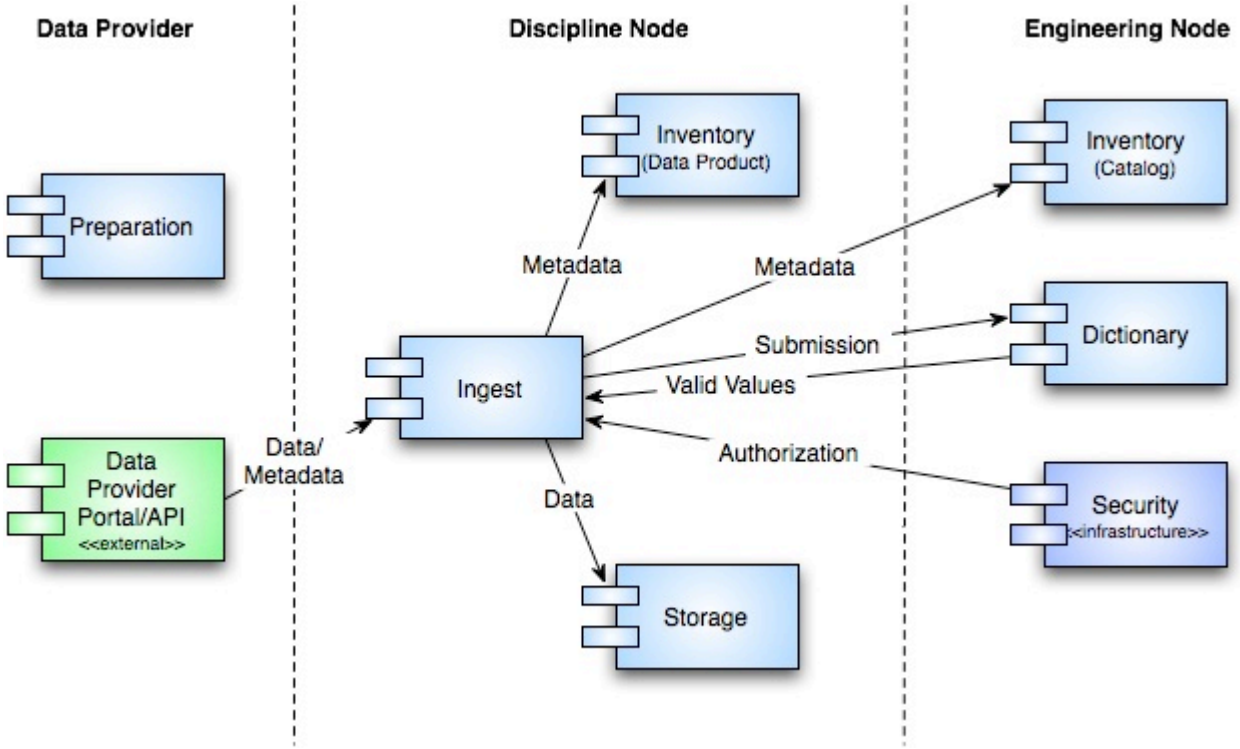


- The service names correspond with the proposed PDS 2010 services for consistency.
- The dashed services are not available at all nodes.

Service Provisioning
Proposed

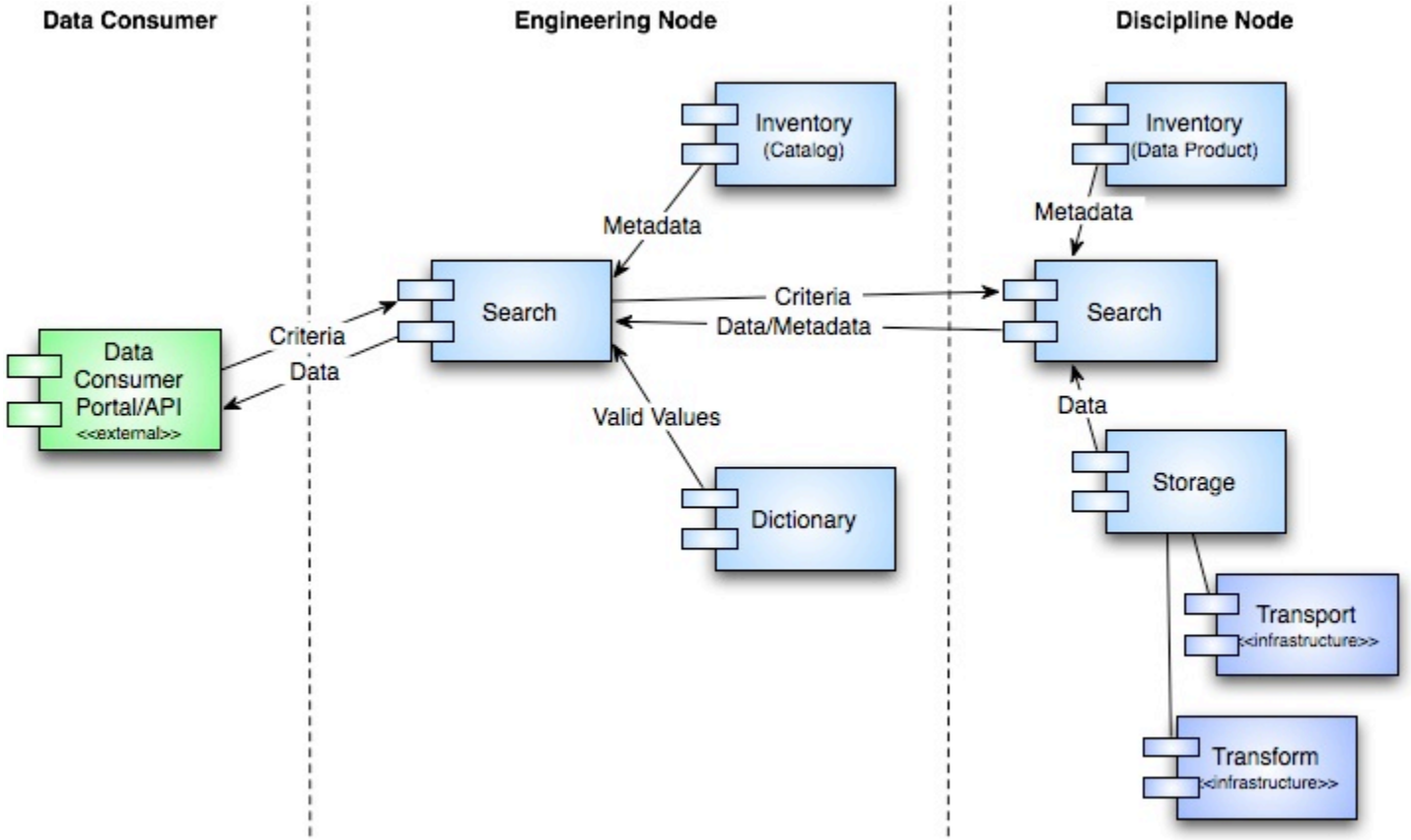


Ingestion Scenario



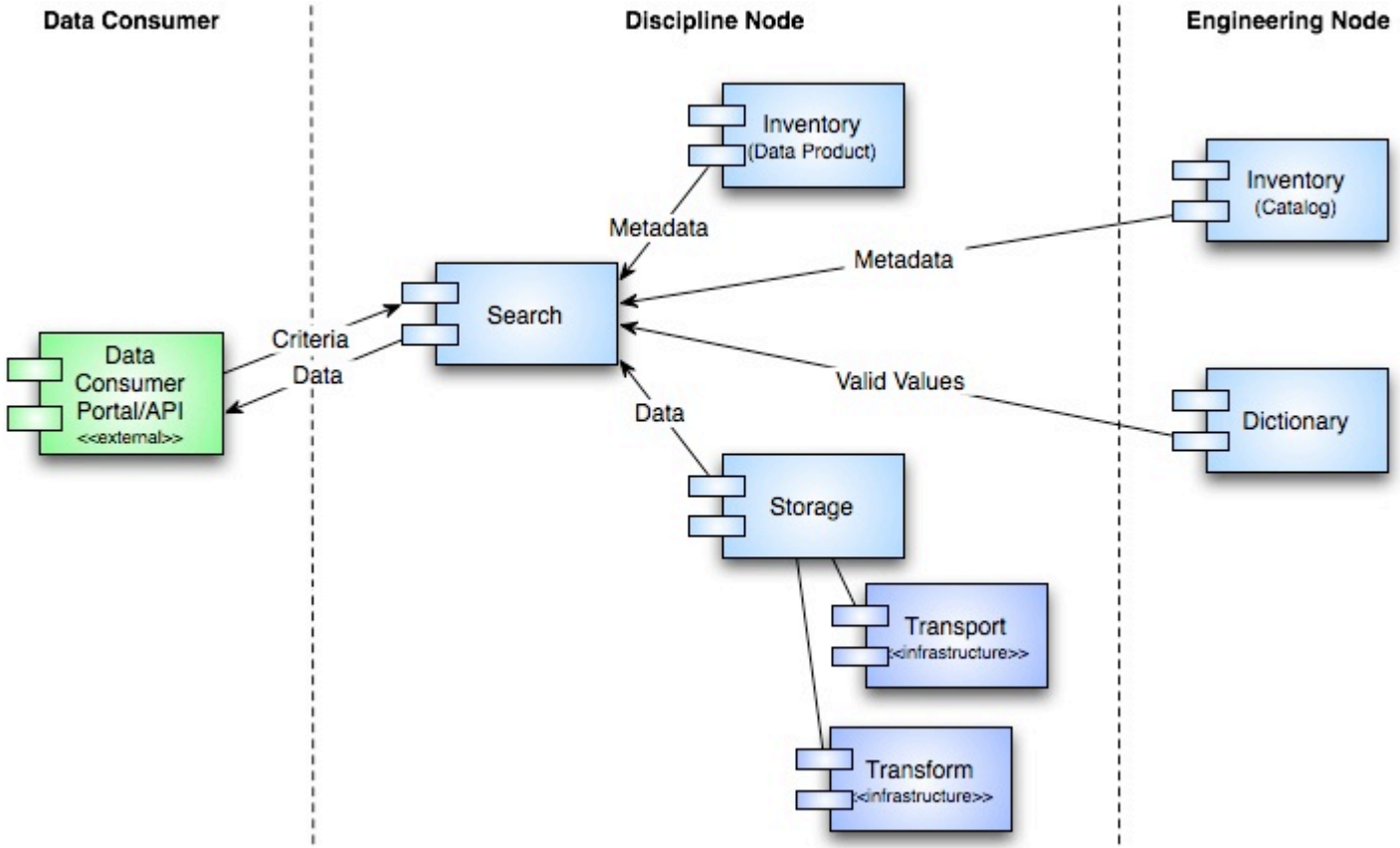
- This scenario details an Ingest service located at the node that manages both catalog and data product ingestion.
- The Preparation service is envisioned as a suite of tools for preparing data submissions.
- Data storage is managed locally at the node but the process does utilize centralized services for authorization and validation against the data dictionary.

Service Provisioning
Search Scenario (From EN)



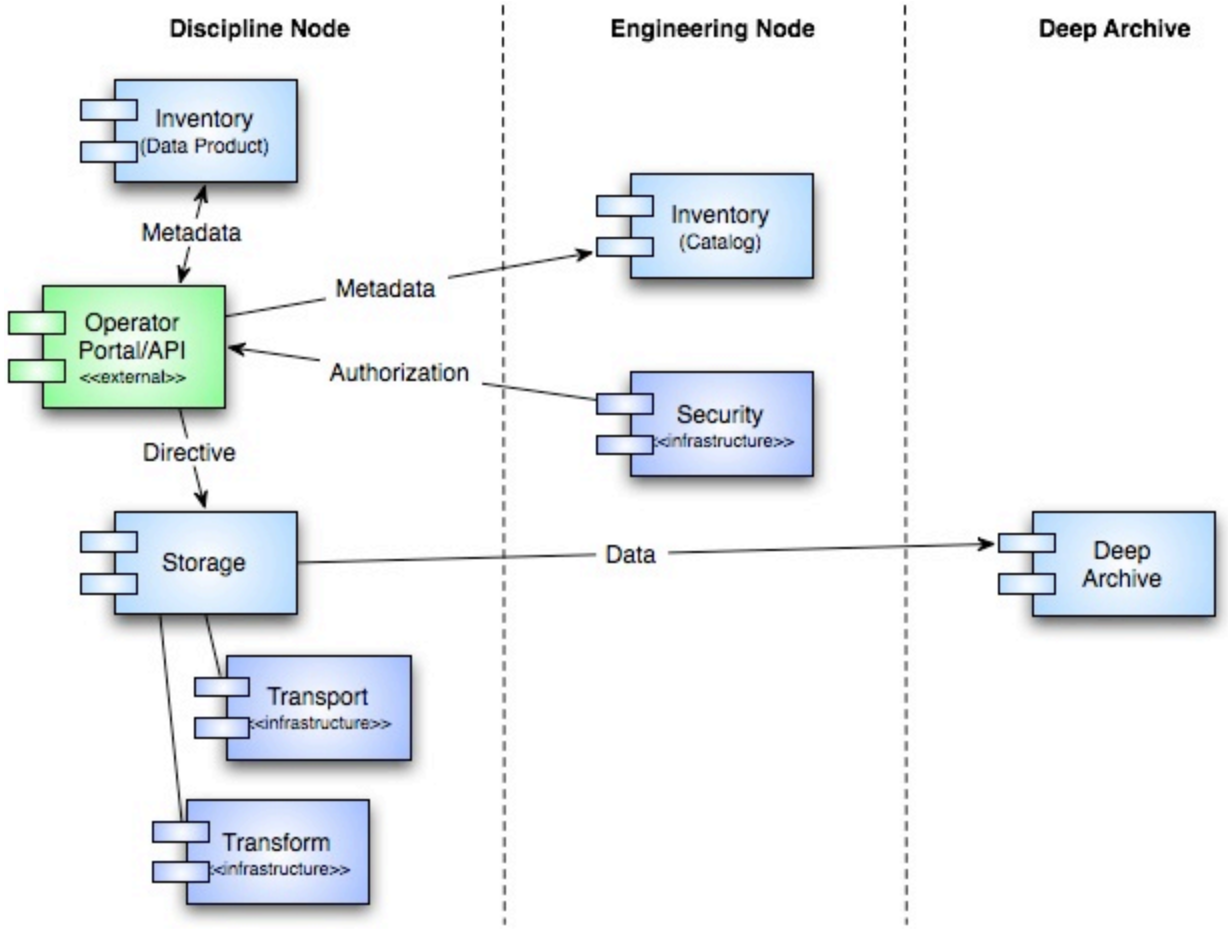
- This scenario details a search for data that originates through the home page portal.
- Catalog metadata aides in data product location.
- Data product metadata and data is served up through a service located at the node.

Search Scenario (From DN)



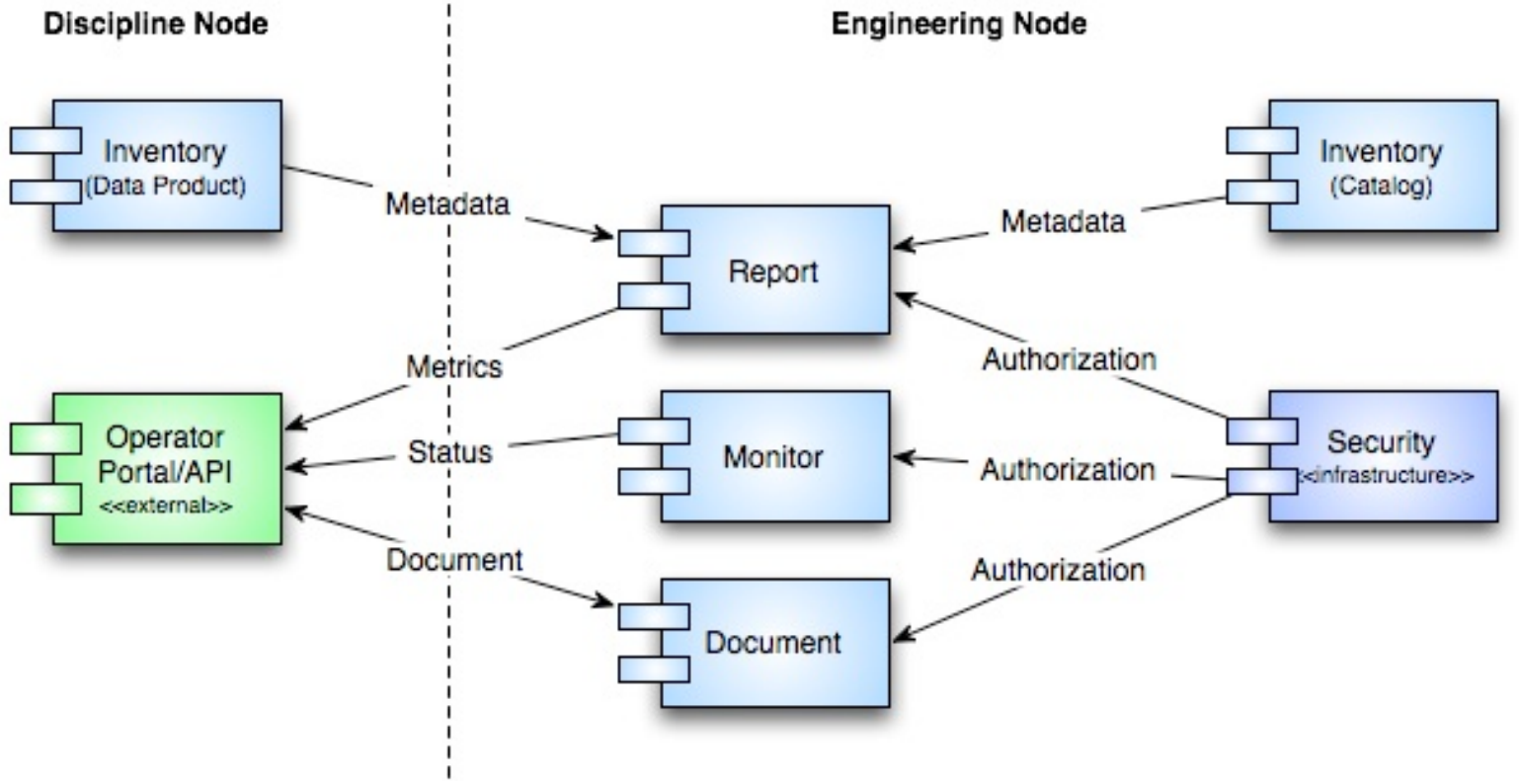
- This scenario details a search for data that originates through a node portal.
- Catalog metadata is available via a centralized service but data product metadata and data is served up through local services.

Deep Archive Scenario



- This scenario details the delivery of data to the deep archive.
- Catalog and data product metadata are updated in the process to reflect delivery.

Miscellaneous Operations Scenarios



- This diagram details miscellaneous operations scenarios where node staff access centralized services for reporting, monitoring and document archive.

Addressing Architectural Drivers

- More Data
  - Addressing storage capabilities and technologies at the system level, offers a more comprehensive solution to increased data volumes.
  - Offering varied transport mechanisms enables efficient movement of large data volumes.
- More Complexity
  - Although this is more of a Data Architecture driver, offering an extensible and adaptable search capability allows users to query based on their desired parameters.
- More Producer Interfaces
  - Efforts already underway to provide a more consistent and easy to use tool suite (e.g., LTDTool and VTool), will aide in the mission interfacing effort.

Addressing Architectural Drivers (cont)

- Greater User Expectations
  - Offering an integrated search capability gets the user to their data of interest faster and more consistently.
  - Providing a transformation capability enables format conversion, subsetting and packaging of search results.
- Limited Funding
  - By building and deploying common services for use by all nodes, the node personnel will be able to focus on discipline-specific solutions for their users.
  - Effective accountability and tracking is facilitated by the architecture freeing up PDS personnel to focus on user needs.
  - The architecture enables phased deployment, allowing development and deployment of the system over time.
- Creating a "System" from the Federation
  - If this effort was about nothing else it was about creating a PDS "system" while allowing the nodes to continue to address discipline-specific needs.

Plans

- Finalize the remaining high-level architecture efforts, focusing on the core service definitions and interfaces.
  - The current draft System Architecture Specification can be found here:
  - http://pds-engineering.jpl.nasa.gov/projects/PDS4/pds2010_sys_arch_spec.pdf
- In January 2009, initiate the Distributed Infrastructure project.
  - Start by defining requirements and developing a design for core services and interfaces.
  - Core services, for example, include Security and Registry.
- Present requirements and design progress at the April MC F2F.
- Present design at the May Tech Session.

Questions/Comments