

DEPARTMENT OF HEALTH & HUMAN SERVICES
Centers for Medicare & Medicaid Services
7500 Security Boulevard, Mail Stop N2-14-26
Baltimore, Maryland 21244-1850



Central Data Administration

Logical Data Design

September 27, 2010

Prepared by:
Office of Information Services/Enterprise Data Group (OIS/EDG)
Division of Business Analysis & Operations (DBAO)
7500 Security Boulevard, Baltimore, Maryland 21244-1850

Revision\Change Description History Log

Revision	Date	Revision\Change Description	Pages Affected
Version 1.0	12/13/2005	Initial Draft	All
Version 2.0	11/13/2009	Added role descriptions and differentiated between local DA and central DA responsibilities.	All
Version 3.0	3/23/2010	Added new reference to Section 1.7.1	pp 25
Version 4.0	9/27/2010	Add an additional Logical Model review after the Design Phase	pp 7 - 30

Table of Contents

1. LOGICAL DATA DESIGN.....	3
1.1. INITIATE DATA DESIGN SERVICES.....	6
1.2. ASSESS DATA DESIGN NEED.....	8
1.3. START THE PROJECT LOGICAL DATA MODEL FROM THE EDM.....	11
1.4. EXTEND THE PROJECT LOGICAL DATA MODEL.....	13
1.5. ASSIGN INFORMATION SECURITY CATEGORIES.....	22
1.6. GENERATE THE PROJECT METADATA REPOSITORY.....	23
1.7. VALIDATE AND APPROVE THE PROJECT LOGICAL DATA MODEL.....	24
1.8. COMPLETE THE PROJECT LOGICAL DATA MODEL.....	26
1.9. UPDATE THE PROJECT LOGICAL DATA MODEL DURING THE DESIGN PHASE.....	27
1.10. LOGICAL DATA DESIGN CONSIDERATIONS FOR PHYSICAL DATA DESIGN.....	29
1.11. REVERSE ENGINEER A LOGICAL DATA MODEL FROM A PHYSICAL SOURCE.....	30

1. Logical Data Design

Introduction:

The focus of logical data design is development of a Logical Data Model. This model represents CMS data within the scope of a system development project and shows the specific entities, attributes, and relationships involved in a business function's view of data.

NOTE: There are references within this section that refer the reader to the Operating Procedures and Guidelines section. Please download the Operating Procedures and Guidelines section to view these references.

The [Logical Data Design process diagram](#) depicts the milestones, control points, and deliverables as they occur during the following steps:

- [Initiate Data Design Services](#)
- [Assess Data Design Needs](#)
- [Start the Project Logical Data Model from the EDM](#)
- [Extend the Project Logical Data Model](#)
- [Assign Data Information Security Categories](#)
- [Generate the Project Metadata Repository](#)
- [Validate and Approve the Project Logical Data Model](#)
- [Complete the Project Logical Data Model](#)

Additional activities related to logical data design are:

- [Logical Data Design considerations for Physical Data Design](#)
- [Reverse Engineering a Logical Data Model from a Physical Source](#)

Logical data design tasks are performed in the *Requirements Analysis Phase* and the *Design Phase* of the *CMS System Development Life Cycle (SDLC)* as indicated in the *CMS Integrated IT Investment & System Life Cycle Framework*, which is available on the CMS Intranet. The same logical data design tasks are also required in alternative development methods.

Key Deliverables:

The *Logical Data Design* process creates the following deliverables:

- Data Design Estimate,
- Conceptual Data Model,
- System Context Diagram,
- Project Metadata Repository,
- Data Issue Documents (where applicable),
- Logical Data Model,
- Data Architecture for Repository Update, and
- Proposed New EDM entries

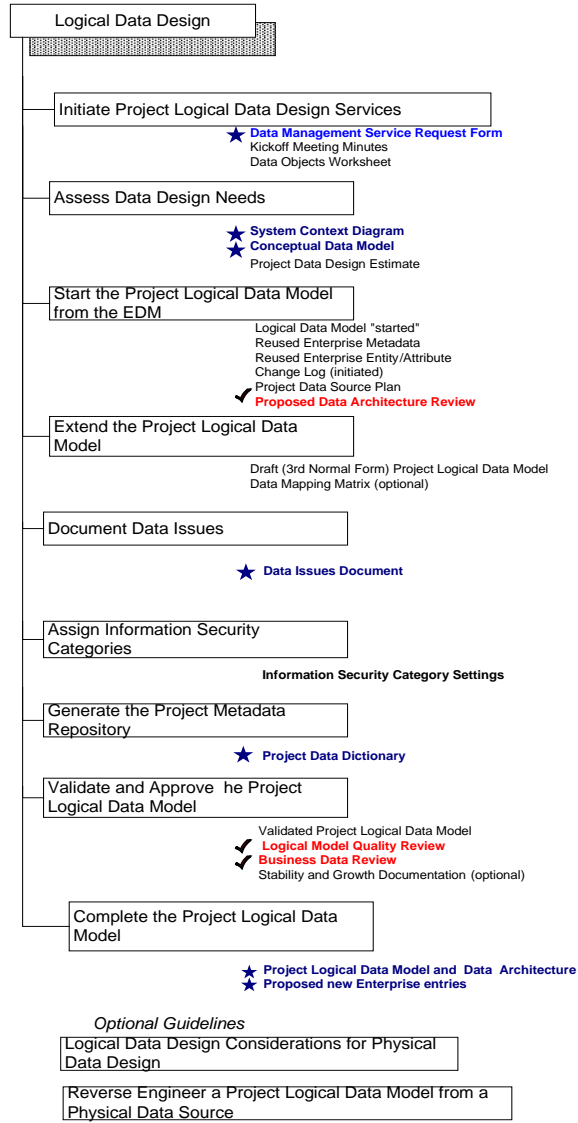
Roles:

The following roles are actors in the processes defined below:

- Business Owner/Partner – The CMS Business Owner of the project.
- CDA Team Lead – The Central Data Administration (CDA) Team Lead is a member of the CDA Team and is primarily responsible for assigning Central DAs to projects.
- Central DA – The CDA Data Analyst (DA) is a member of the CDA Team and is responsible for creating the Conceptual Model, ensuring adherence to CDA data modeling and naming standards, facilitating and verifying appropriate reuse of CMS data assets, etc.
- Database Administrator - The Database Administrator (DBA) is part of the application project team and is responsible for physical database design.
- Data Services Manager – The CMS CDA Team Lead or Government Team Lead (GTL).
- Local DA – The Local Data Analyst (DA) works for the application project team and is responsible for the project Logical Data Model (LDM).
- Project Manager – The Project Manager works for the Business Owner and is responsible for managing the project.

Exhibit 1. Logical Data Design process diagram

Logical Data Design Process



★ Controlled Artifact / Deliverable ✓

1.1. Initiate Data Design Services

Introduction:

This process step supports the gathering of baseline project information so that the appropriate level of data design service resources is assigned. This process occurs during the Initiation Phase of the Investment & System Lifecycle (ILC).

The following processes depict the participant roles, milestones, control points, and deliverables that occur during data service request activities:

- [Assign Data Design Resources](#)
 - [DM G-001 Guideline for Assigning Data Analysts](#)
- [Conduct the Logical Data Design Kick-Off Meeting](#)
 - [DM G-002 Operational Guideline for Conducting the Logical Data Design Kick-Off Meeting](#)

Deliverable(s):

- Kickoff Meeting Minutes
- Data Objects Worksheet (if business requirements are not provided)
- Data Service Request Form

1.1.1 Assign Data Design Resources

- | | |
|------------------------------|--|
| <i>Data Services Manager</i> | A. Receive a completed <i>CDA Services Request Form</i> , indicating a need for logical data design services. |
| <i>CDA Team Lead</i> | B. Assign the appropriate initial <i>Central DA</i> resource according to guidelines in DM G-001 Guideline for Assigning Data Analysts |

1.1.2 Conduct the Logical Data Design Kick-Off Meeting

- | | |
|------------------------------|--|
| <i>Central DA</i> | C. Briefly research the project goals and object classes; and locate any existing data design artifacts e.g. previous versions of the Project Logical Data Model and relevant EDM definitions. The objective is to be somewhat familiar with the project's business concern prior to the kick-off meeting. |
| <i>Data Services Manager</i> | D. Arrange and conduct a kick-off meeting in accordance with DM G-002 Guideline for Conducting the Logical Data Design Kickoff Meeting . |

Logical Data Design

*Project Manager,
System Developer Team,
Central DA,
Local DA,
Data Services Manager
Business Owner/Partner,
(optional invitee) Database
Administrator*

E. Participate in the data design kick-off meeting.

*Project Manager,
Business Owner/Partner*

F. Provide the project's *business requirements* to the Local DA. If formal *business requirements* are not available, provide the Local DAs with a spreadsheet that identifies and describes the entities that are of interest to the project.

1.2. Assess Data Design Need

Introduction:

The purpose of this process step is to analyze the project's data design requirements; and to estimate the data service resources needed to do the work. This process occurs during the Planning Phase of the Investment & System Lifecycle (ILC).

The objectives of the analysis activities are to:

- 1.) Understand enough about the project in order to communicate the project's business purpose and anticipated benefits; and
- 2.) Propose and gain approval for the use of existing data sources, where appropriate.

The objectives of the estimation activities are to:

- 1.) Make a preliminary assessment of the level of data design service needed;
- 2.) Obtain the *Project Sponsor's* approval of the preliminary assessment of data requirements, and approval of data services work estimate; and
- 3.) Establish the coordination of data design services with business process analysis and the respective roles of data analysts and business owners.

The following processes depict the participant roles, milestones, control points, and deliverables that occur during an assessment of data needs:

- [Create the System Context Diagram](#)
 - [DM OP-002 Operating Procedure for Identifying System Interfaces](#)
- [Create the Conceptual Data Model](#)
 - [DM OP-003 Operating Procedure for Developing the Conceptual Data Model](#)
- [Size and Estimate Data Analysis and Design](#)
 - [DM OP-004 Operating Procedure for Estimating Data Management Service Needs](#)
- [Authorize Data Management Services](#)
 - [DM G-003 Guideline for Monitoring Schedule and Cost of Data Services](#)

Deliverable(s):

- System Context Diagram
- Conceptual Data Model
- Project Data Design Estimate

1.2.1 Create the System Context Diagram

- Central DA*
- A. Identify existing or proposed *system interfaces* to the target project system in order to analyze the inputs, outputs, controls, and mechanisms involved in the business function. Then, document the prospective high level data flow between system interfaces using a *System Context Diagram*. See [DM OP-002 Guideline for Identifying System Interfaces](#).

1.2.2 Create the Conceptual Data Model

The purpose of creating the Conceptual Data Model is to understand the project's data entity needs and data boundaries.

- Central DA*
- A. Identify the principal entities and relationships needed to satisfy the project's business requirements and document them in a *Conceptual Data Model*. These entities will be based on the EDM, information provided in the project's business requirement documentation or obtained through interacting with a Business Analyst if a *Requirements* or a *System Design Document* are not available.

See [DM OP-003 Operating Procedure for Developing the Conceptual Data Model](#).

- Central DA*
- B. Save a dated copy of the Conceptual Data Model in case it is needed for later reference in the course of project activities. Make a copy available to the project Local DA.

1.2.3 Estimate Data Design Needs

- Central DA*
- A. Based on the Conceptual Data Model, perform a brief inventory of existing data stores to determine which entities are available from the EDM or alternative physical data sources.
- Local DA*
- B. Estimate the time and effort necessary to complete the project's logical data modeling work, i.e. to create the remaining data artifacts needed to satisfy project needs.

Document the results as the *Project Data Design Estimate*. See [DM OP-004 Operating Procedure for Estimating Data Management Service Needs](#).

1.2.4 Develop Project Data Source Plan

- Central DA*
- A. Prepare a project *Data Source Plan* that identifies existing or new data sources to satisfy the project's business requirements, and indicates limitations and constraints of proposed sources. See [DM OP-007 Operating Procedure for Reuse of Enterprise Data Resources](#).

1.2.5 Review the Project Data Architecture

- Local DA* **B.** Arrange a *Data Architecture Review* with the appropriate CMS Data Architect and respective Data Steward(s). Arrange and conduct the review meeting in accordance with [DM G-007 Guideline for Conducting the Data Architecture Review](#).

Major IT projects require a formal review of the proposed data architecture. It is important to schedule a formal review of the project data architecture as early in the modeling process as possible.

- Data Steward,
Business Owner/Partner,
Data Architect,
Local DA
Central DA* **C.** Participate in the *Data Architecture Review* for the goal of negotiation of appropriate project data sources and confirmation of requirements for new entities.

- Data Architect* **D.** Consider the agency's data management objectives, and IT standards, and data dissemination regulations, when approving the project's overall Data Architecture and *Data Source Plan*.

1.3. Start the Project Logical Data Model from the EDM

Introduction:

The purpose of this process step is to describe the activities needed to “start” the Project Logical Data Model. The activities describe steps to enter model properties, locate reusable entities and attributes, understand the impact of sharing existing data resources, and obtain approval on the planned project data architecture. This process occurs at the beginning of the Requirements Analysis Phase of the Investment & System Lifecycle (ILC).

The following processes depict the participant roles, milestones, control points, and deliverables that occur when starting a logical data model:

- [Set up the Project Logical Data Model](#)
 - [DM OP-005 Operating Procedure for Developing the Logical Data Model](#)
- [Reuse Enterprise Entities, Relationships, and Attributes](#)
 - [DM OP-006 Operating Procedure for Reuse of Enterprise Entities, Relationships, and Attributes](#)
- [Develop Project Data Source Plan](#)
 - [DM OP-007 Operating Procedure for Reuse of Enterprise Data Resources](#)
- [Review the Project Data Architecture](#)
 - [DM G-007 Guideline for Conducting the Data Architecture Review](#)

Deliverables:

- Logical Data Model “started”
- Reused Enterprise Entity/Attribute Change Log (initiated)
- Project Data Source Plan

1.3.1 Set up the Project Logical Data Model

Central DA

- A. Provide the Local DA with a copy of the current, approved system Logical Data Model (LDM), if one exists

Logical Data Design

Local DA

- B. All projects that require new or changed data structures must create or update a *Logical Data Model* to represent the project data needs. See [DM OP-005 Operating Procedure for Developing the Logical Data Model](#).
- a. If an approved LDM exists for the system, copy it to create the new model
 - b. When a logical data model for an existing application system is not available or is outdated, reverse engineer a new model from the project system's database.
 - c. If no prior version of the logical data model exists, or if a new model can't be reverse engineered, a new model must be created. Use a standard data model template, which contains the CMS preferred settings and standard reports. Access to the template and instructions for its use is described in the [Data Model Tool Use Procedure](#).

Local DA

- C. Once you have your project logical data model at hand, enter (or confirm the accuracy of previously entered) *model properties* per DM OP-031 Operating Procedure for Capturing the Standard Logical Data Model Metadata.

1.3.2 Reuse Enterprise Entities, Relationships, and Attributes

Central DA

- A. Compare the *needed entities* in the CDM with the *existing entities* in the EDM to identify candidate reusable entities. Select appropriate candidate entities, attributes, and relationship types from the EDM, using the copy facilities in the standard data modeling tool. See [DM OP-006 Operating Procedure for Reuse of Enterprise Entities, Relationships, and Attributes](#).

1.4. Extend the Project Logical Data Model

Introduction:

The purpose of this process step is to add new entities and attributes that are not available from the EDM. This process occurs during the Requirements Analysis Phase of the Investment & System Lifecycle (ILC).

The processes listed below depict the participant roles, milestones, control points, and deliverables that occur during logical data modeling activities. Refer to the [Data Model Tool Use Procedure document](#) for tool specific details.

- [Create New Entities](#)
 - [DM OP-008 Operating Procedure for Defining Data Entities](#)
 - [DM OP-009 Operating Procedure for Naming Data Entities](#)
 - [DM OP-031 Operating Procedure for Capturing the Standard Logical Data Model Metadata](#)
- [Create New Attributes](#)
 - [DM OP-010 Operating Procedure for Defining Data Attributes](#)
 - [DM OP-011 Operating Procedure for Analyzing Data Attribute Types](#)
 - [DM G-004 Guideline for Designating Representation Term and Data Type](#)
 - [DM G-006 Standard for Assigning Date Formats](#)
 - [DM OP-012 Operating Procedure for Naming Data Attributes](#)
 - [DM OP-031 Operating Procedure for Capturing the Standard Logical Data Model Metadata](#)
- [Model Derived Data](#)
 - [DM OP-013 Operating Procedure for Modeling Derived Data](#)
- [Identify and Model Supertypes and Subtypes](#)
 - [DM G-019 Guideline for Modeling Supertypes and Subtypes](#)
- [Define Relationships](#)
 - [DM OP-015 Operating Procedure for Defining Relationships, Cardinality, and Optionality](#)
- [Determine Primary Identifiers](#)
 - [DM OP-016 Operating Procedure for Assigning a Primary Identifier](#)
- [Normalize the Project Logical Data Model](#)
 - [DM OP-017 Operating Procedure for Normalizing the Project Logical Data Model](#)
- [Define Domain Value Rules](#)
 - [DM OP-018 Operating Procedure for Assigning Domain Value Rules in the Project Logical Data Model](#)
- [Check Completeness of Project Logical Data Model](#)
 - [DM G-008 Guideline for Mapping Data Attribute Sources](#)
 - [DM OP-019 Operating Procedure for Checking Completeness of a Project Logical Data Model](#)

Deliverables:

- Draft (3rd Normal Form) Project Logical Data Model
- Data Mapping Matrix (optional)

1.4.1 Create New Entities

- Local DA* A. Look in the EDM before creating a new entity. If the needed entity is in the EDM, copy it and its attributes into your logical data model.
Entities may also be copied from an alternate approved source. This might be an option when the EDM for the business area is still being developed and a legacy data source is readily available. Such sources might have quality and architectural problems. Therefore, the project's use of alternative sources must be approved by the appropriate Data Architect and Data Steward.
- Local DA* B. Define new entities following policies for creation of well-formed definitions in [DM OP-008 Operating Procedure for Defining Data Entities](#).
- Local DA* C. Before naming new entities, it would be helpful to have a list of approved business *Standard Terms* on hand. This information is available in the Standard Terms and Abbreviations List. The *Standard and Abbreviations Terms List* is available from the Standards Terms page (accessible from the Data Administration home web page). If a needed term is not on the list, follow the procedure outlined on the Standard Terms page.
- Local DA* D. Assign meaningful entity names in familiar business terms according to [DM OP-009 Operating Procedure for Naming Data Entities](#). The objective of naming standards is to foster a common reference of CMS data.
A shortened version of the naming procedure is available as a quick reference in [DM OP-009-QR Quick Reference for Naming Data Entities](#).
- Local DA* E. The Local DA is responsible for assigning the "first cut" physical data names. Assign the physical names in the counterpart physical data model, following the rules outlined in [Assign a Table or File Name](#)
- Local DA* F. Capture the required Entity Metadata as referenced in DM OP-31 Operating Procedure for Capturing the Standard Logical Data Model Metadata.

1.4.2 Create New Attributes

Local DA A. Identify candidate attributes not found in the EDM. These are the new attributes (facts) needed in one of the project entities.

Local DA B. Define the new attribute. The definition of a new attribute shall comply with the Operating Procedure described in [DM OP-010 Operating Procedure for Defining Data Attributes](#).

The above procedure is compliant with a prerequisite standard *ISO IEC 11179-4 Rules and guidelines for the formulation of data definitions*.

Local DA

- C. The other factors to consider when creating a new attribute require data analysis. The purpose of that analysis is to classify new attributes into one of the following categories.

<i>Attribute type</i>	<i>Definition</i>	<i>Example</i>	<i>Description</i>
Prime / Atomic	A basic business fact	Department	Basic information about a business
Derived	A value that can be formulated using values from other attributes.	Invoice Total	Computed from the sum of invoice lines.
Cohesive	Attributes that are usually processed together for business meaning	Employee First Name and Employee Last Name	Neither is very meaningful without the other.
Transaction / Interface	Interface Data	Activity Data	Business required data exchange
Physical Control Data	A system fact used internally to facilitate transactions	Record Count	Processing Data

See [DM OP-011 Operating Procedure for Analyzing Data Attributes Types](#) for methods that can improve how the attributes types are best modeled.

Note that **Physical control data**, such as “next record number”, are not included in the logical data model. For information on how to model such system data, see [Model System Control Data](#).

Local DA

- D. Determine the types of data values that the attribute will eventually represent. Then, identify the appropriate data type for each new attribute. See [DM G-004 Guidelines for Designating Representation Term and Data Type](#)

Local DA

- E. Assign each new attribute a business name. See [DM OP-012-QR Quick Reference for Naming Data Attributes](#).

- Local DA* F. Verify that the new attribute name is compliant using the full Operating Procedure for naming attributes [DM OP-012 Operating Procedure for Naming Data Attributes](#) or the quick reference [DM OP-012-QR Quick Reference for Naming Data Attributes](#).

The objective of naming standards is to foster a common reference of CMS data.

The above procedure is compliant with a prerequisite standard *ISO IEC 11179-5 Naming and identification principles for data elements*.

- Local DA* G. Capture the required Attribute Metadata as referenced in DM OP-31 Operating Procedure for Capturing the Standard Logical Data Model Metadata.

- Local DA* H. Attributes that represent dates must follow the rules outlined in [DM G-006 Standard for Assigning Date Formats](#).

- Local DA* I. The Local DA is responsible for assigning the “first cut” physical data names. Assign the physical names in the counterpart physical data model, following the rules outlined in [Assign a Column or Element Name](#). Consult with the *Project Database Administrator* to confirm the likely DBMS platform for implementation in order to stay in compliance with physical data name length limitations.

- Local DA* J. Consider long-term management for electronic records when adding new attributes to record types. Appropriate classification of data types will facilitate easier archival for those records with federal archival mandates.

1.4.3 Model Derived Data

Local DA

- A. A derived attribute represents a value that can be formulated from other factors.
Analyze the pros and cons of including each derived attribute in the logical data model. Follow the guidelines cited in [DM OP-013 Operating Procedure for Modeling Derived Data](#).

1.4.4 Identify and Model Supertypes and Subtypes

Local DA

- A. Some entities are related through their roles. For example, the *supertype* “Organization” could include the *subtypes* for “Organization for Profit” and “Organization as Non-Profit”.

Analyze any project entities that show similar roles relationships for common characteristics.

Local DA

- B. When common attributes are identified, create a supertype.

Local DA

- C. Follow the guidelines on the identification and creation of supertypes and subtypes in [DM G-019 Guideline for Modeling Supertypes and Subtypes](#).

Local DA

- D. Create a name for the supertype that describes all member subtypes.

Local DA

- E. Place the attributes common to all subtypes in the supertype.

Local DA

- F. Review similar entities with similar attributes for common characteristics.

1.4.5 Define Relationships

- Local DA* A. Review the functional business rules controlling entity relationships.
- Local DA* B. Ensure that all relationship standards are met to model a valid relationship. See [DM OP-015 Operating Procedure for Defining Relationships, Cardinality, and Optionality](#).
- Local DA* C. The IDEF1x standard dictates that when a single attribute is migrated to an entity to express two separate relationships, the attribute must be unified in the child entity. The result is a single attribute migrated through two relationships. To avoid unification, use *rolenames* to distinguish the migrated attributes. This will clarify the distinct relationships in the model. Include the appropriate definitions, cardinalities, and optionality for each *rolename* relationship.
- Local DA* D. Label entity relationships in the Project Logical Data Model with verb phrases and relationship names.
- Local DA* E. Determine and document the cardinality of each relationship, based on business rules in the functional data requirements.

1.4.6 Determine Primary Identifiers

- Local DA* A. The purpose of the *Primary Identifier* is to distinguish one instance of an entity from another instance. Determine the Primary Identifier for each new entity following the criteria in [DM OP-016 Operating Procedure for Assigning a Primary Identifier](#).

1.4.7 Normalize the Project Logical Data Model

- Local DA* A. Normalize the LDM, in order to represent data in a stable format. See the normalization criteria described in [DM OP-017 Operating Procedure for Normalizing the Project Logical Data Model](#).
- Local DA* B. Organize an entity into *first normal form* by moving repeating or multi-valued attributes to a separate entity. Assign a *primary identifier* to each new entity.
- Local DA* C. Organize an entity into *second normal form* by removing attributes that are not dependent on the primary identifier to a separate entity.
- Local DA* D. Organize an entity into *third normal form* by removing transitive relationships. That is, an attribute should be dependent only on the primary identifier and not on another non-identifier attribute. Then ensure that data exists in only one place.
- Local DA* E. Consider further normalization to fourth normal form and fifth normal form when it is beneficial to do so.

1.4.8 Define Domain Value Rules

- Local DA* A. Document the attribute business valid value rules where the business operation requires adherence to a discrete set of permissible data values. Record the business data value rules in the logical model by following [DM OP-018 Operating Procedure for Assigning Domain Value Rules to Attributes](#).

For additional instructions on documenting valid domain value rules in the data modeling tool, see [Data Model Tool Use Procedure](#).
- Local DA* B. Related information describing the use of *reference tables* (also called *look up tables* or *constants*) is available in [DM G-017 Guidelines for Reference Tables](#).

- Local DA* C. The management of domain values through database management system features (e.g. *referential integrity* or *check constraints*) is ultimately decided in the *Physical Data Modeling* by the *Database Administrator*. See [DM G-018 Guidelines for Managing Data Values through Physical Constraints](#).

1.4.9 Check Completeness of the Project Logical Data Model

- Local DA* A. Verify that the logical model meets business needs by mapping the data sources for each project data attribute. See [DM G-008 Guideline for Mapping Data Requirement Sources](#)
- Local DA* B. Verify completeness of the logical model using the criteria in [DM OP-019 Operating Procedure for Checking Completeness of a Project Logical Data Model](#).

1.5. Assign Information Security Categories

Introduction:

This process step provides direction for complying with Title III of the E-Government Act, titled the Federal Information Security Management Act (*FISMA*). *FISMA* requires federal organizations to develop, document, and implement an organization-wide program to provide information security for the information systems that support its operation and assets. This process occurs during the Requirements Analysis Phase of the Investment & System Lifecycle (ILC).

The following processes depict the participant roles, milestones, control points, and deliverables that occur during assignment of information security categories:

- [Assign Information Security Categories](#)
 - [DM OP-021 Operating Procedure for Assigning Information Security Categories](#)

Deliverable(s):

- Information Security Category Settings

1.5.1 Assign Information Security Categories

Local DA A. Analyze the security categories for the Project Logical Model entities and attributes using standards and guidelines in [DM OP-021 Operating Procedure for Assigning Information Security Categories](#).

This procedure supports the federal requirements outlined in *FIPS Publication 199 – Standards for Security Categorization of Federal Information and Information Systems*.

Local DA B. Document the security and levels of impact (low, moderate, and high) in the Project Logical Data Model following the format in [Data Model Tool Use Procedure](#).

1.6. Generate the Project Metadata Repository

Introduction:

The Project Logical Data Model is used to generate the Project Metadata Repository, which reports information about each entity and attribute. This process occurs during the Requirements Analysis Phase of the Investment & System Lifecycle (ILC).

The following processes depict the participant roles, milestones, control points, and deliverables that occur during preparation of a Metadata Repository:

- [Generate the Project Metadata Repository](#)
 - [DM OP-022 Operating Procedure for Generating the Project Metadata Repository](#)

Deliverable(s):

- Project Metadata Repository

1.6.1 Generate the Project Metadata Repository

<i>Local DA</i>	A. Draft a project Metadata Repository following DM OP-022 Operating Procedure for Generating the Project Metadata Repository .
<i>Business Owner / Partner, Data Steward Local DA</i>	B. Validate the Project Metadata Repository. C. Submit the Project Metadata Repository to the project manager prior to or as a part of the quality review in Validate and Approve the Project Logical Data Model .

1.7. Validate and Approve the Project Logical Data Model

Introduction:

After the Project Logical Data Model (PLDM) is drafted, it should be reviewed for adherence to standards and accuracy in representing project data needs. This process occurs towards the end the Requirements Analysis Phase of the Investment & System Lifecycle (ILC).

The validation reviews are opportunities to make corrections and additions. Also, the reviews provide a convenient time to collect information about the amount and growth of data to be collected along with information about anticipated business events that might significantly change the volume of data over the long term.

The following processes depict the participant roles, milestones, control points, and deliverables that occur during review and approval of a project's logical data model:

- [Perform a Quality Review of the Project Logical Data Model](#)
 - [DM OP-025 Operating Procedure for Performing a Quality Review of the Project Logical Data Model](#)
- [Document Data Stability and Growth Information](#)
 - [DM OP-025 Operating Procedure for Collecting Data Stability and Growth Information](#)

Deliverable(s):

- Validated Project Logical Data Model
- Logical Model Quality Review
- Stability and Growth Documentation (optional)

1.7.1 Perform a Quality Review of the Project Logical Data Model

- | | |
|-------------------|---|
| <i>Local DA</i> | A. Arrange a quality review of the Project Logical Data Model to check for standards compliance, completeness, and consistency with CMS Data Management policies/standards, and the Enterprise Data Architecture. |
| <i>Central DA</i> | B. Review the project Logical Data Model following guidelines in New Model Review Procedure . |
| <i>Local DA</i> | C. Research and resolve any issues uncovered in the quality review. |

1.7.2 Document Data Stability and Growth Information

- Local DA* A. Collect information about the anticipated volume and potential growth of all *entity* occurrences. See [DM OP-025 Operating Procedure for Collecting Data Stability and Growth Information](#).

- Local DA* B. Document imminent or expected near-term changes in information for the business enterprise that has implications for the Project Logical Data Model entities.

- Local DA* C. Document any anticipated law or regulatory changes that affects the business enterprise area and suggests volume changes among the entities being collected.

1.8. Complete the Project Logical Data Model

Introduction:

After the development work on the Project Logical Data Model ends, it must be stored to facilitate ongoing analysis of application data and future changes. This process towards the end of the Requirements Analysis Phase of the Investment & System Lifecycle (ILC).

This process step describes the change control activities that catalogs and stores the Project Logical Data Model in the appropriate model library.

Additional activities in this process decide whether any new entities and attributes that were added for the project have the potential for serving other business applications and therefore whether they should be proposed as new Enterprise Data objects.

The following processes depict the participant roles, milestones, control points, and deliverables that occur during completion of a project logical data model:

- [Complete the Project Logical Data Model](#)
 - [DM OP-026 Operating Procedure for Completing the Data Model](#)

Deliverable(s):

- Published Project Logical Data Model

1.8.1 Complete the Project Logical Data Model

Central DA A. Accept the new Project Logical Data Model and publish the model according to instructions in [Production Change Control for Model Management](#).

Note: All models shall be appropriately stored when work is completed (or halted in an incomplete or unapproved status).

Central DA B. Review new data elements in the *Project Metadata Repository* and identify new entities and attributes for inclusion in the EDM. Then, draft a *Proposed New Enterprise Data Report* (when no new EDM appropriate data objects are identified, the report will show that “no new EDM entries are proposed from the respective project”).

Local DA C. Provide any input that might be important to subsequent physical data design. Related activities are described in [Logical Data Design considerations for Physical Data Design](#).

1.9. Update the Project Logical Data Model During the Design Phase

Introduction:

After the Project Logical Data Model (PLDM) is approved at the end of the Requirements Analysis phase, it may need changes during the Design phase to reflect changes in requirements, changes resulting from physical considerations, etc. This process is basically a repetition of several of the steps that occurred in the Requirements Analysis Phase:

- [Extend/Update the Project Logical Data Model](#)
- [Assign Data Information Security Categories](#)
- [Generate the Project Metadata Repository](#)
- [Validate and Approve the Project Logical Data Model](#)
- [Complete the Project Logical Data Model](#)

Deliverable(s):

- Validated Project Logical Data Model
- Logical Model Quality Review
- Stability and Growth Documentation (optional)

1.9.1 Extend/Update the Project Logical Data Model

Local DA A. Update the Project Logical Data Model to reflect new requirements, physical considerations, etc. This may include extension of the PLDM as well as changes to/deletion of previously approved model objects. For additions and changes, use the procedures and standards as described in process 1.4: Extend the Project Logical Data Model.

Local DA B. Document changes as they are made. If the project has a Change Management process that results in formal Change Requests (CRs) or Data Requests (DRs), use it. Otherwise, use the procedure defined in DM OP-033 Operating Procedure for Recording Data Model Changes

1.9.2 Assign Data Information Security Categories

Local DA A. Follow the process defined in section 1.5: Assign Data Information Security Categories.

1.9.3 Generate the Project Metadata Repository

Local DA A. Follow the process defined in section 1.6: Generate the Project Metadata Repository

1.9.4 Validate and Approve the Project Logical Data Model

Central DA A. Review the model according to the process defined in section 1.7: Validate and Approve the Project Logical Data Model. It is only necessary to review the changes that occurred since the initial review, as identified in section 1.9.1B above.

1.9.5 Complete the Project Logical Data Model

Central DA A. Follow the process defined in section 1.8: Complete the Project Logical Data Model

1.10. Logical Data Design considerations for Physical Data Design

Introduction:

This process step guides the logical data modeler in providing comments or suggestions that might assist the physical data designer in the subsequent preparation of an efficient and effective physical data model. The objective is to record information about anticipated physical performance obstacles, and how they might be addressed by the physical data model.

The following processes depict the participant roles, milestones, control points, and deliverables that occur during preparation of a project logical data model for physical design:

- [Preparing the Project Logical Data Model for Physical Design](#)
 - [DM G-009 Guideline for Preparing the Project Logical Data Model for Physical Design](#)

Deliverable(s):

- Comments to be considered in the transformation of the Logical Data Model to a Physical Design.

1.10.1 Preparing the Project Logical Data Model for Physical Design

- | | |
|---|---|
| <i>Local DA</i> | A. To start the preparation for transforming the logical design to physical design, hypothetically assume the Project Logical Data Model is the physical database design, then: <ul style="list-style-type: none"> ▪ View entities as record types or rows, ▪ View attributes as fields or columns, ▪ View relationships as integrity rules. |
| <i>Local DA</i> | B. Using this “hypothetical” physical model, analyze whether the data will be accessible in a manner needed to satisfy each required business data retrieval scenario. For example, sketch a set of SQL queries that could retrieve the data in the manner needed. |
| <i>Local DA</i> | C. Consider the factors outlined in DM G-009 Guideline for Preparing the Project Logical Data Model for Physical Design . |
| <i>Local DA</i> | D. Make notes about factors that might be important to the subsequent physical data design (do not modify the approved project logical data model). |
| <i>Local DA,
Project Database
Administrator</i> | E. Collaborate to: hand-off the approved logical data model to the Project Database Administrator, who will continue the physical data modeling process; discuss any issues that are relevant to physical modeling and data definition. |

1.11. Reverse Engineer a Logical Data Model from a Physical Source

Introduction:

Reverse engineering is the automatic capture of data structure(s) to a physical data model using data modeling tool facilities. The captured physical data model can then serve as the basis for modeling a logical data model.

The reverse engineering of a logical data model from physical sources is an expeditious method to further one or more of the following goals:

- To convey and document existing physical data resources in a logical format that is suitable for business user analysis and planning,
- To assess which existing physical sources of organization data might serve additional business information needs,
- To jump-start a logical data model based on existing un-modeled physical data resources.

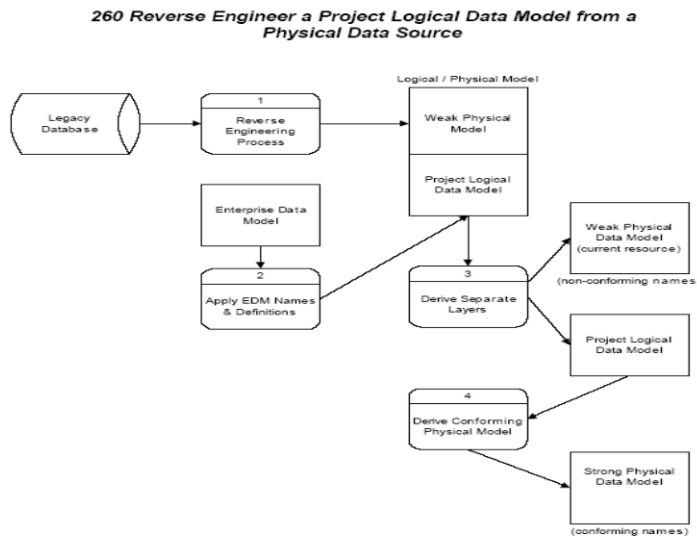
The [Reverse Engineer a Logical Data Model from a Physical Data Source diagram](#) depicts the flow of activities and their outputs as they occur during the following process:

- [Reverse Engineering a Logical Data Model](#)

Deliverable(s):

- A physical data model of existing data resources within a business area.
- Logical data model components based on upward abstraction of all or portions of the above physical data model.

Exhibit 2. Reverse Engineer a Logical Data Model from a Physical Data Source



1.11.1 Reverse Engineering a Logical Data Model

Local DA

A. Contact the database administrator who has assigned responsibility for the existing physical database and request the Data Definition Language (DDL) script from which the data modeling tool can produce a re-engineered physical data model. See the HELP facility of the data modeling tool for more information about how to reverse engineer a physical model.

Local DA

B. Normalize the physical data model and compare it to the Enterprise Data Model for related abstract business entities and attributes. For application projects, apply full business names to the conforming logical pieces
(see [Start the Project Logical Data Model](#)) then add the non-conforming logical pieces
(see [Extend the Project Logical Data Model](#))