

5. Release A Design Component Overview

This section provides an overview of the Release A components. Section 5.1 and 5.2 provide the decomposition of SDPS and CSMS into their Computer Software Configuration Items (CSCI) and Hardware Configuration Items (HWCI). Section 5.3 and Section 5.4 provides a brief summary of the role each CSCI and HWCI plays within the ECS architecture. Section 5.5 discusses the network topologies that interconnect the ECS components. Section 5.6 provides an overview of the changes in the Release A design since the Preliminary Design Review (PDR).

5.1 SDPS Components

The SDPS consists of seven subsystems. Each subsystem consists of one or more Computer Software (CS) or Hardware (HW) Configuration Items (CIs), composed of a logical grouping of software or hardware components. These components consist of Commercial-Off-the-Shelf (COTS) hardware, and custom-developed and OTS software. OTS software may include COTS and/or reuse components. Reuse includes ECS component reuse, reuse of heritage code from other programs, freeware or shareware. Many of the software components are developed by combining OTS and custom-developed software, sometimes referred to as “wrappers” or “glue code”, to integrate and encapsulate the OTS software. Collectively, the CIs provide the functionality identified in the SDPS Requirements Specification, Document number 304-CD-003-001. In addition, some functionality requires the integration of components from several subsystems, including some outside of SDPS. The SDPS subsystem and their CIs are listed below. The Release A CSCIs are described in Section 5.3 and the Release A HWCIs are described in Section 5.4.

Client Subsystem (CLS)

Desktop CSCI (DESKT)

Workbench CSCI (WKBCH)

Interoperability Subsystem (IOS)

Advertising Service CSCI (ADSRV)

Advertising Service HWCI (ADSHW)

Data Management Subsystem (DMS)

Local Information Manager CSCI (LIMGT)

Distributed Information Manager CSCI (DIMGT)

Data Dictionary CSCI (DDICT)

Version 0 Interoperability Gateway CSCI (GTWAY)

Data Management HWCI (DMGHW)

Data Server Subsystem (DSS)

Science Data Server CSCI (SDSRV)

Document Data Server CSCI (DDSRV)

Storage Management Software CSCI (STMGT)

Data Distribution Service CSCI (DDIST)

Access and Control Management HWCI (ACMHW)

Working Storage HWCI (WKSHW)

Data Repository HWCI (DPRHW)

Distribution and Ingest Peripheral Management HWCI (DIPHW)

Ingest Subsystem (INS)

Ingest Services CSCI (INGST)

Ingest Client HWCI (ICLHW)

Planning Subsystem (PLS)

Production Planning CSCI (PLANG)

Planning HWCI (PLNHW)

Data Processing Subsystem (DPS)

Processing CSCI (PRONG)

Science Data Processing (SDP) Toolkit CSCI (SDPTK)

Algorithm Integration and Test CSCI (AITTL)

Science Processing HWCI (SPRHW)

Algorithm Integration and Test HWCI (AITHW)

Algorithm Quality Assurance (QA) HWCI (AQAHW)

5.2 CSMS Components

The CSMS consists of three subsystems; the Management Subsystem (MSS), the Communications Subsystem (CSS), and the Internetworking Subsystem (ISS). Each subsystem consists of one or more Configuration Items (CIs), composed of a logical grouping of software or hardware components. These components consist of Commercial-Off-the-Shelf (COTS) hardware, and custom-developed and COTS software. Many of the software components are developed by combining COTS via custom-developed software, sometimes referred to as glue code. In addition, a significant effort is made in the development of what we refer to as hybrid software, as described in Section 3.1. Collectively, the CIs provide the functionality identified in the CSMS Requirements

Specification, 304-CD-003-002, although often this functionality is accomplished by two or more components of several CIs. In addition, some functionality requires the integration of components from several subsystems, including some outside of CSMS. The CSMS CIs are listed below. The Release A CSCIs are described in Section 5.3 and the Release A HWCI are described in Section 5.4.

Communications Subsystem (CSS)

- Distributed Computing Software CI (DCCI)

- Distributed Communications Hardware CI (DCHCI)

Systems Management Subsystem (MSS)

- Management Software CI (MCI)

- Management Agents CI (MACI)

- Management Logistics CI (MLCI)

- Management Hardware CI (MHCI)

Internetworking Subsystem (ISS)

- Internetworking CI (INCI)

- Internetworking Hardware CI (INHCI)

5.3 Computer Software Configuration Item (CSCI) Description

5.3.1 Client Subsystem

The Release A Client Subsystem contains two CSCIs, Desktop CSCI and Workbench CSCI. These CSCIs are summarized below. For more information on these CSCIs refer to the Release A SDPS Client Subsystem Design Specification (305-CD-005-001).

Desktop CSCI (DESKT)

The Desktop CSCI provides the generic underpinning for the SDPS user interfaces. All science user interfaces, and all developmental operator interfaces will be based on this CSCI. From a design perspective, it provides a set of generic object classes from which the science user interface objects will inherit their behavior, and a number of services for associating programs with icons and user actions.

Workbench CSCI (WKBCH)

The Workbench CSCI provides the user interfaces to SDPS services, as well as a number of basic tools for viewing and/or manipulating SDPS data objects (e.g., guide documents, browse images, production history and quality assurance data).

The WKBCH includes the Release A Client. The Release A Client reuses the V0 Client software as the primary user search agent, receiving inputs from users via the System-V0 CHUI and GUI

interfaces. The Release A Client consists of a Graphical User Interface (GUI) and a Character User Interface (CHUI). The Graphical User Interface (GUI) is a graphical environment which operates under the X-Window system, allowing the user to display multiple windows simultaneously, and supports a mouse for easy user interaction. It also allows search areas to be specified from a global map, and provides an interactive data browse facility and coverage map of data products. This interface requires Internet protocol support to function properly.

The Release A Client provides the following services through its easy to use interface:

- Directory Information for Data Sets - provides brief concise high-level information about datasets from any point in the system.
- Guide Subsystem - provides detailed descriptions about datasets, platforms, sensors, projects, and data centers.
- Inventory - provides descriptions of specific observations or collections of observations of data (granules) that are available for request from a data archive.
- Coverage Maps - are two-dimensional graphical representation of the geographic coverage of selected inventory granules. It displays the Earth in an orthographic projection.
- Browse - allows a user to locate and retrieve reduced resolution images as an aid to data selection. The user may either view the image in the IMS interface or have it staged for FTP pickup.
- Product Request - allows users to view information pertaining to orderable data products, and then construct a request which is forwarded to the relevant archive for order processing.
- Access to the Global Change Master Directory - a multidisciplinary database of information about Earth and space science data. It contains high level descriptions of dataset holdings of various agencies and institutions.

The Character User Interface (CHUI) is intended for users who do not have access to an X-terminal, have a small monitor screen, or are accessing the system via low-bandwidth communications (e.g. via modem). Although it is designed to run on a VT100-standard terminal, it also operates on VT2xx- and VT3xx-class terminals, as well as other systems (e.g., personal computers with telecommunications software) that support VT100 terminal emulation.

No special hardware CI has been defined as host for the Workbench and Desktop CSCI. The two CSCI will be available to scientists for installations on their workstations, but they also will be deployed on workstations within the DAAC in support of normal operations (e.g., those supporting algorithm integration and test).

5.3.2 Interoperability Subsystem

The Release A Interoperability Subsystem contains one CSCI, the Advertising Service CSCI. This CSCI is summarized below. For more information on this CSCI refer to the Release A SDPS Interoperability Subsystem Design Specification (305-CD-006-001).

Advertising Service CSCI (ADSRV)

The Advertising CSCI manages a database of information describing the services and data offered by EOSDIS service providers. The user interfaces to the CSCI are part of the Workbench CSCI. However, the CSCI supports access through Internet protocols (WAIS, http), and thus will be accessible to users who do not have an SDPS Client Subsystem installed, as well.

The Advertising Service provides the interfaces needed to support Client defined interactive browsing and searching of advertisements. Although there will be a single format for submitting advertisements to the service, advertisements should be accessible via several different interfaces to support database searching, text searching, and hyper linked access and retrieval according to several different viewing styles (e.g., plain ASCII text, interactive form, or HTML document).

A data server or other provider will advertise its data collections and services with the Advertising Service. The advertisement will include a listing of all products (and other Earth Science Data Types) available in the collection and a set of product attributes. Advertisements include directory level metadata, therefore, the attributes reflected in the advertising service include the ECS Core Metadata Directory-Level attributes. The workbench will send user queries which access only directory level metadata directly to the advertising service (rather than sending it as a distributed query to the various sites which provided the advertising information). A user who wishes to find out what data sets are available on the network can search (i.e., formulate a query) or browse (i.e., navigate through hyperlinked pages of advertisements) the advertising information. Both types of 'directory searching' are available on the user's desktop; the user can choose whichever approach is most convenient in the current work context.

5.3.3 Data Management Subsystem

The Release A Data Management Subsystem contains one CSCI, the Version 0 Interoperability Gateway CSCI. This CSCI is summarized below. For more information on this CSCI refer to the Release A SDPS Data Management Subsystem Design Specification (305-CD-007-001).

Version 0 Interoperability Gateway CSCI (GTWAY)

The CSCI provides a bi-directional gateway between ECS and Version 0. It enables V0 IMS users to query ECS databases, and users of the ECS Client Subsystem to query Version 0 databases.

Gateway provides interoperability with V0 for directory queries, inventory queries, browse requests and product orders. Version 0 queries originating from Release A Client are sent to a Version 0 gateway which operates at each DAAC. The gateway translates an incoming V0 ODL request into ECS query format and submits it to the local ECS data server. The results are returned through the Gateway, which then reformats it into V0 ODL structures and returns it to the Release A Client. The structure of the V0 ODL messages is documented in "Messages and Development Data Dictionary for v4.5 of IMS Client" (IMSV0-PD-SD-002 v1.0.11 950515). The Gateway uses a database constructed by a gateway administrator using the V0 search parameters, ECS schema and metadata. The Advertising Service (ADSRV) CSCI and Document Data Server (DDSRV) CSCI make use of the gateway database to resolve ECS to V0 mappings.

5.3.4 Data Server Subsystem

The Release A Data Server Subsystem contains four CSCIs: Science Data Server CSCI, Document Data Server CSCI, Storage Management Software CSCI and Data Distribution Service CSCI. These CSCIs are summarized below. For more information on these CSCIs refer to the Release A SDPS Data Server Subsystem Design Specification (305-CD-008-001).

Science Data Server CSCI (SDSRV)

The Science Data Server CSCI (SDSRV) manages the access to ECS science and related data. For example, directory, inventory, and other science metadata are stored in databases managed by the SDSRV. Within a site there may be several instances of an SDSRV, each managing a collection of related science data (called a data collection). Data collections for Release A are defined in the ECS Data Server Taxonomy [Technical Description 420-TD-001-001]. For each data collection, there is a schema defining the science data which make up that collection. The schema does not make any principle distinction between "inventory", "directory", "metadata" and other data concepts traditionally employed in the earth science community, thus implementing the "Data = Data" concept of the SDPS architecture.

Document Data Server CSCI (DDSRV)

The Document Data Server CSCI (DDSRV) manages and provides access to the document holdings at a site. It will be capable of accepting, storing, and delivering documents in several different formats (e.g., PostScript and HTML), and will support popular Internet document access protocols (WAIS, http). It is expected that in the long term, the DDSRV will be merged with the SDSRV, with documents just becoming another type of data managed by an SDSRV. However, in the near term, it seemed prudent to keep the two types of services (data and document management) separated to facilitate the implementation through cost-effective commercial off-the-shelf (COTS) software.

Storage Management CSCI (STMGT)

The Storage Management CSCI manages and provides access to archive data. It also provides a stable interface to the other software within the data server subsystem to insulate them from future changes in storage technology of which ECS will want to take advantage.

The facilities to adapt the physical storage of data in the data server to policy, while minimizing impact to availability, is provided by the Storage Management CSCI (STMGT CSCI). This CSCI provides an isolation layer between the search and access views of the archived data in the clients domain, and the physical storage mechanisms of the data internal to the archive. Through the use of unique data identifiers, the STMGT CSCI externalizes its data holdings to the SDSRV CSCI, while hiding the actual physical storage of its data. This allows the STMGT CSCI to optimize its archive storage and data migration strategies, while maintaining a consistent reference to the data for its clients.

Data Distribution CSCI (DDIST)

Data Distribution CSCI (DDIST) is responsible for providing the distribution services to the data server. DDIST orchestrates the delivery of data to its end destination (e.g., user, DAAC). DDIST

receives tasking, in the form of distribution requests, from the Science Data Server and Document Data Server CSCIs and coordinates the activities of the Storage Management CSCI in transferring the data to the media specified by the requester. DDIST also supports operator management of distribution by allowing operators to view, cancel, suspend/resume, and change the priorities of requests.

Distribution of this data can be via either electronic or physical media.

Electronic distribution may be requested via either push or pull. With push, DDIST uses network resources managed by Storage Management to transfer the data to a remote destination specified by the requester. For pull, the data is placed in an area managed by Storage Management, from which the request originator can retrieve the data.

Physical media distribution can be via 8mm tape, 6250 bpi 9-track tape, or CD-ROM. DDIST uses resources managed by Storage Management to transfer the data to the physical media.

5.3.4 Ingest Subsystem

The Release A Interoperability Subsystem contains one CSCI, the Ingest CSCI. This CSCI is summarized below. For more information on this CSCI refer to the Release A SDPS Ingest Subsystem Design Specification (305-CD-009-001).

Ingest CSCI (INGST)

The Ingest CSCI is responsible for the receipt of data arriving at a site and the physical placement of data into the site's storage hierarchy. These data may be delivered through a wide variety of interfaces (network file transfer, hard media, etc.), with a wide variety of management approaches to these interfaces. This interface heterogeneity and the need to support extendibility and new data/interfaces as algorithms and provider functionality changes, leads to a design in which the ingest functionality is isolated from other subsystems within the segment design.

Each instance of the Ingest CSCI has similar functionality. However, in each instance the CSCI has to deal with the characteristics of the specific interface it is managing. Depending on the interface, data may be transferred by either a data "get" or a data "put." A data get is performed by the Ingest CSCI under Ingest CSCI control. A data put is performed by another data center under that data center's control.

5.3.5 Planning Subsystem

The Release A Planning Subsystem contains one CSCI, the Production Planning CSCI. This CSCI is summarized below. For more information on this CSCI refer to the Release A SDPS Planning Subsystem Design Specification (305-CD-010-001).

Production Planning CSCI (PLANG)

The Production Planning CSCI (PLANG) provides the ability to create, modify, and implement a production plan for a site, and manage all planning related data. A production plan is generated from basic planning information (such as data dependencies, dependencies among different production steps, descriptions of production resources and their availability schedules) with the help of production rules which define, for example, the priorities for various types of processing.

The plan defines a schedule for the pending production requests (i.e., instructions which specify which products should be produced for what periods of time). When completed, the plan consists of series of Data Processing Requests (DPR) which implement the production requests.

Multiple candidate plans can be created, but only one plan can be active at any one time. The CSCI submits the Data Processing Requests in the plan to the Data Processing Subsystem as data becomes available. Execution status is recorded against the plan to assess progress.

5.3.6 Data Processing Subsystem

The Release A Data Processing Subsystem contains three CSCIs: Processing CSCI, SDP Toolkit CSCI and Algorithm Integration & Test CSCI. These CSCIs are summarized below. For more information on these CSCIs refer to the Release A SDPS Data Processing Subsystem Design Specification (305-CD-011-001).

Processing CSCI (PRONG)

The Processing CSCI (PRONG) is responsible for the initiation, managing, and monitoring of the generation of ECS Data Products. An ECS Data Product is generated through the execution of Product Generation Executives (PGEs) which are provided by the instrument teams. The Processing CSCI supports the execution of a PGE by performing the following activities :

- Supports Operations staff interfaces to monitor the Processing environment.
- Interfaces with the Data Server to stage data required by a PGE for execution.
- Allocates hardware resources, i.e., central processing units (CPU), memory, and disk space, required by the PGE for execution.
- Interfaces with the Data Server to destage the data generated by the execution of the PGE.

Requests for processing are transmitted to PRONG from the Planning CSCI (PLANG) in the form of Data Processing Requests (DPR) which describe the details of the processing requirements as defined in the production plan for that product.

SDP Toolkit CSCI (SDPTK)

The SDP Toolkit CSCI provides a set of software libraries which are used to integrate Science Software into the EOSDIS environment. By promoting the POSIX standard, these libraries allow the Science Data Processing environment to support the generation of data products in a heterogeneous computer hardware environment.

This CSCI is part of the incremental track design. Its design is not part of this design specification, but an overview is included to provide subsystem design context. The following documents provide guidance on the roles and responsibilities of the SDP Toolkit to support the execution of science software:

333-CD-001-002	SDP Toolkit Users Guide for the ECS Project, 11/94
193-801-SD4-001	PGS Toolkit Requirements Specification for the ECS Project, FINAL, 10/93 [AKA GSFC 423-16-02]

Algorithm Integration & Test CSCI (AITTL)

The Algorithm I&T CSCI is a set of tools which are used to integrate and test new science software, new versions of science software and user methods into the Science Data Processing operational environment. The CSCI provides the software capabilities needed to transition the science processing algorithms and user methods which have been developed externally within the SCF or at a user site into the operational environment of the DAAC and to validate the results of these algorithms/methods within the operational environment. The CSCI consists for the most part, of OTS tools, including software development environments, test and integration tools (e.g., debuggers), software analysis tools, and the like. The CSCI also includes the user interfaces needed by I&T staff.

5.3.7 Communications Subsystem

The Release A Communications Subsystem contains one CSCI, the Distributed Computing Software CSCI. This CSCI is summarized below. For more information on this CSCI refer to the Release A CSMS Communications Subsystem Design Specification (305-CD-012-001).

Distributed Computing Software CSCI (DCCI)

DCCI is a collection of “middleware” providing additional services in each ECS release. Interim Release 1 provides ftp, virtual terminal and DCE core services: Directory, Security, Time, RPCs. Release A provides mail, bulletin board, event logger, Message Passing and object oriented DCE services along with some enhancements. Both IR-1 and Release A will use a single DCE cell where all the users, platforms, and services are maintained.

DCCI is distributed across all ECS components. On client and server platforms, DCCI provides SDPS and FOS applications with access to legacy services such as mail, bulletin board, file transfer and host access as well as object-oriented infrastructure services upon which to execute client-server operations. Client platforms outside the ECS installation are provided with a subset of DCCI services which are integrated within the ECS Toolkit software. In addition to installation on SDPS and FOS platforms, DCCI services are also installed on CSS and MSS servers and workstations distributed throughout the ECS.

A brief summary of the services provided by DCCI is described here.

Directory Naming Service

The Directory Naming Service provides a reliable mechanism by which distributed applications can associate information with names. Its primary purpose is to allow clients to locate servers. Its capabilities, however, are general-purpose, and it can be used in any application that needs to make names and their attributes available throughout a network.

CSS will provide implementation of both the DNS and the X.500 by supporting BIND and OSF Global Directory Service and OSF Cell Directory Service (CDS). It also provides application programmers the ability to store, retrieve, list information in the locally supported namespaces. The DNS and X.500 namespaces are used to connect the locally supported CDS namespaces. The functionality provided here will be implemented on top of XDS/XOM interfaces. As such,

application programmers can use the above mentioned services (store, retrieve, list) in CDS as well as OSF GDS.

Security Service

The security service provides secure transfer of data on local and wide area networks. It provides mechanisms to verify the identity of users, and to determine whether users are permitted to invoke certain operations (authentication and authorization). Transmission of data is protected through the use of checksums and encryption of data. Authentication is provided by trusted third party (secret key) authentication. Authorization is based on Access Control Lists. The protocol used for authentication is Kerberos. All of these features are implemented within the ECS domain by employing OSF/DCE Security Services.

Multicast

Multicasting is a mechanism through which a single copy of data is transferred from a single point to several places. Multicasting allows a sending application to specify a multicast address and send one copy of the data to that address. This data is then distributed through the Multicast backbone to all the applications listening at that address. This reduces the network traffic and improves the performance.

Multicast is being used by FOS as a Release B service. This is presented for informational purposes. Some FOS testing with multicast service is expected to take place before Release B delivery.

Message Passing Service

The Message Passing Service allows for the exchange of information between applications running on different platforms. Clients send data to servers, which process the data and return the result back to the client. This interaction can be classified into three categories: synchronous, asynchronous, and deferred synchronous.

CSS will provide two implementations of Message Passing. The first model will provide for asynchronous and synchronous message passing—byte streams only—with store and forward, recovery and persistence. It will also include the concept of groups where a list of receivers belong to a group. A message sent to the group will be delivered to all the addresses registered in that local group. The second model will provide for asynchronous and deferred synchronous communication without recovery.

Both implementations are designed to take advantage of OODCE-provided DCE-Pthread class which is used to start and control the execution of a thread. The second mode requires more programmer involvement than the first model. Message Passing Service is generally intended to handle low volumes of data per message. Compare with k/ftp (below) for bulk data transfer.

Thread

A thread is a light weight process without the actual process overhead. Threads provide an efficient and portable way to provide asynchronous and concurrent processing, which is a requirement of network software. Threads can maintain thread specific data and can also share data with other

threads in an application. This service provides functionality to create, maintain (scheduling, locking, etc.) threads.

Time

The Time Service keeps system (host) clocks in the ECS network approximately in sync by adjusting the time kept by the operating system at every host. This service changes the clock tick increments (rather than the actual clock) so that host clocks will be in sync with the some reference time provided by an external time provider. CSS will also provide a way to simulate time by applying a supplied delta time to the actual time. With in ECS, OSF DTS will be used to sync the system clocks.

LifeCycle

Managing a system involves managing individual applications. An operator may want to start a new application, shutdown/suspend a running appellation due to anomalies. An application may not be active all the time to accept requests. In order to effectively use the CPU and memory it is desired to control the applications as well s some objects residing in the application by starting them on demand.

LifeCycle services can be broadly classified into two categories: Application and Object level. LifeCycle services for applications involve Startup, Shutdown, Suspend and Resume functionality on applications. This functionality lets the M&O manage server applications. MSS provides the application related LifeCycle functionality. CSS provides the internal APIs that are needed for the MSS to control the applications. LifeCycle services for objects provide the application programmer with the functionality to create and delete server objects residing in different address spaces.

Distributed Object Framework (DOF)

In an object oriented processing architecture, objects may be distributed in multiple address spaces, spanning heterogeneous platforms. The basic contract between an object and its users is the interface that the object provides and users can use. Objects can be spread across the network for reasons of efficiency, availability of data, etc. From the perspective of the requester of a service, invocation should be the same no matter where the object physically resides.

The distributed object framework will be implemented using OODCE. The set of core DCE services are naming, security, threads, time, rpc. In order to aid the application programmer, another layer of abstractions is provided with OODCE. Four generic classes: DCEObj, DCEInterface, DCEInterfaceMgr, and ESO will be available for application programmers to implement client-server applications.

Electronic Mail (E-Mail)

E-mail is a standard component of Internet systems. It is useful for asynchronous, relatively slow notification of many different types. Also, E-mail is persistent, and will continue to try to deliver even if there are temporary network outages. The CsEmMailRelA class provides object-oriented application program interface (API) to create and send e-mail messages.

File Access-k/ftp

FTP is an internet standard application for file transfers. It allows a user to retrieve or send files from/to a remote server. The files transferred can be either ASCII or binary files. FTP also provides an insecure password protection scheme for authentication. KFTP builds on the standard FTP but adds a layer for strong Kerberos authentication. The CsFtFTPRelA object provides an object for managing FTP sessions between clients and servers to allow programmers to transfer files between machines.

Bulletin Board

Bulletin Board is another internet standard application, however unlike e-mail, Bulletin Board messages are directed to all readers of a named group. It uses the Network News Transfer protocol (NNTP) for sending and receiving messages. The CsBBMailRelA object provides object-oriented application program interface to send e-mail.

Virtual Terminal

Virtual Terminal access refers to remote log on to a machine. This software is provided on all platforms. The server telnetd will listen to incoming telnet clients and will allow remote logons. There is also a secure version of telnet and telnetd using Kerberos authentication which CSS will provide where available. This service is allowed only within ECS due to security considerations.

X is a Graphic User Interface conforming to the X/Open standard. While X is not a specific CSS Release A service this description is listed here for informational purposes. It consists of a client and a server where the client displays the actual interface. Developing applications in X is cumbersome and complex. OSF Motif is another standard, layered on top of X which provides a high level application programming interface to make the application development easier. Applications developed with Motif will work with an X server. The X client/server connection presents some significant security risks; therefore ECS will not support applications where the X client and the X server reside on different platforms. Users can download data from ECS and can use the X application to view the data on their local machines. Alternatively, the program should consider providing dedicated circuit access from a user client to connect to an ECS X application.

Event Log

Event log provides the programmers the capability to record events in to files. Events are broadly classified into two categories: management events and application events. Each event is recorded with all the relevant information for identifying and for later processing. Management events need to be recorded in a history file and on some occasions reported to the Network Node Manager. Application events are only recorded into a programmer specified file. Event log provides a uniform way for the application programmers to generate and report (record) events.

5.3.8 Systems Management Subsystem

The Release A Systems Management Subsystem contains three CSCIs: Management Software CSCI, Management Agents CSCI and Management Logistics CSCI. These CSCIs are summarized below. For more information on these CSCIs refer to the Release A CSMS Systems Management Subsystem Design Specification (305-CD-013-001).

Management Software CSCI (MCI)

The Management Software CSCI provides a variety of management services to support the Systems Management Architecture presented in Section 6. A brief summary of the services provided by MCI is described here.

Fault Management

Fault Management addresses the detection, diagnosis, isolation and resolution of faults associated with the managed objects within ECS. The managed objects comprise networks, hosts and applications. A fault is an unacceptable change in the state of a managed object. Fault Management provides for the detection of changes in state of managed objects in order to be able to distinguish the unacceptable changes that constitute faults from acceptable changes. Fault Management, therefore, provides the capabilities for real-time configuration management to include the startup, shutdown and discovery of ECS applications. Further, since the service maintains the status of resources, it provides the capability to provide the status of these resources, such as processors and associated disks, upon requests from subsystems such as the Planning Subsystem. The section on Errors, Exceptions and Fault Handling in Section 6.4 provides the context for the usage of the mechanisms by developers.

Performance Management

The Performance Management Application Service provides the capability to continuously gather statistical and historical data on the operational states of applications, operating system resources and network components, to analyze the data collected by comparing with established criteria, adjust measurement criteria or initiate other corrective actions as necessary in order to ensure an optimal utilization of resources. The service allows for the benchmarking and trends analysis of network component performance, in addition to collecting performance data on scientific algorithms. The Performance Management Application Service has two instances: one at each of the DAACs and one at the SMC. The site Performance Management Application Service collects and processes performance data local to the site.

Site performance management data is periodically summarized and sent to the SMC for analysis by the SMC Performance Management Application. The SMC Performance Management Application Service, which has capabilities similar to those of the site Performance Application Services, operates on performance data collected system-wide by the various site Performance Management Application Services in order to evaluate system-level performance and system-wide trends. In addition, the SMC Performance Management Application Service is also capable of connecting directly to each of the DAACs as required to monitor the performance of site elements.

Security Management

The mechanisms used to provide security in ECS comprise three distinct parts: network security, distributed communications security, and host-based security. The network security is based on router address filtering. The distributed communications security addresses communications between software entities such as clients and servers employing mechanisms such as Kerberos/DCE for real-time authentication exchange. The host-based security addresses the compliance to established directives (e.g. password usage guidelines) and intrusion detection (e.g. viruses and break-ins).

The Security Management Application Service provides for the management of these three mechanisms that are used to protect and control access to ECS resources. It implements security rules and authentication procedures, maintenance of authorization facilities, maintenance of security logs, intrusion detection procedures. Network security management involves the management of routing tables used for address-based filtering (network authorization). Distributed communications security involves the management of the authentication database (the DCE registry database), the authorization database (DCE Access Control List Managers). Host-based security management addresses the protection of these mechanisms, in addition to the management of compliance to established security policy, and intrusion detection.

Accountability Management

The Accountability Management Service provides the capabilities of User Registration and the generation of reports from audit trails.

ECS provides for two generic classes of users: guest users and registered users. Guest users are users that have not formally registered to become registered users. Registered users are those guest users that have submitted requests for a registered user account, and have had accounts created for them, based on an approval process. Registered users are allowed access to services and products beyond those available to guest users.

User registration provides the operators the capability to create accounts against requests submitted by guest users wishing to become authorized ECS users. The registration service provides the capabilities for the creation, modification and maintenance of accounts with user profiles. The user profile information contains user identification, user class, field(s) of research, investigating group affiliation (if any), and shipping address, electronic mail address (if any). The Accountability Management Service makes the user profile available to the various subsystems, such as the Data Server subsystem for information such as the user's electronic mail address and the shipping address, used for the distribution of data products ordered.

The Audit Trail capability provides the means to verify the integrity of the system. This comprises the generation of a user audit trail and a security audit trail with data collected from a variety of sources.

Physical Configuration Management

The Physical Configuration Management Service (PCMS) provides the capability to track, manage, and control all the physical elements in the network. It integrates graphics with data to create a complete electronic model of the physical infrastructure of the network. It provides tools to locate physical proximity of down nodes, place newly discovered nodes, and manage circuit changes. It supports a variety of network administration applications including inventory, billing, and troubleshooting. It has mechanisms to track everything from maintenance data, network protocol data to software registration. In addition, it provides integration support to several Trouble Ticket applications.

Trouble Ticketing

The Trouble Ticketing Service (TTS) provides the DAACs a consistent means of reporting, classifying and tracking problem occurrence and resolution.

TTS provides several methods for a user to report a trouble ticket. The primary method is through the MsTtUserInterface class. This set of HTML documents allow a graphical form for on which the user can disseminate as much information as possible to characterize the nature of the problem. Additionally, TTS provides a textual electronic mail template for generation of trouble tickets from any e-mail package. Finally, if information regarding a problem is received any other manner, a support staff member may enter a trouble ticket directly into TTS.

To the support staff responsible for handling the trouble tickets, provides a common means of statusing, prioritizing, and categorizing reported problems and the resolutions. However, within this common environment, TTS does offer the flexibility for the support staff at an individual DAAC to customize several aspects of the trouble ticketing process. The two primary functions which provide this flexibility are the definition of escalation rules and active links.

Escalation rules are simply time activated events which execute on trouble tickets which meet a set of specified criteria. Actions which can be taken include notification (of either a user or support staff member), writing to a logfile, setting a field value on the trouble ticket, or even running a custom process. Qualifications can be expressed on any data which tracks for trouble tickets. Additionally, TTS put some pre-defined active links and escalation rules into effect at installation time so as to provide a degree of consistency across the DAACs.

TTS also provides a graphical interface to search the trouble ticket database and produce a variety of reports on the trouble ticket data.

Finally, in addition to the detailed trouble ticket information at each DAAC, TTS summarizes data at the SMC. This data allows reports to be produced which can be used to track more "global" trends in reported problems. An example of this could be a report indicating the number of trouble tickets entered per day across all DAACs.

Management Data Access

The Management Data Access (MDA) Service is responsible for centralizing, processing and providing access to the information which is logged into the ECS Management Data log file on each managed host. This log data includes performance, security audit trail, fault, and ECS application processing information.

MDA will centralize the log file data at each DAAC. It is responsible for transferring these log files to the MSS server from each managed host on a scheduled basis. This schedule is configurable, allowing time intervals, absolute time specification, and or size threshold parameters to be set for each ECS managed host's log file. Any of these parameters may trigger a transfer of the file to the MSS server. The parameters may be updated through MDA's graphical user interface.

For the purposes of shorter term analysis of event data, MDA allows browse access to detailed log file data, through its user interface. Provided a host, time period, and optional selection filter information, MDA will retrieve the requested data and display it using its log file browser. Once displayed, options are given for sorting, additional filtering, and saving this data. It should be noted that while this access is typically used to access DAAC (or SMC/EOC) local log files, it also provides the capability to browse the log file data located at other sites.

For longer term analysis and reporting of management event data, MDA will process and accumulate metric data and load it to the Management RDBMS. In addition to this metric data, some specific event detail information will be loaded to the RDBMS.

Common Management

As documented in 193-00632, DCE Migration Study for the ECS Project, and 193-00156, DME Migration Study for the ECS Project, the Open Software Foundation's (OSF) Distributed Management Environment (DME) is the selected distributed management architecture for ECS project. DME is an open architecture that is capable of evolving with new technologies and offers an integrated Distributed Enterprise System and Network Management Architecture for ECS.

Even though a full DME compliant implementation will not exist for Release A, most industry enterprise management players are adopting these technologies and migrating their existing products toward the DME architecture.

To mitigate risk, a DME precursor product (HP OpenView) has been selected as the ECS Management Framework for Release A. This selection provides an ECS migration path to management applications under the full DME architecture.

Ground Events Planning

Ground Events Planning Service consists of the Production Planning Workbench that is part of the Production Planning applications provided by the Planning and Data Processing Subsystem (PDPS). The Production Planning Workbench provides an interface to submit operations ground events such as maintenance, testing, simulations, training, etc., and develop optimum resource utilization plans and schedules based upon approved system configurations and priorities.

Management Agents CSCI (MACI)

The MSS provides ECS M&O Staff with the capability to manage the ECS enterprise, i.e., to perform network and system management services on all ECS resources, including all SDPS, FOS, and CSMS components.

The enterprise management system is based on the manager-agent model. It consists of management applications, a managed object model, and a management protocol. The management applications reside on managing system(s). They provide the interfaces for the human enterprise manager to perform management tasks. The managed object model consists of managed objects which are defined to represent the resources being managed. The underlying resources can be physical devices, system software or applications. The management agent is the implementation which substantiates the managed objects. It normally resides on each remote host performing monitoring and control functions for the management applications which are on the managing system(s). The management applications communicate with agents through the management protocol.

The MSS is composed of a variety of management applications providing services such as fault, performance, security, and accountability management for ECS networks, hosts, as well as SDPS and FOS applications. The management applications reside on MSS Server. The management

information of remote objects need to be conveyed to the management applications through the Management Agent Service which primarily resides on remote hosts.

The MSS Management Agent Service provides the following functions:

- Enables the management applications to retrieve and to set managed object values.
- Performs local polling on remote hosts to monitor the state of managed resources.
- Handles event logging and notifications.
- Provides instrumentation API to application developers to enable the manageability of ECS applications.
- Defines the managed object model to represent the management characteristics of ECS applications.

SNMP has been chosen as the management protocol since it is the defacto and Internet standard protocol for network management in TCP/IP environment. The MSS management applications pass SNMP requests to the agent to retrieve management information. For setting management information, it uses DCE RPC to send requests to remote agents for security reasons.

MSS management applications need to monitor the state of managed resources. It can be done by polling of remote resources. But, remote polling has certain impact to the network traffic. Therefore, the agent can perform local polling for the management applications to avoid the costly remote polling.

Event handling will be provided by Management Agent Service to satisfy the need to dispatch events for orderly and prompt resolution to fix problems. All events will be logged locally on each host. Performance data will also be logged.

A set of Instrumentation API will be provided to ECS application developers to use for the manageability of ECS applications. ECS applications can be categorized into two general types, OODCE-based or non-OODCE-based applications. Application developers can determine the performance metrics along with their threshold to monitor. Fault types can also be monitored and counted. For managing non-SNMP resources such as COTS, proxy agents will have to be used, and supplied by the resource provider. The front-end of the proxy agent uses the instrumentation API provided by MSS. The back-end of the proxy agent is the interface unique to each resource.

MSS requires that on each managed host, standard SNMP MIB II, Host Resource MIB, and the MIBs of network devices are supported by vendor agents. In addition, a managed object model is defined by MSS for ECS applications in SNMP MIB format. The Management Agent Service implements this application MIB. The information contained in the MIB is composed of different types of attributes: configuration, performance, fault, dynamic, static, and traps.

Management Logistics CSCI (MLCI)

For the TRMM release, the Management Logistics Configuration Item (MLCI) implements the Configuration Management Application Service (CMAS). It provides tools with which ECS staffs

at the DAACs, EOC, and SMC track deployed ECS baselines and control changes to the hardware and software that comprise them.

CMAS maintains electronic stores of baseline data, software, and system change requests that enter the operational environment, making them and a variety of reports available for system maintenance and operations activities. It accepts ECS and algorithm software and non-real time configuration management data from formatted files or via operator interface. M&O staffs, sustaining engineers, and AIT teams rely on CMAS data stores to make, track and audit configuration changes and to help enforce ECS CM rules. They also use CMAS to produce formatted files containing change requests, site baseline records, software, documentation, and reports that can be made available for distribution system-wide via CSS services such as e-mail, ftp, and the ECS bulletin board.

For this release, the MLCI design includes three service managers:

- Baseline Manager
- Software Change Manager
- Change Request Manager

5.3.9 Internetworking Subsystem

The Release A Internetworking Subsystem contains one CSCI, the Internetworking CSCI. This CSCI is summarized below. The Internetworking Subsystem detailed design is not presented in its own subdocument. Instead the Internetworking Subsystem is presented in the DAAC specific subdocuments of DID 305. For more information on this CSCI refer to the DAAC and SMC specific subdocuments listed below.

305-CD-014-001 Release A GSFC DAAC Implementation

305-CD-015-001 Release A LaRC DAAC Implementation

305-CD-016-001 Release A MSFC DAAC Implementation

305-CD-017-001 Release A EDC DAAC Implementation

305-CD-019-001 Release A System Monitoring and Coordination Center Implementation

Internetworking CSCI (INCI)

INCI provides internetworking services based on protocols and standards corresponding to the lower four layers of the OSI reference model as described below.

Transport Protocols

ECS provides IP-based connection-oriented and connectionless transport services. The connection-oriented service is implemented using TCP, while UDP is used for connectionless transport. Higher layer applications use one or the other based on such requirements as performance and reliability.

Transmission Control Protocol (TCP), specified in RFC 793, is a connection-oriented, end-to-end reliable protocol designed to fit into a layered hierarchy of protocols which support multi-network applications. It provides for reliable inter-process communication between pairs of processes in host computers attached to networks within and outside ECS. Because TCP assumes it may obtain potentially unreliable datagram service from the lower level protocols, it involves additional overhead due to the implementation of retransmission and acknowledgment processes.

The User Datagram Protocol (UDP), specified in RFC 768, provides a procedure for application programs to send messages to other programs with minimal overhead. The protocol is transaction oriented and delivery of data is not guaranteed, since there is no acknowledgment process or retransmission mechanism. Therefore, applications requiring ordered and reliable delivery of data would use TCP.

Network Layer Protocols

The network layer provides the functional and procedural means to transparently exchange network data units between transport entities over network connections, both for connection-mode and connectionless-mode communications. It relieves the transport layer from concern of all routing and relay operations associated with network connections.

The Internet protocol (IP), specified in RFC 791, is the ECS-supported network protocol, based on its dominance in industry usage and wide community support. As part of IP support, ICMP and ARP will also be supported. As the IETF-specified new generation of IP becomes available for deployment, it will be supported by ECS networks.

Physical/Datalink Protocols

Physical and datalink protocols describe the procedural and functional means of accessing a particular network topology. For the Release A DAAC and SMC networks, the datalink/physical protocols to be implemented are FDDI and Ethernet. (FDDI is a 100Mbps token-passing network topology, and Ethernet is a 10 Mbps bus topology.) However, other protocols can also be introduced for Release B if network requirements or external interfaces dictate. Examples of such protocols include Asynchronous Transfer Mode (ATM) and High-Performance Parallel Interface (HiPPI).

5.4 Release A Hardware Architecture

5.4.1 Hardware Architecture Overview

SDPS and CSMS subsystems have been subdivided into 14 hardware configuration items (HWCI)s. This section provides a brief description of the hardware configuration items and the roles that each plays within overall ECS context. Sizing and selection rationale for the HWCI)s can be found in the respective subsystem specific subdocument. Specific configurations and candidate hardware selections can be found in the DAAC-specific and SMC subdocuments.

Figure 5.4-1, *ECS DAAC Release A Hardware Architecture*, further details the relationships between the HWCI)s and also illustrates their association with external systems and organizations in the Release A timeframe.

Figure 5.4-1 ECS DAAC Release A Hardware Architecture

Available in a separate file.

Section 5.4.1 describes each HWCI and provides a brief summary of the role each plays within the overall hardware architecture. Section 5.4.2 overviews the performance analysis process used to size the hardware components for Release A. Section 5.4.3 provides the process and rationale for selection of components classes of hardware for Release A. Refer to the DAAC and SMC specific subdocuments for the COTS hardware components for each site configuration (305-CD-014-001 through 305-CD-017-001 and 305-CD-019-001).

5.4.1 Hardware Component Descriptions

5.4.1.1 Interoperability Subsystem

The Release A Interoperability Subsystem contains one HWCI, the Advertising Server HWCI. This HWCI is summarized below. For more information on this HWCI refer to the Release A SDPS Interoperability Subsystem Design Specification (305-CD-006-001).

Advertising Server HWCI (ADSHW)

This HWCI provides the resources to support the advertising services at each site. It provides for any server, disk, as well as operator/database administrator workstations/X-Terminals which are needed for administration of the services.

5.4.1.2 Data Management Subsystem

The Release A Data Management Subsystem contains one HWCI, the Data Management Server HWCI . This HWCI is summarized below. For more information on this HWCI refer to the Release A SDPS Data Management Subsystem Design Specification (305-CD-007-001).

Data Management Server HWCI (DMGHW)

This covers the hardware associated with the LIMGR, DIMGR, GTWAY, and DDICT CSCI. The servers and the operations positions associated with those servers is completely covered by this HWCI. It includes the following:

- The physical server, disk, channel, (etc.) hardware needed to process the service requests and administrative functions associated with these CSCI, and to store the administrative and temporary data required for their operation..
- The workstations, X-Terminals, (etc.) needed to support the operator interfaces to these CSCI at each site. This includes hardware for: DBMS administration, data specialists, user support, phone/mail support, etc.

The HWCI does not include the operations position hardware associated with the data servers at each site. These hardware requirements are covered by a separate HWCI.

5.4.1.3 Data Server Subsystem

The Release A Data Server Subsystem contains five CSCIs: Access Control and Management HWCI, Working Storage HWCI, Document Data Server HWCI, Data Repository HWCI, and the Distribution and Ingest Peripherals HWCI. These HWCI's are summarized below. For more

information on these HWCI's refer to the Release A SDPS Data Server Subsystem Design Specification (305-CD-008-001).

Access Control & Management HWCI (ACMHW)

The Access hardware allows for client access (both the client subsystem and direct "push/pull" user access) to the Data Server subsystem, provides tools and capabilities for system administration, and supports many of the infrastructure requirements of the Data Server. This hardware configuration item controls logical data server access, maintains client sessions, and directs service requests to other appropriate Data Server subsystem configuration items. The Access Control and Management hardware is broken down into two components; Administration Stations (AS) and Access/Process Coordinators (APCs). The number, type, and configuration of the APCs and Administration stations vary according to site needs and number of data servers supported.

Working Storage HWCI (WKSHW)

Working Storage (WS) hardware configuration item of the data server supplies a pool of storage used for temporary file and buffer storage within the data Server architecture. In Release B and later WS may also be used to support the higher levels of a hierarchical storage scheme that utilize other data repositories as lower levels in the storage schema. Any data that resides in WS and is not designated as temporary data will be copied to a permanent data repository (see DRPHW - Data Repository HWCI) and maintained there.

WS provides the disk (Release A) staging capacity for data acquires and inserts. Because of its role at the higher levels of the archiving hierarchy, WS may hold production related data that is to be accessed in the near future to increase performance. WS also improves performance by retaining copies of frequently accessed data that has been copied to the deeper archives and servicing data requests for that data in a faster, more efficient manner. In this regard WS behaves like a cache for frequently accessed portions of the deep archives.

Document Data Server HWCI (DDSHW)

The Document Data Server (DDSRV) provides storage and retrieval services on ESDT related documents and their metadata. Full text and keyword searching is provided, as well as the support for HyperText presentation of document metadata. Document Data Server supports user access to the Guide and Reference Papers.

Data Repository HWCI (DRPHW)

This HWCI provides the permanent storage devices associated with the Data Server Subsystem (and some forms of Ingest Data Servers like the L0 Ingest Client). This includes archive robotics, drives, Data Base repositories (with embedded database software), and file servers. Disk resources used for staging data after retrieval until they are processed or distributed, or after ingest until they are archived, are provided by WS HWCI.

Distribution and Ingest Peripherals HWCI (DIPHW)

This HWCI provides the pool of peripherals needed for hard media data distribution and data ingest (the HWCI is shared by the Data Server and Ingest Subsystems). This makes it possible to

share peripherals which are capable of both input and output across the two subsystems, thus providing for their more cost effective utilization (as requirements permit). The HWCI includes disk, tape and other media ingest and/or preparation devices (e.g., 8mm tape, CD-ROM, printers) as needed to fulfill requirements of the site. The HWCI also covers the workstations needed by ingest and distribution operators.

5.4.1.5 Ingest Subsystem

The Release A Interoperability Subsystem contains one HWCI, the Ingest Client HWCI. This HWCI is summarized below. For more information on this CSCI refer to the Release A SDPS Ingest Subsystem Design Specification (305-CD-009-001).

Note that the Ingest Subsystem includes instantiations of the Data Server Subsystem HWCI's needed for archiving and staging Level 0 Data. The Ingest subsystem also shares Data Server Subsystem input / output peripherals contained in the Distribution & Ingest Peripherals HWCI .

Ingest Client HWCI (ICLHW)

This HWCI covers any servers and/or workstations required for Ingest management, control, monitoring and/or processing. It includes any X-Terminals and/or workstations associated with ingest technician operator positions.

5.4.1.6 Planning Subsystem

The Release A Planning Subsystem contains one HWCI, the Planning HWCI. This HWCI is summarized below. For more information on this CSCI refer to the Release A SDPS Planning Subsystem Design Specification (305-CD-010-001).

Planning HWCI (PLNHW)

This HWCI provides workstations (including user interface hardware), and servers as needed, to support production planning, the maintenance of planning data, and the interaction with and reaction to the processing environment during execution, e.g., to accept and process notifications of PGE completion and submit new DPR.

5.4.1.7 Data Processing Subsystem

The Release A Data Processing Subsystem contains three HWCI's: Science Processing HWCI, Algorithm QA HWCI and Algorithm Integration and Test HWCI. These HWCI's are summarized below. For more information on these HWCI's refer to the Release A SDPS Data Processing Subsystem Design Specification (305-CD-011-001).

Science Processing HWCI (SPRHW)

This HWCI provides all processing pools/strings associated with the following forms of processing: standard, reprocessing (accomplished in Release A through the normal, standard production mechanisms), on-demand, and testing. This includes processing platforms and working storage required during processing. The precise architecture of the disk storage resources used to stage data for processing or after processing for archiving is still under investigation and depends on Release A technology decisions (see WKSHW).

The HWCI also includes workstations for managing the production queues and dispatching processing requests.

Algorithm QA HWCI (AQAHW)

This HWCI provides the workstations, X-Terminals, and other devices needed for algorithm quality assurance (QA). For example, the HWCI supports the manual QA of algorithm results within the DAAC. The HWCI will execute the Client Subsystem, as well as additional user interface software needed to give the QA staff access to QA-related information.

Algorithm Integration and Test HWCI (AITHW)

This HWCI provides the workstations, X-Terminals, and other devices needed by the algorithm I&T staff. Hardware needed to run tests in simulated production mode is part of the SPRHW. The HWCI will execute, for example, software development tools, test and integration tools, and the Client Subsystem.

5.4.1.8 Communications Subsystem

The Release A Communications Subsystem contains one HWCI, the Distributed Computing HWCI. This HWCI is summarized below. For more information on this HWCI refer to the Release A CSMS Communications Subsystem Design Specification (305-CD-012-001).

Distributed Computing HWCI (DCHCI)

The Distributed Communications Hardware CI (DCHCI) logically includes an enterprise communications server, a local communications server, and a bulletin board server. To provide for warm standby, the CSS servers and MSS servers at all DAAC sites and the SMC are cross-strapped and are configured to include the CSS Distributed Computing Software CI (including both OODCE client and server); the MSS Management Software CI; and the MSS Agent Software CI. The complete configuration of the CSS and MSS HWCI, based on the combined requirements of the subsystems and site-specific requirements, are presented in the site-specific subdocuments. Additional detail on the analysis of MSS HWCI sizing and performance is contained in the MSS subdocument.

5.4.1.9 Systems Management Subsystem

The Release A Communications Subsystem contains one HWCI, the Management Hardware HWCI. This HWCI is summarized below. For more information on this HWCI refer to the Release A CSMS Systems Management Subsystem Design Specification (305-CD-013-001).

Management Hardware HWCI (MHCI)

This HWCI provides the servers and workstations needed to host the enterprise monitoring, local management and configuration management software, CM data, and backup copies of all ECS "infrastructure" software.

5.4.1.10 Internetworking Subsystem

The Release A Internetworking Subsystem contains one HWCI, the Internetworking Hardware HWCI. This HWCI is summarized below. Section 5.5, LAN Architecture, provides additional overview material pertaining to the ISS Hardware. The Internetworking Subsystem detailed design is not presented in its own subdocument. Instead the Internetworking Subsystem is presented in the DAAC specific subdocuments of DID 305. For more information on this HWCI refer to the DAAC and SMC specific subdocuments listed below.

305-CD-014-001 Release A GSFC DAAC Implementation

305-CD-015-001 Release A LaRC DAAC Implementation

305-CD-016-001 Release A MSFC DAAC Implementation

305-CD-017-001 Release A EDC DAAC Implementation

305-CD-019-001 System Monitoring and Coordination Center Implementation

Internetworking Hardware HWCI (INCI)

This HWCI provides the networking hardware for the intra-DAAC, DAAC to V0, DAAC to EBnet, SMC, and EOC (at Release B) connectivity including, for Release A, FDDI switches, concentrators and cabling, and Ethernet routers, hubs and cabling, and network test equipment.

5.4.2 Performance Analysis Approach

Our performance analysis leading up to CDR was based on the ECS Technical Baseline, dated June 21, 1995. While the hardware configurations contained in this and the more detailed subsystem and DAAC-specific design specifications are considered final, the procurement of these products is not required until early 1996, therefore, we will continue to monitor the price/performance curves of the selected and alternative hardware components between now and the PO release dates to ensure that we gain from performance/cost advantages arising in the interim.

For each of the hardware CIs, an analysis was performed to determine the performance and storage requirements for the various components. This analysis varied by subsystem, but used the following techniques:

- Dynamic modeling (Processing, Data Server)
- Static modeling and analysis (all)
- Benchmarking (CIDM, MSS, CSS, Ingest, Planning, Data Server)

Issues that were considered in the analysis of performance and storage included failover / back-up strategies driven by RMA requirements, operational requirements (e.g., flexibility of configuration in order to enable test and operations, number of operator staff), and scalability to Release B.

The methodology used to size hardware for each CI is described in the CI's portion of this document. A description of the performance analysis used to size hardware for specific DAACs,

analysis results, and rationale for the choice of Release A hardware components, is contained in the DAAC-specific portions of this document.

5.4.3 Release A Hardware Component Classes

The selection of COTS hardware components for Release A is the result of a major effort to identify and validate candidate products, coordination of vendor demonstrations, hands-on use through the EP process and loaner equipment, and evaluation of vendor stability (past performance, financial status) and support. Other influences included upgradability/expandability, product training and documentation, and lifecycle cost. The results, as illustrated in Figure 5.4-2, *Release A Architecture Component Classes*, show the component classes and vendors selected for the overall DAAC hardware suite. The rationale for these selections is presented in Table 5.4-1, *HWCI Class Selection Rationale*.

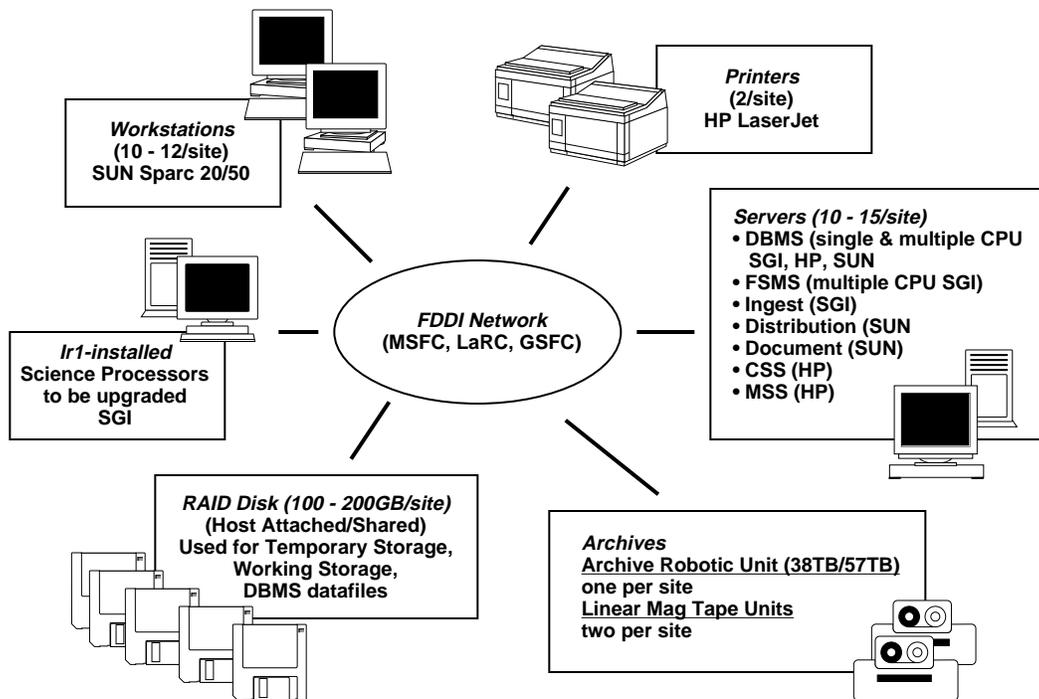


Figure 5.4-2. Release A Architecture Component Classes

Table 5.4-1 HWCI Class Selection Rationale (1 of 2)

Hardware CI	Platform Family	Rationale
Advertising Server HWCI (ADSHW)	See Data Management HWCI	
Data Management HWCI (DMGHW)	HP K200 - Servers HP 715 - Ops Workstation Sun Sparc 20 - Workstations	Software compatibility; Software compatibility; Price/Performance
Access Control & Management HWCI (ACMHW) <i>(Data Server Subsystem)</i>	SGI Challenge L - Servers SGI RAID - Storage Sun Sparc 20 - Workstations	Software compatibility with regards to total Data Server solution; Price/Performance
Distribution & Ingest Peripherals HWCI (DIPHW) <i>(Data Server Subsystem)</i>	Sun Sparc 20 - Servers HP LaserJet 4M - Printers	Software compatibility; Price/Performance
Data Repository HWCI (DRPHW) <i>(Data Server Subsystem)</i>	SGI Challenge XL - Servers Archive Robotics -Archives Sun Sparc 20 - Doc. Server	Software compatibility (archive solution primary processor is SGI); Robotics selected via Request For Proposal (RFP); Document server selected based on software compatibility.
Working Storage HWCI (WKSHW) <i>(Data Server Subsystem)</i>	SGI RAID - Storage	Software and hardware compatibility with data server/ingest overall solution.
Ingest Client HWCI (ICLHW) <i>(Ingest Subsystem)</i>	SGI Challenge L - Server	Software and hardware compatibility with data server overall solution.
Algorithm Integration & Test HWCI (AITHW) <i>(Data Processing Subsystem)</i>	Sun Sparc 20 - Workstations HP LaserJet 4M - Printers	Price/performance; Price/performance.
Algorithm QA HWCI (AQAHW) <i>(Data Processing Subsystem)</i>	Sun Sparc 20	Price/performance.
Science Processing HWCI (SPRHW) <i>(Data Processing Subsystem)</i>	SGI Power Challenge XL - SP SGI RAID Sun Sparc 20 - Workstations	Suitability and scalability; Software and hardware compatibility; Price/performance.
Planning HWCI (PLNHW) <i>(Planning Subsystem)</i>	Sun Sparc 20 - Servers and Workstations	Software compatibility and price/performance.
Distributed Computing HWCI (DCHCI) <i>(Communications Subsystem)</i>	HP 755 - Servers Sun Sparc 2 - BB Server HP RAID	Software compatibility and price/performance.

Table 5.4-1 HWCI Class Selection Rationale (2 of 2)

Hardware CI	Platform Family	Rationale
Management Hardware HWCI (MHCI) <i>(Management Subsystem)</i>	HP 755 Servers, Sun Sparc 20 Workstations HP RAID	Software compatibility and price/performance.
Inter networking Hardware HWCI (INCI) <i>(Inter networking Subsystem)</i>	CISCO Routers Synoptics FDDI Concentrators, Cabletron Hubs	Result of RFP; Result of RFP; Result of RFP; Result of RFP.

Notes:

1. The following elements were taken into consideration in determining the vendor platforms:
 - Processing and storage capacities were determined for Release A and extrapolated out to Release B (this identified the scalability required for a particular platform);
 - COTS SW was assigned to each platform;
 - A survey across the Release A DAACs with regards to platforms currently in place and in use by the science community and the DAAC operations staff was taken to determine which platforms could easily be integrated into the solution (suitability);
 - A common denominator based on equipment class (i.e. workstation, DBMS server, Science Processor), suitability and COTS SW availability was identified (this was done in order to obtain the best possible pricing (a much better price is achieved when many of the same configurations are ordered) on the most suitable and scalable platforms that were identified.):

workstation	SUN Sparc 20/50
server	HP 755; HP K200; SGI Challenge series; SUN Sparc 20/71
science processors	SGI Power Challenge XL; SGI Indy
 - Mass storage for Release A (RAID) could be supplied by platform vendors when required in the design;
 - Price/performance was the determination for the print selection
2. Justification for each vendor was performed early on in the project, as the end result of the Request for Proposal (RFP) process.

5.5 Release A LAN Architecture Overview

5.5.1 DAAC LAN Architecture

This section provides an overview of the DAAC network architecture during Release A. Detailed network topologies on a per-subsystem basis are contained in the subsystem-specific subdocuments of this document, and DAAC and SMC-specific topologies are presented in DID 305-CD-014-001 through 305-CD-017-001 and 305-CD-019-001.

Of the four Release A DAACs, EDC will not have additional DAAC LAN network equipment, but rather will utilize its existing Ethernet-based network from the IR-1 release. This network is discussed in detail in 305-CD-017-001.

The three remaining Release A DAACs (GSFC, LaRC, and MSFC) all receive complete networks for Release A. The overall architecture for these DAACs consists of a central high-performance FDDI switch connecting several FDDI networks, with each FDDI network supporting one or two DAAC subsystems. Figure 5.5-1 shows this generic topology. (Refer to Section 3.4.1 of DID 305-CD-014-001 through 305-CD-016-001 for the DAAC-specific aspects of the design.)

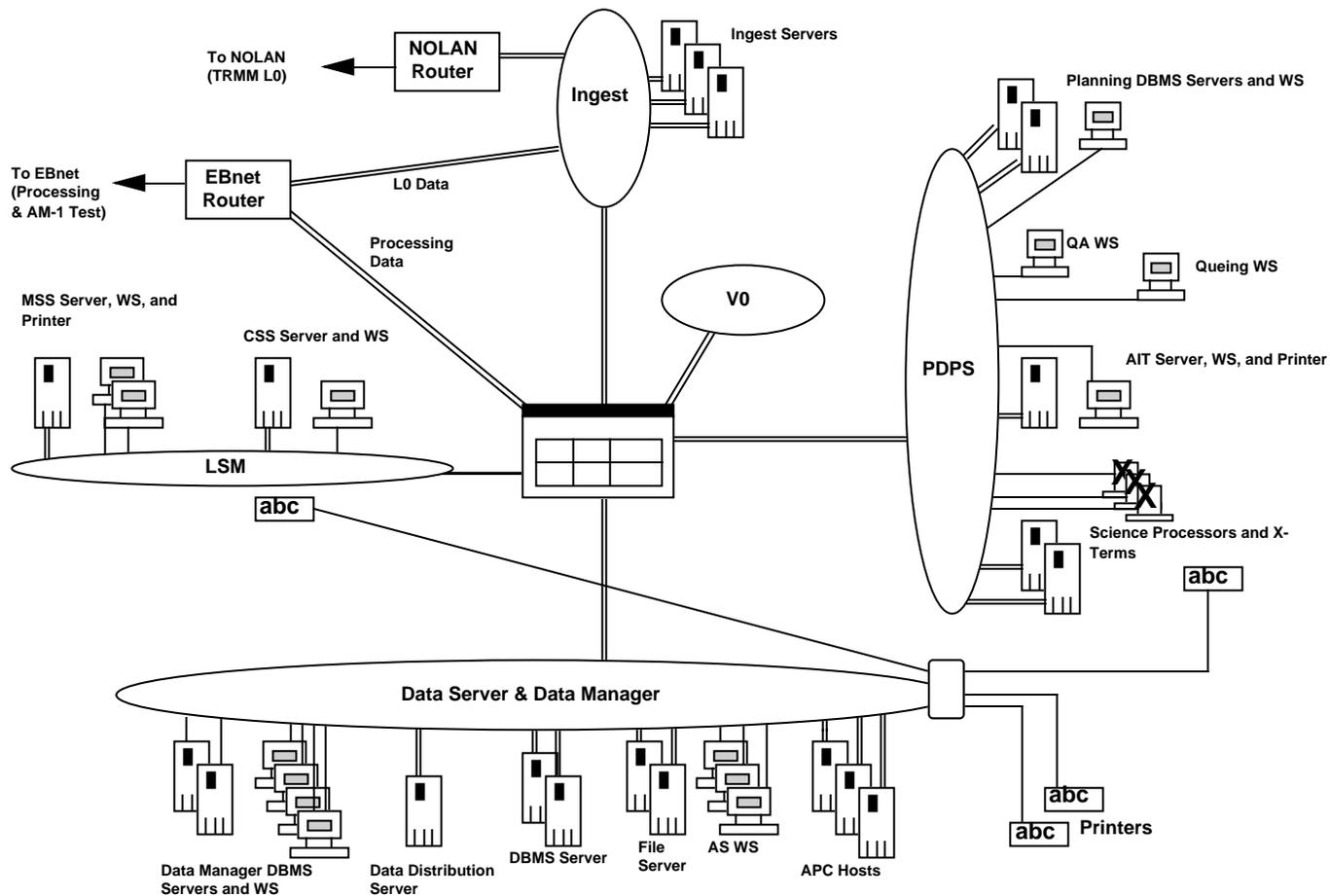


Figure 5.5-1 Release A DAAC Networks: Generic Architecture

The network consists of six FDDI rings supporting the DAAC subsystems and connections to external systems. The Ingest, PDPS, and LSM subsystems are contained on individual FDDI rings, while Data Server and Data Manager have been combined onto a single ring. There are also dedicated FDDI rings for external connections to the V0 network and to the EBnet router. The FDDI switch is the central device connecting the rings together.

Combining Data Server and Data Manager onto a single ring provides several benefits. It is justified by traffic flow estimates for Release A, which show that the traffic from both subsystems can be supported on a single FDDI. Also, both subsystems have the same network-related security implementation (see Section 5.5.2 below), making it easy to combine them on a single interface on the FDDI switch. Since the FDDI switch, when procured, will have at least eight FDDI interfaces, this topology allows the extra interfaces to be used for parallel testing in preparation for Release B. Thus, parallel testing can be performed on separate networks not involved in DAAC operations.

The individual FDDI rings will be implemented with FDDI concentrators to provide ease of wiring and central points of management. All DAAC hosts will have FDDI interfaces and will be attached directly to the FDDI rings. Workstations will have single-attached FDDI cards, whereas the high-performance servers and processors will have dual-attached FDDI cards to provide redundancy. Dual-attached hosts will be dual-homed to two separate FDDI concentrators to provide an additional level of redundancy in the event of a hub failure. (Section 5.5.6 below details network backup and failure recovery scenarios.) The only Ethernet devices in the DAAC LANs will be printers, which will be connected to the FDDI rings via FDDI-to-Ethernet hubs. (Note that the detailed network implementation on a subsystem basis is presented in the subsystem-specific design subdocuments.) Extra FDDI concentrators will be supplied to each DAAC to support parallel testing in preparation for Release B.

All three Release A DAACs will have connections to the existing V0 network and to EBnet. The V0 network will be directly connected to the FDDI switch, primarily to facilitate V0 data migration to the Data Server subsystem. Although the detailed implementation of EBnet is still in progress (see Section 5.6 below), Figure 5.5-1 reflects current understanding of the interface. The DAAC connection to EBnet will be via separate FDDI interfaces on a single router. One interface will connect directly to the Ingest subsystem in order to provide direct delivery of L0 data (bypassing the FDDI switch), thereby decreasing complexity of the data flow and correspondingly increasing availability. The second interface to the EBnet router will carry both DAAC-to-DAAC processing flows as well as user traffic to/from NSI. This interface will connect directly to the FDDI switch which will route data to the proper subsystem (Data Manager, Data Server, LSM, etc.).

5.5.1.1 DAAC Addressing and Routing Architecture

Details relating to the specific DAAC addressing scheme and routing architecture are currently TBD due to the EBnet consolidation. However, the DAAC LAN architecture contains sufficient flexibility to accommodate various addressing and routing schemes without impact to the design. This is in part due to the flexibility of the FDDI switch and its support for "virtual LANS," which refers to mechanisms by which addressing and routing domains can be combined and/or separated independently from their physical connection to the FDDI switch. Possible addressing and routing

schemes have been evaluated against the current DAAC LAN design, and the FDDI switch will support and implement, without redesign, whatever scheme is required.

5.5.2 SMC Network Architecture

The SMC network architecture, as illustrated in Figure 5.5-2, consists of two FDDI LANs. The Enterprise Communications Server (ECS) and the Enterprise Management Server (EMS) connect directly to one of the FDDI rings, and the Management Workstations and printers are attached to Ethernet networks bridged to the FDDI ring via an Ethernet-to-FDDI hub. Since the Bulletin Board Server (BBS) is accessible by the general public, it is attached to a separate FDDI ring to facilitate increased security and to segregate BBS traffic from the rest of the SMC. (Section 5.5.3 below discusses network security in more detail.)

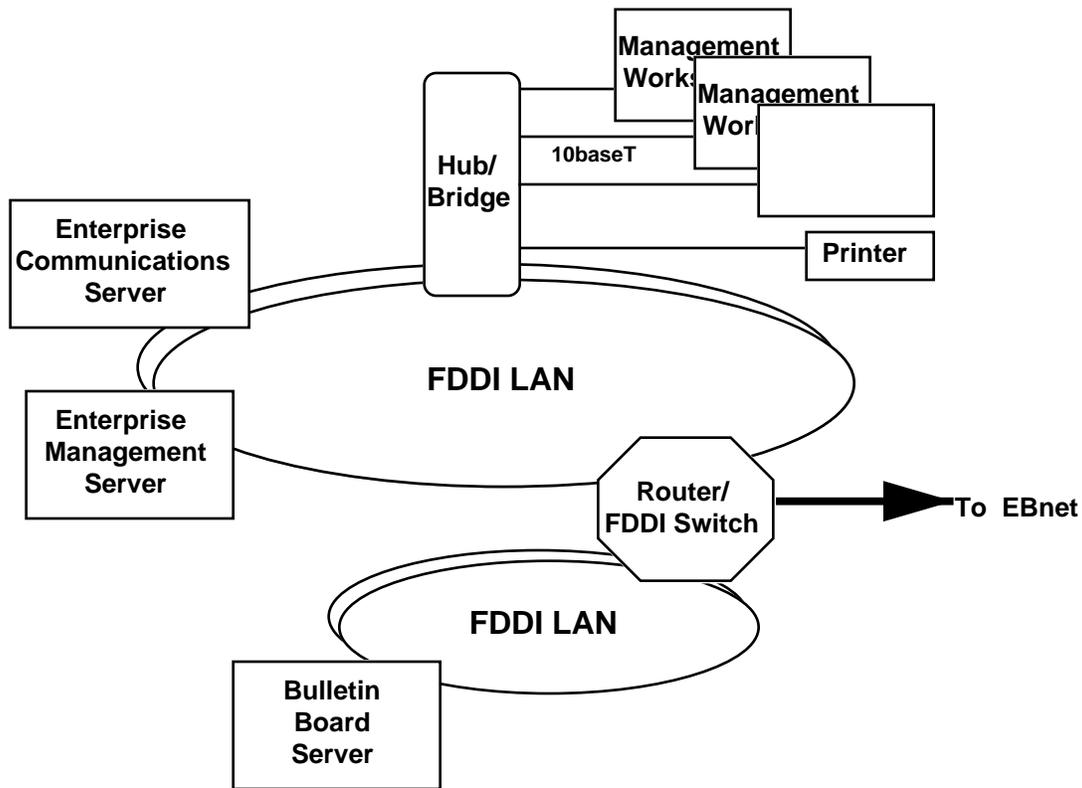


Figure 5.5-2. SMC Network Architecture

Because the SMC has been assigned requirements dictating very high availability, the FDDI LANs will be implemented via physically wired rings as opposed to concentrators. Physical rings eliminate concentrator hardware from the network and create a less complex topology, thereby increasing availability. The use of physical rings is feasible in this case due to the very small number of hosts on the FDDI network (two hosts on one ring and one host on the other). Of course, the workstations and printers will be attached to the Ethernet-to-FDDI hub, which will in turn be part of the physically wired FDDI ring.

5.5.3 Network-based Security Architecture

The Release A network architecture will provide basic levels of security to isolate and protect particular hosts and subsystems within the DAACs and SMC. Note that this section describes only network-based security; ECS has implemented other security measures, such as DCE-based authentication and authorization, Kerberized telnet and FTP, and DCE access control lists (ACLs), which are described elsewhere in this document.

For the Release A DAACs, network-based security will be implemented via network and transport-layer filters in the DAAC FDDI switch. These filters control what traffic passes through the switch, and they are able to control access to individual hosts as well as to whole subsystems. Figure 5.5-3 shows a graphical representation of the DAAC network security architecture.

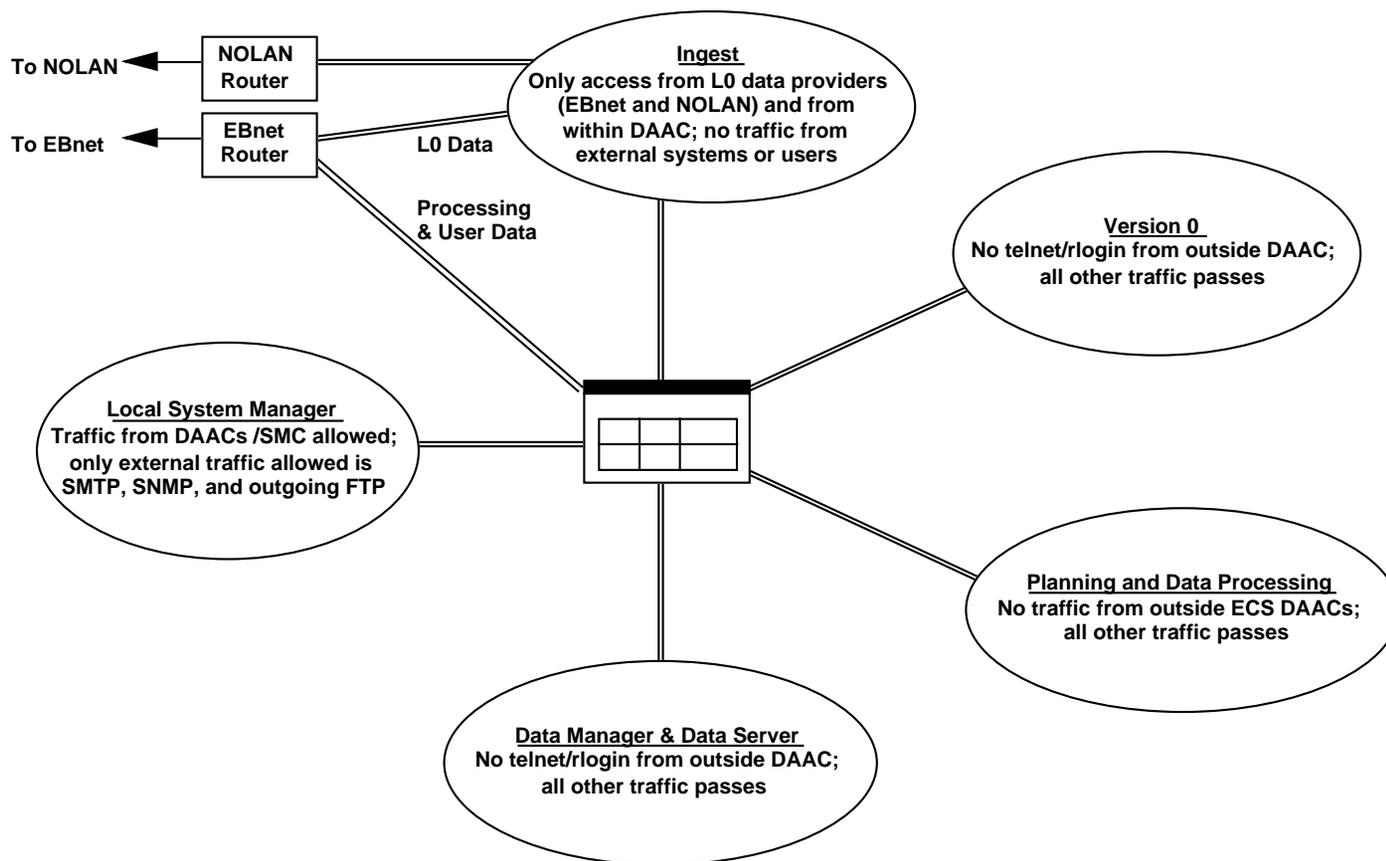


Figure 5.5-3 Network-based Security Architecture for Release A DAACs

The highest level of network security is associated with the Ingest subsystem, because of its role in receiving critical L0 data. No direct user access to Ingest will be permitted, and the only permissible traffic from outside the DAAC itself will be from the L0 data providers, such as EDOS (through EBnet) and SDPF (through NOLAN). All other traffic to Ingest will be filtered by the FDDI switch. For the LSM subsystem, no filtering will be performed on traffic originating from within ECS (e.g., other DAACs, EOC, SMC), but the only permissible external traffic is SNMP (for network management interaction with external network providers such as NSI), SMTP (for mail messages to external systems), and FTPs originating from within the LSM. The PDPS subsystem is also secure, in that no traffic from outside ECS will be allowed. For the Data Manager and Data Server subsystems and the interface to V0, only telnet and rlogin traffic originating from outside the DAAC will be prohibited; all other traffic will be allowed

No filtering is performed on outgoing traffic originating from within the DAAC.

The network security architecture for the SMC is illustrated in Figure 5.5-4. The design is similar to that of the DAACs, since the primary security mechanism is preventing interactive traffic from networks outside the SMC. The Enterprise Management Server and Enterprise Communications Server will allow all traffic originating from within ECS (e.g., other DAACs and the EOC), but will only allow SNMP and SMTP from external systems. Only telnet and rlogin will be blocked for the Bulletin Board Server, since its purpose is to provide information to the public.

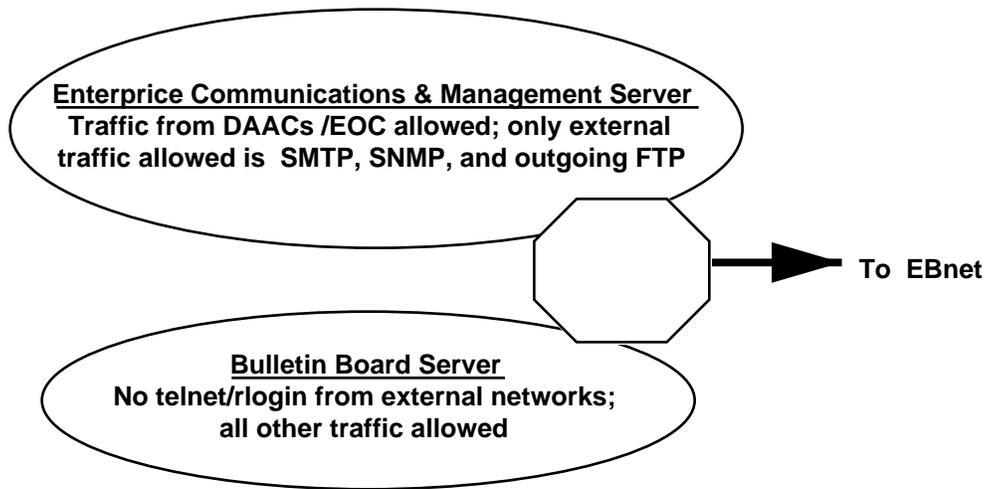


Figure 5.5-4. Network-Based Security Architecture for the SMC

5.5.5 Release A Network COTS Hardware

The Release A DAAC and SMC LANs will contain three types of COTS hardware: FDDI concentrators, FDDI-to-Ethernet hubs, and FDDI switches. As described above, the FDDI rings within the DAACs will be implemented via FDDI concentrators, and the FDDI switch will be used to connect multiple FDDI rings together (refer to Figure 5.5-1). The FDDI-to-Ethernet hubs will be used only to connect printers in the DAACs (and also the SMC workstations).

Table 5.5-1 shows the selected COTS hardware. Both the FDDI concentrators and the FDDI-to-Ethernet hubs are pre-configured, self-contained, stackable units. (For the purposes of this document, "stackable" refers to devices that are pre-configured with a specific number of ports, as opposed to devices that contain chassis with multiple slots into which various interface cards are inserted.) Stackable hubs were chosen due to the DAAC and SMC topologies: Because the DAACs consist of multiple FDDI rings containing a relatively small number of devices, it is more efficient to implement the networks with smaller and cheaper units. These hubs are also simple to configure and maintain, and they are easily replaced in the event of failure. Larger chassis hubs, such as the Cabletron MMAC/MMAC-Plus line and the Bay Networks System 3000/5000 line, contain increased complexity and provide additional capabilities that are not needed.

Although the FDDI switch selection is currently under evaluation, the unit will provide both high-speed switching and routing between multiple FDDI ports, and will allow network and transport layer filtering in order to limit access to certain DAAC subsystems, such as Ingest and Processing.

Table 5.5-1. Release A Network COTS Hardware

Item	Vendor and Model	Capacity	Description
FDDI Concentrator	Bay Networks System 2000 Model 2914-04	12 M Ports plus 1 A/B Port	All-FDDI Stackable Hub with MIC Interfaces
FDDI-to-Ethernet Hub	Cabletron MicroMAC-22E with BRIM-F6 Module	1 A/B FDDI Port and 12 Shared Ethernet Port	Stackable 10baseT HUB to connect Ethernet to FDDI
FDDI Switch	TBD - RFP Evaluation In Progress	At least 8 FDDI Ports	High-Performance FDDI Switch

5.5.6 Network Backup and Recovery

The DAAC and SMC networks are designed to provide exceptional availability coupled with quick and straightforward failure-recovery procedures. The required level of redundancy is determined by F&PRS (Level 3) functional RMA requirements. Specific RMA analysis for these functional requirements can be found in TBD.

As mentioned briefly in the previous sections, the DAAC LAN FDDI rings will be implemented with FDDI concentrators. Each ring will have at least two concentrators, and each high-end server or processor will be dual-homed to separate concentrators (e.g., each server/processor will be connected to two different units). This allows complete and uninterrupted connectivity to exist in the event of a concentrator failure or in the event of a damaged or severed FDDI cable. The switch-over to a backup concentrator for a dual-homed host is basically instantaneous and results in no data loss; in fact, the application processes will be unaware of the event and will continue as normal.

Since workstations are used primarily for administrative tasks and are thus not part of the critical processing flow, they are single-attached to single FDDI concentrators. In the event of a concentrator failure, several recovery procedures could be performed. The most simple, perhaps, is to simply move the FDDI cable of the effected workstation from the failed concentrator to the

backup. This is possible because every FDDI ring will have at least two concentrators, each generally having spare capacity. If no extra capacity is available on the remaining concentrator, then another workstation can be used or the failed unit can be replaced.

Replacing the FDDI concentrator is simple and straightforward. The units are capable of performing properly at power-up without prior configuration (i.e., they are "plug and play"). (Note, however, that configuration would be required to configure the unit's SNMP management capabilities, but such configuration is not required for proper operation.) Thus, replacing a failed unit is simply a matter of transferring cables to the replacement unit. This is also the case in replacing a failed FDDI-to-Ethernet hub.

The FDDI switch will be highly redundant. It will have n+1 redundant, load-sharing power supplies and cooling fans to allow the unit to operate if a power supply or fan fails. The FDDI switch will generally have redundant control processors and hot-swappable interface cards, and may also have redundant interface cards and backplanes. In the event of an interface card failure, the card will be replaced (while the rest of the switch operates as normal) and normal operation will continue without further reconfiguration (because the configuration data is contained in the control processor).

If a control process fails, the redundant control processor will take over if one is available. If there is no redundant control processor, then the card will be replaced. The unit will be reconfigured from backup configuration files contained separate from the switch under configuration management. Thus, because no manual reconfiguration is required, the switch will be operational in a relatively short time (given that the most stringent RMA requirement using the FDDI switch has a MDT of 2 hours.) Note that because L0 data is routed directly to the Ingest hosts without passing through the FDDI switch, the critical receipt of L0 data is not affected by switch failures.

Even though the FDDI switch will be highly redundant and will have high reliability, it is conceivable, although very unlikely, that an entire switch could fail beyond its ability to be repaired. In this case, the unit would need to be replaced, which would not happen within the two hour MDT. The DAAC LAN architecture can be reconfigured in such an instance to provide complete connectivity. This will be accomplished by removing the FDDI switch from the network and connecting each subsystem together directly. This involves physically connecting the FDDI concentrators together to form one large FDDI ring to which all DAAC hosts will be attached. From the host's point-of-view, they will communicate exactly as before (i.e., there will be no reconfiguration required on the hosts), but they will be sharing the same 100 Mbps FDDI and may therefore experience less throughput. This reconfiguration allows complete network connectivity to exist, albeit in a somewhat degraded mode, until the FDDI switch is replaced.

5.6 Summary of Changes Since PDR

At the PDR a preliminary design for SDPS and CSMS was presented. This design has continued to evolve through the detailed design process. Changes to the preliminary design have occurred due to various factors including COTS selections, completion of trade studies and prototypes, changes to the technical baseline, and refinement of the object models. The major changes in the Release A subsystems are summarized below.

CLS Subsystem

An implementation plan for reusing the Version 0 Client as the Release A Client was produced. The plan details the modifications required to the Version 0 Client, as well as ECS components such as EOSView. The implementation plan is described in the document "Implementation Plan for the Release A Client" (441-TP-001-001).

IOS Subsystem

The SYBASE Replication Server product was chosen as the mechanism for replicating the advertising database at each DAAC. This allows for a search of all advertisements to be performed at each DAAC without cross-DAAC communications.

The advertising service will server as the interface signature repository for ECS services. The complete interface signature (e.g., arguments, types, etc.) will be a part of the advertisement. This supports the capability to dynamically find and bind to ECS services. Advertisements for services that are accessed by Mime types (e.g., HTTP, telnet, etc.) will not contain signature information.

DMS Subsystem

The Data Dictionary CSCI (DDICT) was removed from Release A. The primary user of the data dictionary service was the ECS Client. Since the Version 0 client is being reused for Release A, the requirement for data dictionary support moved entirely to Release B.

The V0 Gateway CSCI (GTWAY) added a public interface that allows other CSCIs to use the GTWAY's mapping service. The mapping service provides 2-way translation between Version 0 and ECS terms and attributes. This public interface will be used by the Document Data Server (DDSRV) CSCI and the GCMD exporter component of the Advertising (ADSRV) CSCI.

Data Server Subsystem:

The detailed design activities between PDR and CDR have not significantly altered the design presented at PDR. The design has changed only in the natural evolution of a preliminary design into a detailed design: Design decisions have been made, COTS products selected and Hardware components have been specified.

The design of the Data Server software interface is a set of classes that have been constructed to act as an interface to a dynamic set of data types and data type services. However, in Release A the V0 Gateway does not have a need for a dynamic interface; the data types and services are well-defined and finite. Therefore, the dynamic interface to the Data Server has been extended and specialized to provide a "static layer" to the V0 Gateway.

The FSMS product selected for Release A is the Archival Management and Storage System (AMASS) product marketed by EMASS Inc. The VFS-linked separate file system design allows all UNIX File System (UFS) access methods to be employed (e.g., ftp, rcp, uucp, nfs, RPC, native, etc.) while removing some of the limitations of the UFS. Primary among these is reliance on UNIX Index Node (inode) structures. AMASS maintains all inode information in database files rather than in associated disk structures. This minimizes or eliminates many of the file search problems inherent in searching large numbers of files in multiple directories. In addition, AMASS

organizes files as groups of blocks which can be individually retrieved. This differs from UFS resident systems which require staging the entire file. AMASS utilizes a disk based I/O buffer for communications rate matching between disk and tape resources. The I/O buffer scheme fits the AMS paradigm of encapsulating HSM storage and data viability functions while allowing the Data Server Storage Management CSCI manage the associated Working Storage.

EMASS Automated Tape Library has been selected for the Data Repository component of the Data Server. The driving selection factor was the library's ability to accommodate multiple media form factors. This crucial ability enables recording technology migration with minimum migration cost associated.

The Document Server will support stateless connections in order to maximize use of COTS.

The Document Server will use Data Management's V0-ECS mapping service for its keywords.

Ingest Subsystem

The Ingest design was modified to support an increase in the TSDIS Data Rate by storing this data in RAID. Hughes team is performing a cost analysis on the impact of purchasing the RAID.

The Ingest Subsystem design was modified to interface with a gateway translating external messages (tcp/ip) into OODCE requests for ECS servers. The Gateway Architecture is described in Section 6.

Use of a HTML/http based GUI for ingest requests on the Release A Client.

Planning Subsystem

The basic capabilities of the Planning subsystem described in the preliminary design have not changed. Rather, these capabilities have been enhanced by the AutoSys/AutoXpert Job Scheduling COTS selection. What has changed is the division of responsibilities within the Planning and Data Processing subsystems. A major goal in re-architecting the system after the selection was to ensure that the interfaces to COTS are within one subsystem. This ensures that interfaces to the COTS can be appropriately encapsulated to give later flexibility augmenting, modifying or replacing the underlying COTS as ECS matures.

The Job Scheduling COTS is appropriately made a component of the Data Processing subsystem, which accounts for the reallocation of some of the Planning capabilities. The key changes to the Planning design since PDR are within the Production Management capabilities. The two main activities performed here were:

1. to coordinate the production by providing a Data Processing Request to the Data Processing subsystem when all the data required for the task are present at the Data Server,
2. to provide a display of the active production, and it's status according to the plan.

These two capabilities have been split into separate CSCs. The (new) subscription manager CSC, which is part of this subsystem, provides the first of these capabilities. The graphic capabilities of AutoXpert in the Data Processing subsystem provides for the second.

Other changes that have been brought on due to the selection of AutoSys and AutoXpert pertain to the sequence of events “activating” a plan. This now involves rolling a portion of the “long term” plans generated in the Planning subsystem into AutoSys.

Processing Subsystem

After the Preliminary Design Specification, a COTS product(s) has been selected which will fulfill the majority of Level 3 and Level 4 requirements has had a tremendous impact on the detailed design as presented here. The following information will summarize these design modifications and their rationale. The selected COTS products are Platinum Technology's **AutoSys** and **AutoXpert**. They will be integrated into PDPS to provide the basis for the monitoring and management of ECS' science data production facility.

All design decisions have been driven by a desire to minimize custom code development, tempered by the need to provide proper encapsulation of the COTS to insure later flexibility of adding or modifying the underlying COTS product as ECS matures and evolves.

As a result of these efforts, some design elements which were mapped to the Planning CSCI at PDR have since moved to the Processing CSCI. This has resulted in a clearer division of the roles and responsibilities of the Planning and Processing CSCIs has been developed. As a side effect, the Planning and Processing CSCIs are now more loosely coupled. This will insure greater flexibility in the future.

The following summarizes the design decisions and provides a top-level view of the current Planning and Data Processing Subsystem Architecture.

1. PDPS will share a common database, i.e., one instance of a SYBASE RDBMS. This will allow PDPS to eliminate the large amount of common persistent data structures which existed in the PDPS preliminary design. For detailed information on the PDPS Database, please refer to Section 4.6.6, PDPS Database CSC, in the Planning Subsystem Preliminary Design Specification
2. The Production Management CSC which was mapped to the Planning CSCI has been divided between the Planning and Processing CSCIs. As presented at PDR, the Production Management CSC provided two important functions; managing of subscription notifications from the Data Server and Ingest and managing the active plan by receiving status feedback from the Processing CSCI. Since the AutoXpert product provides mechanisms for monitoring and managing the active plan, it was decided to encapsulate the COTS products into a single COTS CSC within the PRONG CSCI. This will provide a more consistent and simpler design with fewer interfaces needed between the Planning and Processing CSCI. Therefore, active plan management is now within the Processing CSCI, whereas, the management of subscription notifications remains in the Planning CSCI.
3. The interface between the Planning and Processing CSCIs has been modified. This change involves when a Data Processing Request is made visible to the Processing CSCI. At PDR, the approach amounted to not providing a Data Processing Request to the Processing CSCI until all the data subscriptions, sometimes called data dependencies, were fulfilled for a Data Processing Request. Because of the selection of AutoSys and its capabilities to

manage job dependencies, this approach has been changed to consist of all Data Processing Requests being fed into AutoSys at the beginning of the day. The Data Processing Requests which do not have all data dependencies fulfilled would be kept in a "HELD" state until the dependencies are fulfilled. Upon the meeting of all data dependencies, the Planning CSCI would release the job.

4. The software components of the Science Data Pre-Processing CSCI as defined in the Preliminary Design Specification have been mapped into the Processing CSCI or Ingest CSCI, based on what is the optimal location to perform these operations. Within the Processing CSCI, the Science Data Pre-Processing functions have been mapped to a CSC called Data Pre-Processing.

MSS Subsystem:

The following changes have occurred in the MSS design since PDR.

- 1) Change from SNMP Traps to RPCs for Fault and Security notifications. Unreliability of SNMP Traps could result in lost messages due to the underlying protocol (UDP). Change provides guaranteed delivery between the host computer and MSS Server.
- 2) Change from SNMP to RPCs for all controlling interfaces with hardware devices due to the lack of robust security in SNMP. This provides an additional level of security against IP spoofing for all control actions for ECS managed objects.
- 3) Trouble Ticketing has been added to the capabilities in Release A based on discussions at PDR. A COTS Trouble Ticketing package (REMEDY) is being provided for Release A.
- 4) Physical Configuration Management has been added to the capabilities in Release A based on discussions at PDR. At PDR MSS briefed the use of Office Automation tools to maintain physical location and configuration information, this has been replaced by a COTS package.
- 5) DCE Cell Management was briefed as a manual process at the PDR, utilizing a combination of DCE capabilities, scripting, and command line interfaces. Release A will have a graphical DCE Cell Management capability provided by a COTS cell manager.
- 6) Additional Management capabilities have been added for ECS applications, at PDR monitoring was provided for all ECS applications, but only custom developed applications were to be managed. Release a will now include complete life cycle services (startup, shutdown, discovery, reporting, and control), for all ECS applications. This includes the capability to send notifications from management applications to ECS applications and retrieving application-specific data from ECS applications.
- 7) Accountability has been enhanced through the use of UUID pairs being logged with each event to provide end to end traceability of transactions, and more robust error tracking to the source of each reported error condition.

- 8) At PDR Virus checking was presented as a Release A capability. This has been moved to Release B based on the limited availability of Virus checkers for all ECS UNIX platforms, the cost of those available for limited platforms, and lack of a truly integrated solution.

CSS Subsystem

This section addresses the major CSS design changes since PDR. These design changes have occurred due to:

- additional information being available from the major platform vendors, software COTS vendors, and technology consortiums;
- additional design detail is now available from the other ECS subsystems so that more accurate usage/need of CSS services is understood; and
- additional prototyping of CSS services and evaluations of potential COTS have been completed.

The major Release A changes include the use of DCE version 1.0.3, DCE cell architecture, Remote File Access, Event Service, and Message Passing service. These changes are detailed in the following sections.

DCE Version and Cell Architecture Changes. The DCE encapsulation prototype and associated benchmarking of OODCE services has provided insight into the performance of OODCE, which is a key contributor to the development of performance allocations to satisfy Level 3 requirements. At PDR, DCE 1.1 was expected to be marketed by November 1994. However, DCE 1.1 has not yet been delivered by vendors. Currently vendors are planning to ship DCE 1.1 in late Fall/Winter 1995. Due to this timing change, Release A can not rely on using DCE 1.1. Instead, in order to reduce the risk, ECS has decided to use the existing DCE 1.0.3 for Release A.

The DCE cell configuration trade resulted in the decision to provide one cell per DAAC/site configuration, with an additional "Isolation cell" to isolate the above configurations from external networks. This trade, which is summarized in "Trade-off Studies Analytical Data" (ECS DID 211/SE3), provides the basis for placement of an OODCE server at each ECS site. However, DCE 1.0.3 doesn't support authentication of foreign users (cross cell authentication) and the OODCE doesn't support foreign identities (users from other cells) in authorization. Lack of these features would prevent service providers from offering their services to users in other cells. ECS will use a single cell for Release A. Studies have indicated that DCE cells can reasonably support up to 50k users. Release A is not expecting more than 25k users. ECS architecture also includes maintaining (Directory & Security) replicas at each DAAC, and the EOC to improve performance. The multi-cell architecture to provide scalability will be implemented in Release B, provided DCE 1.1 and the corresponding OODCE are available in a timely manner.

Remote File Access Changes. We have prototyped Distributed File System (DFS) and compared the features and performance with Network File System (NFS). Another remote file access considered is the Andrew File System (AFS). DFS has a rich set of functionality and provides complete security to the file level. DFS integrates well into the existing infrastructure

(OODCE). NFS doesn't provide security but tests indicate that NFS has better performance. AFS provides limited security. AFS is available and is being used in many systems. Due to the lack of security NFS is not being considered any further. While AFS is available, it involves extra cost and maintenance and only provides limited security. DFS has performance problems, but future releases of DCE (1.1 and 1.2) are expected to include significant enhancements to improve the performance. Since Release A doesn't require the use of an Remote File Access product, we recommend waiting for DFS enhancements, and revisiting the selection process in Release B time frame. While DFS is not provided as a deliverable in Release A, it is being studied in parallel with Release A for prototyping purposes.

Event Services Changes. Event services are essentially the same as asynchronous message passing with the exception of the decoupling of senders and receivers. At PDR time frame, it was perceived that MSS would have to use this service to send notifications (traps) to the Network Node Manager. MSS design currently uses RPCs between the Agent and the NNM for reliable transmission of notifications. There hasn't been any other need identified in Release A for the Event Services. As such, Event Services is not being supported in Release A. It will be revisited in Release B to see if it will be needed. If needed, it may then be absorbable into the more general Message Passing Service.

Message Passing Changes. During PDR a trade study ("Trade-off Studies Analytical Data" ECS DID 211/SE3) was done to find the best way to provide Asynchronous Message Passing Service and it was determined that ECS would buy COTs products and develop custom features like Deferred Synchronous Message Passing on top of the COTs products. Since then, CSS has evaluated several COTs products and found that none of them are reliable or thread safe; They also have other limitations like message size and lack of security. It has been decided at CDR to custom develop this service on top of OODCE. Message passing will be developed on top of OODCE using threads and message queues.

ISS Subsystem

The ESN WAN network (carrying DAAC-DAAC processing flows) and the Ecom network (carrying L0 data to the DAACs from EDOS) have been consolidated into a single network called the EOSDIS Backbone Network (EBnet).

The ramifications of this consolidation are still being considered, and ECS is working with Ecom and EOSDIS to complete the new topology. For the purposes of the DAAC LAN designs presented in this document, ECS has assumed, based on discussions with Ecom, that the DAAC interface to EBnet will be via a single router with two interfaces to the DAAC. One interface will carry L0 data for ingest, and the other interface will carry DAAC-to-DAAC processing traffic as well as user traffic to/from NSI. (Refer to Figure 5.5-3 for clarification). The detailed specification of these interfaces will continue to be worked, but the DAAC LAN design has sufficient flexibility to accommodate changes that may arise during the process without significant impact to the overall DAAC LAN architecture.

The EBnet consolidation may also potentially impact the SMC LAN design, because the exact method by which the SMC, EOC, and the DAAC at GSFC interface with EBnet is currently being worked. This impacts the SMC design only in terms of the exact router configuration used to

connect the SMC to EBnet; the internal SMC LAN architecture will remain constant (see Figure 5.5-4 for clarification).

Hardware Changes

Several changes have been made to the hardware CI designs since PDR due to changes in operations planning, updated processing, data rates and storage requirements, and newly adopted assumptions including. The action that caused these changes and resulting action is presented in Table 5.6-1.

Table 5.6-1. Hardware Changes Since PDR

Action	Result
Changes in the hours of operations	Data server data rate increases Pull area (FSMS) for processing Processing disk space Processing I/O bandwidth Processing, SLOC estimates (no lights out processing, except AI&T)
DAAC QA Hardware	One fifth pull of daily production
Increase in ingest data rate	Operations refinements on interfaces
Planning and Queuing	COTS selections Prototyping Better detail on sizing
Adoption of V0 gateway and V0 client	Better ops concept, more detailed design

This page intentionally left blank.