|  |
| --- |
| Florida department of transportation |
| Data / Database Standards |
| Naming and Validation Standards |
|  |
| **FDOT Data Administration** |
| **7/7/2014** |

|  |
| --- |
| Document the naming standards used in Florida Department of Transportation (FDOT) data models and databases. |

Contents

[Purpose 3](#_Toc393051232)

[Global Rules 3](#_Toc393051233)

[Data Architecture 3](#_Toc393051234)

[NAMING STANDARDS 3](#_Toc393051235)

[1. Data Model 3](#_Toc393051236)

[2. Attribute and Column names 3](#_Toc393051237)

[Naming Rules 5](#_Toc393051238)

[3. Naming Component Notation 5](#_Toc393051239)

[4. Artifact Type Codes 6](#_Toc393051240)

[5. Global Rules 7](#_Toc393051241)

[6. Base and Standard Names for Artifacts 7](#_Toc393051242)

[7. Database Name 8](#_Toc393051243)

[8. Key and Index Name 8](#_Toc393051244)

[9. Referential Integrity Rule and Foreign Key Name 9](#_Toc393051245)

[10. FOCUS and EDA Master and Access Files 10](#_Toc393051246)

[11. SAS View/Access File 10](#_Toc393051247)

[validation standards 12](#_Toc393051248)

[12. Model 12](#_Toc393051249)

[13. Data Confidentiality and Sensitivity 13](#_Toc393051250)

[14. Database 14](#_Toc393051251)

[15. Entity and Table 15](#_Toc393051252)

[16. Table Constraint 15](#_Toc393051253)

[17. Overlap between a Model Table and an External Table 16](#_Toc393051254)

[18. Attribute and Column 16](#_Toc393051255)

[19. Attribute and Column Domain, Data Type and Data Class 17](#_Toc393051256)

[20. Attribute and Column Constraint 18](#_Toc393051257)

[21. Key and Index 18](#_Toc393051258)

[22. Relationship 18](#_Toc393051259)

[23. Function 19](#_Toc393051260)

[24. Sequence 19](#_Toc393051261)

[25. Stored Procedure 19](#_Toc393051262)

[26. Tablespace and Filegroup 19](#_Toc393051263)

[27. Trigger 20](#_Toc393051264)

[28. View 20](#_Toc393051265)

[CLASS WORD DOMAIN DATA TYPE CROSSWALK 21](#_Toc393051266)

[Change History 25](#_Toc393051267)

# Purpose

This standard documents the data and database design standards used by the Florida Department of Transportation (FDOT). This procedure does not address the full capability of any of the DBMS’ supported by the FDOTdatabase management team. This document speaks to the constraints FDOT places upon the features and functions of each DBMS in order to provide optimal, efficient, and consistent support for all DBMS’ used in the FDOT environments.

This standard documents the validation rules applied to the construction of data models for databases created and maintained within the FDOT infrastructure.

# Global Rules

All exceptions to any standard herein must be submitted the the FDOT Data Administration Team for consideration and approval or rejection.

# Data Architecture

The FDOT Data Architecture requires the use of the following components:

1. The CA-ERwin Data Modeling Tool for logical and physical data model design,
2. The FDOT Meta Data Repository Glossary for abbreviation of business or logical names,
3. The DOTCODES reference data repository of code lists, values and descriptions,
4. The Enterprise Document Management System (EDMS) for storage of all electronic documents.
5. Existing logical and physical model and database information which already exists in FDOT databases as the Department’s source of record for this information:
	1. Staff Repository System of all internal and external staff,
	2. Transportation Vendor Information system of all FDOT Vendors,
	3. Organization Codes for the FDOT organizational units,
	4. Work Program and Financial Projects for the FDOT construction projects,
	5. Contract Information for the FDOT Contracts.

The number and scope of utilized components in a data and database design depends upon the information required by the application system.

# NAMING STANDARDS

## Data Model

The data model must have a name.

## Attribute and Column names

**Attribute Names (logical)**

* 1. Name words or abbreviations are ordered from left to right:
		1. The prime word for the object being referenced.
		2. The modifier words (as needed).
		3. The class word.
	2. Attribute names must be limited to no more than 32 characters.
	3. Attribute name words must be separated by a space character.

**Column Names (physical)**

* 1. Column names must be based on approved abbreviations for the words used to create the attribute name.
	2. The order of abbreviations in a column name must be the same as the order of words in the attribute name.
	3. Attribute words must be translated into column name abbreviations using the meta data repository glossary provided with the ERwin starter model.
	4. Column names must be limited to no more than 18 characters.
	5. Column name abbreviations must be separated by an underscore.
	6. The maximum length of a column name of data type of VARCHAR or VARCHAR2 is 17 characters when z/OS COBOL Copybooks are needed.
	7. Existing column names must be reused.

# Naming Rules

## Naming Component Notation

The notation symbols for each component of an artifact name are identified in the table below:

|  |  |
| --- | --- |
| Notation Symbol | Component Name |
| [a] | Artifact Type, e.g. table, view, tablespace, index, … |
| [c] | Content grouping. Default is the application acronym. |
| [e] | Environment Type. e.g. DB2, Oracle, Microsoft SQL Server |
| [l] | Lifecycle identifier or sub-grouping of the content grouping component. |
| [n] | A number in the range 001 - 999 assigned to an artifact. This number may or may not be sequentially assigned depending on the artifact type. |
| [s] | A sequentially assigned number in the range 01 - 99 or 001 - 999 used to provide uniqueness to the artifact name. |
| [t] | View/Access type for SAS. |

## Artifact Type Codes

The complete list of artifact type codes in order by type code value.

|  |  |  |
| --- | --- | --- |
| Type Code | Name | Maximum Length |
| -- | Attribute\* | 32 |
| -- | Column\* | 18 |
| -- | Database | 8 |
| -- | Entity | 32 |
| -- | FOCUS Master\*\* | 8 |
| CK | Check Constraint | 18 |
| D | Default Value Constraint | 18 |
| F | Flat, Fixed or Standard File | 40\*\*\* |
| G | Trigger | 18 |
| I | Oracle Index Tablespace | 8 |
| J | Multi-table Joined Master | 8 |
| K | Package | 18 |
| M | Materialized or Cached View | 18 |
| N | Function | 18 |
| P | Plan (DB2) | 18 |
| Q | Sequence | 18 |
| R | Procedure | 18 |
| T | Table | 18 |
| TS | Tablespace | 8 |
| V | View | 18 |
| X | Index | 8 |

\* See the **Attribute, Column and Element** **Naming** section for Attribute and Column naming.

\*\* See the **FOCUS and EDA Master/Access File** section for FOCUS Master naming.

\*\*\* z/Enterprise Server flat file names may be up to 5 nodes of 8 characters each. Each node is separated by a “.” character.

## Global Rules

* 1. When the same name exists in multiple schema, it must represent the same thing.
	2. The content grouping part of each name must be the application acronym assigned to the model.
	3. Artifact rules must override global rules where there is overlap or where the global rule is restated for the artifact.
	4. Object names for this standard must contain only alphabetic characters, numeric characters and embedded underscores.
	5. All names must be in uppercase.

## Base and Standard Names for Artifacts

The standard naming format is used for the majority of artifacts for which a naming standard is prescribed.

For those instances where an artifact naming standard is prescribed but compliance not clearly possible, an exception to standard must be submitted to the FDOT Data Administration team for review and acceptance or rejection.

* 1. A standard name is composed of a content group, artifact type, number and additional descriptive text up to the maximum character length specified for the artifact type. The format is: [ccc] [a] [nnn] [\_] [xxx].
	2. Tables
		1. If a table exists across multiple platforms and databases, it must have the same name and represent the same data in all databases where it is found.
		2. All tables having the same name must represent the same data regardless of database where the table name is found.
		3. For DB2: the content group, [ccc], and number, [nnn], components of the name must be the same for the table and the tablespace associated to that table.
	3. Views
		1. If a view exists across multiple platforms and databases it must have the same name and represent the same view of the same data in all databases where it is found.
		2. All views having the same name must represent the same view of the same data regardless of database where the view name is found.
	4. Packages
		1. The Oracle package specification and the package body have the same name.

Reference book***: “Oracle Database 10g, The Complete Reference*** “ by **Kevin Loney**, pg 168.

* 1. Sequences
		1. A sequence object must be used for one and only one table.
		2. The number component for the sequence name must be the same as the number component of the table name to which the sequence object is associated.

## Database Name

Database names are assigned by the FDOT Physical DBA (DBAT) group.

* 1. Each database name must be unique to a DBMS.
	2. Each database name must be unique within the FDOT computing environment.
	3. Each database name must be eight (8) characters long.
	4. DB2 Database Name
		1. A DB2 database name is formatted as [ccc] + “DB2” + [nn].
		2. The number component of the name must be a two (2) character value assigned by the FDOT Physical DBA Group.
	5. SQL Server Database Name
		1. A SQL Server database name is formatted as [ccc] + “SQL” + [s] where [s] is a sequentially assigned number.
		2. The sequence component of the name must be a sequentially assigned number in the range 1 - 9.
	6. Oracle Database Name
		1. An Oracle database name is formatted [cc] [l] [s] where [s] is a sequentially assigned number.
		2. The content group component of the name represents the district identifier for the database.
		3. The values for the life cycle, or sub-grouping, component of the name are in the table below:

|  |  |
| --- | --- |
| Sub-group | Name |
| PRD | Production |
| TST | System Test |
| UT | Unit Test |

* + 1. The sequence component of the name is a two (2) digit sequentially assigned number in the range 01 - 99.

## Key and Index Name

* 1. The key name and the index name must be the same name.
	2. An index name must be eight (8) characters long and formatted as [ccc] [a] [nnn] [s].
	3. The artifact type value must be “X”.
	4. The number component of the name must be a three (3) digit number in the range 001 – 999.
	5. The value for the number component of the name must have the same value as the number component of the table name to which the index is associated.
	6. The value for sequence component of the name must be “1” for the primary key index of a table.
	7. The value for the sequence component of the name must be assigned sequentially from the range 2-9 and then A-Z for all indexes which are not the primary key index for the table.
	8. The table columns which compose the index members must be the same in all databases where it is found.
	9. The order of the table columns which compose the index members must be the same for the table index in all databases where it is found.
	10. Each index for a table must represent the same access path in all schema where the table index is implemented.

## Referential Integrity Rule and Foreign Key Name

The referential integrity rule name is assigned to the entity and the foreign key name is assigned to the table associated to that entity.

* 1. A referential integrity rule name must be formatted as [ccc] [nnn] [ss].
	2. The referential integrity rule name and the foreign key name must be the same name.
	3. The number component of the relationship name must have the same value as the number component of the table name which is the parent of the relationship.
	4. The sequence component of the relationship name must be a two (2) digit sequentially assigned number in the range 01 - 99.
	5. Each referential integrity rule must have the same parent entity and the same child entity in all schema where the name is found.
	6. Each referential integrity rule must represent the same business relationship or business rule in all schema where the name is found.

## FOCUS and EDA Master and Access Files

FDOT Data Administration is responsible for the generation and documentation of FOCUS and EDA masters.

* 1. FOCUS and EDA master and access file names must be eight (8) characters long and formatted as [ccc] [e] [a] [nnn].
	2. The environment type component of the name is defined in the table below:

|  |  |
| --- | --- |
| Type Code | Name |
| D | DB2 |
| F | Flat, Fixed or Standard File |
| O | Oracle |

* 1. The artifact type component of the name is defined in the table below:

|  |  |
| --- | --- |
| Artifact Type Code | Artifact Type Name |
| F | Flat, Fixed or Standard File |
| J | Multi-table Joined Master |
| T | Table |
| V | View |

* 1. The number component of the name must be a three (3) digit number in the range 001 – 999.
	2. A FOCUS or EDA master must have the same number component value as the underlying object type to which it is associated, unless that number produces a duplicate name.
	3. A FOCUS or EDA master must be created for one, and only one, object.
	4. Each FOCUS or EDA master with the same name must be associated with the same underlying object in all database platforms and environments against which the master is intended to execute.

## SAS View/Access File

FDOT Data Administration is not currently creating new SAS Views. The recommended option to replace the SAS View is to use SQL Pass-Through.

* 1. A SAS view name is DB2 specific. It is eight (8) characters long formatted as [ccc] [t] [a] [nnn].
	2. The access type is defined in the table below:

|  |  |
| --- | --- |
| Type Code | Name |
| A | Access Descriptor |
| V | View Descriptor |

* 1. The artifact type is defined in the table below:

|  |  |
| --- | --- |
| Type Code | Name |
| T | Table |
| V | View |

* 1. The number component of the name must be a three (3) digit number in the range 001 – 999.
	2. Each artifact in this group must represent the same thing in all databases where the name is found.

# validation standards

 All Data models developed and maintained by, and for, FDOT must use the ERwin data modeling tool.

## Model

* 1. The model must have a model name. The components for the name are:
		1. The application acronym for the model.
		2. Model Type Abbreviation.

|  |  |
| --- | --- |
| Model Type | Model Type Abbreviation |
| Combined Logical and Physical Model | CMB |
| Conceptual Model | CON |
| Dimensional Model | DIM |
| Logical Only Model | LOG |
| Physical Only Model | PHY |
| Reverse Engineered Model (Physical) | REV |

* + 1. DBMS Acronym.

|  |  |
| --- | --- |
| Database Management System | DBMS Abbreviation |
| IBM z/Os DB2 | DB2 |
| Oracle  | ORA |
| Microsoft SQL Server | SQL |

* + 1. The model must include the application acronym, for object naming, in the model level physical view user defined property named ‘Acronym’.
	1. The model must include the author and/or company preparing the model.
	2. The model must include a definition or description of the model, including the purpose for creating the model.
	3. The model must use the Information Engineering notation option for the logical model view.
	4. The model must use the Information Engineering notation option for the physical model view.
	5. The model must use the Dimensional Modeling notation option for the dimensional or data warehouse physical model view.
	6. The model must enable the Data Movement option in the model properties.



* 1. Object names must be unique within the scope of the object type.
	2. The model name must comply with the FDOT naming standards for model and database objects.
	3. All database objects and object properties must conform to DBMS limits for the chosen DBMS.
	4. The model must not contain hidden objects.
	5. All database designs must be in at least 3rd Normal Form.
	6. All names must be in upper case only.

## Data Confidentiality and Sensitivity

FDOT Data Administration is required to document confidentiality and sensitivity information for data stored in FDOT databases. Currently this applies to tables, columns, views and view columns. The application and data owners are the best source for specifying data sensitivity designations, and should provide citations of the Federal Law, Florida Statute, Florida Administrative Rule, Department Policy and/or Department Procedure which govern any data confidentiality or sensitivity designation other than Public Access.

* 1. The Basis for Protection must be one of the following values:

|  |  |
| --- | --- |
| Basis for Protection Value | Description |
| DOT | Departmental Policy or Procedure |
| FED | Federal Law |
| FS | Florida Statute Other Than Chapter 119 |
| FSE | Florida Statute Public Records Exemption (Chapter 119) |
| N/A | Not Analyzed or Identified (Default Value) |
| PUB | Public Access |

* 1. The Degree of Protection must be one of the following values:

|  |  |
| --- | --- |
| Degree of Protection Value | Description |
| CND | Conditionally Confidential, Sometimes Public Records Exempt |
| CON | Confidential, Always Public Records Exempt |
| N/A | Not Analyzed or Identified (Default Value) |
| NGA | Not Generally Accessible |
| PUB | Public Access |
| TIM | Confidential For Specified Time Period and then Public Records Available |

The Sensitivity Date must be a valid date.

* 1. Data sensitivity explanation or reference must contain a citation of the specific statute, policy, procedure or administrative rule governing confidentiality *when* either the value for basis for protection or the value for degree of protection is not “PUB” for public.
	2. Data confidentiality and sensitivity information must be provided for:
		+ - Tables
			- Table Columns
			- Regular, Cached and materialized Views
			- View Columns

**

## Database

* 1. Each generated database within the data model must be associated to at least one table.

## Entity and Table

* 1. All entities must have a definition.
	2. The order of attributes in the entity must match the order of the columns in the associated table(s).
	3. If design layering is used, then the attribute order and the column order must be the same for all tables generated from the entity.
	4. Key unification must not occur in a physical schema, and must be resolved no later than the physical design step of the database design.

**

* 1. All entities must have at least one attribute.
	2. All tables must have at least one column.
	3. An entity or table must have at least one relationship.
	4. All entities and tables must contain populated volumetric information.
	5. The volumetric parameters for a table must not contain a zero (0).
	6. All entities and tables must contain primary key attributes/columns.
	7. All entities and tables must comply with the entity and table naming standards.
	8. All tables which will exist in a Gen model must contain at least one (1) non-key column.
	9. Entity attributes and table columns must be ordered as follows:
		1. Required attributes in order of size from small to large.
		2. Required foreign keys.
		3. Optional foreign keys.
		4. Optional attributes in order of size from small to large.

## Table Constraint

* 1. Constraints must use SQL syntax as defined by the SQL92 and newer standards.
	2. Each table constraint must comply with the constraint naming standard.

## Overlap between a Model Table and an External Table

When data defined in a model table duplicates, replicates, or overlaps with an existing source, the existing source data must be fetched into the model.

* 1. When the relationship between the external table and the model table will be implemented:
		1. The external table must be included in the model as a model entity.
		2. The external table must be marked as ‘Do Not Generate’ so they do not participate in DDL generation.
		3. The relationship between the external table and the model table must not be marked as ‘Do Not Generate’.
		4. The external table must be the parent in the relationship.
		5. The model table must be the child in the relationship.
	2. When the relationship between the external table and the model table will not be implemented:
		1. The external table must be included into the model as a data source table.
		2. Each overlapping external table column must be documented as a data source column for the model table column.
	3. The model must not extend an external table into a model table.

When the model table is extending the external table:

* + 1. The relationship between the external table and the model table must be One-to-One.
		2. The relationship between the external table and the model table must use the process/steps in 17.1 of this section when the relationship will be implemented.
		3. The relationship between the external table and the model table must use the process/steps defined in 17.2 of this section when the relationship will not be implemented.
	1. The data sensitivity for the key columns in the model table must be the same as the data sensitivity of the same columns in the external table where they overlap.
	2. The model must not overlap non-key data between an external table and a model table.

## Attribute and Column

* 1. Each attribute must have a definition.
	2. Each attribute name must comply with the naming standard for attributes.
	3. Each column name must comply with the naming standard for columns.
	4. All Attributes and columns must comply with the data definition standards.
	5. Each foreign key (child) column data type must be the same as the contributing parent column data type.
	6. The length of the attribute name must not exceed 32 characters.
	7. The length of the column name must not exceed 18 characters.
	8. The length of the column name must not exceed 17 characters if the column is to be included in a z/OS COBOL copybook and the data type is VARCHAR or VARCHAR 2 in the database.
	9. DB2 and Oracle table columns must not be designated as identity columns.
	10. The class word or class abbreviation for an attribute or column name must be found in the FDOT Class Word Crosswalk document.
	11. Each class abbreviation in the column name must correctly correspond to the class word or class abbreviation in the attribute name.
	12. Each attribute data type must be compatible with the corresponding column data type.
	13. Each word or abbreviation in an attribute or column name must be found in the glossary.
	14. Each non-standard abbreviation must be approved by FDOT Data Administration.
	15. Each column name must not contain a Null Value Constraint name.
	16. Foreign Key Migration
		1. The child attribute name representing a foreign key must be the same as the attribute name of the corresponding parent primary or alternate key attribute.
		2. The child column name representing a foreign key must be the same as the column name of the corresponding parent primary or alternate key column.
	17. Each attribute and column must have the same data type definition for all occurrences of the attribute or column in the model.
	18. The use of varchar(max) for a SQL Server for a column data type is not allowed.

## Attribute and Column Domain, Data Type and Data Class

FDOT does not allow all possible domains and data types. We concentrate on those that provide the most benefit and can be consistently implemented across all of our DBMSs. Refer to the Class Word / Domain / Data Type Crosswalk standards document for valid combinations of domain, data type and class word.

* 1. Each attribute and column data type must be valid for the attribute and column class word or class abbreviation.
	2. Each attribute and column data type must be valid for the designated DBMS chosen for the model.
	3. Each attribute and column data type must be valid for the attribute domain.
	4. Each attribute and column data type must be an allowable data type for the selected domain.
	5. Each attribute and column must be associated to a currently approved domain.
	6. Each attribute and column data type must be a preferred data type for the selected domain. A written justification must be submitted to the FDOT Data Administration group for an exception to this standard.
	7. Each attribute and column must be associated to the same domain for all occurrences of the attribute and column in the model.

## Attribute and Column Constraint

* 1. Each column constraint must be compliant to the SQL92 format standard.
	2. Each column constraint must comply with the constraint naming standard.
	3. The constraint name for the attribute must be the same as the constraint name for the column the attribute is associated to.

## Key and Index

* 1. Each key attribute must be an attribute in the model.
	2. Each index column must be a column in the model.
	3. Each key must contain at least one attribute.
	4. Each index must contain at least one column.
	5. Each entity must contain a primary key.
	6. Each table must contain a primary key index.
	7. Each table must contain a primary key constraint.
	8. Each foreign key must contain a foreign key constraint.
	9. Overlapping indexes must not be generated.
	10. The primary key member order must match the order of the attributes in the entity.
	11. The primary key index column order must match the order of the columns in the table.
	12. The key member order and index column order for foreign keys must match the order in the entity and table contributing the foreign key.
	13. Each index must be associated to a tablespace or filegroup.
	14. Each index must comply with the index naming standard.

## Relationship

* 1. Each identifying relationship must not allow nulls in the child foreign key column(s).
	2. Each relationship must have valid parent and child entities and tables.
	3. Each relationship foreign key name must comply with the relationship naming standard.
	4. A many-to-many relationship must not be included in either a logical model or a physical model.
	5. Assertion
		1. Each relationship must have a parent-to-child phrase describing the business rule (an assertion) represented by the relationship.
		2. Each relationship must have a child-to-parent phrase describing the business rule (an assertion) represented by the relationship.
	6. The parent or child phrase length must not exceed 32 characters in length for Gen.
	7. No relationship can be marked as logical only or physical only.

## Function

* 1. Each function name must comply with the function naming standard.

## Sequence

* 1. Each sequence name must comply with the sequence naming standard.

## Stored Procedure

* 1. Each stored procedure must have a definition to describe its purpose.
	2. Each stored procedure must be associated to at least one database object present in the model.
	3. Each implemented stored procedure must have the procedure code included in the model.
	4. Each stored procedure name must comply with the stored procedure naming standard.

## Tablespace and Filegroup

* 1. Each tablespace name must comply with the tablespace naming standard.
	2. DB2
		1. Each physical design must contain at least one tablespace object.
		2. Each tablespace must be associated to a single DB2 table.
		3. Each tablespace must contain allocation values.
	3. Oracle
		1. Each physical design must contain at least two tablespaces.
		2. The design must contain a tablespace for data.
		3. The design must associate each table to the data tablespace.
		4. The design must contain a tablespace for indexes.
		5. The design must associate each index to the index tablespace.
		6. Each Oracle tablespace must be associated to an Oracle Physical Storage object.
		7. Each Oracle tablespace must contain allocation values.
		8. Each Oracle tablespace must specify an extent size value.
	4. SQL Server
		1. Each physical design must contain at least two storage files.
		2. The design must have a primary data file in the primary filegroup.
		3. The design must associate each table to the primary data file.
		4. The design must have an index file in a secondary filegroup.
		5. The design must associate each index to the index file.

## Trigger

* 1. Each trigger name must comply with the trigger naming standard.

## View

* 1. Each view must have a definition.
	2. Each view name must comply with the view naming standard.
	3. Each view must have at least one (1) column.
	4. Each view must have SQL to define it whether it is generated by the modeling tool or is supplied as user defined SQL.

# CLASS WORD DOMAIN DATA TYPE CROSSWALK

|

|  |
| --- |
| **Class Word** |

 |

|  |
| --- |
| **Abbrev** |

 | **Domain Key** | **Default Data Type** | **Default Length** | **DB2** | **Oracle** | **SQL SRV** | **Flat File** |
| AMOUNT | AMT | MONEY | DECIMAL | 15.2 | DECIMAL | NUMBER | MONEY | NUMERIC |
|  |  |  |  |  |  |  | DECIMAL | DECIMAL |
|  |  |  |  |  |  |  |   |  |
| AREA | AREA | DECIMAL | DECIMAL | 7.2 | DECIMAL | NUMBER | DECIMAL | NUMERIC |
|  |  |  |  |  |  |  |   | DECIMAL |
|   |   |   |  |  |  |  |   |  |
| BINARY | BIN | BINARY | BLOB | 0 | BLOB | BLOB | BLOB |   |
|  |  |  |  |  |  |  |   |  |
| CIRCUMFERENCE | CRCM | DECIMAL | DECIMAL | 7.2 | DECIMAL | NUMBER | DECIMAL | NUMERIC |
|  |  |  |  |  |  |  |   | DECIMAL |
|   |  | INTEGER | INTEGER | 0 | INTEGER | NUMBER | INTEGER (INT) | NUMERIC |
|  |  |  |  |  | SMALLINT |  |  |  |
|  |  |  |  |  |  |  |  |  |
| CODE | CD | CODE DOT | CHAR | 8 \* | CHAR | CHAR | CHAR | CHAR |
|  |  |  |  |  |  |  |  |  |
|  |  | CODE TBL | CHAR | 8 | CHAR | VARCHAR2 | VARCHAR | CHAR |
|  |  |  |  |  |  | CHAR | CHAR |  |
|  |  |  |  |  |  |  |  |  |
| COMMENT | CM | TEXT | VARCHAR | 255 | VARCHAR | VARCHAR2 | VARCHAR | CHAR |
|  |  |  |  |  | <255, CHAR | CLOB |   |  |
|  |  |  |  |  | CLOB |  |  |  |
|  |  |  |  |  |  |  |  |  |
| COUNT | CNT | INTEGER | INTEGER | 0 | INTEGER | NUMBER | INTEGER (INT) | NUMERIC |
|  |  |  |  |  | SMALLINT |  |  |  |
|  |  |  |  |  |  |  |   |  |
| DATE | DT | DATE | DATE | 0 | DATE | DATE | DATE | CHAR |
|  |  |   |   |   | CHAR | CHAR | CHAR |   |
|  |  |  |  |  |  | VARCHAR2 | VARCHAR |  |
|  |  |  |  |  |  |  |   |  |
| DEPTH | DPTH | DECIMAL | DECIMAL | 7.2 | DECIMAL | NUMBER | DECIMAL | NUMERIC |
|  |  |  |  |  |  |  |   | DECIMAL |
|  |  |  |  |  |  |  |   |  |
|  |  |  |  |  |  |  |  |  |
| DESCRIPTION | DS | DESC DOT | CHAR | 30 \* | CHAR | CHAR | CHAR | CHAR |
|  |  | DESC TBL | CHAR | 30 | CHAR | VARCHAR2 | VARCHAR | CHAR |
|  |  |   |   |   | VARCHAR | CHAR | CHAR |   |
|  |  |  |  |  |  |  |  |  |
|  |  | TEXT | VARCHAR | 255 | VARCHAR | VARCHAR2 | VARCHAR | CHAR |
|  |  |  |  |  | <255, CHAR | CLOB |  |  |
|  |  |  |  |  | CLOB |  |  |  |
|  |  |  |  |  |  |  |  |  |
| DIAMETER | DIAM | DECIMAL | DECIMAL | 7.2 | DECIMAL | NUMBER | DECIMAL | NUMERIC |
|  |  |  |  |  |  |  |   | DECIMAL |
|   |  | INTEGER | INTEGER | 0 | INTEGER | NUMBER | INTEGER (INT) | NUMERIC |
|  |  |  |  |  | SMALLINT |  |  |  |
|  |  |  |  |  |  |  |   |  |
| DISTANCE | DSTNC | DECIMAL | DECIMAL | 7.2 | DECIMAL | NUMBER | DECIMAL | NUMERIC |
|  |  |  |  |  |  |  |   | DECIMAL |
|   |  | INTEGER | INTEGER | 0 | INTEGER | NUMBER | INTEGER (INT) | NUMERIC |
|  |  |  |  |  | SMALLINT |  |  |  |
|  |  |  |  |  |  |  |   |  |
| HEIGHT | HT | DECIMAL | DECIMAL | 7.2 | DECIMAL | NUMBER | DECIMAL | NUMERIC |
|  |  |  |  |  |  |  |   | DECIMAL |
|   |  | INTEGER | INTEGER | 0 | INTEGER | NUMBER | INTEGER (INT) | NUMERIC |
|  |  |  |  |  | SMALLINT |  |  |  |
|  |  |  |  |  |  |  |   |  |
|  |  |  |  |  |  |  |   |  |
| IDENTIFIER | ID | ID | CHAR | 8 | CHAR | VARCHAR2 | VARCHAR | CHAR |
|  |  |  |   |   | INTEGER | NUMBER | INTEGER (INT) | NUMERIC |
|  |  |  |  |  |  |  | CHAR |  |
|  |  |  |  |  |  |  |  |  |
| LENGTH | LNGTH | DECIMAL | DECIMAL | 7.2 | DECIMAL | NUMBER | DECIMAL | NUMERIC |
|  |  |  |  |  |  |  |   | DECIMAL |
|   |  | INTEGER | INTEGER | 0 | INTEGER | NUMBER | INTEGER & (INT) | NUMERIC |
|  |  |  |  |  | SMALLINT |  |  |  |
|  |  |  |  |  |  |  |   |  |
| NAME | NM | TEXT | VARCHAR | 255 | VARCHAR | VARCHAR2 | VARCHAR | CHAR |
|  |  |  |  |  | <255 CHAR | CLOB |  |   |
|  |  |  |  |  | CLOB |  |  |  |
|  |  |  |  |  |  |  |   |  |
| NUMBER | NUM | DECIMAL | DECIMAL | 7.2 | DECIMAL | NUMBER | DECIMAL | NUMERIC |
|  |  |  |  |  |  |  |   | DECIMAL |
|  |  | INTEGER | INTEGER | 0 | INTEGER | NUMBER | INTEGER (INT) | NUMERIC |
|  |  |  |  |  | SMALLINT |  |  |  |
|  |  |  |  |  |  |  |   |  |
| PERCENT | PCT | DECIMAL | DECIMAL | 7.2 | DECIMAL | NUMBER | DECIMAL | NUMERIC |
|  |  |  |  |  |  |  |   | DECIMAL |
|   |  | INTEGER | INTEGER | 0 | INTEGER | NUMBER | INTEGER (INT) | NUMERIC |
|  |  |  |  |  | SMALLINT |  |  |  |
|  |  |  |  |  |  |  |   |  |
| PHONE | PH | PHONE | CHAR | 10 | CHAR | VARCHAR2 | VARCHAR | CHAR |
|  |  |  |  |  |  | CHAR | CHAR |  |
|  |  |  |  |  |  |  |  |  |
| RADIUS | RAD | DECIMAL | DECIMAL | 7.2 | DECIMAL | NUMBER | DECIMAL | NUMERIC |
|  |  |  |  |  |  |  |   | DECIMAL |
|   |  | INTEGER | INTEGER | 0 | INTEGER | NUMBER | INTEGER (INT) | NUMERIC |
|  |  |  |  |  | SMALLINT |  |  |  |
|  |  |  |  |  |  |  |   |  |
| RATE | RATE | DECIMAL | DECIMAL | 7.2 | DECIMAL | NUMBER | DECIMAL | NUMERIC |
|  |  |  |  |  |  |  |   | DECIMAL |
|  |  |  |  |  |  |  |  |  |
| RATIO | RTIO | DECIMAL | DECIMAL | 7.2 | DECIMAL | NUMBER | DECIMAL | NUMERIC |
|  |  |  |  |  |  |  |   | DECIMAL |
| SEQUENCE | SQ | INTEGER | INTEGER | 0 | INTEGER | NUMBER | INTEGER (INT) | NUMERIC |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |   |  |
| TEXT | TXT | TEXT | VARCHAR | 255 | VARCHAR | VARCHAR2 | VARCHAR | CHAR |
|  |  |  |  |  | <255, CHAR | CLOB | CHAR |  |
|  |  |  |  |  | CLOB |  |  |  |
|  |  |  |  |  |  |  |  |  |
| TIME | TM | TIME | TIME | 0 | TIME | DATE | TIME | CHAR  |
|   |  |   |   |   | CHAR | CHAR | CHAR |   |
|  |  |  |  |  |   | VARCHAR2 | VARCHAR |   |
|  |  |  |  |  |  |  |  |  |
| TIMESTAMP | TMS | DATETIME | TIMESTMP | 0 | TIMESTMP | DATE | DATETIME2 | CHAR |
|  |  |   |  |  |  | TIMESTMP | DATETIME  |  |
|  |  |  |  |  |  |  |  |  |
| WEIGHT | WT | DECIMAL | DECIMAL | 7.2 | DECIMAL | NUMBER | DECIMAL | NUMERIC |
|  |  |  |  |  |  |  |   | DECIMAL |
|   |  | INTEGER | INTEGER | 0 | INTEGER | NUMBER | INTEGER (INT) | NUMERIC |
|  |  |  |  |  | SMALLINT |  |  |  |
|  |  |  |  |  |  |  |   |  |
| WIDTH | WDTH | DECIMAL | DECIMAL | 7.2 | DECIMAL | NUMBER | DECIMAL | NUMERIC |
|  |  |  |  |  |  |  |   | DECIMAL |
|   |  | INTEGER | INTEGER | 0 | INTEGER | NUMBER | INTEGER (INT) | NUMERIC |
|  |  |  |  |  | SMALLINT |  |  |  |
|  |  |  |  |  |  |  |   |  |
| YEAR | YR | YEAR | SMALLINT | 0 | SMALLINT | NUMBER(4) | SMALLINT | NUMERIC(4) |
|   |   |   |   |   | CHAR(4) | CHAR(4) | CHAR(4) | CHAR(4) |
|  |  |  |  |  |  | VARCHAR2(4) | VARCHAR(4) |  |
|  |  |  |  |  |  |  |  |  |
| ZIP  | ZIP | ZIP  | CHAR | 9 | CHAR | VARCHAR2 | VARCHAR | CHAR |
|  |  |   |  |  | VARCHAR | CHAR | CHAR |  |
| Grey shading represents the preferred Data Type for the DBMS |  |  |  |  |  |
| \* Max length. Can be less if necessary |  |  |  |  |  |  |  |

# Change History

|  |  |
| --- | --- |
| **Effective Date:** July 7, 2014 |  |
| **Create Date**: July 7, 2014 | **Created By:** Sarah Close |

|  |  |  |
| --- | --- | --- |
| **Last Update Date** | **Last Updated By** | **Reason for Change** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |