Department of Health and Human Services
Centers for Medicare and Medicaid Services
7500 Security Boulevard, Mail Stop N2-14-26
Baltimore, Maryland 21244-1850

Data Administration & Engineering Services

Data Management Operating Procedures and Guidelines

November 26, 2018

Revision/Change Description History Log

Revision	Date	Revision/Change Description	Pages Affected
Original		Baseline	N/A
Version 3.1	08/31/2009	Add cover page, history log, and table contents. Insert page numbers.	All
Version 4.0	11/30/2009	Add DM OP-031 and DM OP-032. Revise cover page. Remove DM G-021 and DM G-022; renumber DM G-023	pp. 40-47, 70-88
Version 5.0	03/08/2010	Update DM OP-010 to include new operating procedure on foreign key definitions. Update DM OP-031 to address an additional UDP to capture National/International Standard Element Names. Replace DM OP-023 with New Model Review Procedure.	pp. 16, 29, 41-42
Version 5.1	04/27/2010	Update DM OP-008 to further clarify entity definitions. Update DM OP-012 to require representation terms to be consistent with definitions.	pp. 14, 19
Version 5.2	05/24/2010	Update DM G-010 and DM G-011 to differentiate between LDM/PDM vs. separate LDM and PDM.	pp. 55-57
Version 5.3	07/26/2010	Update DM OP-023 by adding URL for New Model Review Procedure.	p. 29
Version 5.4	09/27/2010	Remove DM OP-024. Add DM OP-033.	pp. 30, 45
Version 6.0	05/03/2011	Update DM OP-009 and DM OP-012 to reflect change in use of Object Class Terms.	pp. 15-16, 19-21
Version 6.1	10/19/2011	Update DM OP-015 to clarify entity relationship documentation requirements.	p. 21
Version 7.0	10/30/2012	Update DM OP-006, OP-034, and G-007 to capture new procedures for reusing and updating the Enterprise Logical Data Model (ELDM).	pp. 12, 46, 55

Revision	Date	Revision/Change Description	Pages Affected
Version 8.0	09/01/2013	Restructure the DM OP&G document so that each operating procedure consist of steps and applicable standards and guidelines. Merge all guidelines (G-xxx) with an existing operating procedure (OP-xxx) or include them in a new operating procedure. Add a Standard Metadata Matrix that defines required metadata in a data model by data model type under OP-031.	All
Version 8.1	12/01/2013	Add an example to DM OP-035.	p. 48
Version 9.0	05/06/2014	Add Table of Contents.	p. 4
Version 10.0	02/23/2015	Update standard #5 in DM OP-010, Operating Procedure for Defining Attributes. Add standard to DM OP-015, Operating Procedure for Defining Relationships, Cardinality, and Optionality.	pp. 17, 21
Version 10.1	08/05/2015	Update standard #2 in DM OP-008, Operating Procedure for Defining Entities.	pp. 15, 35
	09/16/2015	Update DM OP-029 to include dimensional models.	
Version 11.0	01/01/2016	Add standard #8 in DM OP-012. Update DM OP-043 guideline #6.	pp. 21, 60
Version 12.0	01/03/2018	Add new attribute UDP.	p. 42
Version 13.0	11/26/2018	Reformat document. Update name of document owner (DAES). Correct phrasing for clarity. Update DM OP-005 to reflect discontinuation of NSM file support.	All

Table of Contents

Data Management Operating Procedures and Guidelines

The full text of operating procedures, guidelines, and standards, including their underlying *rationale* and *level of enforcement* are enclosed for reference.

Numbering schemes follow this format with *xxx* as a sequence number (e.g. 001):

DM OP *xxx* Operating Procedure

An Operating Procedure specifies a sequence of tasks that are performed to achieve a specific goal. Standards and guidelines that are applicable to the tasks are described within the operating procedure. Standards' compliance is required guidelines are optional.

The Operating Procedures are referred to in the following Data Administration and Engineering Services documents:

1. Logical Data Design
2. Physical Data Design
3. Data Model Tool Use Procedure
4. New Model Review Procedure
5. Dimensional Model Review Procedure
6. Enterprise Data Planning
7. Model Management
8. Geographic Template Read-Me Document

Data Management Operating Procedures

DM OP-001 Operating Procedure for Requesting Data Management Services

Rationale

The Data Services Manager needs certain information to adequately plan and assign data management service resources in support of CMS projects. That information is collected through the Data Management Service Request Form, which is designed to capture it in a clear and organized format.

Operating Procedure

A request for data design services shall be initiated through submission of the Data Management Service Request Form. The form is in the CDA Forms Download file at http://www.cms.gov/Research-Statistics-Data-and-Systems/CMS-Information-Technology/DataAdmin/index.html.

DM OP-002 Operating Procedure for Identifying System Interfaces

Rationale
The System Context Diagram concisely illustrates the system boundary in relation to its environment and documents the actual or planned data flows into and out of the system.

Operating Procedure
Business application interfaces shall be identified using a System Context Diagram (Figure 1 below).

Figure 1: System Context Diagram

Other System 1		Person
Input data flow 1	System	Output data flow 2
Input data flow 2		Output data flow 1
Organization 1		Organization 2
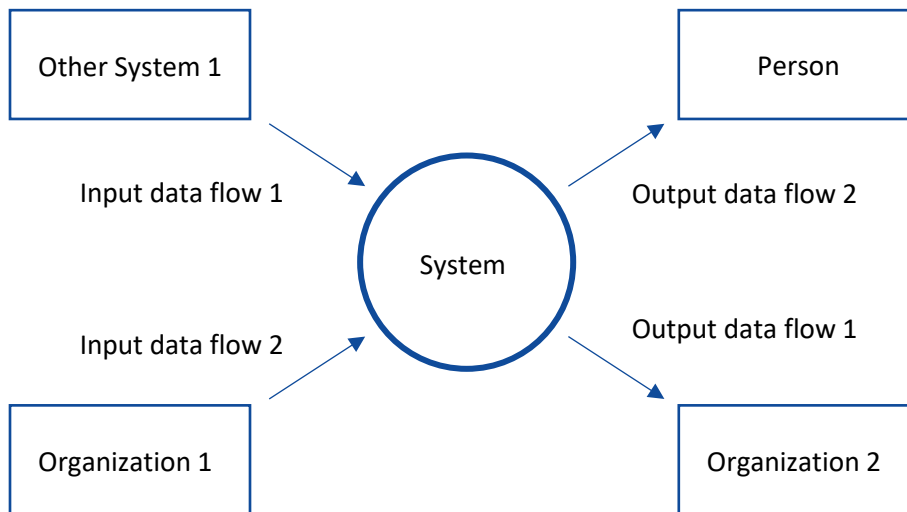
DM OP-003 Operating Procedure for Developing the Conceptual Data Model

Rationale

The Conceptual Data Model diagram illustrates the major entities about which the business enterprise needs information. This diagram will assist in the identification of existing data sources and help in determining the need for new entities.

Operating Procedure

1. Business applications that create, update, or replicate data shall have high-level data requirements documented in a Conceptual Data Model.
2. Conceptual Data Models shall represent business entity relationships that are within the scope of the target business function(s) in a manner depicted by Figure 2.
3. Major IT projects may require review by a Data Architect.

Figure 2: Example Conceptual Data Model

```
┌──────────────┐                              ┌──────────────┐
│              │                              │              │
│   Employer   │ ──── contracts with ──▶      │   Insurer    │
│              │                              │              │
└──────────────┘                              └──────────────┘
                                                      │
                                                    offers
                                                      │
                                                      ▼
┌──────────────┐                              ┌──────────────┐
│              │                              │              │
│   Employee   │ ──── subscribes to ──▶       │ Benefit Plan │
│              │                              │              │
└──────────────┘                              └──────────────┘

┌ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ┐
  Business rule: An employer contracts with one or more insurance
  companies to offer health care benefits to any employee who elects to
  subscribe.
└ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ┘
```
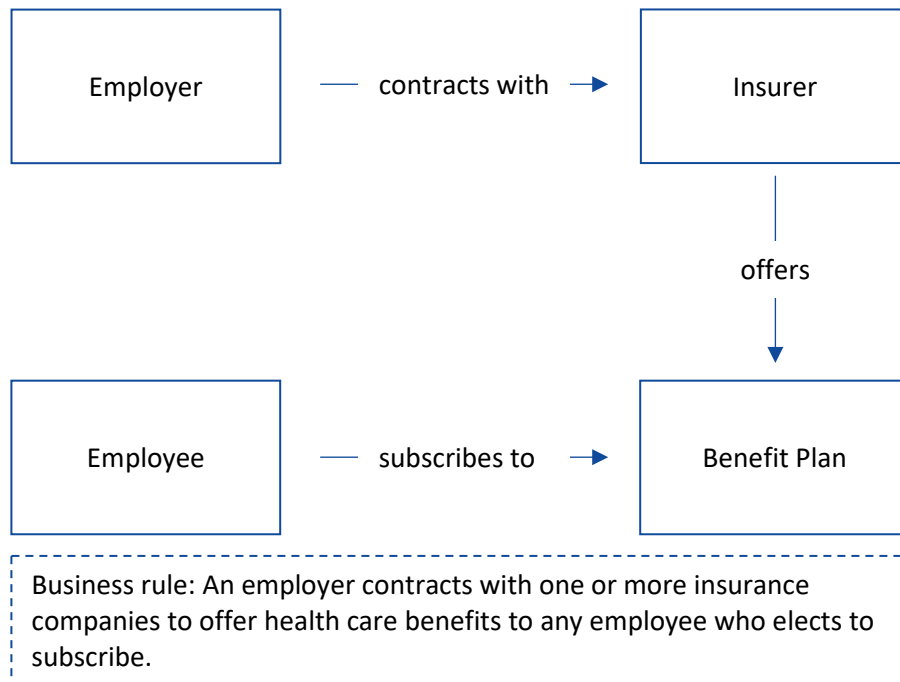
A Conceptual Data Model

- includes the important entities and the relationships among them,
- does not specify any attributes, and
- Does not specify any primary keys.

DM OP-004 Operating Procedure for Estimating Data Management Service Needs

Rationale

The project sizing and estimated cost provides the business enterprise the information for calculation of potential return on investment (ROI).

Operating Procedure

Prior to starting data management services, the Project Sponsor shall ensure development of an estimate of the effort (e.g. the dollar amount or hours, data management resources) and schedule of the data services work necessary to satisfy project needs. Data Management services shall be negotiated with the Data Services Manager.

DM OP-005 Operating Procedure for Developing the Logical Data Model

Rationale

Business requirements are to be documented in a standard notation for consistent documentation with a view for prospective data reuse, and effective communication of business data requirements.

Operating Procedure

1. Select the model type according to project needs. (Refer to DM OP-029.)
2. Refer to the Data Model Tool Use Procedure document for data model tool instructions on
   a. using the appropriate UDP Template from the CMS/DA website based on the selection of model type,
   b. importing the CSV file to the data model (refer to Data Model Tool Use Procedure document), and
   c. populating metadata.
3. Verify that the model notation is a CMS standard.

Standards

1. Business applications that create, update, or replicate data shall have their data requirements documented in a Logical Data Model.
2. The Logical Data Model shall be developed in the standard data modeling tool or in a format that can be directly imported into the standard data modeling tool.
3. The Logical and Physical Data Models shall use IDEF1X notation.
4. The text that identifies the model diagram shall be located in the upper left corner and include the project name; subject area name, if applicable; update date; and modeler identification.

DM OP-006 Operating Procedure for Reuse of Enterprise Entities, Relationships, and Attributes

Rationale
Shared business data artifacts must be managed according to enterprise data stewardship objectives.

Operating Procedure
Project Data Analysts shall use shared model objects where available rather than duplicating them for exclusive use by each project. Reuse of enterprise entities and attributes requires observance of the following requirements:
1. Compare the *needed entities* in the project Conceptual Data Model (CDM) with the *existing entities* in the Enterprise Logical Data Model (ELDM) to identify candidate reusable entities.
2. Select appropriate candidate entities, attributes, and relationship types from the ELDM
3. When a project selects and incorporates in a project model an entity or attributes from the ELDM, keep a log of changes throughout the project lifecycle.
4. Work with the Data Administration & Engineering Services team to ensure correct interpretation and use of the existing data objects in the ELDM.

Standards
1. New data definitions shall not redundantly define information already in the Enterprise Metadata Repository. The goal is to ensure that a new data artifact does not duplicate an existing one.
2. The project ELDM log shall include entries for the following instances:
    a. altered entity and attribute definitions,
    b. changes to cardinality or optionality, and
    c. deletions, migrations, or changes of any kind.
3. If applicable, ELDM entity names must be either used as the entity name in the application data model or be documented in the entity UDP "entity ELDM name."
4. If applicable, ELDM attribute names must be either used as the attribute name in the application data model or be documented in the attribute UDP "attribute ELDM name."
5. If the application data model contains an ELDM entity or attribute, the application data model must use the ELDM entity's or attribute's definition.

DM OP-007 Operating Procedure for Reuse of Enterprise Data Resources

Rationale
Shared data resources must be managed according to enterprise data management objectives.

Operating Procedure
1. Project Data Analysts shall make use of existing enterprise data resources.
2. Develop a project Data Source Plan, which identifies candidate sources for the data required to satisfy the project's business requirements.

Standards
For each candidate data source, the following shall be documented:
1. Name
2. Description
3. Business Owner or steward
4. Project data needs satisfied
5. Limitations (e.g., data quality issues)
6. Constraints (e.g., operational impact of using source)

DM OP-008 Operating Procedure for Defining Data Entities

Rationale
A well-formed definition makes a data object's contents apparent to business users.

Operating Procedure
1. At the time an entity is added to the Logical Data Model it shall be defined in accordance with the following standards and guidelines.
2. Definitions are to be developed by subject matter experts who understand the business meaning, in conjunction with the data analyst. If the data analyst develops the definition, it is important that subject matter experts review and confirm the accuracy of the definition.

Standards
1. An entity shall be defined in a manner that distinguishes its unique role within the business enterprise; i.e. two entities with different names shall have different definitions.
2. An application entity that reuses an enterprise entity must have the same definition. An application entity is one in any logical data model other than the Enterprise Logical Data Model (ELDM). An enterprise entity is one defined in the Enterprise Logical Data Model that is maintained by CMS/DA (Data Administration).
3. The definition shall be clear, concise, and unambiguous. Examples or exclusions may be added to the definition to improve clarity.
4. The definition shall describe a single occurrence of the entity.
5. The definition shall start with either 'A' or 'An', e.g. "A person who…", or "An individual that…" Corollary: the definition shall not start with a verb, e.g. "Tracks the audits…", "Captures beneficiary…," etc.
6. The definition shall not include references to technology or media. For example, references to "tapes" or "disks" are not appropriate in the definition of a business entity.
7. All acronyms and abbreviations must be spelled out in the definition the first time they appear.
8. If an acronym appears in the entity name it must be spelled out in the entity definition.

Guidelines
1. The definition should exclude references to time-bound or process-bound circumstances. For example, "the information the agency receives quarterly" should not be included in a definition. It is better to state "the information the agency receives for the reporting period." However, some time-bounded circumstances that are legislated may be documented. For example, states are required to submit their expenditure reports every 30 days.
2. The definition should not include references to application privileges, interface behavior, or internal database auditing since data entities are not generally technological concepts.

DM OP-009 Operating Procedure for Naming Data Entities

Rationale

A good data entity name is meaningful and self-documenting. It provides the first source of information about the entity's purpose and contents.

Operating Procedure

1. At the time an entity is added to the Logical Data Model it shall be named in accordance with the following Common Logical Naming Standards and Entity Naming Standards.
2. Refer to DM OP-035 Operating Procedure for Updating the Standard Terms List for description of Object Class Term and Qualifier Term.
3. Refer to CMS Standard Terms Glossary at http://www.cms.gov/Research-Statistics-Data-and-Systems/CMS-Information-Technology/DataAdmin/index.html for valid terms and a classification of terms into object class and qualifier term.
4. If a standard business name exceeds the maximum length allowed by the modeling tool (including embedded spaces), progressively remove the least significant *qualifier term(s)*.

Common Logical Naming Standards

1. A standard term, other than an integer, shall be defined in the CMS Standard Terms Glossary.
2. Names are made of one or more terms separated by single spaces.
3. Term abbreviations that are not acronyms shall only be used in a name when it is necessary to fit the name within the modeling tool character limitation.
4. Integer terms (e.g., five), integer numbers (e.g., 5), ordinal terms (e.g., fifth) and ordinal numbers (5th) cannot be the first term of a name.

Entity Naming Standards

1. An entity name shall be unique throughout the Enterprise Data Model and any individual Project Data Model.
2. An entity name shall be composed of the following types of terms:

Term Type	Cardinality
Object Class	1
Qualifier	0-n

3. A name shall represent a single occurrence of the entity.
4. The object class term shall be fully spelled out in the entity.
5. An entity modeled for a look-up table shall use REFERENCE as the last term in the entity name.

Entity Naming Guideline

Avoid use of DATA as the object class term for a reference entity. Select an object class term that best describes the type of data, e.g. Claim Adjustment Reason.

DM OP-010 Operating Procedure for Defining Data Attributes

Rationale
A well-formed definition makes an attribute's contents apparent to business users.

Operating Procedure
1. At the time an attribute is added to the Logical Data Model it shall be defined in accordance with the following standards and guidelines.
2. Definitions are to be developed by subject matter experts who understand the business meaning, in conjunction with the data analyst. If the data analyst develops the definition, it is important that subject matter experts review and confirm the accuracy of the definition.
3. Allow foreign key attributes to inherit their definitions from the parent attribute. Refer to DM OP-015 to document why the foreign key exists in the child entity.

Standards
1. An attribute shall be defined in a manner that distinguishes its unique role within the business enterprise.
2. An attribute definition shall be clear, concise, and unambiguous. Examples or exclusions may be added to the definition to improve clarity.
3. The definition shall describe a single occurrence of the attribute.
4. If an acronym appears in the attribute name it must be spelled out in the attribute definition unless the acronym has already been spelled out in the entity definition, except in the case of a primary key. If an acronym appears in the attribute name of a primary key it must be spelled out in the attribute definition.
5. The definition shall not include references to technology or media. For example, references to "tapes" or "disks" are not appropriate in the definition of a business attribute.
6. An attribute that has the representation term of 'Switch' shall be defined in terms of the "true" value, e.g. for the attribute Prescription Switch, "Whether the drug is only available by prescription." Thus a "true" value associated with the attribute would be interpreted as "drug is only available by prescription" and a "false" value would be interpreted as "drug is not only available by prescription."

Guidelines
1. An attribute definition should
   a. state the essential meaning of the concept,
   b. be able to stand alone,
   c. be expressed without embedding rationale, functional usage, domain information, or procedural information,
   d. avoid circular reasoning, and
   e. use the same terminology and consistent structure for related definitions.
2. The definition should exclude references to time-bound or process-bound circumstances. For example, "the information the agency receives quarterly" should not be included in a definition. It is better to state "the information the agency receives for the reporting period." However, some time-bound circumstances that are legislated may be documented. For example, states are required to submit their expenditure reports every 30 days.

DM OP-011 Operating Procedure for Analyzing Types of Data Attributes

Rationale
Attributes can be named and defined with additional consideration for how they are used and therefore, enhance their recognition and capability for storing business data values.

Operating Procedure
Define the attribute in the Logical Data model by the following criteria:
1. **Prime** attributes are basic business facts. They are always added to the model.
2. **Derived** attributes can be formulated using values from other attributes. They may be added to the Project Logical Data Model if useful to business communication but not necessarily transformed to physical data design.
3. **Transaction/Interface transaction elements** are included only if they are to be transformed to the physical data design.

Standards
Physical control data, such as "next record number," shall not be included in the Logical Data Model.

Guidelines
1. Derived data may be included in the Logical Data Model as a convenient way to communicate the meaning of calculated business data.
2. If the source data is very dynamic, it is better to just derive it every time.
3. Derived data may or may not be included in the Physical Data Model, depending on the benefit of storing the derived product. Use the following criteria in guiding whether to store derived data.

Pro → Store Derived Attribute	Con → Derive Attribute at Processing Time
Original factors not always available.	Original Factors always available.
Retrieval performance is important.	Updates are frequent.
Original factors are static.	Original factors are dynamic.
Up-to-date accuracy is not important.	Up-to-date accuracy is essential.
Derivation is costly or difficult.	Derivation is cheap and easy.

DM OP-012 Operating Procedure for Naming Data Attributes

Rationale

A good data attribute name is meaningful and self-documenting. It provides the first source of information about the data attribute's purpose and contents.

Operating Procedure

1. At the time an attribute is added to the Logical Data Model it shall be named in accordance with the following Common Logical Naming Standards and Attribute Naming Standards.
2. Refer to DM OP-035 Operating Procedure for Updating the Standard Terms List for description of Qualifier Term, Property Term and Representation Term.
3. Refer to CMS Standard Terms Glossary at http://www.cms.gov/Research-Statistics-Data-and-systems/CMS-Information-Technology/DataAdmin/index.html for valid terms and a classification of terms into qualifier terms, property terms, and representation terms.
4. If a standard business name exceeds the maximum allowed by the modeling tool (including embedded spaces), progressively remove the least significant Property Term(s) or Qualifier Term(s).

Common Logical Naming Standards

1. A standard term, other than an integer, shall be defined in the CMS Standard Terms Glossary.
2. Names are made of one or more terms separated by single spaces.
3. Term abbreviations that are not acronyms shall only be used in a name when it is necessary to fit the name within the modeling tool character limitation.
4. Integer terms (e.g., five), integer numbers (e.g., 5), ordinal terms (e.g., fifth) and ordinal numbers (5th) cannot be the first term of a name.

Attribute Naming Standards

1. An attribute name shall be unique in the entity.
2. A data attribute shall be composed of the following types of terms:

Term Type	Cardinality
Property	0-n
Qualifier	0-n
Representation	1

3. A Representation Term shall be the last term in the attribute name.
4. A Representation Term shall be completely spelled out in the name; no abbreviations or acronyms are to be used.
5. A Representation Term shall be consistent with the definition of the attribute. For example, if the definition of the attribute is "the date the employee is born," then the Representation Term shall be 'Date'.
6. The name of an entity-owned attribute that is part of the primary key shall start with the entity name. For example, the key of Beneficiary should be 'Beneficiary Identifier', not just 'Identifier'. **Exception**: For entities that end with the term 'Reference', the primary key attribute name shall omit the term 'Reference'. For example, the primary for the entity 'Status reference' should be 'Status Code', not 'Status reference Code'. This standard ensures uniqueness and clear lineage when a primary key is used as a foreign key in another entity.

7. Abbreviations that are not acronyms shall only be used when it is necessary to fit the attribute name within character limitations.
8. The representation term "Switch" shall only be used in a name when the attribute logically represents a Boolean value (e.g. T/F, Y/N).

REMOVED: DM OP-013 Operating Procedure for Modeling Derived Data

DM OP-013 has been merged with DM OP-011 Operating Procedure for Analyzing Types of Data Attributes.

DM OP-015 Operating Procedure for Defining Relationships, Cardinality, and Optionality

Rationale

The labeling of entity relationships in the data model aids the understanding of model details. Also, cardinality labels define the minimum and maximum number of entity occurrences that can participate in a relationship. Optionality labels define the existence of the relationship as being optional or mandatory.

Operating Procedure

At the time an entity relationship is added to the Logical Data Model it shall be defined in accordance with the following standards and guidelines.

Standards

1. No redundant relationships between entities are allowed.
2. Each relationship shall be named with a verb or verb phrase to express the relationship in the parent-child direction. Examples: "authorizes," "executes," "submits," "enrolls," "justifies," "elects," "certifies." All relationships between a parent entity and an associative entity shall use the verb phrase "is associated by."
3. Cardinality shall be specified for each direction of a relationship.
4. Optionality shall be specified for each direction of a relationship.
5. The definition of the entity parent to child relationship shall be documented if there is need to clarify the relationship beyond the verb or verb phrase or describe the conditions of the relationship and any special conditions under which it exists.

Guidelines

1. Relationship names may be expressed for each direction of the relationship. If relationship verb phrases are designated for both relationship directions, the two verb phrases should be complementary. Example: "authorizes"/"authorized by."
2. Non-descriptive relationship names such as "belongs to," "may have," "is associated with," or "is a part of" should not be used, except for relationships between a parent entity and an associative entity.

DM OP-016 Operating Procedure for Assigning a Primary Identifier

Rationale
An entity must be defined in a way that the business can distinguish unique occurrences.

Operating Procedure
Designate a unique identifier for the entity type in accordance with the following standards and guidelines.

Standards
In the Project Logical Model, every entity type shall have a unique identifier composed as one of the following options:
1. one attribute,
2. multiple attributes,
3. an identifying relationship and one attribute, or
4. two identifying relationships.

Guidelines
An assigned number may be used as an identifier. This is rarely justified but is permitted for practicality when it represents a unique abbreviated way of referring to an occurrence of an entity type or needed to compensate for other unusual circumstances. Examples:
1. A generated sequential number may be used to identify an entity where the combination of the identifying attributes and relationship is much too cumbersome for practical use as an identifier.
2. During a company merger, records from two sources are to be combined and their existing primary identifiers might result in duplicate identifiers without an assigned number, which acts as a tiebreaker.

DM OP-017 Operating Procedure for Normalizing the Project Logical Data Model

Rationale

The purpose of data normalization is to ensure that (1) each attribute is assigned to the appropriate entity, (2) redundant storage of information is minimized, and (3) data structures are stable and receptive to updates, reducing the risk for introduction of anomalies.

Operating Procedure

1. The Project Logical Model shall be normalized to *First Normal Form.* All attributes of every entity must be atomic, indivisible, and not part of a repeating group.
2. The Project Logical Model shall be normalized to *Second Normal Form.* All attributes of every entity must depend on the full primary key of that entity only.
3. The Project Logical Model shall be normalized to *Third Normal Form (3NF).* All attributes of every entity must depend on the full primary identifier of that entity only and not be functionally dependent on some other attribute.
4. The Project Logical Model may be normalized to *Fourth Normal Form*. The model is in *3NF* and has no sets of multi-valued dependencies.
5. The Project Logical Model may be normalized to *Fifth Normal Form* such that every join dependency is made through candidate identifiers.

Standards

A Project Logical Model shall be in Third Normal Form.

DM OP-018 Operating Procedure for Documenting Domain Value Rules

Rationale

Value domain assignments support business data definition and quality by restricting data to a certain set of permitted values. A *reference table* enables one to (1) categorize, classify, qualify, and constrain other data; (2) provide a method of coding and decoding; and (3) ensure that only valid values are entered into a database.

Operating Procedure

1. Determine if an attribute needs a discrete set of permissible values by applying the rule in Standard #4 below.
2. Using Standard #1 and the Guideline below, determine if a reference entity should be created.
3. Create a reference entity if deemed appropriate following Standard #2 below.
4. Document valid values or data source in the data model.

Standards

1. If there is an external source of valid values, e.g. an industry standard, then there shall be a separate reference entity.
2. A reference entity shall have a Code as the primary key and a non-key attribute for Description.
3. An entity data source shall be documented for a reference entity.
4. An attribute with the Representation Term of 'Indicator' and an attribute with the Representation Term 'Code' that is not modeled in a reference entity shall have valid values and descriptions documented in the data model.

Guidelines

In general, it is better to create reference entities as opposed to maintaining reference values as domains in the data model, especially if those domains are volatile, i.e. the set of values may be subject to change. If, however, the set of valid values is fixed (and, typically, small), then it may be appropriate to maintain in the data model.

REMOVED: DM OP-019 Operating Procedure for Checking Completeness of a Project Logical Data Model

DM OP-019 has been replaced by the New Model Review Procedure document. Standards were moved to OP-005 Operating Procedure for Developing the Logical Data Model and OP-017 Operating Procedure for Normalizing the Project Logical Data Model.

DM OP-020 Operating Procedure for Documenting Data Issues

Rationale
Sometimes it is not possible to express a data issue in the data modeling tool however, it is still necessary to communicate the issue and track its resolution.

Operating Procedure
1. Use the *Data Issues Notes* to report the following:
2. Document business rules that cannot be expressed using the data modeling tool.
3. Document data quality issues of any existing project data that might be in opposition to stated business rules.
4. Document unusual database physical platform location requirements.
5. Document data conversion challenges and any other issues that require special attention from DBA or business users.
6. Document rules according to standards below.

Standards
Document Data Issues following the example below, including the underlined sections.

Data Issue: The current production table column *BENEFICARY. INDV_WGHT_AMT* has a defined data type of CHAR(3), which has resulted in entry and storage of invalid weight data. Project "*CMIS: Contractor Management Information System"* requires that the column be changed to *smallint*. Therefore, existing values that are non-compliant with the revision, must be corrected before it can be reloaded to the revised table column.

Date Recorded: 2004/08/24

Resolution: A conversion program has been developed to correct invalid weight data in table column BENEFICARY. INDV_WGHT_AMT. Doug Nichols, the project manager will coordinate correction of invalid data with the assigned database administrator, prior to production change implementation.

Date Resolved: 2004/08/25 by Mable Hanks, assigned Project Data Analyst

DM OP-021 Operating Procedure for Assigning Information Security Categories

Rationale

The Federal Information Security Act (FISMA) requires all federal agencies to document the appropriate security categories of asset and operational data.

Operating Procedure

1. Obtain the appropriate security setting of each entity from the data business owner.
2. Document the security setting by entity in the Logical Data Model. (See DM OP-031 Operating Procedure for Capturing the Standard Logical Data Model Metadata.)

Standards

1. Enterprise data shall be assigned with appropriate security settings in compliance with *FIPS Publication 199 – Standards for Security Categorization of Federal Information and Information System* as shown in the table below*.
2. Each entity in a Logical Data Model shall be assigned a potential impact level for the following security objectives: confidentiality, integrity, and availability.

	Potential Impacts		
Security Objective	**Low**	**Moderate**	**High**
Confidentiality Preserving authorized restrictions on information access and disclosure, including means for protecting personal privacy and proprietary information. [44 USC, SEC 3542]	The unauthorized disclosure of information could be expected to have a **limited** adverse effect on organizational operations, organizational assets, or individuals.	The unauthorized disclosure of information could be expected to have a **serious** adverse effect on organizational operations, organizational assets, or individuals.	The unauthorized disclosure of information could be expected to have a **severe or catastrophic** adverse effect on organizational operations, organizational assets or individuals.
Integrity Guarding against improper information modification or destruction, including ensuring information non-repudiation and authenticity. [44 USC, SEC 3542]	The unauthorized modification or destruction of information could be expected to have a **limited** adverse effect on organizational operations, organization assets, or individuals.	The unauthorized modification or destruction of information could be expected to have a **serious** adverse effect on organizational operations, organization assets, or individuals.	The unauthorized modification or destruction of information could be expected to have a **severe or catastrophic** adverse effect on organizational operations, organization assets, or individuals.

	Potential Impacts		
Security Objective	**Low**	**Moderate**	**High**
Availability Ensuring timely and reliable access to and use of information. [44 USC, SEC 3542]	The disruption of access to or use of information or an information system could be expected to have a **limited** adverse effect on organizational operations, organizational assets, or individuals.	The disruption of access to or use of information or an information system could be expected to have a **serious** adverse effect on organizational operations, organizational assets, or individuals.	The disruption access to or use of information or an information system could be expected to have a **severe or catastrophic** adverse effect on organizational operations, organizational assets, or individuals.

DM OP-022 Operating Procedure for Generating the Project Metadata Repository

Rationale

The Metadata Repository documents the understanding of business data and facilitates the communication about the information that is represented in the Logical Data Model.

Operating Procedure

The Metadata Repository holds the following information:
1. For each entity,
    a. Entity Name
    b. Entity Definition
    c. Degree of required Confidentiality
    d. Degree of required Integrity
    e. Degree of required Availability
2. For each attribute,
    a. Attribute Name
    b. Definition
    c. Primary Key Indicator
    d. Foreign Key Indicator
    e. Data type
    f. Valid Values
    g. Edit Criteria
    h. Business transformation rules

REMOVED: DM OP-023 Operating Procedure for Performing a Quality Review of the Project Logical Data Model

DM OP-023 has been replaced by the New Model Review Procedure. Please see the Logical Design page http://www.cms.gov/DataAdmin/03_LogicalDataDesign.asp#TopOfPage.

REMOVED: DM OP-024 Operating Procedure for Approval of Data Models based on Business Data Views

DM OP-024 has been replaced by the New Model Review Procedure. Please see the Logical Design page http://www.cms.gov/DataAdmin/03_LogicalDataDesign.asp#TopOfPage.

DM OP-025 Operating Procedure for Collecting Data Stability and Growth Information

Rationale

Based on predictions by subject matter experts (SMEs), resources can be better chosen to accommodate near- term volume expectations. This procedure encourages collection of data stability and growth information that might be used by the Data Base Manager in selection of the appropriate Data Base Management System (DBMS) platform. (This information serves only as preliminary prediction. Additional, detailed planning for physical storage allocations will be performed by the assigned data base administrator (DBA) in the course of physical database implementation.)

Operating Procedure

This information is typically captured on the ERWIN Volumetric Sizing Information table property tab within the PDM (Physical Data Model) for each table. Document the expected growth rate of a new table predicted by the SMEs, as an annual increase in the number of rows. For example, if the SME predicts 100,000 more records in 2 years, the estimated annual growth is 50,000 records which can be specified in the Grow By property value in the Erwin Volumetric Sizing Information table property tab.

DM OP-026 Operating Procedure for Completing the Data Models

Rationale
The data models are stored and cataloged to benefit ongoing enterprise data analysis and business data analysis.

Operating Procedure
1. All data models shall be stored and cataloged in DDMSS model management libraries, according to their state of completion.
2. All models shall be appropriately stored when work is completed (or halted in an incomplete or unapproved status).

DM OP-027 Operating Procedure for Granting Project Data Model Library Access

Rationale
Storing models in DDMSS libraries provides protected storage for data models and data management documents.

Operating Procedure
1. Each project shall have designated model management libraries:
   a. 2 libraries in Model Manager; one for development models, the other for production models.
   b. 1 folder in the DDMSS Change Control library to hold project Data Issue Documents and Change Control Documents.
2. Full update privileges for development libraries and viewing privileges to project production libraries are available to authorized Project Data Analysts and Data Base Administrators.
3. Update privileges for Model Manager and DDMSS production model libraries are exclusive to the Model Manager Administrator.
4. Assign a name and provide a description for each Model Manager library in the following manner:
5. The library name consists of: *application acronym + application business name + phase designation + approval status*.
   a. Example: *CMIS: Contractor Management Information System Approved*.
6. The application library description should be brief and may include the names of the assigned data analysts, project manager, and government task leaders (GTL).
   a. Example: *As an intended replacement for the CROWD system, CMIS allows CMS to effectively manage, monitor, and report on the performance of its Medicare fee-for-service contractors. Project Manager is Doug Nichols of Acme Systems, Inc. The Project Data Analyst is Mable Hanks. The Project Database Administrator is Nancy Owens.*
7. Create a folder in the *DDMSS Change Control Library*.
8. The folder name convention is: *application acronym + application business name + phase designation + YYYYMMDD*.
   a. Example: *CMIS: Contractor Management Information System 20030401*.

DM OP-028 Operating Procedure for Naming and Defining Data Models

Rationale

Consistent names and description formats are important to the cataloging of data models, managing them as data architecture resources, and exploiting their potential for reuse.

Operating Procedure

1. At the time a data model is being created the model name and definition shall adhere to the following standards.
2. The name of the data model is changed when the approved data model is updated to implement version control.

Standards

1. A data model shall be named in the following manner:
   system acronym + ("relational"/"dimensional") + model type (EDM/CDM/LDM/PDM) + approval date (or the storage date for models in development) in YYYYMMDD format.
   Example: **CMIS relational LDM 20040726**
2. The model definition shall describe the purpose and status of the model in a few sentences of text.
   Example: **An integrated view of core data required for the main business functions of Centers of Medicare and Medicaid. Current approved LDM.**

DM OP-029 Operating Procedure for Selecting Model Type

Rationale

The CMS standard data modeling tool, *All-Fusion ERwin Data Modeler*, provides convenient functions that help the data modeler to create models that support the various levels of CMS data architecture.

Operating Procedure

Select the model type according to project needs. Logical models capture data requirements. Physical models support the development of a database design in parallel with or separately from the associated conceptual or logical model development.

1. The following table lists the allowed model source relationships for logical and physical models:

Parent Model Type	Child Model Type	Notes
Logical	Logical	For example, enterprise to project logical, or project conceptual to project logical.
Logical	Physical	You can derive a physical model from a logical model. The result is a pair of linked models with the logical model as the parent and the physical model as the child.
Physical	Physical	For example, generic physical to target server-specific physical.

2. A third model type, logical/physical, combines one logical model and one physical model that are tightly coupled. Use a logical/physical model when you require only one physical implementation for a given logical model and do not need to continue logical modeling while the physical design is being implemented. Logical/physical models can also be created by reverse engineering an implemented database or a DDL script. Logical/physical models are used in design layer architectures, but their usage is a little different. The allowed model source relationships that result from splitting a logical/physical model are:

Parent Model Type	Child Model Type	Notes
Logical/Physical	Logical	The logical level and model properties such as the naming standards model and data type standards model may be passed to the child model.
Logical/Physical	Physical	The physical level, and model properties such as the naming standards model and data standards model, may be passed to the child model.

3. In addition to the model parent types (logical, physical, or logical/physical) models are either relational or dimensional. Note if a dimensional model is constructed using a star schema.

Standards

A dimensional data model shall be identified as such in the modeling tool.

REMOVED: DM OP-030 Operating Procedure for Conducting a Physical Data Model Review

DM OP-030 has been deleted. It is no longer a DAES operating procedure.

DM OP-031 Operating Procedure for Capturing the Standard Logical Data Model Metadata

Rationale

Capturing a standard set of metadata associated with the Logical Data Model provides the necessary details to foster communication within and across projects.

Operating Procedure

1. When creating and updating a Logical Data Model, capture the model metadata as defined in the Standard Metadata Matrix below for required and conditional metadata corresponding to the type of data model.
2. When creating and updating an entity in a Logical Data Model, capture the entity metadata as defined in Standard Metadata Matrix below for required and conditional metadata corresponding to the type of data model.
3. When creating and updating an attribute in a Logical Data Model, capture the attribute metadata as defined in the Standard Metadata Matrix below for required and conditional metadata corresponding to the type of data model (logical only or logical/physical).
4. Refer to Data Model Tool Use Procedure document for the Erwin access path used to capture the metadata in an application data model.

Standards

All metadata shown as 'Required' in the Standard Metadata Matrix shall be populated in the corresponding type of data model.

Metadata	Logical	Log/Phys	Physical	Format/Description
Model-level				
Model Name	Required	Required	Required	See DM OP-028 Operating Procedure for Naming and Defining Data Models
Model Author	Required	Required	Required	The first and last name of the Local or Central Data Architect as well as the company name of the person responsible for creating and/or maintaining the data model. Example: Bob Smith, XYZ Company
Model Type	Required	Required	Required	See DM OP-29 Operating Procedure for Naming and Defining Data Model
Model Definition	Required	Required	Required	See DM OP-028 Operating Procedure for Naming and Defining Data Models
Model Version Effective Date	Required	Required	Required	The date that identifies a version of a model (can be a system release date)
Model Central DA Name	Required	Required		The name of the person who is responsible for certifying the compliance of the model components according to CMS Data Administration standards.
Model Business Owner Name	Required	Required		The name of the organization/person who is responsible for approving the definitions in the model. Example: R. R. Kirk (CMS/CMM)
LDM CDA Signoff Date			N/A	The date when this version of the logical data model received its Central DA sign-off.
Project Name				The name of the application project (e.g. Integrated Data Repository)
Project Release Number				The project release number.
Entity-level				
Entity Name	Required	Required	N/A	See DM OP-009 Operating Procedure for Naming Data Entities
Entity Definition	Required	Required	N/A	See DM OP-008 Operating Procedure for Defining Data Entities

Metadata	Logical	Log/Phys	Physical	Format/Description
Entity Requirement ID	Required	Required	N/A	A reference to the requirement(s) or change request identifier(s) that justify the existence of the entity in the model. Valid entry formats are as follows: 1. DOORS or requirement document Tracking ID use – BR-#### for business requirements, FR-#### for functional requirements, and SR-#### for system requirements. 2. CR-#### for change requests 3. DR-##### for data change requests 4. RT_###### for Remedy Tickets 5. MAPD_#### for MAPD
Entity Security Category Description	Required	Required	N/A	A reference to the *FIPS Publication 199 – Standards for Security Categorization of Federal Information and Information System*, which describes the risk of unauthorized access, unauthorized modification, or unavailability of the data represented by the Entity. The format of this UDP contains three values separated by semicolons, e.g. CONFIDENTIALITY=<impact>; INTEGRITY=<impact>; AVAILABILITY=<impact>, where <impact> has a value from the list. Low, Moderate, High, N/A. Example: CONFIDENTIALITY=HIGH; INTEGRITY=MEDIUM; AVAILABLITY=LOW See DM OP-021 Operating Procedure for Assigning Information Security Categories
Entity ELDM Name	Conditional	Conditional	N/A	The name of the corresponding entity in the Enterprise Logical Data Model if different from the project's entity name. See DM OP-006 Operating Procedure for Reuse of Enterprise Entities, Relationships and Attributes.
Logical Only Entity Indicator			N/A	An indication that the entity will not correspond to a physical table(s).
Entity Data Source	Conditional	Conditional	N/A	An external system or document source for a Reference entity – it is (1) manual or automated system sources of code values and descriptions or (2) departmental document sources of code values and descriptions. See DM OP-018 Operating Procedure for Documenting Domain Value Rules.
Attribute-level				
Attribute Name	Required	Required	N/A	See DM OP-012 Operating Procedure for Naming Data Attributes

Metadata	Logical	Log/Phys	Physical	Format/Description
Attribute Definition	Required	Required	N/A	See DM OP-010 Operating Procedure for Defining Data Attributes
Attribute Primary Key Indicator	Required	Required	N/A	An indication that the attribute is primary key.
Attribute Domain Name			N/A	The name of a generic domain that can be used to standardize attribute data type, i.e. the data type assigned to a domain will apply to all attributes with the same domain.
Attribute Logical Only Indicator			N/A	An indication that the attribute will not correspond to a physical column.
Attribute Valid Values	Conditional	Conditional	N/A	A list of all valid codes and descriptions of a code or indicator attribute. See DM OP-018 Operating Procedure for Documenting Domain Value Rules.
Attribute Required Indicator			N/A	An indication of whether or not the Attribute must assume a non-null value when an entity instance is created.
Attribute Derivation Text			N/A	The narrative explanation of any non-trivial logical used to transform information from one or more other attributes into the value for this attribute.
Attribute Requirement ID	Conditional	Conditional	N/A	A reference to the requirement(s) or change request identifier(s) that justify the existence of the attribute in the model or changes to the attribute. Entry is required only when the attribute has been added as a result of a requirement that differs from the requirement entered in the Entity Requirement ID of the entity to which the attribute belongs. If the requirement for the attribute is the same as that of the entity, the Attribute Requirement ID is not required. Valid entry formats are as follows: 1. For DOORS or requirement document Tracking ID, use BR-#### for business requirement, FR-##### for functional requirement, SR-#### for system requirements. 2. For Change Requests, use CR-####. 3. For Data change requests, use DR-#####. 4. For Remedy Tickets, use RT_######. 5. For MAPD, use MAPD_####.

Metadata	Logical	Log/Phys	Physical	Format/Description
Attribute Data Source Name	Required	Required	N/A	The CMS database, external data feed, manual data entry process, or software process from which the column takes its value. Valid entry formats are as follows: 1. CMS sources – database.table.column or filename.recordtype.field. 2. External system or document sources – These are (1) manual or automated system sources of data element names or (2) departmental document sources of data element types. The format is org.specification.pubdate.record.subpart. field. Subpart is used where there are multiple record formats or levels of element structure, and may be omitted if there is only one flat format for the record formats or levels of element structure, and may be omitted if there is only one flat format for the record specification. The following example shows a column data source field which is located three levels deep within an XML document. Example: 'HealthLevelSEven.CDAL1R1.20061219.<level one_1.0.xsd>.<clinical_document_header>. <provider>.<type_cd>'. 3. Software processes – application name, service name, or 'System Generated'. 4. Manual data entry processes – application-formname.fieldname, or 'User-Supplied'. If this level of detail is not available, indicate the source in terms of a user role, type of stakeholder, organization, business process, system, program, database, or file.
Attribute Alias Name			N/A	Any applicable alternate business name used to refer to the attribute.
Attribute ELDM Name	Conditional	Conditional	N/A	The name of the corresponding attribute in the Enterprise Logical Data Model if different from the project's attribute name. See DM OP-006 Operating Procedure for Reuse of Enterprise Entities, Relationships, and Attributes.

Metadata	Logical	Log/Phys	Physical	Format/Description
ANSI/ISO Standard Element Name			N/A	The name of the corresponding data element defined in a national or international standard. The National/International Standard Element Name(s) preceded by the standard that corresponds to the CMS attribute name. For example, the CMS attribute name = Beneficiary Birth Date would be reflected as follows: ANSI/ISO Standard Element Name: HIPAA, Subscriber Date of Birth; C83, Person Date of Birth. Note: Multiple standards are separated by semicolons.
Attribute Personally Identifiable Information (PII)	Conditional	Conditional		Whether the attribute is Personally Identifiable Information (PII). 'Y' means it is. 'N' or blank is not.
Table-level				
Table Name	N/A	Required	Required	See DM OP-045 Operating Procedure for Construction First Cut Physical Table or File Names from the Logical Data Model.
Table Comment	N/A	Required	Required	The narrative explanation of the meaning of a row in the Table. Example: TUS_CNTRCT – A binding agreement between CMS and an Organization that enables eligible Medicare beneficiaries to obtain medical services from the organization in exchange for monthly payments. Both current and historical information is retained.
Physical-Only Table Indicator	N/A			An indication that the table will not correspond to an entity.
Logical Entity Name	N/A	N/A		The CDA-approved logical entity name that is equivalent to the table's physical name. It may also be the name of the corresponding entities defined in the Logical Data Model, which corresponds to this Physical Data Model.

Metadata	Logical	Log/Phys	Physical	Format/Description
Table Requirement ID	N/A			A reference to the requirement(s) or change request identifier(s) that justify the existence of the Table in the model. Valid entry formats are as follows: 1. For DOORS or requirement document Tracking ID use BR-#### for business requirement, FR-##### for functional requirement, SR-#### for system requirements. 2. For change requests, use CR-####. 3. For data change requests, use DR-#####. 4. For Remedy Tickets, use RT_######. 5. For MAPD, use MAPD_####.
Table Security Category Description	N/A			A reference to the *FIPS Publication 199 – Standards for Security Categorization of Federal Information and Information System*, which describes the risk of unauthorized access, unauthorized modification, or unavailability of the data represented by the Table. The format of this UDP contains three values separated by semicolons, e.g. CONDIENTIALITY=<impact>; INTEGRITY=<impact>; AVAILABILITY=<impact>, where <impact> has a value from the list: Low, Moderate, High, N/A. Example: CONFIDENTIALITY=HIGH; INTEGRITY=MEDIUM; AVAILABILITY=LOW. See DM OP-021 Operating Procedure for Assigning Information Security Categories.
Column-level				
Column Name	N/A	Required	Required	See DM OP-046 Operating Procedure for Constructing Physical Column or Element Names.
Column Comment	N/A	Required	Required	The narrative explanation of the meaning of an instance of the column. Example: PLAN_PBP_NAME – The name assigned by the contractor to the Plan Benefit Package.
Primary Key Indicator	N/A	Required	Required	An indication of whether the column is part of the table's primary key.
Column Domain Name	N/A			The name of the domain which defines the data type, default value, and valid values of the column.
Physical-Only Column Indicator	N/A			An indication that the column will not correspond to an attribute.
Logical Attribute Name	N/A	N/A		The CDA-approved logical attribute name that is equivalent to the column's physical name.

Metadata	Logical	Log/Phys	Physical	Format/Description
Column Data Source Name	N/A			The CMS database, external data feed, manual data entry process, or software process from which the column takes its value. Valid entry formats are as follows. 1. CMS – database.table.column or filename.recordtype.field. 2. External – org.specification.pubdate.record.subpart. field. Subpart is used where there are multiple record formats or levels of element structure and may be omitted if there is only one flat format for the record specification. The following example shows a column data source field which is located three levels deep within an XML document. Example: 'HealthLevelSeven.CDAL1R1.20061219.<level one_1.0.xsd>.<clinical_document_header>. <provider>.<type_cd>'. 3. Software processes – application name, service name, or 'System Generated'. 4. Manual data entry processes – application.formname.fieldname or 'User-Supplied'. If this level of detail is not available, indicate the source in terms of a user role, type of stakeholder, organization, business process, system, program, database, file, or data exchange standard.

Metadata	Logical	Log/Phys	Physical	Format/Description
Column Requirement ID	N/A			A reference to the requirement(s) or change request identifier(s) that justify the existence of the column in the model or changes to the column. Entry is needed only when the column has been added as a result of the requirement that is different from the requirement entered in the Table Requirement ID of the table to which the column belongs. If the requirement for the column is the same as that of the table, the Column Requirement ID is not required. Valid entry formats are as follows: 1. For DOORS or requirement document Tracking ID use BR-#### for business requirement, FR-##### for functional requirement, SR-#### for system requirements. 2. For change requests, use CR-####. 3. For data change requests, use DR-#####. 4. For Remedy Tickets, use RT_######. For MAPD, use MAPD_####.
Column Derivation Text	N/A			The narrative explanation of any non-trivial logic used to transform information from one or more other columns or fields into the value for this column.
Relationship-level				
Relationship Parent-to-Child Verb	Required	Required	Required	Express the relationship in the parent-child direction.
Relationship Cardinality	Required	Required	Required	Define the minimum and maximum number of entity occurrences that can participate in a relationship.
Relationship Optionality	Required	Required	Required	Define the existence of the relationship as being optional or mandatory.
Relationship Definition				The description of the relationship explains the existence of the foreign key(s) in the child entity, as opposed to overwriting the foreign key attribute definition.

DM OP-032 Operating Procedure for Capturing the Standard Physical Data Model Metadata

Rationale

Capturing a standard set of metadata associated with the Physical Data Model provides the necessary details to foster communication within and across projects.

Operating Procedure

1. When creating and updating a Physical Data Model, capture the model metadata as defined in the Standard Metadata Matrix above for required and conditional metadata corresponding to the type of data model.
2. When creating and updating an entity in a Physical Data Model, capture the table metadata as defined in the Standard Metadata Matrix above for required and conditional metadata corresponding to the type of data model.
3. When creating and updating an attribute in a Physical Data Model, capture the column metadata as defined in the Standard Metadata Matrix above for required and conditional metadata corresponding to the type of data model.
4. Refer to Data Model Tool Use Procedure document for the Erwin access path used to capture the metadata in an application data model.

DM OP-033 Operating Procedure for Recording Data Model Changes

Rationale

A Project Logical Data Model (PLDM) will often undergo successive periods of change after the initial form of the model is reviewed and approved by Central Data Administration. These changes will be recorded on an ongoing basis by the Project Local DA. The benefits of recording such changes include:

1. Being able to present Central DA with a clear list of the changes that Central DA needs to review for a specific version of the Project Logical Data Model.
2. Providing the Project team with a concise historical record of when and why the valuable resource that the Project Logical Data Model constitutes was altered.

Operating Procedure

Record changes to the PLDM objects using the template "Model Change Report Template" on http://www.cms.gov/DataAdmin/03_LogicalDataDesign.asp#TopOfPage. The template contains instructions on how to fill out the fields.

DM OP-034 Operating Procedure for Updating the ELDM

Rationale

To ensure that the Enterprise Logical Data Model (ELDM) continues to represent data objects from an enterprise perspective it is necessary to update the ELDM with new or revised business entities, attributes, and relationships. These revisions will be a result of creating or revising Project Logical Data Models.

Operating Procedure

There are two scenarios for identifying changes that may impact the ELDM.

1. Creation of a new Project Logical Data Model (PLDM)
   The "Checklist of Potential New ELDM Objects," which is a product of the Logical Data Design Process, will serve as the initial document for identifying and determining the need to add or modify the ELDM. This document coupled with the 3NF PLDM will be provided to the Central Data Administration Organization. From these documents the Central Data Administrator will create a "Proposed New Enterprise Data Report" that will be distributed to the individuals responsible for maintaining the ELDM.

2. Revisions to the Project Logical Data Model (PLDM)
   Changes to an existing PLDM will be recorded as stated in DM OP-033 (Operating Procedure for Recording Data Model Changes). The "Model Change Report Template" document will be provided to the Central Data Administration Organization responsible for maintaining the ELDM. The Data Administrators in this organization will review the change and determine if a change to the ELDM is required.

DM OP-035 Operating Procedure for Requesting a New Standard Term.

Rationale

A consistent use of business terms leads to a common vocabulary and understanding of business data. Therefore, it is important that the business taxonomy be managed and used appropriately to label data entities and attributes.

Operating Procedure

If terms in the *Standard Term List* are inadequate for naming a data entity or attribute, a request may be made to add a new standard term to the list. Integers are implicitly standard terms and do not have to be added to the *Standard Term List*.

1. Requestor fills out a Standard Term Request Form at http://www.cms.gov/Research-Statistics-Data-and-Systems/CMS-Information-Technology/DataAdmin/index.html - Download Forms.zip.
   a. Complete Section I – Data Analyst Information
   b. Fill in Section II – Standard Term Information as follows:
      i. Indicate if the term is new or a change to an existing term in the *Standard Term List*.
      ii. Enter the Proposed TERM in accordance with the Term Name Standards and Guidelines below.
      iii. Enter a Proposed TERM Abbreviation in accordance with the Term Abbreviation Standards below.
      iv. Indicate in which category(s) the proposed term belongs:
         1. Object Class Term (compliance with ISO/IEC 11179 – 5): An Object Class Term is a noun that represents an important entity classes or entity subclasses in an enterprise data subject area. Examples of Object Class Terms: Account, Beneficiary, Claim, Contract, Provider.
         2. Property Term (compliance with ISO/IEC 11179 – 5): A Property Term expresses a discrete category. Examples of Property Terms: City, Gender, Spouse, Report, Revenue, Color
         3. Qualifier Term (compliance with ISO/IEC 11179 – 5): A Qualifier Term can be used to further modify or specify an Object Class Term or Property Term. Examples of Qualifier Terms: Minimum, Net, Last, Temporary.
         4. Representation Term (compliance with ISO/IEC 11179 – 5). Examples of Representation Terms: Code, Name, Identifier, Date, Quantity
      v. Provide a definition of the term.
      vi. Provide an example of an entity or attribute name using the term.
   c. Leave Section III blank.
2. Requestor emails the Standard Term Request Form to DataAdmin@cms.hhs.gov
3. Requestor will receive confirmation of receipt and acceptance from DataAdmin.

Term Name Standards

1. A term shall contain one or more words or integers separated by spaces.
2. Valid term characters are alphabetic (a-z) and integer (0-9).
3. A term shall not include a period (.), slash (/) and hyphen (-) or other special characters.
4. A term shall not be a numeric representation of an ordinal number (e.g., 5th is not a valid term, "Fifth" is a valid term).

5. Each word in a term shall start with a capital letter and the rest shall be lower case.
6. A Representation Term shall be one word in singular form, e.g. DATE, not DATES.

Term Name Guidelines

1. Avoid addition of synonyms to the list, except in those instances where the synonym's use is justified for business communication. For example, when the term "Responsibility" is an existing standard term, resist adding the synonym "Duty."
2. Avoid negative terms such as "Nonparticipant", "Unqualified", "Ineligible", etc. Rationale: the use of negative terms invariably leads to confusing names; also, they have the potential for creating double negative conditions.

Term approval criteria	Example of data name using negative term	Example of data value	Likely business interpretations or questions
Avoid negative terms	Person Ineligibility Switch	"N"	The person is eligible, or are they?
		"Y"	Is the person eligible?
Allow positive terms	Person Eligibility Switch	"N"	The person is not eligible.
		"Y"	The person is eligible.

3. When a word can be expressed in different forms to indicate possession or direction, multiple forms of a term can be accepted, e.g. "Payee," "Payer," and "Payment."
4. Representation Term describes the valid form of data content. Add new terms only when a new form of data cannot be accommodated or expressed by the data types associated with existing Representation Terms. (See DM OP-040 Standard for Designating Representation Term and Data Type.)

Term Abbreviation Standards

1. Term abbreviations shall be in all capital letters.
2. All integers are implicitly valid abbreviations (e.g. 101). The term is the English name of the integer (e.g., One Hundred One) without the 'and' conjunction.
3. Words that are four letters or less are not abbreviated unless there is a widely-recognized and common abbreviation available.
   Example: AGE [AGE]
   There are exceptions such as: DATE [DT], CODE [CD]
4. Abbreviations are limited to eight characters. Abbreviations of 4 characters are preferred.
   Discouraged example: RESPONSIBILITY [RSPNSBLTY]
   Suggested example: RESPONSIBILITY [RESP]
5. Abbreviations must be unique in order to support clarity of usage except for the widely accepted acronyms.
   First term example: RESPONSIBILITY [RESP]
   Second term example: RESPONSE [RSPN]
   First acronym Example: Plan Benefit Package [PBP]
   Second acronym Example: Provider Based Physician [PBP]

6. An abbreviation may not form a word. If the abbreviation forms a word, replace letters to make the abbreviation not be a word while maintaining uniqueness.
   Prohibited example: RETENTION [RENT]
   Suggested example: RETENTION [RTNT]

Term Abbreviation Guidelines

1. Condense all double consonants to a single consonant.
   Example: Enrollment [ENRL]
2. Eliminate vowels except: (a) when it is a leading vowel, (b) when the vowel is needed for clarity, (c) when the vowel is needed to provide uniqueness, and (d) when a vowel is part of a commonly used abbreviation.
   Example of a: Employee [EMPY]
   Example of b: Exposition [EXPO]
   Example of c: Library [LIB] and Liability [LIAB] (An "A" is added to distinguish Liability's abbreviation.)
   Example d: Product [PROD]
3. Abbreviated words that are widely known and consistently used in the agency should be defined as an acronym.
   Example: Health Service Area [HSA]

DM OP-036 Operating Procedure for Assigning Data Analysts

Rationale
The *quality* of project data analysis and design deliverables relies on the capabilities of the Project Data Analyst.

Operating Procedure
Assign Project Data Analysts that have practical experience that is commensurate with the project's data analysis needs and the anticipated design complexity.

Guidelines
Suggested Skill Set:
1. A minimum of five years of experience in Relational and/or Dimensional Modeling techniques.
2. Skills for compiling business requirements analysis and documentation.
3. Experience interfacing with health care professionals and business customers.
4. Skills and experiences with Medicare subject matter domains.
5. Knowledge of the health care insurance industry.
6. Experience with standard CMS data modeling tools.

DM OP-037 Operating Procedure for Conducting the Logical Data Design Kickoff Meeting

Rationale

The purpose of this guideline is to encourage an understanding of participant roles and responsibilities, discuss major project goals, and confirm the level of agency sponsorship including project budget authorization. The meeting also provides an opportunity to discuss the data design process, the checkpoint reviews, and the location and scope of the data administration standards.

Operating Procedure

Follow the outlined agenda and exchange information as indicated. Agenda:
1. Introduce team members and roles.
2. Explain data design process.
3. Identify level of service needed and available funding sources.
4. Identify schedule requirements.
5. Collect data requirements documents from Project Manager.
6. Estimate time frame for follow-up response.

DM OP-038 Operating Procedure for Defining a Subject Area

Rationale

Data architecture is more helpful to business personnel when it is organized in a manner that speaks to business purposes. That is the objective of a Subject Area: to serve as the broadest classification for a business line's data.

Operating Procedure

Create a Subject Area by the following the guidelines below.

Guidelines

1. Name and define a Subject Area according to business line. Use a work session to allow business users to discuss their processes and come up with an overall name for a grouping of enterprise entities. Good subject area names usually reflect a major business grouping of data. Occasionally, a subject area name might be the name of a particular business activity that an organization keeps information about, for instance, *Beneficiary Enrollment*.
2. *Enterprise Entities* are exclusive to a single *Subject Area.* Do not duplicate enterprise entity or subject area names.
3. All *Enterprise Entities* must have an assigned *Subject Area.*
4. *Data Stewards* should be stakeholders from all parts of the Subject Area business line. Try to assign Data Stewards from the organization unit that is as close as possible to the initial point of data capture of the most important Enterprise Entities in the *Subject Area*.
5. If a business change means that certain business functions end, determine whether Enterprise Entities can be removed from Subject Area(s) and the Enterprise Data Model.

DM OP-039 Operating Procedure for Conducting the Data Architecture Review

Rationale

The purpose of this procedure is to obtain an agreement on the sources of project entity-level data as early in the data design process as possible. Ideally, an early agreement will give all the impacted parties the time to accommodate the new data needs.

Operating Procedure

Follow the outlined agenda and exchange information as indicated.

Required Materials

1. A draft Project Logical Data Model, which, at a minimum, shows the project entities (Detailed level attributes are not required for the Architecture Review.)
2. A Data Source Plan
3. An Impact Analysis Report

Agenda

1. Introduce the project principals and briefly discuss the project's purpose.
2. Based on the Data Source Plan,
   a. review the ELDM candidate entities that are being proposed for reuse,
   b. review the non-ELDM data sources that are being proposed for reuse, and
   c. discuss/justify the need for any new entities.
3. Based on the Impact Analysis Report, discuss the pros and cons for production application systems that might be impacted by the project application's shared use of existing data structures.
4. Request timely approval of the Data Source Plan (or an alternative recommendation) from the responsible Data Architect(s) and Data Steward(s).

DM OP-040 Operating Procedure for Designating Representation Term and Data Type

Rationale
Data type and attribute names must be compatible to enhance recognition of data values.

Operating Procedure
When designating a representation term and data type, follow the guidelines below.

Guidelines
1. There may be several physical data types that could be used to represent business data values for a domain. For example, an integer data type (INTEGER, SMALLINT, TINYINT) could be used to store a domain whose valid values are integers between 1 and 20. Alternatively, a decimal data type could be used.
2. In selecting the data type, consider the way the RDBMS of choice stores values and the application processing efficiency.
3. The Representation Term and data type must not conflict; for example, an attribute ending with the representation term *Description* must not have a data type of *Decimal* assigned.
4. More examples of compatibility between Representation Term and data type are shown in the table below.

Term	Abbreviation	Definition	Example Data Types
ADDRESS	ADR	Alphanumeric data which identifies street name and number, suite number, etc. May also be used for Post Office box, email address, or Internet address (URL).	String, Varchar, Char(50), Decimal, (6,2)
AMOUNT	AMT	A number that is expressed as a currency denomination.	Currency
AUDIO	AUD	A type of Object that can store a stream of audible sounds.	String, Varchar
CODE	CD	A short character string used to replace a more descriptive value.	String, Char(50)
DATE	DT	A calendar date (month, day, and year) documenting when something happened.	Numeric
MESSAGE	MSG	A brief explanation or warning of a condition such as an edit validation error.	String, Varchar
QUANTITY	QTY	A type of numeric value that is a non-monetary measure of a dimensional extent, volume, etc.	Numeric, Integer
TIMESTAMP	TS	Date and time combined, documenting when something is recorded by a system.	Datetime

DM OP-041 Operating Procedure for Assigning Date Formats

Rationale

For date and time attributes, Data Administration has adopted the International Standard for the representation of dates and times, ISO 8601. Its full reference number is ISO 8601: 2000 (E), and its title is "Data elements and interchange formats - Information interchange - Representation of dates and times." ISO 8601 describes many date/time formats. The standard is based on the 24-hour timekeeping system. To reduce complexity, only the following date/time formats are being supported.

Operating Procedure

When assigning date formats, follow the guideline below.

Content Represented	Format	Example
Complete date	YYYY-MM-DD	2003-10-08
Year and month	YYYY-MM	2003-10
Year	YYYY	2003
Complete time	HH:MM:SS	18:33:56
Hour and minute	HH:MM	18:33
Hour	HH	18

DA has approved these formats for logical design; this guideline is not intended to constrain physical implementation.

DM OP-042 Operating Procedure for Modeling Supertypes and Subtypes

Rationale

It may be helpful to partition an entity type into subtype entities when it helps business communication to show a subtype entity with a more specialized meaning or when a subtype entity requires different integrity treatment.

Operating Procedure

When modeling supertypes and subtypes, follow the guidelines below.

Guidelines

1. A supertype or subtype is named according to entity naming standards.
2. A subtype has at least one attribute or one relationship that is not in the supertype.
3. Sometimes it is simpler to combine a supertype and subtype within an entity. However, when subtypes are represented within a single entity, the attributes specific to the subtype are defined as *optional* (nulls allowed). This permits the entity to exist when the subtype does not, as illustrated in Example 1 below.

Example 1: Combined supertype/subtype	Example 2: partitioned supertype/subtype

Employee
  Employee Number (required)
  Employee Name (required)
  Employee Job Code (required)
  Employee Street Address (required)
  Employee Temporary Type Code (required)
  Employee Temporary Company Name (optional)
	Employee
	  Employee Number (required)
	  Employee Name (required)
	  Employee Job Code (required)
	  Employee Permanent Address (required)
	Employee Temporary
	  Employee Number (required)
	  Employee Temporary Company Name (required)

4. Note the attribute *Employee Temporary Type Code*. This attribute distinguishes whether the record holds data for a temporary employee and is referred to as a *type code* or a *discriminator*. *Discriminators* are needed when subtypes are generalized into the supertype.
5. Balance the need for subtyping by looking at business rules. Create a subtype when there is a distinct need for subtype *referential integrity*; otherwise, generalize the subtype within the supertype.
6. Observe normalization rules in order that the Logical Data Model transforms appropriately into a Physical Data Model.

DM OP-043 Operating Procedure for Managing Data Values through Physical Constraints

Rationale

The integrity of data values in data resources is important in providing an organization with quality information. Database Management Systems provide various methods that can automatically check and ensure that values comply within an established range.

Operating Procedure

When managing data values through physical constraints, follow the guidelines below.

Guidelines

The following list describes types of constraints.

1. *Primary Key* Constraint: The *primary key* of a table holds a value that uniquely identifies each row in the table. It is defined when a table is created or altered. A table may contain only one *primary key*.
2. *Foreign Key* Constraint: A *foreign key* is a column or combination of columns that form a relationship between the data in two tables. The relationship is created between two tables by adding the column or columns that hold one table's primary key values to a like column or columns in the other table. The second table holds the *foreign key*.
3. *Unique* Constraint: A *unique* constraint ensures that no duplicate values are entered in specific columns that do not participate in a *primary key.*
4. *Check* Constraint: A *check* constraint enforces domain value integrity by limiting the values that are permitted in a column. Like *foreign key* constraints, *check* constraints control the values that are placed in a column. In contrast, *check* constraints limit the valid values to a *declared* list of values.
5. *Default* Constraint: The *default* constraint determines the value to be stored when no value is explicitly assigned.
6. *Null* Constraint: The *null* constraint determines whether a column must contain a value.
7. *Integrity* Constraint: *Integrity* constraints attempt to enforce data consistency and prevent accidental damage to the database during updates. Use the table below as a guideline for when to apply alternative constraint types.

Managerial Circumstance	Constraint Type			
	Assertions	Check Constraints	Foreign Key (FK) Constraints	Triggers
A stable set of permitted values		X		
Occasional change to the set of permitted values		X	x	
The data value is dependent on variable conditions.	x			x

DM OP-044 Operating Procedure for Preparing the Project Logical Data Model for Physical Design

Rationale

The logical data modeler and physical data modeler may or may not be the same individual. In either case, the logical data modeler should anticipate what the physical data modeler needs to know in order to develop an efficient physical data model and capture these suggestions as comments in the entity definition.

Operating Procedure

1. Consider the performance of record retrieval based on anticipated record volume and the caching, indexing, and clustering facilities available in the anticipated DBMS.
2. Consider the mandated enterprise data security settings and any conflicts with the proposed business data functional requirements, for example, whether data will be distributed to multiple platforms or outside of firewalls.
3. Make notes of additional concerns and insights, which can later be considered in physical data design.
4. Do not modify the project logical data model to reflect physical design compromises such as denormalization to roll down supertype attributes into subtype entities. Information pertinent to physical design considerations are to be recorded as comments.
5. Provide the first-cut physical names using the approved abbreviated terms and construction rules defined in DM OP-045 Operating Procedure for Constructing Physical Table and File Names or DM OP-046 Operating Procedure for Constructing Physical Column or Element Names.

DM OP-045 Operating Procedure for Constructing Physical Table or File Names

Rationale

Physical data names represent the technical implementation of important business concepts. Therefore, it is important that the names of physical data can be easily recognized and associated with the business concept. Physical data names representing system operational requirements are important to database administration. Therefore, it is also important that physical-only objects can be easily tracked, understood, and compared.

Operating Procedure

General Steps to Attain First Cut Physical Table or File Names:

- Step 1 (For logical/physical models the first two steps are done automatically. For physical only models they must be done manually.)
    - o Translate the logical data model to a physical data model by using CMS Standard Terms and Abbreviation List.
    - o Replace space between terms with an underscore.
    - o Analyze the length of table names and proceed to step 2 if they exceed the DBMS table name length limitation. Otherwise proceed to step 3.
- Step 2
    - o Eliminate the least important terms until the target length is attained.

- Step 3
    - o Ensure table names are unique within the physical data model.

Example	
Logical Entity Name	**Physical Table Name**
ORGANIZATION ENROLLMENT	ORG_ENRLMT
ORGANIZATION	ORG
BENEIFCIARY DUAL STATUS	BENE_DUAL_STUS

Standards

1. Physical data name lengths must comply with the name length limitation established by Central DBA standards for the target database management system (DBMS).
2. Approved acronyms must be used in place of standard abbreviations. For example, Centers for Medicare and Medicaid Services must be abbreviated as CMS, not CTR_MDCR_MDCD_SRV.

DM OP-046 Operating Procedure for Constructing Physical Column or Element Names

Rationale

Physical data names represent the technical implementation of important business concepts. Therefore, it is important that the names of physical data can be easily recognized and associated with the business concept. Physical data names representing system operational requirements are important to database administration. Therefore, it is also important that physical-only objects can be easily tracked, understood, and compared.

Operating Procedure

General Steps to Attain First Cut Physical Column or Element Names:

- Step 1 (For logical/physical models the first two steps are done automatically. For physical only models they must be done manually.)
    - Translate logical data model to physical data model by using CMS Standard Terms and Abbreviation List.
    - Replace space between terms with an underscore.
    - If the last term in the physical column or element name is an acronym, append the approved abbreviation of the representation term to the physical name. For example, translate logical attribute *Beneficiary Identification Code* to physical column *BIC*. Append abbreviation for *Code* (representation term) to the column or element name resulting in *BIC_CD*.
    - Analyze the length of column names and proceed to step 2 if they exceed the DBMS table name length limitation. Otherwise, proceed to step 3.
- Step 2
    - Eliminate the least important terms until the target length is attained. Representation terms may not be removed.
- Step 3
    - Ensure column names are unique within the corresponding table.
- Step 4
    - For an LDM/PDM, update the physical name in the PDM if the default name has been changed.
    - For separate LDM and PDM, there are two situations:
        - If no PDM has yet been created, then the use of the Erwin "Derive a new model" feature in DM OP-045 above to create an Erwin logical/physical model from the separate LDM will result in the physical view of this Erwin logical/physical model having names that can serve as the first-cut column names.
        - If the project team has already produced a separate PDM, then there is no need for the Central DA to provide first-cut physical column names.
    - For separate LDM and PDM, update Physical Column Name UDP for attribute in LDM.

Example	
Logical Attribute Name	**Physical Column Name**
ORGANIZATION ENROLLMENT DATE	ORG_ENRLMT_DT
PERSON LAST NAME	PERSN_LAST_NAME
PROVIDER NATIONAL PROVIDER IDENTIFIER	PRVDR_NPI_ID

Standards

1. Physical data name lengths must comply with the name length limitation established by Central DBA standards for the target database management system (DBMS).
2. Approved acronyms must be used in place of standard abbreviations. For example, Centers for Medicare and Medicaid Services must be abbreviated as CMS, not CTR_MDCR_MDCD_SRV.
3. Every column and element name must have a representation term.

DM OP-047 Operating Procedure for Assigning Data Planning Participants

Rationale
Business management and IT management must work together to create an Enterprise Architecture (EA) for CMS business functions.

Operating Procedure
Assign CMS personnel and responsibilities as described below.

Guidelines
1. **CMS Enterprise Architect (FEA liaison)**: maintains the enterprise architecture to ensure that business processes and information systems are aligned
2. **Enterprise Data Architect**: identifies data design and integration issues
3. **Enterprise Process Architect**: identifies process design and integration issues
4. **Business Owners/Partners**: provide business information requirements and security category specifications
5. **Data Stewards**: ensure that all processes enforce data quality

DM OP-048 Operating Procedure for Conducting Enterprise Data work sessions

Rationale
This is a procedure for conducting Information Technology Systems Planning work sessions related to Enterprise Data Architecture.

Operating Procedure
Follow the outlined agenda and exchange information as indicated in the guidelines below.

Guidelines
1. Schedule design sessions with Business Intelligence Council and meeting facility.
2. Conduct orientation and training for design session participants.
3. Prepare the materials, room, and software aids.
4. Customize the design session agenda.
5. Conduct the kickoff meeting.
6. Agenda
    a. Introduce BIC team members and roles.
    b. Designate the BIC Chairperson.
    c. Establish the need for alignment between business functions with data architecture.
    d. Select team members for subsequent detailed definition component and break-out sessions.
    e. Explain how the EA project fits in the strategic direction of the agency.
    f. Identify level of service needed and available funding sources.
    g. Explain the definition design process.
    h. Confirm the general requirements for the business functions areas affected, including the scope, goals and objectives.
    i. List the main processes within each affected business function area.
    j. Develop blueprints for and descriptions of the process and enterprise entities in use (the as-is).
    k. Identify the information needed for supporting the processes and depict the enterprise entities in a logical data model (the to-be).
    l. Document expected individual contributions.
    m. Identify schedule requirements.
    n. Estimate the timeframes for follow-up responses.
    o. Publish work session minutes.

DM OP-049 Operating Procedure for Creating a Business Process Diagram

Rationale
This is a procedure for development of a Business Process Diagram.

Operating Procedure
1. Define the process scope.
   a. Describe what is being done.
   b. Tie the process to an organization and business application.
2. Set the level of detail.
   a. Scope the diagram to represent a major business process at *Level 0.* (This is equivalent to a FEA-*BRM* chartered function such as *Maintain Medicare Beneficiary Enrollment*.)
   b. Decompose the *Level 0* process to 8-10 *Level 1* sub-processes.
   c. If needed to show all business recognized tasks, decompose respective *Level 1* sub-processes to *Level 2* sub-processes.
3. Present the processes in graphical format.
   a. It is not necessary to show sequence.
   b. It is not necessary to express conditions that initiate a process.
   c. Show organization interaction (who does what).
   d. Show information exchanges (what is needed to do this).
   e. Identify the FEA-BRM alignment.

Data Management Guidelines

REMOVED: DM G-001 Guideline for Assigning Data Analysts

DM G-001 has been replaced by DM OP-036 Operating Procedure for Assigning Data Analysts.

REMOVED: DM G-002 Guideline for Conducting the Logical Data Design Kickoff Meeting

DM G-002 has been replaced by DM OP-037 Operating Procedure for Conducting the Logical Data Design Kickoff Meeting.

REMOVED: DM G-003 Guideline for Monitoring Schedule and Cost of Data Services

DM G-003 has been deleted.

REMOVED: DM G-004 Guideline for Designating Representation Term and Data Type

DM G-004 has been replaced by DM OP-040 Operating Procedure for Designating Representation Term and Data Type.

REMOVED: DM G-005 Guideline for Constructing Physical Data Names

DM G-005 has been merged into DM OP-045 Operating Procedure for Constructing Physical Table or File Names and DM OP-046 Operating Procedure for Constructing Physical Column or Element Names.

REMOVED: DM G-006 Guideline for Assigning Date Formats

DM G-006 has been replaced by DM OP-041 Operating Procedure for Assigning Date Formats.

REMOVED: M G-007 Guideline for Conducting the Data Architecture Review

DM G-007 has been replaced by DM OP-039 Operating Procedure for Conducting the Data Architecture Review.

REMOVED: DM G-008 Guideline for Mapping Data Requirement Sources

DM G-008 has been deleted. It is covered in DM OP-031 Operating Procedure for Capturing the Standard Logical Data Model Metadata.

REMOVED: DM G-009 Guideline for Preparing the Project Logical Data Model for Physical Design

DM G-009 has been replaced by DM OP-044 Operating Procedure for Preparing the Logical Data Model for Physical Design.

REMOVED: DM G-010 Guideline for Constructing Physical Table or Files Names

DM G-010 has been replaced by DM OP-045 Operating Procedure for Constructing Physical Table or File Names.

REMOVED: DM G-011 Guideline for Constructing Physical Column or Element Names

DM G-011 has been replaced by DM OP-046 Operating Procedure for Constructing Physical Column or Element Names.

REMOVED: DM G-012 Guideline for Assigning Data Planning Participants

DM G-012 has been replaced with DM OP-047 Operating Procedure for Assigning Data Planning Participants.

REMOVED: DM G-013 Guideline for Conducting Enterprise Data work sessions

DM G-013 has been replaced with DM OP-048 Operating Procedures for Conducting Enterprise Data Work Sessions.

REMOVED: DM G-014 Guideline for Creating a Business Process Diagram

DM G-014 has been replaced by DM OP-049 Operating Procedures for Creating a Business Process Diagram.

REMOVED: DM G-015 Guideline for Defining a Subject Area

DM G-015 has been replaced with DM OP-038 Operating Procedures for Defining a Subject Area.

REMOVED: DM G-016A Guideline for Analyzing and Approving Standard Terms

DM G-016A has been deleted. It was merged with OP-035 Operating Procedure for Creating a New Standard Term.

REMOVED: DM G-016B Guideline for Abbreviating a Standard Term

DM G-016B has been deleted. It has been merged with OP-035 Operating Procedure for Creating a New Standard Term.

REMOVED: DM G-017 Guidelines for Reference Tables

DM G-017 has been deleted. It was merged with DM OP-018 Operating Procedure for Documenting Domain Value Rules.

REMOVED: DM G-018 Guideline for Managing Data Values through Physical Constraints

DM G-018 has been replaced with OP-043 Operating Procedures for Managing Data Values through Physical Constraints.

REMOVED: DM G-019 Guideline for Modeling Supertypes and Subtypes

DM G-019 has been replaced with DM OP-042 Operating Procedure for Modeling Supertypes and Subtypes.

REMOVED: DM G-020 Guideline for Using Erwin Complete Compare to Create a New Data Model

DM G-020 has been deleted. It is covered by the Data Model Tool Use Procedure document. Standards were moved to DM OP-005 Operating Procedure for Developing the Logical Data Model.

REMOVED: DM G-021 Guideline for Naming Reference Entities

DM G-021 has been deleted. It was merged with DM OP-009 Operating Procedure for Naming Data Entities.