

311-CD-521-002

## **EOSDIS Core System Project**

# **Release 5B.06 INGEST Database Design and Schema Specifications for the ECS Project**

December 2000

Raytheon Company  
Upper Marlboro, Maryland

**Release 5B.06**  
**INGEST Database Design and Schema Specifications**  
**for the ECS Project**

**December 2000**

Prepared Under Contract NAS5-60000  
CDRL Item #050

**RESPONSIBLE ENGINEER**

<u>Peter MacHarrie /s/</u>	<u>11/5/00</u>
Peter MacHarrie	Date
EOSDIS Core System Project	

**SUBMITTED BY**

<u>William Knauss /s/</u>	<u>11/6/00</u>
William Knauss, Development Manager	Date
EOSDIS Core System Project	

**Raytheon Company**  
Upper Marlboro, Maryland

This page intentionally left blank.

# Preface

---

This document describes the data design and database specification for the subsystem. It is one of nine documents comprising the detailed database design specifications for each of the ECS subsystems.

The subsystem database design specifications for the as delivered system include:

- 311-CD-520-002 Data Management (DM) Subsystem Database Design and Database Schema Specifications for the ECS Project
- 311-CD-521-002 Ingest Subsystem Database Design and Database Schema Specifications for the ECS Project
- 311-CD-522-002 Interoperability Subsystem (IOS) Database Design and Database Schema Specifications for the ECS Project
- 311-CD-523-002 Planning and Data Processing Subsystem (PDPS) Database Design and Database Schema Specifications for the ECS Project
- 311-CD-524-002 Science Data Server (SDSRV) Subsystem Database Design and Database Schema Specifications for the ECS Project
- 311-CD-525-002 Storage Management (STMGT) Subsystem Database Design and Database Schema Specifications for the ECS Project
- 311-CD-526-002 Subscription Server (SUBSRV) Subsystem Database Design and Database Schema Specifications for the ECS Project
- 311-CD-527-002 Management Support Subsystem (MSS) Database Design and Database Schema Specifications for the ECS Project
- 311-CD-528-002 Registry Database Design and Database Schema Specifications for the ECS Project

This submittal meets the milestone specified in the Contract Data Requirements List (CDRL) of NASA Contract NAS5-60000. It is a formal contract deliverable with an approval code 2. As such, it does not require formal Government approval, however, the Government reserves a time-limited right of disapproval (45 days) for the initial submittal. Once approved, contractor changes to this document are handled in accordance with Class I and Class II change control requirements described in the EOS Configuration Management Plan, and changes to this document shall be made by document change notice (DCN) or by complete revision

Entity Relationship Diagrams (ERDs) presented in this document have been exported directly from tools and some cases contain too much detail to be easily readable within hard copy page constraints. The reader is encouraged to view these drawings on-line using the Portable Document Format (PDF) electronic copy available via the ECS Data Handling System (ECS) on the world-wide web at <http://edhs1.gsfc.nasa.gov>.

Any questions should be addressed to:

Data Management Office  
The ECS Project Office  
Raytheon Company  
1616 McCormick Drive  
Upper Marlboro, MD 20774-5301

# Abstract

---

This document outlines Release 5B.06 “as-built” database design and database schema of the INGEST database including the physical layout of the database and initial installation parameters.

**Keywords:** data, database, design, configuration, database installation, scripts, security, data model, data dictionary, replication, performance tuning, SQL server, database security, replication, database scripts

This page intentionally left blank.

# Change Information Page

---

<b>List of Effective Pages</b>			
<b>Page Number</b>	<b>Issue</b>		
Title ii thru xii 1-1 and 1-2 2-1 and 2-2 3-1 and 3-32 4-1 through 4-4 5-1 through 5-4 6-1 through 6-2 A-1 through A-6 AB-1 through AB-2	Submitted as Final Submitted as Final Submitted as Final Submitted as Final Submitted as Final Submitted as Final Submitted as Final Submitted as Final Submitted as Final		
<b>Document History</b>			
<b>Document Number</b>	<b>Status/Issue</b>	<b>Publication Date</b>	<b>CCR Number</b>
311-CD-521-001 311-CD-521-002	Submitted as Final Submitted as Final	April 2000 December 2000	00-0335 00-1131



This page intentionally left blank.

# Contents

---

## Preface

## Abstract

## 1. Introduction

1.1	Purpose and Scope .....	1-1
1.2	Document Organization .....	1-1

## 2. Related Documents

2.1	Applicable Documents .....	2-1
2.2	Information Documents .....	2-2

## 3. Data Design

3.1	Database Overview .....	3-1
3.1.1	Physical Data Model Entity Relationship Diagram .....	3-1
3.1.2	Tables .....	3-2
3.1.3	Columns .....	3-15
3.1.4	Column Domains .....	3-26
3.1.5	Rules .....	3-26
3.1.6	Defaults .....	3-26
3.1.7	Views .....	3-26
3.1.8	Integrity Constraints .....	3-26
3.1.9	Triggers .....	3-28
3.1.10	Stored Procedures .....	3-29

3.2	File Usage .....	3-31
3.2.1	Files Definitions.....	3-31
3.2.2	Attributes.....	3-31
3.2.3	Attribute Domains.....	3-31

## 4. Performance and Tuning Factors

4.1	Indexes .....	4-1
4.2	Segments.....	4-3
4.3	Named Caches .....	4-3

## 5. Database Security

5.1	Approach .....	5-1
5.2	Users .....	5-1
5.3	Groups.....	5-2
5.4	Roles .....	5-2
5.5	Object Permissions.....	5-3

## 6. Scripts

6.1	Installation Scripts .....	6-1
6.2	De-Installation Scripts.....	6-1
6.3	Backup/Recovery Scripts .....	6-1
6.4	Miscellaneous Scripts .....	6-2

## List of Figures

3-1	ERD Key.....	3-1
5-1	Sybase Approach to SQL Server Security .....	5-1

## List of Tables

Table 3-1. Data Tables Listing .....	3-2
Table 3-2. EcDbVersions.....	3-3
Table 3-3. InCurrentDataTypeMap.....	3-3
Table 3-4. InDataTypeTemplate .....	3-4
Table 3-5. InEDPAddressMap.....	3-4
Table 3-6. InESDTMap.....	3-5
Table 3-7. InExternalDataProviderInfo .....	3-5
Table 3-8. InFileTypeTemplate.....	3-6
Table 3-9. InGranuleServerInfo .....	3-7
Table 3-10. InMediaCheckin .....	3-7
Table 3-11. InMediaType.....	3-7
Table 3-12. InNextAvailableID .....	3-8
Table 3-13. InRequestFileInfo .....	3-8
Table 3-14. InRequestProcessData .....	3-9
Table 3-15. InRequestProcessHeader .....	3-9
Table 3-16. InRequestSummaryData.....	3-11
Table 3-17. InRequestSummaryHeader .....	3-11
Table 3-18. InSourceMCF .....	3-12
Table 3-19. InSystemParameters .....	3-13
Table 3-20. InValDataGranuleState.....	3-13
Table 3-21. InValGranuleServerUR .....	3-13
Table 3-22. InValIngestType .....	3-14
Table 3-23. InValNotifyType.....	3-14
Table 3-24. InValParameterClass .....	3-14
Table 3-25. InValRequestState .....	3-14
Table 3-26. Dependencies on Table: InCurrentDataTypeMap .....	3-27
Table 3-27. Dependencies on Table: InDataTypeTemplate.....	3-27

Table 3-28. Dependencies on Table: InExternalDataProviderInfo .....	3-27
Table 3-29. Dependencies on Table: InMediaType .....	3-27
Table 3-30. Dependencies on Table: InRequestProcessHeader .....	3-27
Table 3-31. Dependencies on Table: InValDataGranuleState .....	3-27
Table 3-32. Dependencies on Table: InValGranuleServerUR.....	3-27
Table 3-33. Dependencies on Table: InValIngestType .....	3-28
Table 3-34. Dependencies on Table: InValNotifyType .....	3-28
Table 3-35. Dependencies on Table: InValParameterClass.....	3-28
Table 3-36. Dependencies on Table: InValRequestState.....	3-28
Table 3-37. Trigger Listing .....	3-28
Table 4-1. Index Type Key .....	4-1
Table 4-2. Index List .....	4-1
Table 4-3. Segment Descriptions .....	4-3
Table 5-1. Permission Key .....	5-3
Table 5-2. Object Permissions .....	5-3
Table 6-1. Installation Scripts .....	6-1
Table 6-2. De-Installation Scripts .....	6-1
Table 6-3. Backup and Recovery Scripts .....	6-2
Table 6-4. Miscellaneous Scripts and Input Data Files.....	6-2

## **Appendix A. Entity Relationship Diagram**

### **Abbreviations and Acronyms**

# 1. Introduction

---

## 1.1 Purpose and Scope

The purpose of INGEST Database Design and Database Schema Specification document is to describe the database design and schema specifications implemented to support the data requirements of Release 5B.06 INGEST CSCI.

## 1.2 Document Organization

Section 1 provides information regarding the identification, purpose, scope and audience of this document.

Section 2 provides a listing of the related documents, which were used as a source of information for this document.

Section 3 contains the database overview for the INGEST physical data model which includes a description of the database tables, triggers, stored procedures, flat files, and attributes.

Section 4 provides a description of database performance and tuning factors such as indexes, caches, and segments.

Section 5 provide a description of the database security infrastructure used and list of the users, groups, roles, and permissions available upon initial installation.

Section 6 provides a description of scripts used for installation, de-installation, backup/recovery, and other miscellaneous functions.

This page intentionally left blank.

## 2. Related Documents

---

### 2.1 Applicable Documents

The following documents, including Internet links, are referenced in this document, or are directly applicable, or contain policies or other directive matters that are binding upon the content of this volume.

305-CD-510	Release 5B Segment Design Specification for the ECS Project
920-TDG-009	DAAC Hardware Database Mapping/GSFC
920-TDN- 009	DAAC Hardware Database Mapping/NSIDC
920-TDE-009	DAAC Hardware Database Mapping/EDC
920-TDL-009	DAAC Hardware Database Mapping/LARC
920-TDS-009	DAAC Hardware Database Mapping/SMC
920-TDG-010	DAAC Database Configuration/GSFC
920-TDN-010	DAAC Database Configuration/NSIDC
920-TDE-010	DAAC Database Configuration/EDC
920-TDL-010	DAAC Database Configuration/LARC
920-TDS-010	DAAC Database Configuration/SMC
920-TDG-011	DAAC Sybase Log Mapping/GSFC
920-TDN-011	DAAC Sybase Log Mapping/NSIDC
920-TDE-011	DAAC Sybase Log Mapping/EDC
920-TDL-011	DAAC Sybase Log Mapping/LARC
920-TDS-011	DAAC Sybase Log Mapping/SMC
922-TDG-013	Disk Partitions/GSFC
922-TDN-013	Disk Partitions/NSIDC
922-TDE-013	Disk Partitions/EDC
922-TDL-013	Disk Partitions/LARC
922-TDS-013	Disk Partitions/SMC



These documents are maintained as part of the ECS baseline and available on the world-wide web at the URL: <http://cmdm.east.hitc.com/baseline>. Please note that this is a partial mirror site in that some items are not available (they are identified) since this is OPEN to all. This site may also be reached through the EDHS homepage. Scroll page to the connections line and click on the ECS Baseline Information System link.

## **2.2 Information Documents**

The following documents, although not directly applicable, amplify or clarify the information presented in this document. These documents are not binding on this document.

313-CD-510	Release 5B CSMS/SDPS Internal ICD for the ECS Project
609-CD-510	Release 5B Operations Tools Manual for the ECS Project
611-CD-510	Release 5B Mission Operation Procedures for the ECS Project

These documents are accessible via the EDHS homepage.

# 3. Data Design

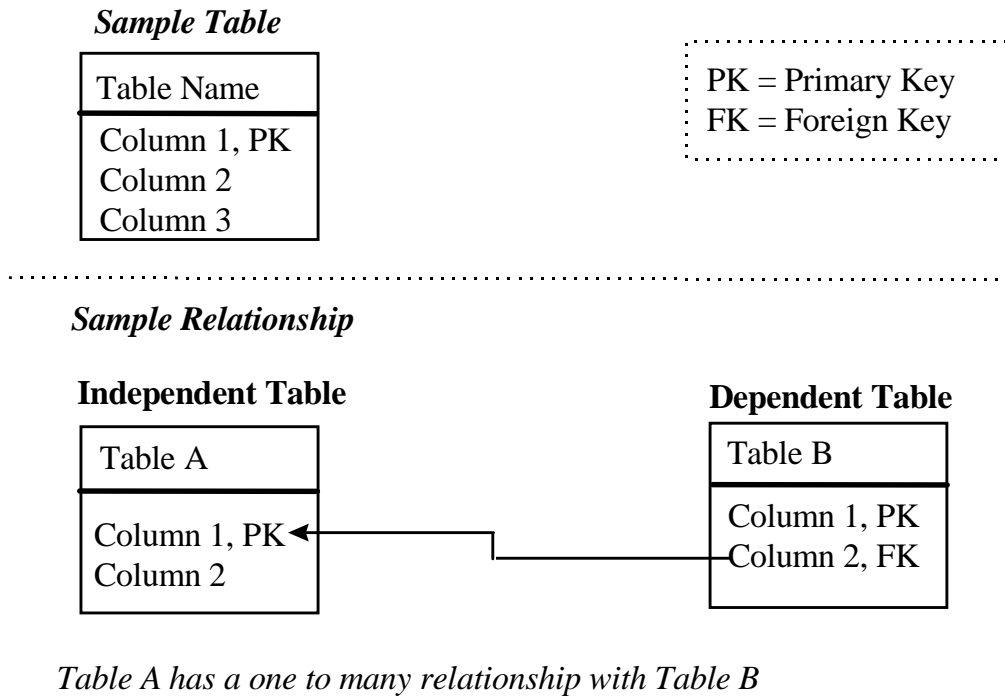
---

## 3.1 Database Overview

The INGEST database implements the large majority of the persistent data requirements for the INGEST subsystem. The database is designed in such a manner as to satisfy business policy while maintaining data integrity and consistency. Database tables are implemented using the Sybase Relational Database Management system (RDBMS). All components of the INGEST database are described in the section which follow, in sufficient detail to support maintenance needs.

### 3.1.1 Physical Data Model Entity Relationship Diagram

The Entity Relationship Diagram (ERD) presents a schematic depiction of the INGEST physical data model. The ERDs presented here for the INGEST database were produced using the Power Designer Data Architect Computer Aided Software Engineering (CASE) tool. ERDs represent the relationship between entities or database tables. On ERDs, tables are represented by rectangles and relationships are represented as arrow (see Figure 4-1).



**Figure 3-1. ERD Key**

### 3.1.2 Tables

A listing of each the tables in the INGEST database is given here. A brief definition of each of these tables follows including a listing of the columns comprising the table. The column list indicates if the column is part of the primary key for the table, that is, if the columns can be used alone or in combination with other primary key columns to uniquely identify a single row in the table. The column list also indicates whether the column is a mandatory attribute that must be included in every row.

Data requirements for INGEST fall into logical categories:

1. Active requests – data on in progress Ingest requests
2. Completed requests – data on completed Ingest requests
3. Configuration Data – data that defined how specific datatypes are provided and processed
4. Primary Key management – the next available primary key value
5. Validation Data – Domain definitions for codes used by INGEST software.

**Table 3-1. Data Tables Listing (1 of 2)**

Table Name	Logical Grouping
EcDbVersions	Configuration Data
InCurrentDataTypeMap	Configuration Data
InDataTypeTemplate	Configuration Data
InEDPAddressMap	Configuration Data
InESDTMap	Configuration Data
InExternalDataProviderInfo	Configuration Data
InFileTypeTemplate	Configuration Data
InGranuleServerInfo	Configuration Data
InMediaCheckin	Configuration Data
InMediaType	Validation Data
InNextAvailableID	Primary Key Management
InRequestFileInfo	Configuration Data
InRequestProcessData	Active Requests
InRequestProcessHeader	Active Requests
InRequestSummaryData	Completed Requests
InRequestSummaryHeader	Complete Requests
InSourceMCF	Configuration Data
InSystemParameters	Configuration Data
InValBypassPreproc	Validation Data
InValDataGranuleState	Validation Data
InValGranuleServerUR	Validation Data

**Table 3-1. Data Tables Listing (2 of 2)**

Table Name	Logical Grouping
InValIngestType	Validation Data
InValNotifyType	Validation Data
InValParameterClass	Validation Data
InValRequestState	Validation Data

Table 3-2 identifies the current version level of the Ingest database.

**Table 3-2. EcDbVersions**

Column Name	Data Type	PK Column	Mandatory Column
EcDbSchemaVersionID	Smallint	Yes	Yes
EcDbDropVersion	Char(64)	No	Yes
EcDbDropDescription	Varchar(255)	No	Yes
EcDbCurrentVersionFlag	Char(1)	No	Yes
EcDbDatabaseName	Varchar(255)	No	No
EcDbDropInstallDate	Datetime	No	No
EcDbSybaseVersion	Varchar(255)	No	No
EcDbSybaseServer	Varchar(255)	No	No
EcDbComments	Varchar(255)	No	No
EcDbUpdateProcess	Varchar(255)	No	No

Table 3-3 holds a data provider and the current version id.

**Table 3-3. InCurrentDataTypeMap**

Column Name	Data Type	PK Column	Mandatory Column
DataType	Varchar(32)	Yes	Yes
VersionID	Varchar(16)	No	Yes

Table 3-4 defines valid Earth Science Data Types (ESDTs) that Ingest is capable of ingesting

**Table 3-4. InDataTypeTemplate**

Column Name	Data Type	PK Column	Mandatory Column
DataType	nvarchar(32)	Yes	Yes
DataTypeByPassPreproc	Varchar(20)	No	Yes
ExpeditedDataType	nvarchar(32)	No	No
ExpeditedVersionID	varchar(16)	No	No
FileTypeTemplateKey	nvarchar(32)	No	Yes
GranuleServerURKey	tinyint	No	Yes
IngestFtpKey	Varchar(30)	No	Yes
OutputDestination	char(40)	No	No
PrimaryFlag	tinyint	No	Yes
SdsrvUR	Nvarchar(255)	No	No
SecondaryDataType	nvarchar(32)	No	No
ServerType	Char(5)	No	No
StorageMgmtKey	varchar(30)	No	Yes
TestDataType	nvarchar(32)	No	No
VersionID	varchar(16)	Yes	Yes

Table 3-5 associates an External Data Provider with his Internet address.

**Table 3-5. InEDPAddressMap**

Column Name	Data Type	PK Column	Mandatory Column
IPAddress	varchar(255)	Yes	Yes
External Data Provider	Varchar(20)	No	Yes

Table 3-6 holds the information of the Earth Science data type (ESDT) and the Client data type. It defines what a user can do with (a Class of) Granules.

**Table 3-6. InESDTMap**

<b>Name</b>	<b>Data Type</b>	<b>PK Column</b>	<b>Mandatory Column</b>
ClientDataType	varchar(15)	No	Yes
DataDescriptor	varchar(15)	No	No
ESDT	nvarchar(32)	No	Yes
ExternalDataProvider	varchar(20)	Yes	Yes
VersionID	Varchar(16)	No	Yes

Table 3-7 holds the configuration and current processing status information for a data provider.

**Table 3-7. InExternalDataProviderInfo (1 of 2)**

<b>Column Name</b>	<b>Data Type</b>	<b>PK Column</b>	<b>Mandatory Column</b>
CDSEntry	varchar(255)	No	No
CurrentRequests	int	No	Yes
CurrentVolume	float	No	Yes
DataProvMediaStorageMgmtKey	varchar(30)	No	No
EmailAddress	varchar(255)	No	No
ExternalDataProvider	varchar(20)	Yes	Yes
FTPPassword	binary(30)	No	No
FTPPasswordSize	int	No	No
FTPUsername	varchar(10)	No	No
HTMLPassword	binary(30)	No	No
HTMLPasswordSize	int	No	No
IngestPriority	varchar(10)	No	Yes
IngestType	varchar(40)	No	Yes
MaximumRequests	int	No	Yes
NotifyFTPDirectory	varchar(255)	No	No
NotifyFTPNode	varchar(255)	No	No

**Table 3-7. InExternalDataProviderInfo (2 of 2)**

Name	Code	Type	PK	Mandatory
NotifyFTPPassword	NotifyFTPPassword	binary(30)	No	No
NotifyFTPPasswordSize	NotifyFTPPasswordSize	int	No	No
NotifyFTPUsername	NotifyFTPUsername	varchar(10)	No	No
NotifyNamingConv	NotifyNamingConv	varchar(10)	No	No
NotifyOperator	NotifyOperator	tinyint	No	No
NotifyType	NotifyType	varchar(10)	No	Yes
PostTransferSizeCheck	PostTransferSizeCheck	tinyint	No	Yes
ProviderByPassPreProc	ProviderByPassPreProc	varchar(20)	No	No
TransferFlag	TransferFlag	tinyint	No	Yes
UUID	UUID	char(36)	No	No
VolumeThreshold	VolumeThreshold	Float(48)	No	Yes

Table 3-8 defines all valid Files that make up a DataType.

**Table 3-8. InFileTypeTemplate**

Name	Data Type	PK Column	Mandatory Column
ArchivalFlag	char(1)	No	No
AttributeName	varchar(255)	No	No
ExtConvFileName	varchar(48)	No	No
ExtConvType	varchar(32)	No	No
FileClass	char(4)	No	No
FileType	nvarchar(32)	Yes	Yes
FileTypeTemplateKey	nvarchar(32)	Yes	Yes
InternalFileType	nvarchar(32)	No	No
LineDelimiter	char(1)	No	No
Maximum	tinyint	No	Yes
MetadataSpecialization	varchar(48)	No	No
Minimum	tinyint	No	Yes
ParameterClassDefault	varchar(8)	No	No
PVSeparator	char(1)	No	No
RequiredFlag	char(1)	No	Yes
ScienceSpecialization	varchar(48)	No	No
SourceMCF	varchar(32)	No	No
StringDelimiter	char(1)	No	No

Table 3-9 holds information about the server into which granules will be ingested.

**Table 3-9. InGranuleServerInfo**

Name	Data Type	PK Column	Mandatory Column
GranuleServerURKey	tinyint	Yes	Yes
TotalGranuleThreshold	int	No	Yes
VolumeThreshold	numeric(21)	No	Yes

Table 3-10 holds information about the different types of media on which data will come in.

**Table 3-10. InMediaCheckin**

Name	Code	Type	PK	Mandatory
CheckinTime	CheckinTime	datetime	No	Yes
ExternalDataProvider	ExternalDataProvider	varchar(20)	No	Yes
MediaType	MediaType	varchar(10)	No	Yes
State	State	char(15)	No	Yes
VolumeID	VolumeID	varchar(40)	Yes	Yes

Table 3-11 holds the valid values of the media type available that can be ingested.

**Table 3-11. InMediaType**

Name	Code	Type	PK	Mandatory
MediaType	MediaType	varchar(10)	Yes	Yes
StackReq	StackReq	tinyint	No	Yes
StackSlotReq	StackSlotReq	tinyint	No	Yes



Table 3-12 holds the next available RequestID to be given.

**Table 3-12. InNextAvailableID**

<b>Name</b>	<b>Code</b>	<b>Type</b>	<b>PK</b>	<b>Mandatory</b>
NextID	NextID	int	Yes	Yes

Table 3-13 holds the file information for the ingest request.

**Table 3-13. InRequestFileInfo**

<b>Name</b>	<b>Code</b>	<b>Type</b>	<b>PK</b>	<b>Mandatory</b>
CompletionTime	CompletionTime	int	No	Yes
DataGranuleID	DataGranuleID	int	Yes	Yes
FileID	FileID	varchar(245)	Yes	Yes
FileNumber	FileNumber	int	No	Yes
FileSize	FileSize	int	No	Yes
FileState	FileState	char(15)	No	No
FileStatus	FileStatus	smallint	No	No
FileType	FileType	nvarchar(32)	No	No
RequestID	RequestID	int	Yes	Yes
SourceDirectoryID	SourceDirectoryID	varchar(255)	No	Yes
TimeStamp	TimeStamp	datetime	No	No

Table 3-14 holds the granule information for the ingest request, this table might have different granules for a single request.

**Table 3-14. InRequestProcessData**

Name	Code	Type	PK	Mandatory
DataDescriptor	DataDescriptor	varchar(60)	No	No
DataGranuleID	DataGranuleID	int	Yes	Yes
DataGranuleState	DataGranuleState	varchar(20)	No	Yes
DataGranuleVolume	DataGranuleVolume	float(48)	No	No
Data Type	Data Type	datatype	Yes	Yes
GranuleCompleted	GranuleCompleted	smallint	No	Yes
GranuleHandle	GranuleHandle	char(36)	No	No
GranuleRpciID	GranuleRpciID	nvarchar(255)	No	No
NodeName	NodeName	varchar(255)	No	No
ProcessingEndDateTime	ProcessingEndDateTime	datetime	No	No
ProcessingStartDateTime	ProcessingStartDateTime	datetime	No	No
RequestID	RequestID	requestid	Yes	Yes
RetryCount	RetryCount	smallint	No	No
StagingTagID	StagingTagID	varchar(255)	No	No
TimeToArchive	TimeToArchive	int	No	No
TimeToPreprocess	TimeToPreprocess	int	No	No
TimeToXfer	TimeToXfer	int	No	No
TotalFileCount	TotalFileCount	int	No	No
VersionID	VersionID	varchar(16)	No	Yes

Table 3-15 holds information on how the request data will be processed such as the number of granules, the number of files, the path where the DAN file exits, the External Data Provider name, priority, etc. This table holds one row per RequestID.

**Table 3-15. InRequestProcessHeader (1 of 2)**

Name	Code	Type	PK	Mandatory
ArchPercentComplete	ArchPercentComplete	smallint	No	Yes
CDSName	CDSName	varchar(255)	No	No
DANFileName	DANFileName	varchar(255)	No	Yes
DDNDestination	DDNDestination	int	No	No

**Table 3-15. InRequestProcessHeader (2 of 2)**

<b>Column Name</b>	<b>Column Code</b>	<b>Data Type</b>	<b>PK Column</b>	<b>Mandatory Column</b>
ExpirationDateTime	ExpirationDateTime	datetime	No	No
ExpiredFlag	ExpiredFlag	smallint	No	Yes
ExternalDataProvider	ExternalDataProvider	varchar(20)	No	Yes
IngestType	IngestType	varchar(40)	No	Yes
InitialRpcID	InitialRpcID	nvarchar(255)	No	No
Mission	Mission	varchar(60)	No	No
PreprocPercentComplete	PreprocPercentComplete	smallint	No	Yes
ProcessingEndDateTime	ProcessingEndDateTime	datetime	No	No
ProcessingStartDateTime	ProcessingStartDateTime	datetime	No	No
ReqMgrRpcID	ReqMgrRpcID	nvarchar(255)	No	No
RequestID	RequestID	int	Yes	Yes
RequestPriority	RequestPriority	varchar(10)	No	Yes
RequestStateKey	RequestStateKey	tinyint	No	No
SequenceID	SequenceID	int	No	Yes
SpecProc	SpecProc	smallint	No	Yes
TotalDataVolume	TotalDataVolume	float	No	No
TotalFileCount	TotalFileCount	int	No	No
TotalGranuleCount	TotalGranuleCount	int	No	No
TransferFlag	TransferFlag	smallint	No	Yes
UUID	UUID	char(36)	No	No
XferPercentComplete	XferPercentComplete	smallint	No	Yes

Table 3-16 contains a historical summary of ingest request process level data. Data is copied to this table from InRequestProcessData table when processing has been completed on an ingest request for more than the MonitorTimeForCompletedRequests in the InSystemParameters table. When the data is copied, the granule information for the ingest request is deleted from the InRequestProcessData table.

**Table 3-16. InRequestSummaryData**

Name	Code	Type	PK	Mandatory
DataGranuleID	DataGranuleID	int	Yes	Yes
DataGranuleState	DataGranuleState	varchar(20)	No	No
DataGranuleVolume	DataGranuleVolume	float(48)	No	No
DataType	DataType	varchar(32)	Yes	Yes
NodeName	NodeName	varchar(255)	No	No
ProcessingEndDateTime	ProcessingEndDateTime	datetime	No	No
ProcessingStartDateTime	ProcessingStartDateTime	datetime	No	No
RequestID	RequestID	Int	Yes	Yes
RetryCount	RetryCount	Smallint	No	No
TimeToArchive	TimeToArchive	Int	No	No
TimeToPreprocess	TimeToPreprocess	Int	No	No
TimeToXfer	TimeToXfer	Int	No	No
TotalFileCount	TotalFileCount	Int	No	No

Table 3-17 contains a historical summary of ingest request summary level data. Data is copied to this table from InRequestProcessHeader table when processing has been completed on an ingest request for more than the MonitorTimeForCompletedRequests in the InSystemParameters table. When the data is copied the granule information for the ingest request is deleted from the InRequestProcessHeader table.

**Table 3-17. InRequestSummaryHeader (1 of 2)**

Name	Code	Type	PK	Mandatory
DANFileName	DANFileName	varchar(255)	No	Yes
ExternalDataProvider	ExternalDataProvider	varchar(20)	No	Yes
IngestType	IngestType	varchar(40)	No	Yes
Mission	Mission	varchar(60)	No	No
ProcessingEndDateTime	ProcessingEndDateTime	datetime	No	No

**Table 3-17. InRequestSummaryHeader (2 of 2)**

<b>Name</b>	<b>Code</b>	<b>Type</b>	<b>PK</b>	<b>Mandatory</b>
ProcessingStartDateTime	ProcessingStartDateTime	datetime	No	No
RequestID	RequestID	int	Yes	Yes
RequestPriority	RequestPriority	varchar(10)	No	No
RequestStateKey	RequestStateKey	tinyint	No	No
TimeToArchive	TimeToArchive	int	No	No
TimeToPreprocess	TimeToPreprocess	int	No	No
TimeToXfer	TimeToXfer	int	No	No
TotalDataVolume	TotalDataVolume	float	No	No
TotalFileCount	TotalFileCount	int	No	No
TotalGranuleCount	TotalGranuleCount	int	No	No
TotalSuccessfulGranules	TotalSuccessfulGranules	int	No	No

Table 3-18 is, initially, pre-populated with the valid metadata types for each FileType. It is the mapping that “points” you to the (meta)data and indicates “how” to handle the data in a standard (ODL) format.

**Table 3-18. InSourceMCF**

<b>Name</b>	<b>Code</b>	<b>Type</b>	<b>PK</b>	<b>Mandatory</b>
CSDT	CSDT	varchar(32)	No	No
DateTimeFormat	DateTimeFormat	varchar(32)	No	No
DateTimeValueFormat	DateTimeValueFormat	varchar(32)	No	No
FieldLength	FieldLength	int	No	No
FieldLocationOffset	FieldLocationOffset	int	No	No
GroupLabel	GroupLabel	varchar(32)	No	No
MandatoryFlag	MandatoryFlag	char(1)	No	No
ParameterClass	ParameterClass	varchar(8)	No	No
ProductSpecific	ProductSpecific	varchar(48)	No	No
SourceID	SourceID	int	Yes	Yes
SourceMCF	SourceMCF	varchar(32)	No	Yes
SourceParameter	SourceParameter	varchar(255)	No	No
SpecialProcessing	SpecialProcessing	varchar(8)	No	No
TargetParameter	TargetParameter	varchar(255)	No	No

Table 3-19 holds the system parameters used to manage ingest processing. There is only one entry in this table.

**Table 3-19. InSystemParameters**

Name	Code	Type	PK	Mandatory
CommunicationRetryCount	CommunicationRetryCount	int	No	Yes
CommunicationRetryInterval	CommunicationRetryInterval	int	No	Yes
CurrentTotalRequests	CurrentTotalRequests	int	No	Yes
CurrentTotalVolume	CurrentTotalVolume	float(48)	No	Yes
IngestFTPPassword	IngestFTPPassword	binary(30)	No	No
IngestFTPPasswordSize	IngestFTPPasswordSize	int	No	No
IngestFTPUsername	IngestFTPUsername	varchar(10)	No	No
MaximumTotalRequests	MaximumTotalRequests	int	No	Yes
MaximumTotalVolume	MaximumTotalVolume	float(48)	No	Yes
MediaStorageMgmtKey	MediaStorageMgmtKey	varchar(30)	No	Yes
MonitorTimeForCompletedRequest	MonitorTimeForCompletedRequest	int	No	Yes
ScreenUpdateInterval	ScreenUpdateInterval	int	No	Yes
SysParameterID	SysParameterID	int	Y	Yes

Table 3-20 defines all the valid values for a data granule state.

**Table 3-20. InValDataGranuleState**

Name	Code	Type	PK	Mandatory
DataGranuleState	DataGranuleState	varchar(20)	Yes	Yes

Table 3-21 defines all the valid values for the granule server UR

**Table 3-21. InValGranuleServerUR**

Name	Code	Type	PK	Mandatory
GranuleServerUR	GranuleServerUR	varchar(40)	No	Yes
GranuleServerURKey	GranuleServerURKey	tinyint	Yes	Yes

Table 3-22 defines all the valid values for an ingest type.

**Table 3-22. InValIngestType**

<b>Name</b>	<b>Code</b>	<b>Type</b>	<b>PK</b>	<b>Mandatory</b>
IngestType	IngestType	varchar(40)	Yes	Yes

Table 3-23 defines all the valid values for a notify type.

**Table 3-23. InValNotifyType**

<b>Name</b>	<b>Code</b>	<b>Type</b>	<b>PK</b>	<b>Mandatory</b>
NotifyType	NotifyType	varchar(10)	Yes	Yes

Table 3-24 defines all the valid values for a parameter class

**Table 3-24. InValParameterClass**

<b>Name</b>	<b>Code</b>	<b>Type</b>	<b>PK</b>	<b>Mandatory</b>
ParameterClass	ParameterClass	varchar(8)	Yes	Yes

Table 3-25 defines all the valid values for a request state.

**Table 3-25. InValRequestState**

<b>Name</b>	<b>Code</b>	<b>Type</b>	<b>PK</b>	<b>Mandatory</b>
RequestState	RequestState	varchar(15)	No	Yes
RequestStateKey	RequestStateKey	tinyint	Yes	Yes

### 3.1.3 Columns

Brief definitions of each of the columns present in the database tables defined above are contained herein.

Column Name	Column Description	Valid Values
ArchivalFlag	Boolean flag to indicate if the file is needed to be archived or not.	Y = Yes, N = No
ArchPercentComplete	This is the percentage of files inserstions into the earchive completed for the requested data.	0 -100
AttributeName	The name of the metadata attribute as defined in the Core Metadata Model. Valid names/attributes are either core or product specific.	See 420-TP-021
CDSEntry	The name identified in a CDS entry. (i.e.: EcCsLandsat7 Gateway)	
CDSName	The name of a CDS component.	
CheckinTime	This is the date and time when the media was checked in.	
ClientDataType	This is the client data type.	
CommunicationRetryCount	This holds the number of times that a user retries a communication.	
CommunicationRetryInterval	This is the interval between communication retries of a user.	
CompletionTime	This is the calculated length of time for completion for a RequestID's (Granule) File.	
CSDT	This is the Computer Science Data Type (CSDT), (i.e., int, float, double, short, string, LittleEndian_float, LittleEndian_Int, etc.)	See 420-TP-021
CurrentRequests	Keeps a running total of the number of requests currently in the system for an External Data Provider.	



Cont'd

<b>Column Name</b>	<b>Column Description</b>	<b>Valid Values</b>
CurrentTotalRequests	This is the total requests that are current in the system.	
CurrentTotalVolume	This is the total volume of ingested data for all RequestIDs that are currently in the system.	
CurrentVolume	A running total of the volume of RequestIDs currently in the system for an External Data Provider.	
DANFileName	File name of Data Availability Notice (DAN), transmitting availability notification information for a single granule.	
DataDescriptor	This is the data descriptor.	
DataGranuleID	This is the data granule identifier.	
DataGranuleState	This is the state of a data granule.	ArchErr, Archived, Cancelled, New, PreprocErr, Preprocessed, Terminated, Transferred, XferErr.
DataGranuleVolume	Total data volume to be ingested for a data granule in an ingest request. The total data volume for the data granule is determined by summing the data volumes for the files comprising the data granule.	
EcDbComments	Notes or comments on the database version level.	
EcDbCurrentVersionFlag	Flag indicating if this row represents the current database version entry	1= yes, 0 = no
EcDbDatabaseName	The name of the database for which this database version level is applied.	
EcDbDropDescription	The official description of the ECS software drop for this database version level.	
EcDbDropInstallDate	The date and time that the database version level was installed.	
EcDbDropVersion	The official name of the ECS software drop for this database version level.	

Cont'd

Column Name	Column Description	Valid Values
EcDbSchemaVersionId	The subsystem-specific identifier for this database schema version.	
EcDbSybaseServer	The name of the baseline Sybase SQL server controlling this database.	See 920-TDx-009
EcDbSybaseVersion	The software release version of the Sybase SQL server in place when this database version level was initially installed.	
EcDbUpdateProcess	The installation method by which this database version level was installed	
DataProvMediaStorageMgmtKey	This is the data provider media storage management key.	
DataType	This holds primary ESDT short-name of a ECS data type that is handled by a particular data server. (i.e.,AM1 L0, SAGEIII L0, Radat ALT L0, Landsat7 L0R, SeaWinds,Ancillary, etc.)	See column ShortName in 420-TP-021
DataTypeBypassPreProc	This column determines whether a data type is to have preprocessing bypassed or not when the ProviderBypassPreproc column in the InExternalDataProviderInfo table = MIXED.	NONE – Normal preprocessing is done DIRECT_INSERT – no preprocessing is done
DataVersion	This holds the data version.	
DateTimeFormat	This is the date time format that the file contains; required for standard handling by Science Data Server (i.e., yy-mm-dd - hh:ii:ss, yyyy-mm-ddThh:ii:ss.ssss, etc.)	
DateTimeValueFormat	This is the value of the date time format. (i.e., %06.0f, etc.)	
DDNDestination	This column identifies the destination where the Data Delivery Notices (DDN) were placed for a given ingest request.	

Cont'd

Column Name	Column Description	Valid Values
EmailAddress	This is the email address of the external data provider.	
ESDT	The short name of an Earth Science Data Type. An ECS type that Science Data Server holds.	See column Shortname in 420-TP-021
ExpeditedDataType	This is the name of the expedited data type.	See column Shortname in 420-TP-021
ExpeditedVersionID	This is the version identifier of the expedited datatype.	
ExpirationDateTime	Date/time by which the corresponding ingest request must be completed (i.e., archive insertion complete and response returned to the External Data Provider).	
ExpiredFlag	A boolean flag indicating whether data has passed the expiration date.	"Y"=yes, "N" = No
ExtConvFileName	This holds the provider's external file name to be converted.	
ExtConvType	This is the external file type to be converted	Script, SharedObject
ExternalDataProvider	This is the name of the External data provider.	
FieldLength	This is the field length of the file.	
FieldLocationOffset	This is the integer location offset for the field.	
FileClass	This holds the 3-letter acronym name class of the file.,(i.e., TEX, HDF).	
FileID	The unique identifier of the ingest file.	
FileNumber	This is the file number indicator.	
FileSize	This attribute represents the size of the individual file.	
FileState	This is the current processing state of the file being ingested.	
FileStatus	Final error status for the ingest processing of a data granule.	TBI

Cont'd

Column Name	Column Description	Valid Values
FileType	This holds the valid file type of the ingest file	SCIENCE, METADATA, BROWSE, NATIVE, HTML, TEXT, PDF, POSTSCRIPT, RTF, DANFILE, DOCUMENT, DATA, IMAGE1, IMAGE2, IMAGE3, IMAGE4, IMAGE5, IMAGE6, IMAGE7, IMAGE8, CALIBRATION, MSCD, PCD, ALGORITHM
FileTypeTemplateKey	The unique grouping of all related FileTypes.	
FTPPassword	This is the FTP user password used by the external data provider. This is used to access the External Data Provider's system.	
FTPPasswordSize	This is the FTP user password size used by the external data provider. This is used to access the External Data Provider's system.	
FTPUsername	This is the FTP user name used by the external data provider. This is used to access the External Data Provider's system.	
GranuleCompleted	This indicates if the granule Ingest was completed.	
GranuleHandle	This is the name of the granule handle. The granule handle is used to identify the granule to be accessed by the software.	
GranuleRpcID	This is the remote procedure call identifier associated with the granule for a given ingest request.	
GranuleServerUR	This is the CDS name of the granule server.	EclnGran, InGranServer, InGranServerN, InGranServerNsgi, EclnGranN, EclnGranNsgi, N = Number
GranuleServerURKey	This holds the Granule ServerID that is mapped to a specific Granule Server's name.	

Cont'd

<b>Column Name</b>	<b>Column Description</b>	<b>Valid Values</b>
GroupLabel	This is a label for a group that a file contains.	BEGIN- END START- STOP, DATASTART- DATAEND, SUBSTART- SUBSTOP, STARTDATETIME- ENDDATETIME
HTMLPassword	This is the HTML user password used by the external data provider.	
HTMLPasswordSize	This is the HTML user password size used by the external data provider.	
IngestFTPKey	This is the ingest FTP Key.	
IngestFTPPassword	This is the ingest FTP user password.	
IngestFTPPasswordSize	This is the ingest FTP user password size.	
IngestFTPUsername	This is the ingest FTP user name.	
IngestPriority	This is the ingest priority assigned to a request.	Normal, High, Low
IngestType	The type of Ingest processing requested.	Auto, Interactive, Media, Polling_w/DR, Polling_wo/DR
InitialRpcID	The first RPC id created for an Ingest request at the time when the request is received by Ingest. As a request moves through Ingest, its RPC id changes as rpcs to other servers are made. The InitialRpcID is used for fault recovery so that a request which is warm started has the same sequence of RPC ids as it initially had.	
InternalFileType	This attribute identifies the type of internal file.	Metadata, Science, Browse, ScienceN, BrowseN, Native N = number
LineDelimiter	This attribute will define the symbol used to indicate the end of a parameter-value metadata statement.	
MandatoryFlag	Flag indicating if a field in the MCF is mandatory or not.	0 = not mandatory, 1 = mandatory
Maximum	The maximum number of fields in the file comprising a given defined datatype.	

Cont'd

Column Name	Column Description	Valid Values
MaximumRequests	The maximum requests available for the external data provider.	
MaximumTotalRequests	The maximum requests that the system can hold at a time.	
MaximumTotalVolume	This is the maximum volume of data that the system can hold at a time.	
MediaStorageMgmtKey	A System's Parameter, this is the valid storage management key of where the data will be stored.	<HWCI>_<mode> i.e. HWCI2_TS1
MediaType	This is the description of the media type.	4mm Tape, 8mm Tape, D3 Tape
MetadataSpecialization	This attribute holds the specialization of the metadata in the file.	SCENE, InBOMMetaData, InPVMetadata, InFDDMetadata, InFDDMetaData, InODLMetadata, InODLMetaData, InBUFRMetadata, InNCEPMetadata, InISSCPMetadata, InOZONESBMetadata, InNCEPT62MetaData
Minimum	The minimum number of fields in the file.	
Mission	This is the name of the mission which generated the data to be ingested. (i.e., AM-1).	
MonitorTimeForCompletedRequest	This is the length of time a request is held before the request information can be moved to the archive (InRequestSummary) tables.	
NextID	Automatically generated in a sequential order by the database, this provides the unique RequestID.	
NodeName	This holds the path where the data granule exists.	
NotifyFTPDirectory	This is the directory where the notify FTP exists.	
NotifyFTPNode	This is the path where the notify FTP exists.	
NotifyFTPPassword	This is the notify FTP password.	

Cont'd

<b>Column Name</b>	<b>Column Description</b>	<b>Valid Values</b>
NotifyFTPPasswordSize	This is the notify FTP password size.	
NotifyFTPUsername	This the notify FTP user name.	
NotifyNamingConv	The naming convention governing a notification.	
NotifyOperator	This is number for the operator to be notified.	
NotifyType	This is the valid Notify Type.	Buffer , EDOS, PVL
OutputDestination	This is the name of a subsystem where the output data will be placed. (i.e., SDSRV)	
ParameterClass	This is the Parameter Class of the file.	OBJ, PV, TOOLKIT
ParameterClassDefault	This is the default for a parameter class.	
PostTransferSizeCheck	The size of the file after transfer is complete.	
PreprocPercentComplete	This is the percentage of pre processing completed.	
PrimaryFlag	This is the flag for the primary data type.	0 = not a primary data type 1= a primary data type
ProcessingEndDateTime	This is the processing end date and time for an ingest of a data granule.	
ProcessingStartDateTime	This is the processing start date and time for the ingest of a data granule.	
ProductSpecific	This is the granule's product specification.	
ProviderBypassPreProc	This column determines whether the Ingest preprocessing of data from the external data provider is to be normal preprocessing, SIPS preprocessing, cross-mode Ingest preprocessing, no preprocessing or different types of preprocessing depending on the data type.	
PVSeparator	This attribute will define the separator symbol used in between the parameter-values (i.e., = ).	

Cont'd

Column Name	Column Description	Valid Values
RequestID	This is the request identifier automatically generated from the InNextAvailableID table.	
ReqMgrRpclId	The remote procedure call identifier of the Request Manager for an ingest request.	
RequestPriority	The information that determines the order in which an ingest request will be processed relative to other ingest request waiting to be processed. This priority is provided by the InExternalDataProviderInfo for each external data provider.	Normal, NORMAL, High, Low, XPRESS
RequestState	The current processing state of an ingest request.	Active, Suspending, Suspended, SettingPriority, Canceling, Resuming, Successful, Cancelled, Failed, Partial_Failure, Terminated
RequestStateKey	A numeric identification of the current state of an ingest request.	1 - Active, 2 - Suspending, 3 - Suspended, 4 - SettingPriority, 5 - Canceling, 6 - Resuming, 7 - Successful, 8 - Cancelled, 9 - Failed, 10 - Partial_Failure, 11 - Terminated, 12 - TBI
RequiredFlag	This is a flag that is set if the file is to be required for a granule.	0 = not required, 1 = required
RetryCount	This holds the number of attempts to retry ingest of the data granule.	
ScienceSpecialization	This attribute holds the specialization of the science data in the file.	See ECSTopicKeyword, ECSTermKeyword, ECSVariableKeyword, ECSPParameterKeyword in 420-TP-021
ScreenUpdateInterval	A System's Parameter, this is the length of time (in minutes) between updates to the GUI screens.	
SdSrvUR	The universal reference for the Science Data Server where this Data Type will be stored.	
SecondaryDataType	This holds the secondary ESDT short-name of a DataType that is handled by a particular Data Server.	See Shortname in 420-TP-021



Cont'd

Column Name	Column Description	Valid Values
SequenceID	This is the sequence identifier of the request in relation to other requests.	
ServerType	Specifies either Science Data Server (Sdsrv) or Document Server (Dcsrv) to connect to using the SdsrvUR.	
SourceDirectoryID	The source directories where the files can be found.	
SourceID	Unique identifier of a Source Metadata Configuration File.	
SourceMCF	The acronym used to identify a source Metadata Configuration File (MCF). (i.e., AST_L1,L7OR1,TRMMEPH1,ODL11,etc.)	See Column Shortname in 420-TP-021
SourceParameter	The name that the external data provider uses for a metadata attribute or field.	
SpecialProcessing	This row is to specify if the request to be processed is special.	
SpecProc	This is the number of special processes.	
StackReq	This holds the stack required for the media type.	
StackSlotReq	This holds the stack slot required for the media type.	
StagingTagID	This is an identifier to indicate the staging of the data granule.	
State	Current state of External Data Provider's request processing (i.e.: Active, Suspended)	
StorageMgmtKey	This holds the valid Storage Management Key (<HWCI>) of where the data will be stored (i.e.: DRP1, SGI, HWCI1, ICL1).	
StringDelimiter	This attribute will define the symbol used to indicate the end of a parameter-value metadata string.	
SysParameterID	The unique identifier of system parameter entries.	

Cont'd

<b>Column Name</b>	<b>Column Description</b>	<b>Valid Values</b>
TargetParameter	The name that ECS uses for a metadata attribute or field.	
TestDataTypes	This is the name of the test data type.	
TimeStamp	This is the current date and time of when a File's State is completed.	
TimeToArchive	Time (in seconds) from submit of archive request to Data Server to receipt of completion status (success or fail).	
TimeToArchive	This is the date and time to archive the data.	
TimeToPreprocess	Time (in seconds) from start of preprocessing of granule to time of completion (success or fail) of preprocessing.	
TimeToXfer	Time (in seconds) from start of transfer for 1st file in granule to time of receipt of status (success or fail) for last file in granule.	
TotalDataVolume	This is the total data volume of the granule.	
TotalFileCount	This is the total number of files for the request.	
TotalGranuleCount	This is the total number of granules for the request.	
TotalGranuleThreshold	This is the total number of data granules that a server can hold.	
TotalSuccessfulGranules	This is the total number of data granule successful ingested.	
TransferFlag	This is a flag to indicate the transfer of data.	
UUID	The user id for an external data provider.	
VersionID	This holds the version identifier of the data type.	
VolumeID	This is the volume identifier for the media check in.	
VolumeThreshold	This is the volume limit available for the External Data Provider.	
XferPercentComplete	This is the percentage transfer of granule complete	

### **3.1.4 Domains**

Sybase supports the definition of specific data types, domains, to further limit the format of data for given column. User-defined data types are no longer used in the INGEST database.

### **3.1.5 Rules**

Sybase supports the definitions of rules. Rules provide a means for enforcing domain constraints on a given column. There are no rules defined in Sybase for the INGEST database.

### **3.1.6 Defaults**

Defaults are used to supply a value for a column when one is not defined at insert time. The defaults defined in Sybase for the INGEST database are described herein.

TBI

### **3.1.7 Views**

Sybase allows the definition of views as a means of limiting an application or users access to data in a table or tables. Views create a logical table from columns found in one or more tables. There are no views defined in Sybase for the INGEST database.

### **3.1.8 Integrity Constraints**

Sybase allows the enforcement of referential integrity via the use of declarative integrity constraints. Integrity constraints allow the SQL server to enforce primary and foreign key integrity checks automatically without requiring programming. Sybase constraints support “restrict-only” operations. This means that a row can not be deleted or updated if there are rows in other tables having a foreign key dependency on that row. Cascade delete and update operations can not be performed if a declarative constraint has been used. All declarative integrity constraints defined in the INGEST database are described herein.

**Table 3-26. Dependencies on Table: InCurrentDataTypeMap**

Referenced by	Primary Key	Foreign Key
InDataTypeTemplate	DataType, VersionID	DataType

**Table 3-27. Dependencies on Table: InDataTypeTemplate**

Referenced by	Primary Key	Foreign Key
InESDTMap	DataType	ESDT
InRequestProcessData	DataType	DataType

**Table 3-28. Dependencies on Table: InExternalDataProviderInfo**

Referenced by	Primary Key	Foreign Key
InESDTMap	ExternalDataProvider	ExternalDataProvider
InMediaCheckin	ExternalDataProvider	ExternalDataProvider
InRequestProcessHeader	ExternalDataProvider	ExternalDataProvider

**Table 3-29. Dependencies on Table: InMediaType**

Referenced by	Primary Key	Foreign Key
InMediaCheckin	MediaType	MediaType

**Table 3-30. Dependencies on Table: InRequestProcessHeader**

Referenced by	Primary Key	Foreign Key
InRequestProcessData	RequestID	RequestID

**Table 3-31. Dependencies on Table: InValDataGranuleState**

Referenced by	Primary Key	Foreign Key
InRequestProcessData	DataGranuleState	DataGranuleState

**Table 3-32. Dependencies on Table: InValGranuleServerUR**

Referenced by	Primary Key	Foreign Key
InGranuleServerInfo	GranuleServerURKey	GranuleServerURKey

**Table 3-33. Dependencies on Table: InValIngestType**

Referenced by	Primary Key	Foreign Key
InExternalDataProviderInfo	IngestType	IngestType
InRequestProcessHeader	IngestType	IngestType

**Table 3-34. Dependencies on Table: InValNotifyType**

Referenced by	Primary Key	Foreign Key
InExternalDataProviderInfo	NotifyType	NotifyType

**Table 3-35. Dependencies on Table: InValParameterClass**

Referenced by	Primary Key	Foreign Key
InFileTypeTemplate	ParameterClass	ParameterClassDefault
InSourceMCF	ParameterClass	ParameterClass

**Table 3-36. Dependencies on Table: InValRequestState**

Referenced by	Primary Key	Foreign Key
InRequestProcessHeader	RequestStateKey	RequestStateKey

### 3.1.9 Triggers

Sybase supports the enforcement of business policy via the use of triggers. A trigger is best defined as set of activities or checks that should be performed automatically by Sybase whenever a row is inserted, updated, or deleted from a given table. Sybase allows the definition of insert, update, and delete triggers for each table. Description of each the triggers in the INGEST database is given here. Trigger code may vary as new drops or test executables for Release 5B.06 are installed into the implemented database. For this reason trigger code listings are no longer included in this documentation but may be reviewed on-line using the installed database.

**Table 3-37. Trigger Listing**

Table Code	Trigger Name	Description
InCurrentDataTypeMatch	InCDTUpdateTrig	Referential Integrity Enforcement

### 3.1.10 Stored Procedures

Sybase also includes support for business policy via the use of stored procedures. Stored procedures are typically used to capture a set of activities or checks that will be performed on the database repeatedly to enforce business policy and maintain data integrity. Stored procedures are parsed and compiled SQL code that reside in the database and may be called by name by an application, trigger or another stored procedure. A listing of each the stored procedures in the INGEST database is given here. A brief definition of each of these stored procedures follows. Stored procedure code may vary as new drops or test executables for Release 5B.06 are installed into the implemented database. For this reason stored procedure code listings are no longer included in this documentation but may be reviewed on-line using the installed database.

#### Procedure List (1 of 2)

Name	Description
Datawarning	Notifies DBA when data segment threshold is crossed.
InCDTInsert	Inserts records into the InCurrentDataTypeMap table.
InCDTSelectByType	Selects datatypes from InCurrentDataTypeMap table.
InDTTUpdGSURkey	Update Value of Granule Server UR Key
InEDPIUpdate	Update all columns of the InExternalDataProviderInfo table for one ExternalDataProvider
InEDPIUpdCurr	Update the CurrentVolume and CurrentRequests columns of the InExternalDataProviderInfo table for one ExternalDataProvider and the CurrentTotalVolume and CurrentTotalRequests columns of the InSystemParameters table.
InGetID	To get the next available number for a new ingest request and increment the number for the next request
InGSIUpdate	Update the values of VolumeThreshold and TotalGranuleThreshold in the InGranuleServerInfo table.
InRFIInsert	Insert a record into the InRequestFileInfo table.
InRFIUpdGranuleStatus	Update the value of InRequestFileInfo.FileStatus for all files for a given Granule.
InRFIResetGran	Reset value of InRequestFileInfo.DataGranuleID.
InRFIUpdState	Update the value of InRequestFileInfo.FileState
InRFIUpdStateStatus	Update the value of InRequestFileInfo.FileState and InRequestFileInfo.FileStatus
InRFIUpdStatus	Update the value of InRequestFileInfo.FileStatus
InRPDInsert	Insert a record into the InRequestProcessData table.
InRPDRecov	To determine if a granule was completed and if not, to set its RetryCount and/or the FileStatus for the associated files appropriately.

## Procedure List (2 of 2)

Name	Description
InRPDSetGranComp	Update the value of InRequestProcessData.GranuleCompleted to 1
InRPDUpdEndTime	Update the value of InRequestProcessData.ProcessingEndDateTime
InRPDUpdHandle	Update the value of InRequestProcessData.ProcessingEndDateTime
InRPDUpdStaging	Update the value of InRequestProcessData.StagingTagID.
InRPDUpdStagReq	Update the value of InRequestProcessData.StagingTagID.
InRPDUpdStartTime	Record ProcessingStartDate and Time for a given Request.
InRPDUpdStateArch	Update the value of InRequestProcessData.DataGranuleStateand the appropriate time based on the value the state.
InRPDUpdStateDefault	Update the value of InRequestProcessData.DataGranuleState and the appropriate time based on the value the state.
InRPDUpdStatePreproc	Update request state to "pre-processing".
InRPDUpdStateXfer	Update request state to "transferring".
InRPHColdStart	Deletes all records from the InRequestProcessHeader, InRequestProcessData, and InRequestFileInfo tables.
InRPHDelCompReq	Delete and archive completed requests.
InRPHInsert	Insert a record into the InRequestProcessHeader table.
InRPHUpdDDNDest	Update Data Delivery Notice (DDN) destination information for a given Request.
InRPHUpdEndTime	Record Request .Processing End Date and Time
InRPHUpdExpired	Set status of Request to "expired".
InRPHUpdFailReq	Set status of Request to "failed".
InRPHUpdPercentCompleteArch	Provide update to Archival percentage complete status
InRPHUpdPercentCompletePre	Provide update "pre-processing" percentage complete status.
InRPHUpdPercentCompleteXfer	Provide update "transfer" percentage complete status.
InRPHUpdPriority	Change the priority of a given Request.
InRPHUpdPriorityDataProv	Change the priority of all requests received from a given Etxernal Data Provider.
InRPHUpdState	Set the Request state for a given Request as indicated.
InRPHUpdStateChng	Set the Request state for a given Request as indicated.
InRPHWarmStart	Removes all Request from the database that have an invalid request state.
InSPUpdAll	Updates system parameters found in the InSystemParameters table.
Logdump	Dump the log when log segment threshold is crossed.
Logwarning	Notify the DBA when log segment approaches capacity threshold.

## **3.2 File Usage**

There are cases when the implementation of a persistent data requirement is better suited to a flat file than to a database table. A typical example of such data is system configuration information. System configuration information is fairly static and usually has no explicit relationship to other data in the enterprise. Another common use of files in ECS is as an interface mechanism between ECS and the external world. Files utilized in INGEST are described herein.

### **3.2.1 Files Definitions**

TBI

### **3.2.2 Attributes**

TBI

### **3.2.3 Attribute Domains**

TBI



This page intentionally left blank.

## 4. Performance and Tuning Factors

---

### 4.1 Indexes

An index provides a means of locating a row in a database table based on the value of a specific column(s), without having to scan all data in the table. When properly implemented, indexes can significantly decrease the time it takes to retrieve data, thereby increasing performance. Sybase allows the definition of two types of indexes, clustered and non-clustered.

In a clustered index, the rows in a database table are physically stored in sequence-determined by the index. Clustered indexes are particularly useful, when the data is frequently retrieved in sequential order. Only one clustered index may be defined per table.

Non-clustered indexes differ from their clustered counterpart, in that, data is not physically stored in sorted order—newly added rows are stored at the end of the related database table.

A key of the types of indexes found in Ingest is provided in Table 4-1 Index Type Key. A list and description of each of the defined indexes is given in Table 4-2 Index List.

**Table 4-1. Index Type Key**

Index Type Key	Description
PK	Primary Key
FK	Foreign Key
U	Unique – Only one for the column code combination
C	Clustered or non-clustered index
Sort	ASC (ascending) of DESC (descending) order

**Table 4-2. Index List (1 of 2)**

Table Code	Index Code	PK	FK	U	C
InDataTypeTemplate	pk_indatatypeemplate	Yes	No	Yes	Yes
InESDTMap	pk_inesdtmap	Yes	Yes	Yes	Yes
InExternalDataProviderInfo	pk_inexternaldataproviderinfo	Yes	No	Yes	Yes

**Table 4-2. Index List (2 of 2)**

<b>Table Code</b>	<b>Index Code</b>	<b>PK</b>	<b>FK</b>	<b>U</b>	<b>C</b>
InFileTypeTemplate	InFTTPrimary	No	No	Yes	No
InGranuleServerInfo	pk_ingranuleserverinfo	Yes	Yes	Yes	Yes
InMediaCheckin	pk_inmediacheckin	Yes	No	Yes	Yes
InMediaType	pk_inmediatype	Yes	No	Yes	Yes
InNextAvailableID	pk_innextavailableid	Yes	No	Yes	Yes
InRequestFileInfo	pk_inrequestfileinfo InRFIGranuleID	Yes No	No No	Yes No	Yes No
InRequestProcessData	pk_inrequestprocessdata	Yes	No	Yes	Yes
InRequestProcessHeader	pk_inrequestprocessheader InRPHDataProvider	Yes No	No Yes	Yes No	Yes No
InRequestSummaryData	pk_inrequestsummarydata InRSDPrimary	Yes No	No No	Yes No	Yes No
InRequestSummaryHeader	pk_inrequestsummaryheader InRSHDataProvider InRSHProcessingStartDateTime	Yes No No	No No No	Yes No No	Yes No No
InValDataGranuleState	pk_invaldatagranulestate	Yes	No	Yes	Yes
InValGranuleServerUR	pk_invalgranuleserverur	Yes	No	Yes	Yes
InValIngestType	pk_invalingesttype	Yes	No	Yes	Yes
InValNotifyType	pk_invalnotifytype	Yes	No	Yes	Yes
InValParameterClass	pk_invalparameterclass	Yes	No	Yes	Yes
InValRequestState	pk_invalrequeststate	Yes	No	Yes	Yes

## 4.2 Segments

Sybase supports the declaration of segments. A segment is a named pointer to a storage device(s). Segments are used to physically allocate a database object to a particular storage device. Segments defined for the INGEST and all other subsystem databases are described in Table 4-3.

**Table 4-3. Segment Descriptions**

Segment Name	Description
Default	Default data segment used if no other segment specified in the create statement.
Logsegment	SYSLOGS, Transaction Logs
Systemsegment	System tables and indexes.
INSOPSDAT01	INGEST OPS mode data segment.
INSOPSIDX01	INGEST OPS mode index segment.
INSTS1DAT01	INGEST TS1 mode data segment.
INSTS1IDX01	INGEST TS1 mode index segment.
INSTS2DAT01	INGEST TS2 mode data segment.
INSTS2IDX01	INGEST TS2 mode index segment.

## 4.3 Caches

A cache is a block of memory that is used by Sybase to retain and manage pages that are currently being processed. By default, each database contains three caches:

Data cache – retains most recently accessed data and index pages

Procedure cache – retains most recently accessed stored procedure pages

User transaction log cache – transaction log pages that have not yet been written to disk for each user

The size of each of these default caches is a configurable item which must be managed on a per DAAC basis. These caches may be increased or decreased by the DAAC DBA as needed.

The data cache can be further subdivided into named caches. A *named cache* is a block of memory that is named and used by the DBMS to store data pages for select tables and/or indexes. Assigning a database table to named cache causes accessed pages to be loaded into memory and retained. The named cache does not need to be allocated to accommodate the entire database table since the DBMS manages the cache according to use. Named caches greatly increase performance by eliminating the time associated for disk input and output (I/O). There are no named caches that are currently defined for the INGEST Subsystem database. Named caches may be defined as the memory usage of the INGEST database becomes more well known and the DAACs move into an operational environment. As named caches are defined this portion of the document will be updated.

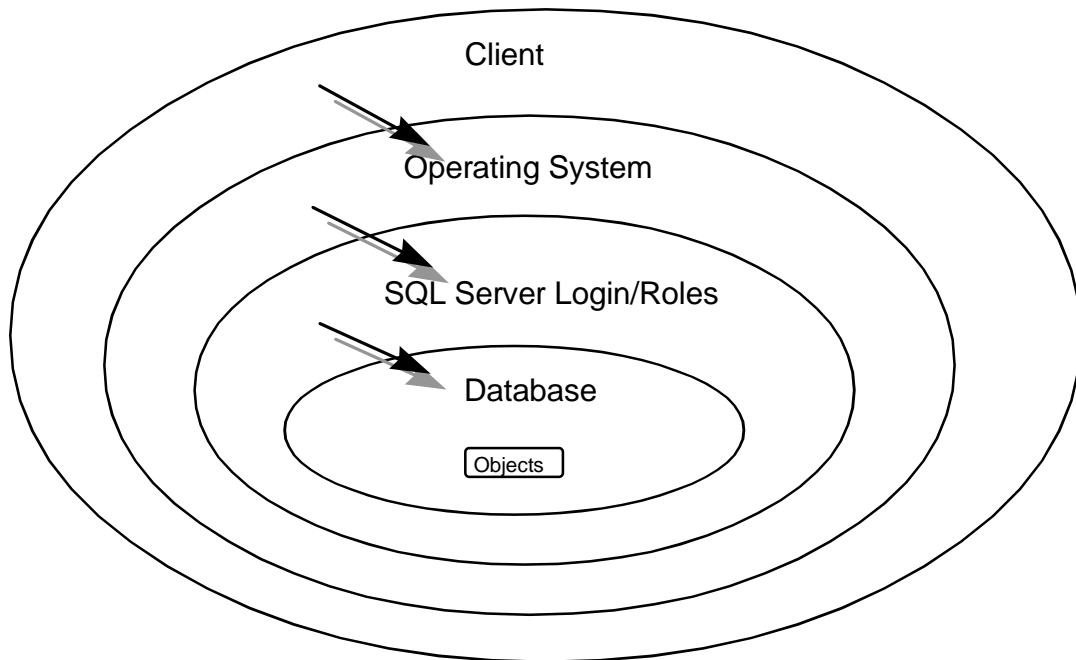
This page intentionally left blank.

# 5. Database Security

---

## 5.1 Approach

The database security discussed within this section is bounded to security implementation within the Sybase SQL Server DBMS. A Sybase general approach to security is adopted as illustrated in Figure 5-1.



**Figure 5-1. Sybase General Approach to SQL Server Security<sup>1</sup>**

## 5.2 Users

The client (user) requires a SQL Server login to access the DBMS. The login is assigned to a user with certain related permissions for gaining access to particular objects (e.g., database tables, views, commands) within the database. The System Administrator may grant or revoke objects permissions for a login individually or based on defined group or roles.

---

<sup>1</sup> Reference Sybase Student Guide: *Advanced SQL Server Administration*.

## 5.3 Groups

Groups are a means of logically associating users with similar data access needs. Once a group has been defined, object and command permissions can be granted to that group. A user who is member of a group inherits all of the permissions granted to that group. No groups have been initially defined in the INGEST Subsystem “default database. The DAACs should define database groups to support the database security requirements of their individual DAACs. Assigning each user to the appropriate group should control security for local DAAC users.

## 5.4 Roles

Roles were introduced in Sybase to allow a structured means for granting users the permissions needed to perform standard database administration activities and also provide a means for easily identifying such users. There are six pre-defined roles that may be assigned to a user. A definition of each of these roles follows, as well as a description of the types of activities that may be performed by each role.

**System Administrator** (*sa\_role*): This role is used to grant a specific user permissions needed to perform standard system administrator duties including:

- installing SQL server and specific SQL server modules
- managing the allocation of physical storage
- tuning configuration parameters
- creating databases

**Site Security Officer** (*sso\_role*): This role is used to grant a specific user the permissions needed to maintain SQL server security including:

- adding server logins
- administrating passwords
- managing the audit system
- granting users all roles except the *sa\_role*

**Operator** (*oper\_role*): This role is used to grant a specific user the permissions needed to perform standard functions for the database including:

- dumping transactions and databases
- loading transactions and databases

**Navigator** (*navigator\_role*): This role is used to grant a specific user the permissions needed to manage the navigation server.

**Replication** (*replication\_role*): This role is used to grant a specific user the permissions needed to manage the replication server.

**Sybase Technical Support** (*sybase\_ts\_role*): This role is used to grant a specific user the permissions needed to execute *database consistency checker (dbcc)*, a Sybase supplied utility supporting commands that are normally outside of the realm of routine system administrator activities.

The DAACs should review these roles and assign them to the appropriate login and/or groups.

## 5.5 Login/Group Object Permissions

During initial database installation logins used by the ECS custom code were created and permissions assigned for access to the INGEST Subsystem database. In addition, special database installation login, *ingest\_role*, was created to support database installation needs. For each login, the level of access is limited to that associated with their login, group or assigned group/role. Object Permissions are set within the installation scripts of the INGEST Subsystem for each object and group/role.

Permissions are identified in Table 5-1. A specification of the object permissions is contained in Table 5-2.

**Table 5-1. Permission Key**

Permission	Description
A	All
S	Select
I	Insert
U	Update
D	Delete
E	Execute

**Table 5-2. Object Permissions (1 of 2)**

Name	Delete	Insert	Select	Update
InDataTypeTemplate	X	X	X	X
InESDTMap	X	X	X	X
InExternalDataProviderInfo	X	X	X	X
InFileTypeTemplate	X	X	X	X
InGranuleServerInfo	X	X	X	X
InMediaCheckin	X	X	X	X
InMediaType	X	X	X	X



**Table 5-2. Object Permissions (2 of 2)**

<b>Name</b>	<b>Delete</b>	<b>Insert</b>	<b>Select</b>	<b>Update</b>
InNextAvailableID	X	X	X	X
InRequestFileInfo	X	X	X	X
InRequestProcessData	X	X	X	X
InRequestProcessHeader	X	X	X	X
InRequestSummaryData	X	X	X	X
InRequestSummaryHeader	X	X	X	X
InSourceMCF	X	X	X	X
InSystemParameters	X	X	X	X
InValDataGranuleState	X	X	X	X
InValGranuleServerUR	X	X	X	X
InValIngestType	X	X	X	X
InValNotifyType	X	X	X	X
InValParameterClass	X	X	X	X
InValRequestState	X	X	X	X

# 6. Scripts

---

## 6.1 Installation Scripts

Scripts used to support installation of the INGEST Subsystem database are listed in Table 6-1.

**Table 6-1. Installation Scripts**

Script File	Description
EcInDbBuild	Create a new initialized INGEST database
EcInDbPatch	Upgrade an existing INGEST database to the next valid database version level
EcInDbDump	Dump a specified INGEST database on demand
EcInDbLoad	Load a specified INGEST database on demand
EcDbDesc	List and detail the structure of all database objects in the specified ECS database
EcDbChecksum	Provide row count totals for each of the tables in a specific ECS database

## 6.2 De-Installation Scripts

Scripts used to support de-installation of the INGEST Subsystem database are listed in Table 6-2.

**Table 6-2. De-Installation Scripts**

Script File	Description
EcInDbDrop	Drop all objects in the specified INGEST database.

## 6.3 Backup and Recovery Scripts

Scripts developed to perform backup and recovery of the INGEST Subsystem database are listed in Table 6-3. These scripts should be configured to run automatically using the Unix cron facility. It is recommended that, transaction logs dumps (incremental dumps) are performed a minimum of 3 times each day. It is recommended that database dumps (full database dumps) are performed a minimum of once each day. Backup and recovery are M&O activities. At their discretion, DAACs may modify these backup/recovery scripts or utilize backup/recovery scripts developed by their local M&O staff.

**Table 6-3. Backup and Recovery Scripts**

<b>Script File</b>	<b>Description</b>
EcCoDbSyb_DumpDb	Dumps all databases for managed by the SQL server instance.
EcCoDbSyb_DumpTran	Dumps the transaction log for all databases managed by the SQL server.

## 6.4 Miscellaneous Scripts

Miscellaneous scripts applicable to the INGEST Subsystem database are listed in Table 6-4.

**Table 6-4. Miscellaneous Scripts and Input Data Files**

<b>Script</b>	<b>Description</b>
EcDdmMonitorServer	Monitors segment usage and user levels for a selected SQL server. Superseded by DbVision COTS.
EcDdmSegmentUse	Monitors segment usage. Used by EcDdmMonitorServer. Superseded by DbVision COTS.
EcDdmUserCounts	Monitors user access. Used by EcDdmMonitorServer. Superseded by DbVision COTS.
EcCoDbSyb_CkErrorLog	Checks the error log for error messages warranting DBO attention. Superseded by DbVision.
EcCoDbSyb_DbStat	Updates index statistics for each table in the selected database.
EcCoDbSyb_DboMail	Emails DBA error notification via e-mail. Used by EcCoDbSyb_DumpDb/Tran and EcCoDbSyb_CkErrorLog scripts.

# Appendix A. Entity Relationship Diagram

---

This page intentionally left blank.

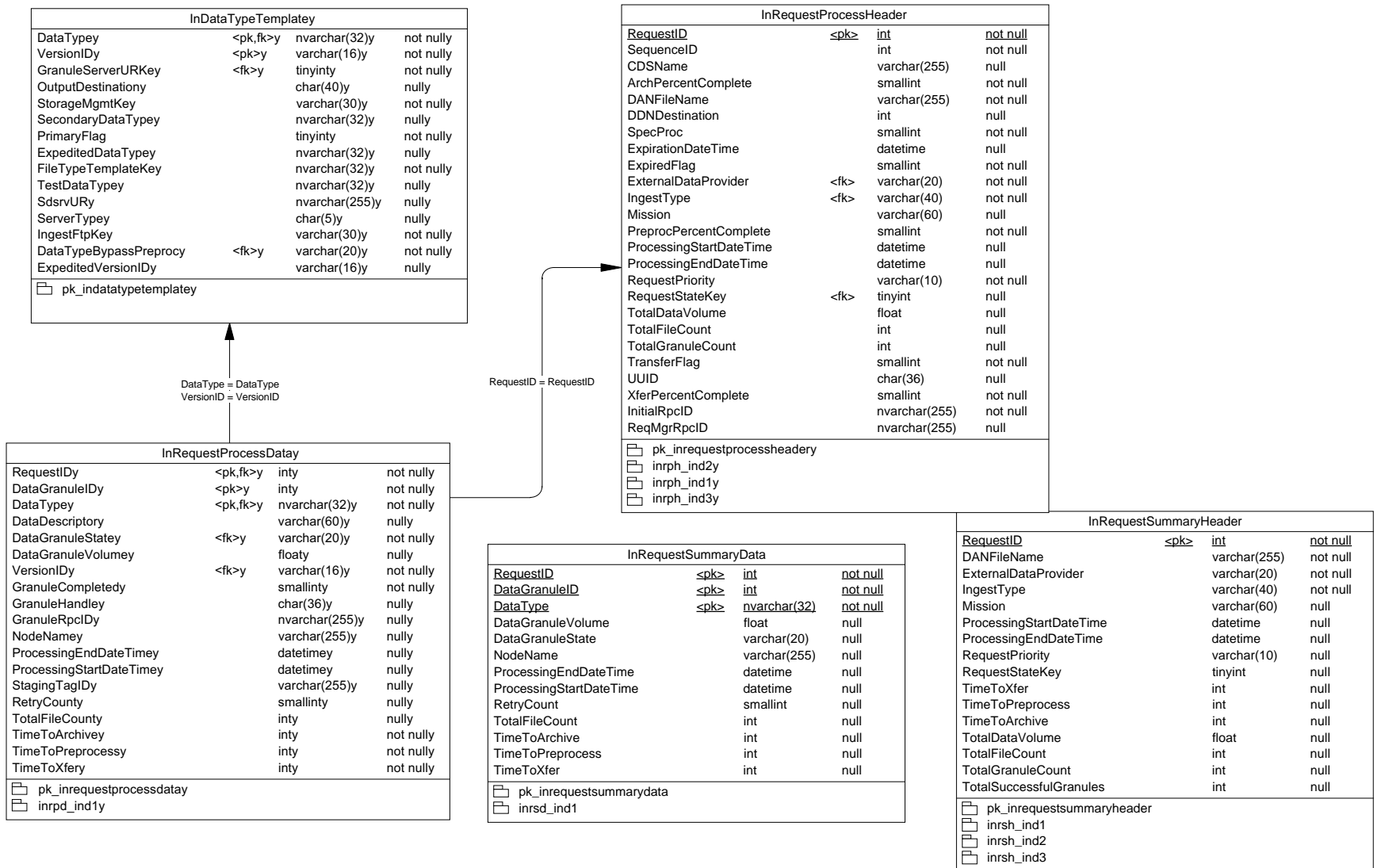


Figure A-1. INGEST Requests

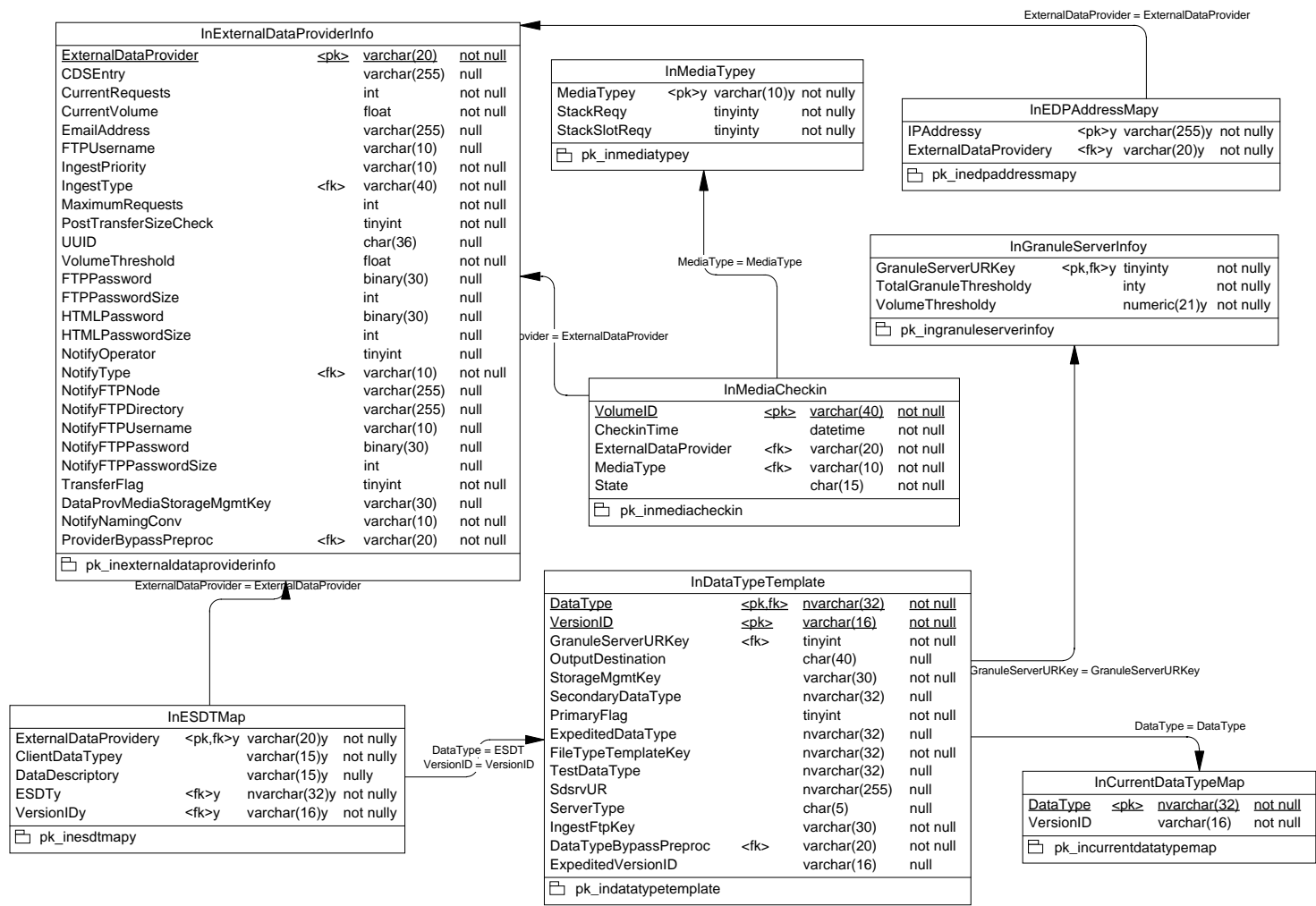
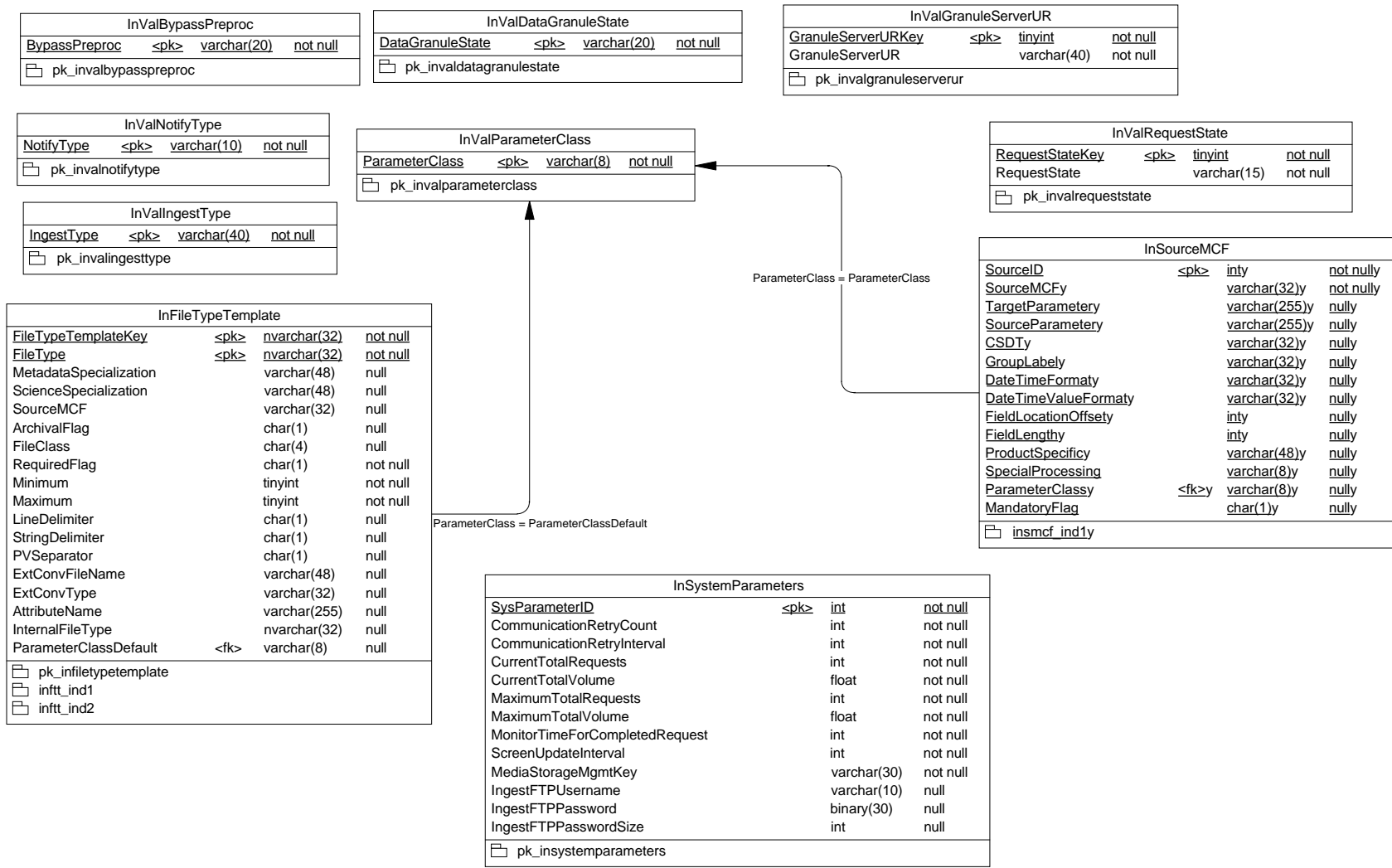


Figure A-2. Data Type Information



**Figure A-3. Validation/Configuration Data**



EcDbVersions			
<u>EcDbSchemaVersionID</u>	<pk>	<u>smallint</u>	<u>not null</u>
EcDbDropVersion		char(64)	not null
EcDbDropDescription		varchar(255)	null
EcDbCurrentVersionFlag		char(10)	not null
EcDbDatabaseName		varchar(255)	null
EcDbDropInstallDate		datetime	null
EcDbSybaseVersion		varchar(255)	null
EcDbSybaseServer		varchar(255)	null
EcDbComments		varchar(255)	null
EcDbUpdateProcess		varchar(255)	null

**Figure A-4. Database Versioning**

# Abbreviations and Acronyms

---

ADSRV	Advertising Service CSCI
ANSI	American National Standards Institute
CASE	Computer Aided Software Engineering
CD	contractual delivery 214-001
CDRL	contract data requirements list
CDS	cell directory service
CI	configuration item
COTS	commercial off-the-shelf (hardware or software)
CSCI	computer software configuration item
CSDT	Computer Science Data Type
CSMS	Communications and Systems Management Segment (ECS)
CSS	Communications Subsystem
DAAC	Distributed Active Archive Center
DBMS	Database Management System
DDICT	Data Dictionary CSCI
DDIST	Data Distribution Services CSCI
DDN	Data Delivery Notice
DID	data item description
DM	Data Management
DMS	Data Management Subsystem
DP	Data Provider
DPS	Data Processing Subsystem
DSS	Data Server Subsystem
ECS	EOSDIS Core System
EDC	EROS Data Center
EDHS	ECS Data Handling System
EDOS	EOS Data and Operations System
EOS	Earth Observing System

EOSDIS	Earth Observing System Data and Information System
EROS	Earth Resources Observation System
ESDIS	Earth Science Data and Information System (GSFC)
ESDT	Earth science data types
FK	Foreign Key
FTP	File Transfer Protocol
GSFC	Goddard Space Flight Center
GUI	graphic user interface
HTML	Hypertext Markup Language
HTTP	Hypertext Transport Protocol
HWCI	Hardware Configuration Item
ICD	interface control document
ID	identification
INGEST	Ingest Services CSCI
IOS	Interoperability Subsystem
IP	Internet Protocol
ISS	Internetworking Subsystem
IV&V	independent verification and validation
LaRC	Langley Research Center (DAAC)
MCF	Metadata Configuration File
MSFC	Marshall Space Flight Center
MSS	Management Support Subsystem
PDPS	Planning and Data Processing Subsystem
PK	Primary Key
PLANG	Production Planning CSCI
PLS	Planning Subsystem
RPC	Remote Procedure Call
STMGT	Storage Management Software CSCI
SUBSRV	Subscription Server
UR	Universal Reference
WWW	World-Wide Web