Database Programming Style Guidelines

Version 1.1, April 2001 Copyright © D-Bross

Table of Content

Introduction

Standardization is Important

Good Points

When a project tries to adhere to common standards a few good things happen:

- programmers can go into any code and figure out what's going on
- new people can get up to speed quickly
- people new to programming are spared the need to develop a personal style and defend it to the death
- people new to programming and design are spared making the same mistakes over and over again
- people make fewer mistakes in consistent environments
- programmers have a common enemy :-)

Bad Points

Now the bad:

- the standard is usually stupid because it was made by someone who doesn't understand programming
- the standard is usually stupid because it's not what I do
- standards reduce creativity
- standards are unnecessary as long as people are consistent
- standards enforce too much structure
- people ignore standards anyway

Discussion

Are standards necessary for success? Of course not. But they help, and we need all the help we can get! Be honest, most arguments against a particular standard come from the ego. Few decisions in a reasonable standard really can be said to be technically deficient, just matters of taste. So be flexible, control the ego a bit, and remember any project is fundamentally a team effort.

Interpretation

The use of the word "shall" or "must" in this document requires that any project using this document must comply with the stated standard.

The use of the word "may" or "should" designates optional requirements.

Tables

This naming convention recognizes four different types of tables. All tables names should be in uppercase, separated with underscore where is necessary.

Some restrictions for naming tables:

- No acronyms or abbreviations
- No proper names or words which unduly limit the data which can be entered
- Should not imply more than one subject
- Should be plural

Primary Tables

Primary Tables and their instances should be named with simple plural nouns. **People** and **Employees** are names typical of a primary table and instance.

Linking Tables

The names of Linking tables will contain first the name of the primary table followed by the names of the linking table. Verbs may be used to clarify the nature of the link, and the table names may be changed to singular form for readability

PeopleAddresses
OrganizationPhones
UserGroups

Lookup Tables

Lookup or attribute table names should be in the singular form and clearly describe the attribute.

OrganizationType
MaritalStatus
QualityDescriptor

Collection Tables

Collection tables are a denormalized form. That is, the structure of a one collection table might differ by only one field from the structure of another table.

Fields

Field names should always be in the singular. All names should be in mixed case, first letter on each word should be in uppercase. Use names that are meaningful to your customers.

Some restrictions for naming fields:

- No acronyms
- Use abbreviations only if clear and meaningful
- Should not imply more than one subject
- Should be singular

Key Fields

The name of each primary key field shall be named **ID** and shall be auto-incremental field.

The name of each foreign key field shall be the singular form of the table name followed by _ID.
Employee_ID
Person_ID
Account_Type_ID

Attribute Fields

The name of each attribute field shall be the name that the customer uses to refer to the attribute.

Last_Name
Debit_Amount
Shipment_Number

Queries

The name of each query shall begin with the source of the recordset followed by the name of the SQL Keyword that dictates the action taken when the query is executed. The remainder of the name should indicate the contents of the chosen recordset and the sort order if any.

Organizations_SELECT_Alpha
Shipment_UPDATE_to_Transfers
Shipment_Number

Structured Query Language (SQL)

An SQL statement consists of only two things: Objects (Tables, Recordsets, etc.) and actions taken on those objects (SQL Keywords).

All logical parts of a SQL statement must open new line in the statement. An exception of this rule might be if the statement is short enough to fit in 80 symbols. SQL statements must look like this:

SELECT DISTINCT Person_ID, Last_Name, First_Name, Middle_Name FROM People

```
ORDER BY Last_Name, First_Name

SELECT * FROM People
WHERE Person_ID  = 10 AND
      First_Name = "John" OR
      Last_Name  = "Smith"
ORDER BY Last_Name

INSERT INTO People (Person_ID, Last_Name, First_Name, Middle_Name)
VALUES (15, "Smith", "John", "Junior")

UPDATE People
SET First_Name = "John",
    Last_Name  = "Smith"
WHERE Person_ID = 10
```

SQL Keywords

The name of each SQL Keyword shall be in uppercase.

```
SELECT
FROM
WHERE
```

Recommendation for design

Date and Time Fields

Fields for data and time shall be in CHAR(8) or CHAR(14) format

Date fileds are CHAR(8) : YYYYMMDD
Time fileds are CHAR(14): YYYYMMDDHHMMSS