FLORIDA DEPARTMENT OF TRANSPORTATION

Data and Database Design Standards for SQL Server

Office of Information Technology (OIT)

2/4/2016

Data and Database Design Standards for SQL Server

Table of Contents

PURPOSE 3

DATA ARCHITECTURE 3

FDOT Systems of Record 3

DESIGN STANDARDS 4

1. Global Rules 4
2. Database 4
3. Table 4
4. Table Constraint 4
5. Column 4
6. Key and Index 5
7. Relationship 5
8. Physical Storage 5
9. View 5

DOMAIN DATA TYPE CROSSWALK 6

CHANGE HISTORY 9

Data and Database Design Standards for SQL Server

PURPOSE

This document contains the database design standards applied to ***all*** application system development, maintenance, and enhancement efforts commissioned by the Florida Department of Transportation (FDOT) to ensure optimal, efficient, and consistent use of and support for Microsoft's SQL Server database platform across all environments.

These standards are in effect for all new development efforts beginning 2/4/2016.

The use of Oracle or DB2 as a RDBMS platform is no longer allowed for newly developed applications. Existing systems that are already implemented in either of these contained RDBMS platforms may continue to use them for maintenance or enhancement projects only; Any work performed on these existing systems is covered by the standards published under Standard Set 'G' [http://www.dot.state.fl.us/OIS/AppDevDocsAndGuidelines.shtm].
If development teams elect to re-implement an application's database objects from Oracle or DB2 into SQL Server, compliance with these standards is mandatory.

These standards are not applicable to COTS application systems purchased by the Department. However, these standards shall apply to all customizations the Department adds to COTS application systems implemented in SQL Server.

Naming validation rules and data design requirements applied to the construction of data objects created and maintained within the FDOT infrastructure are also included in this document.

All requested exceptions to these standards shall be submitted to the BSSO Standards and Technical Work Group (BSTWG) in writing. The BSTWG will review each exception request and send a recommendation to the Application Services Manager, who will make the final decision.

DATA ARCHITECTURE

FDOT Systems of Record

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

1. The FDOT Meta Data Repository Glossary[a] for acronyms, abbreviation of business or logical names, and approved key words for naming objects,
2. The DOTCODES reference data repository of code lists, values and descriptions,
3. The Enterprise Document Management System (EDMS) for storage of all electronic documents.
4. Department sources of record:
   4.1. Staff Repository System of all internal and external staff,
   4.2. Transportation Vendor Information system of all FDOT Vendors,
   4.3. Organization Codes for the FDOT organizational units,
   4.4. Work Program and Financial Projects for the FDOT construction projects,
   4.5. Contract Information for the FDOT Contracts.

[a] A standardized process exists to request additions to Glossary words. Please see Meta Data Repository

Glossary Update Process.

DESIGN STANDARDS

All database objects addressed in this document must comply with the following standards.

1. Global Rules

1.1. Standards governing creation of logical and physical object names may be found in Logical and Physical Object Naming Standards for SQL Server and this document.

1.2. When the same name exists in multiple schema, it must represent the same thing, the same view of data, or manipulate data in the same way.

1.3. Database object rules shall override global rules where there is overlap, or where the global rule is restated for the data object.

2. Database

2.1. Each generated schema within a database must store at least one table.

3. Table

3.1. Key unification must not occur in a physical schema, and must be resolved no later than the physical design step of the database design.

3.2. All tables must have at least one column.

3.3. All tables must be assigned estimated volumetric information applicable to the first year of use.

3.4. All tables must contain primary key columns.

3.5. Columns shall be in the following order for database logging efficiency:

- 3.5.1. Primary key columns
- 3.5.2. Frequently changed values
- 3.5.3. Infrequently changed values
- 3.5.4. Very large columns [VARCHAR(2000), for example]

4. Table Constraint

4.1. Constraints must use SQL syntax as defined by the ANSI SQL92, or newer, standards.

4.2. FDOT does not implement column-level encryption or decryption.

5. Column

5.1. Each column data type must be selected from the authorized data type list.

5.2. FDOT does not allow the use of all possible database platform data types. Limiting use to authorized data types enables consistent implementation and long-term performance benefits.

- 5.2.1. Refer to the authorized domain–data type crosswalk.

5.3. Each foreign key (child) column data type must be the same as the contributing (parent) column data type.

5.4. Table columns shall not be defined as IDENTITY columns.

5.5. Names shall only be assigned to those column constraints defined for columns where application rules dictate specific behavior or values.

5.6. The use of the varchar(max) column data type in SQL Server is prohibited.

5.7. Each column shall have the same data type definition for all occurrences of the column in the database.

6. Key and Index

6.1. Each index must contain at least one column.

6.2. Each table must contain a primary key index.

6.3. Each table must contain a primary key constraint.

6.4. Each foreign key must contain a foreign key constraint.

6.5. Overlapping indexes shall not be generated.

6.6. The primary key index column order must match the order of the columns in the table which comprise the primary key.

6.7. Foreign key column order must match the column order of the table contributing the foreign key.

7. Relationship

7.1. Each identifying relationship shall not allow nulls in the child foreign key column(s).

7.2. Each relationship must have valid parent and child tables.

7.3. Many-to-many relationships shall not occur.

7.4. Each relationship shall be enforceable.

8. Physical Storage

8.1. Physical storage file names and allocations shall be assigned by the FDOT Database Administration Technical Team (DBAT).

8.2. Each physical design shall contain at least two storage files.

8.2.1. Primary default filegroup used to store all implemented physical objects

8.2.2. Database Log

8.3. The primary filegroup shall have a primary data file.

9. View

9.1. Each view must include at least one (1) column retrieved from a table or other view present in the database.

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	
			BINARY		BINARY	BINARY	BINARY	
			BIT	0	BIT (string)	BIT (string)	BIT (integer)	*see CODE
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
				0		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

Data and Database Design Standards for SQL Server

SEQUENCE	SQ	INTEGER	INTEGER	0	INTEGER	NUMBER	INTEGER (INT)	NUMERIC
TEXT	TXT	TEXT	VARCHAR	255	VARCHAR	VARCHAR2	VARCHAR	CHAR
					<255, CHAR	CLOB	CHAR	
					CLOB			
		TEXT	XML		XML	XMLType	XML	CHAR
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

**** FDOT does not implement databases using international or unicode data types. (Normal data types with an "N" as a prefix character for the name.**

CHANGE HISTORY

Effective Date: 2016 February 4	
Create Date: 2016 February 4	**Created By:** Jolene

Last Update Date	Last Updated By	Reason for Change