Corporate Services for the Natural Resource Sector

Information Management Branch

Code Management Standards for the Natural Resource Sector

Last Updated:	April , 2013
Version:	1.0.1
Document:	NRS Code Management Standards_1.0.1

Document History

Version	Description of Change, Review or Approval	Author	Date
0.9	First draft of the guidelines consolidated from Natural Resource Sector data modelling standards • Guide S7 – Modelling Standards for Relational Applications (MFR) • Oracle Designer Standards V 2.1 (MOE/MAL) • Oracle Designer Standards V 2.0 (ILMB) • Programming Standards (Ministry of Small Business, Technology, & Economic Development and Ministry of Energy Mines and Petroleum Resources) – **Obsolete December 31, 2011**	Jeremey Janzen (Forests Author) Randy Hoffner (MoE/MAL Author)	December 2011
1.0	Consolidated multiple ministry standards where standards were the same to create one NRS code management standard. Did not consolidate where standards deviated.	Kristin Lefler	May 2012
1.0.1	Change occurrences of Data Administration to Data Architecture. Update links to NRS standards to new SDLC site	Bill Holland	April 2013

Contents

1 Introduction

This document was created by the Data Architecture/Data Management (DA/DM) section of Information Management Branch (IMB) as an initial consolidation and refinement of code table naming and usage standards for the natural resource ministries.

Through this consolidation effort, the DA/DM section will look for opportunities to move towards a single set of standards. Although this document is in a "final" version, changes to this document should be expected.

1.1 Purpose & Scope

The purpose of this guide is to outline the various code table naming and usage standards used within the Natural Resource Sector ministries and to provide one location where stakeholders can find this information. Standards listed in this guide depict the current state of code table related standards and this guide has been formatted to display existing standards for each ministry in its own section of this guide.

Included in this guide:

- Definition of codes and why they are used
- Current Ministry of Forests, Lands and Natural Resource Operations (FLNR) code standards
- Current Ministry of Environment (MoE), Ministry of Agriculture (MAL), and Integrated Land Management Bureau (ILMB) code standards

The goal will be to provide a more comprehensive set of standards and guidelines that can be adopted by all ministries within the sector.

If you have any questions about this guide or any code table naming and usage at for the Natural Resource Sector (NRS), contact Data Architecture within IMB.

1.2 How to Use this Guide

This guide contains a listing of code table naming and usage standards specific to each ministry within the Natural Resource Sector.

As the IMB DA/DM section works with our stakeholders to build a consolidated guide for all Natural Resource Sector applications, it may be assumed that existing applications will use the

current standards and any net new sector applications will use the standard determined by the project team.

1.3 Intended Audience

The intended audience for this document is:

- Sector data custodians
- Business area staff (ex; Data Standards Managers, Data Stewards)
- Systems analysts and designers
- Contracted developers

1.4 Items Audience Must Understand

Audience Member	Component of Guide
Business Portfolio Managers	SDLC process and how the code naming and usage standards document fits into the overall project
Data Designers (Data Modellers / Application Developers)	Must understand this guide to develop codes for the Natural Resource Sector
Data Administrators / Database Administrators	Will use these standards for data model reviews and sector-wide data design

1.5 Related Standards and Documents

Name	**Description**
Systems Development Life Cycle (SDLC)	http://www.nrs.gov.bc.ca/sdlc/
NRS Data Modelling Standards	http://www.nrs.gov.bc.ca/sdlc/NRS_standards/NRS_Data_Modelling_Standards.pdf
NRS Data Naming Standards	http://www.nrs.gov.bc.ca/sdlc/NRS_standards/NRS_Data_Naming_Standards.pdf
Guidelines for Best Practices in Data Management	http://www.for.gov.bc.ca/his/datadmin/DataMgmtRolesResp-2010Sept-FINAL-Approved.pdf

2 - Classifying Sector Information

The Natural Resource Sector classifies some of its information through use of codes and code tables.

This section explains why we classify our data, what has been done in the past, some of the problems with historical data classifications and how to avoid these problems in the future.

2.1 What Are Codes?

Codes are defined as abbreviated values given to highly repeated business information. For example; status codes such as Approved, Completed, and Deferred could be coded to **APP**, **COM**, and **DEF**; respectively.

The purpose of using codes within the Sector is to provide a set of meaningful values that staff can use to help them deal with the information they use in their day-to-day jobs. However, to maintain a useful and reliable set of codes, the sector must properly plan and manage the code information.

Along with proper management of Sector codes, standardization is crucial to ensuring that code values are easy to understand and that similar code values are consistent within the Sector. This is extremely important in integrated environments where corporate data is shared; however, there are some instances where non-shared, stand-alone databases contain codes that are defined elsewhere with one code but use a different code to mean the same thing. For example; the status of **Complete** may be coded as **COM** or **CMP**. Regardless of how a code is named, it must be properly described so that it is easy to know what the code means. This is further explained in section **2.3 – Importance of Information Classification**.

In addition to standardization, the usage of codes has many other benefits. These include, but are not limited to:

- Auditing
- Data Analysis potential
- Reporting (ex; gathering information from a database to show the number of records related to a specified code)
- Displaying information on an application screen or printed report (ex; display the code versus the longer description to earn more space on the screen or report to display other information)

...continued on next page

- Consistent data capture with consistent meaning
    - Use codes to ensure the same values are entered into the database in the same format, versus using a free-form text field where the risk of entering varying information will likely occur (which opens up further risks of data integrity issues and the inability to audit or report on data in a reliable manner).
    - Use codes to enter data into an application more efficiently versus the longer descriptions
- Saving on storage space by storing a code versus an entire description over and over again (ex; in a table where there might be a couple million rows, a lot of space can be saved by storing a code of **A18** instead of storing the full meaning '**Forestry Licence to Cut with Cutting Permits'**.)
- Data Sharing potential. Once a code has been defined, there is opportunity to share that code amongst other applications within the Sector.

2.2 When Are Codes Used?

Codes are used to deal with the following instances:

- When consistent business meaning is needed for the information item being captured
- When the same code/value is being used across one or more business areas and/or applications
- When a value is intended to mean the same thing

There are a few instances where the above may be true and a code table may not be necessary. For example; In the MoE/MAL and ILMB structure, column check constraints may be used in the database in lieu of setting up a code table. Although a value is used like a code, it does not need to be separated into its own code table because the code values are as they are described (ex; fall, winter, spring, summer).

Note: Column check constraints are not going to be the new practice and this process will be phased out.

2.3 Importance of Information Classification

To get the best value out of the information collected by sector business areas it is often necessary to classify this information.

Where consistent meaning of possible data values is necessary for data sharing, data analysis and data reporting, it is necessary to codify data. If data is left as free-form text it is not possible to consistently capture it, globally understand it, share it or consistently analyse it.

Using Ministry of Forests, Lands and Natural Resource Operations data as an example; imagine if there were no standard classification scheme for tree species and it was captured as free form text, this is what would happen when you:

- Attempt to consistently capture all Douglas Fir stands - they will be captured as F, FIR, Douglas Fir, DF, Fir - Douglas, FD, D. Fir, etc - as many ways as you can think of
- Attempt to get consistent understanding that what is being captured is a species of tree - tree species, brush, shrubs, etc will be captured
- Attempt to share data consistently with other business areas - they won't always know what DF is, or what kind Fir is and all business areas will capture Douglas Fir differently
- Attempt to consistently analyse and report on data - what percentage of the tree stands are Douglas Fir? What would you count?

If data values are defined specifically and the same values are captured and used consistently throughout the ministry (e.g. code **FD** always means Douglas Fir) comprehensive understanding, data sharing, data analysis, data reporting, etc. are possible.

2.4 Historical Patterns for Creating Classification Codes

Historically, codes have often been created to satisfy the needs of a specific business area or a specific business function within the sector. For example:

- A business area needed to know what species of tree was being harvested so they created a tree species code and values for harvesting. Another business area needed to know what kind of tree species was being used for re-planting so they created a different tree species code and values for silviculture.
- A business area needed to know what types of pests were damaging trees so they created a pest species code and values.

2.5 Problems with Historical Classification Codes

Historical methods of classifying ministry data have caused some serious problems. Having different codes for the same thing has caused great confusion. For example, a field worker might be determining tree species for harvesting and capture Douglas Fir as **FD**, and another time be determining tree species for re-planting and have to capture Douglas Fir as **DF**. There is a pretty good chance the field worker won't notice the code difference and the tree species captured for re-planting will be FD. This kind of error is due to the inconsistency in codes between the two business areas. It gets really complicated when FD in the re-planting tree species means Flowering Dogwood and you wanted to order Douglas Fir seedlings for re-planting.

Another historical problem with classification of data is embedding multiple, often hierarchical, meaning within one code. There are many examples of this kind of classification structure within the sector, and just about everywhere else information has been classified. Embedding meaning can cause real problems. Often the embedded meaning has to be parsed out of the code, or worse we run out of values to continue the embedded meaning when new codes are added. Running out of values means we either add non-consistent code values which are misleading, or we re-assign the values of the entire code causing (often massive and complex) data conversion, or we re-structure the code data expanding its field size to add levels to the hierarchy and cause expensive application maintenance.

2.6 Ways to Avoid Historical Problems

We still must live with many codes created in the past which cause these and other problems, or we must resolve the problems (often at great expense). There are some things we can do now and in the future to avoid or lessen code problems.

Identify specifically what it is you are dealing with and want to capture consistent, standard data values for. Ensure it reflects base data which the business area is responsible for. Create code values for the business data items using the guidelines below:

- Create meaningful codes where possible. A code value of 1 for Proposed and 2 for Completed is not very meaningful. Better to go with PRO for Proposed and COM for Completed; something that is intuitive.
- Make your code long enough to handle potential future values without making them meaningless. One or two character codes are often too restrictive. For example, you have a code that is two characters; one of the values is OG for Other Government. You want to add a code for Oil and Gas Commission. The intuitive code of OG is not available. A three character code would allow a value of OGC for Oil and Gas Commission.
- Make your codes as simple as possible. Some coding schemes are so complex that you literally have to take a course to understand them. Simple codes with good, meaningful and easily understood descriptions will make your data much more useable and shareable.
- Ensure each code value is unique across all business areas that need to record and share data the code identifies. Most shared codes should be unique anyway as there is a business area, and a data custodian, responsible for the data the code identifies. This won't always be the case though, as in the past data custodian responsibilities were not identified and codes were created without much thought about sharing.
- If you need to group the codes you have identified, use a separate code to designate the grouping. Do not embed multiple meanings within codes. When multiple meanings (so called smart codes) are embedded, the codes created tend to be very narrow in their business usefulness, very inflexible when business changes or enhancements are

necessary and very susceptible to the slightest business change causing expensive maintenance.

The sector has a number of standard code tables in place for implementing classification codes.

2.7 Implementing Codes

The process of implementing codes within the Natural Resource Sector varies based upon which Ministry makes the request. This is because each Ministry had different methods of implementation prior to the Sector merger.

The goal for implementing codes in the future will be to align current procedures within the Sector.

Until then, the following sections outline how the Ministries within the Sector currently implement their codes.

For further questions/comments on this matter, please contact Data Architecture.

2.7.1 Ministry of Forests, Lands and Natural Resource Operations

Within FLNR, the Data Architecture (DA) section of IMB maintains classification codes on behalf of the business area data custodians. Although the majority of code implementation is conducted by the DA section and IMB, the responsibility resides with the data custodians to determine the codes they want to use and how they want to standardize business data within their area of expertise. When a code is requested by a data custodian, the DA section will work with that business area to confirm whether that code already exists and if it can be re-used. This will ensure that there are no duplicates of codes being used. If the DA determines that the code is new, the code will be reviewed to ensure it meets naming standards and will be added to the corporate code listing.

Because the usage of FLNR codes is often cross-program to ministry-wide, they are held in corporate code tables with additions/updates/deletions managed carefully due to cross-ministry impacts. Having codes stored in this fashion also allows for easy sharing of code values amongst various business area applications within the Ministry.

To assist Ministry staff with understanding codes, all code values and descriptions have been added to a corporate code dictionary so that staff can access the information. This dictionary, and the values within it, is maintained by the Data Architecture section of IMB.

2.7.2 Ministry of Environment, Ministry of Agriculture, and ILMB

Currently within the MoE/MAL and ILMB, codes are not maintained centrally by Data Architecture. Each instance of a code implementation is determined by the current project team involved.

2.7.3 Ministry of Energy and Mines / Ministry of Aboriginal Relations and Reconciliation

Currently within the MEM and MARR, codes are not maintained centrally by Data Architecture. Each instance of a code implementation is determined by the current project team involved.

3 - FLNR Codes

This section explains the standards and conventions both for using corporate codes, and for the use of standard corporate code tables and code lookups that all applications must follow at the Ministry of Forests, Lands and Natural Resource Operations (FLNR).

The Data Architecture section maintains classification codes for different business areas throughout the ministry. When ministry business area staff need codes for use within their applications they must request these codes be added to the corporate code dictionary.

3.1 Approval Process

All logical and physical code names must be approved by Data Architecture staff prior to any tables being built. **This includes names for codes which are formally approved at the data model review.**

In general, the process for approving codes for the Ministry is the same; however, there are some process variances when dealing with Standard Information Codes. Further information on this can be found in section **3.4.2 – Standard Information Codes**.

3.2 Application Code Tables

In the Oracle environment, individual **application code tables** must be used for code validation. The source for all application code tables is the corporate code dictionary. Application developers are responsible for the design and creation (DDL submission) of Application Code tables. Individual application code tables are loaded from codes within the corporate code dictionary by data architecture staff.

It is strongly recommended that projects use application code tables to enforce validations and build in referential integrity.

3.2.1 Application Code Table Use

Corporate code values from the source corporate code dictionary are used to populate and update the Application Code Table values on an as needed basis. The Data Architecture section manages the update of Application Code Tables on behalf of Data Custodians.

Application Code Tables are based on a standard FLNR design. Application developers are responsible for creating the physical structure (DDL) of the Application Code Tables, using the standard design, through the ministry data modelling process.

3.2.2 Process

The business area defines appropriate codes matching the business needs. All requests for new codes are submitted to Data Architecture staff through the appropriate Data Standards Manager. See section **3.2.3 – Submitting Application Code Table Changes** for a sample format for proposing Application Code Table values.

Both the Data Standards Manager and Data Architecture staff must review and approve all new code values. Once approved, Data Architecture staff will manage the update process to populate Application Code Tables.

3.2.3 Submitting Application Code Table Changes

The following format can be used for submitting proposed Application Code table code values to DA for review. Three examples are provided:

In the first example, Appraisal_Category_Code, the project is proposing a brand new application code table, with new code values.

In the second example, Appraisal_Culvert_Type_Code, the project will create a new subset from an existing list of codes. If additional code values were needed, these could also be added as new code values to the master list (with Data Custodian approval) and to the new subset.

In the third Point_Of_Origin_Code, example, the project will use an existing code list and subset as is. In this case it is not necessary to list all of the code values.

Application Code Table (physical table name)	Code Value	Code Description	Code Source (provide existing code list source name or indicate new)	Code Subset Source (provide existing subset name or indicate new)
APPRAISAL_CATEGORY_CODE	N	Normal	new	new
	R	Reappraisal		
	A	Audit		
APPRAISAL_CULVERT_TYPE_CODE	W	Wooden	CULVERT_TYPE_CODE	new
	M	Metal		
	T	Tabular		
POINT_OF_ORIGIN_CODE	AGAM	Agamemnon	BC_LOCATION_CODE	PT_OF_ORIGIN_ST
	ALAR	Alice Arm		
	(Cont.)			

Where a new code list or new subset is required, Data Architecture will assign a new name.

3.3 Using Cross Reference Tables for Codes

Cross Reference tables allow for the cross-referencing of codes. These are needed for situations where selection of a value from one code determines the values which are valid from a second code. For example, selection of a code for part-time temporary employment could guide the list of employment benefit codes available. This would result in a valid combination of two or more codes (concatenated keys). Concatenated keys (to code tables) cannot be supported, and should be implemented as a cross reference table.

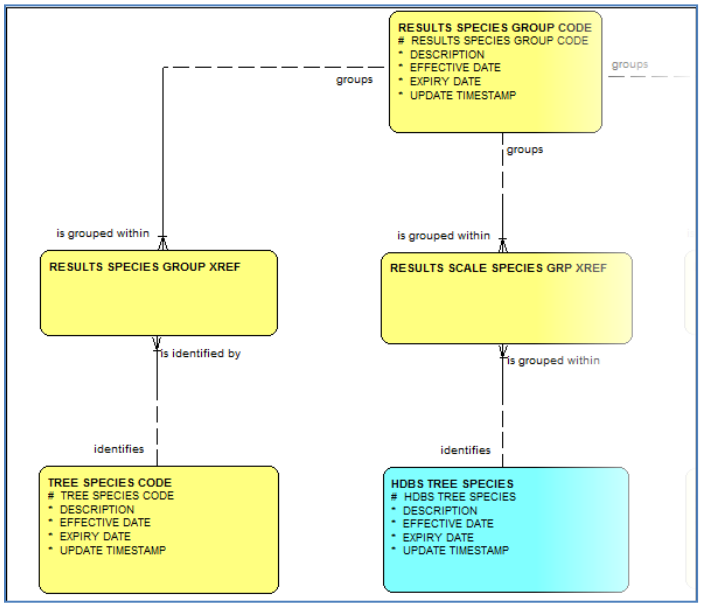
As well, hierarchies of codes should be implemented in cross reference tables. For the most part, hierarchies can be split out. Ex; a two-level hierarchy can be split out into two code lists, plus one cross reference table (XREF).

Projects maintain their own cross reference tables with the application's data tables. Insert scripts are to be submitted by the project for populating these cross reference tables.

NOTE: Data Architecture manages the “feeder” tables for the cross reference tables, but not the cross reference tables themselves.

Example:

In RESULTS, there is a cross reference table called RESULTS SPECIES GROUP XREF. This table was created at the request of the business area as they had a reporting need that required them to create their own combinations of codes from 2 different code table sources. As seen in the diagram below, the RESULTS SPECIES GROUP XREF consists of code values from RESULTS SPECIES GROUP CODE and TREE SPECIES CODE.



3.4 Code Value Types

3.4.1 Status Codes

All status codes within the ministry are defined to a particular standard, displayed in Integrated Data Dictionary (IDD) under the code name STATUS_CODE. Any systems requiring use of status codes must access them through their own defined names, implemented in an application code table. This is to ensure that across systems, all screen entries for status codes remain consistent, so that users do not continually have to change the codes they use for the same status result. The standard status codes are three-character mnemonic (ex; ACT = Active, COM = Complete, EXP = Expired, etc).

3.4.2 Standard Identification Codes

Part of the ministry's need for information can be satisfied by well-designed code structures. Exchange of information is facilitated, understanding is increased, consistent answers are obtained, and operational effectiveness is improved by using codes that are standardized throughout the ministry. Benefits of *corporate* codes are many. In contrast, the drawbacks of using different codes for the same thing, or the same codes for different things, can act as a drag on all sorts of ministry processes, significantly reducing the effectiveness of ministry staff's efforts.

For these reasons, standard codes are developed for use in the ministry (see Ministry Policy 7.1: Standard Identification Codes and Symbols). This document provides more detail about the particular code sets that have been through a formal approval process and are designated *ministry standard codes*. Not all codes used across the ministry need to go through a formal

process to be designated as ministry standard; only those that significantly affect ministry operations, are used by multiple areas of the ministry, or where significant confusion or significant disagreement exists.

3.4.2.1 Description of Standard Identification Codes

A standard identification code (SIC) is a combination of letters and/or digits that has the same meaning Ministry-wide. An SIC allows ministry information to be transmitted, stored, and retrieved quickly and efficiently from all areas of the ministry. The use of SICs helps reduce the inconsistency and confusion that results from various ministry areas using different codes for the same thing. For example, the standard Organizational Unit code represents all offices, sections, and programs within the Ministry.

The design structure of SICs and the elements and characters that comprise them must be compatible with general data architecture standards, including:

- conciseness
- uniform size and format (using standard patterns)
- expandability
- ease of recognition
- automated processing capability

A code should not be developed for its own sake, but to produce greater economy and efficiency within the Ministry.

3.4.2.2 When to Use Standard Identification Codes

Standard identification codes will only be defined for codes that:

- significantly affect Ministry operations
- are used by multiple areas of the Ministry, or have the potential to be used by more than one area

New automated and manual systems for all program areas and activities must use authorized standard identification codes where they exist.

3.4.2.3 Approval of New Standard Identification Codes

New standard identification codes (and changes to existing ones) that are recommended by a Standard Identification Codes (SIC) working group, are subject to review and authorization by the SIC Committee prior to active use. The committee reviews and authorizes SIC based on proper and reasonable code structure.

Each SIC will be sponsored by a single Branch Director, designated as the custodian for that SIC.

3.4.2.4 Standard Identification Code Index

The following Standard Identification Codes have been sponsored and are formalized as corporate codes:

- Organizational Unit
  - See Appendix A
- Timber Supply Area
  - See Appendix B
- Tree/Log Species Conversion
  - See Appendix C
- Code Conversion Rules
  - See Appendix D

3.5 Code Naming Conventions

3.5.1 Naming Code Tables

All code tables must follow these naming guidelines:

- The name of an Application Code Table should reflect the business content of the table. It is preferable to spell out abbreviated words.
- The table name must end with "CODE".
- The Application Code Table name may be up to 30 characters.
- The first column of the Application Code Table is the SAME name as the Application Code Table. This column forms the primary key for the Application Code Table.
- When deciding on the length of the code primary key, the length should be that of the longest code value.
- The column order for the attributes must be as shown below.
- All columns must be mandatory.

3.5.2 Naming Code Table Fields

Each code table must contain the following fields:

- Primary key is the name of the code table and contains the code values (ex; CODE), the length of the code field should be equal to the length of the largest code being used, with some consideration for possible expansion where a code could become a few characters larger; mandatory
- Contains and field called DESCRIPTION, which contains the description of the code value, length must be 120 characters; mandatory

- Contains an field called EFFECTIVE DATE , which contains the date the code was added, this value is used in code lookup functions; mandatory
- Contains an field called EXPIRY DATE, which contains the date the code value is to be expired, default date is 9999-12-31, this value is used in code lookup functions; mandatory
- Contains and field called UPDATE TIMESTAMP, which contains the most recent date for when the code value data was updated; mandatory

Example

```
CREATE TABLE TIMBER_FILE_STATUS_CODE
        (  TIMBER_FILE_STATUS_CODE      VARCHAR2 (5)        NOT NULL
          , DESCRIPTION                 VARCHAR2 (120)      NOT NULL
          , EFFECTIVE_DATE              DATE                NOT NULL
          , EXPIRY_DATE                 DATE                NOT NULL
          , UPDATE_TIMESTAMP            DATE                NOT NULL
        ) ;
```

The effective/expiry dates allow for historical codes to remain on the list for lookup but be restricted for data capture. The default for EFFECTIVE DATE is the date the code is added and the default for EXPIRY DATE is 9999-12-31. The structure is such that there can only be one valid date range for each code value (DESCRIPTION).

The UPDATE TIMESTAMP is used by the Database Architecture group in the physical update process that maintains the Application Code Tables.

4 - MOE/MAL (ILMB) Codes

This section outlines the code usage and naming standards that have been defined for the Ministry of Environment (MOE), Ministry of Agriculture (MAL), and the Integrated Land Management Bureau (ILMB) database structure.

Within this structure, there are a number of ways codes and code tables are utilised. To reflect these differences, codes have been classified into 3 usage level categories (Corporate, Custodial, and Application) and 2 table types (Simple, and Complex). These are further explained in the sections below.

To simplify the process of modelling codes within application data models, the various structures have been defined within the Ministry's Corporate Oracle Repository under the following models:

- CORP_CSD Corporate codes
- CORP_CSD Custodial codes
- CORP_MOF Corporate codes
- CORP_MOF Custodial codes
- CORP_OracleSpatialTypes2

The abovementioned models can be referenced one of three ways;

- from within the Oracle Data Model Repository through the Repository Object Browser (ROB),
- internally within the repository, or
- requested as Oracle .dmp or .dat files

4.1 Approval Process

As there is no formal code approval process for the Ministry, it is included with a data model approval.

For further assistance with codes within this Ministry contact Data Architecture.

4.2 Code Table Types

Code tables within the MOE/MAL and ILMB structure are classified based upon the level in which the code tables are shared within the Ministry. There are three classified levels of code table types; corporate code tables, custodial code tables, and application code tables. This section defines each of these table types, as well as the applicable naming standards.

4.2.1 Corporate Code Table

A **corporate code table** is defined as a widely shared table that contains information relevant to many business areas within the sector. These code tables, which are named with a CORP prefix, are defined at the sector or government level and are placed in a corporate schema called CORP (for operational databases) or in the BC Geographic Warehouse under the WHSE_CORP schema.

The main purpose of a corporate code table is to provide a consistent set of business code values across the organization to enable sharing and avoid redundancies.

When a corporate code value already exists and is used to perform business code lookups and/or validations, it must be used in place of creating a new code or subset.

A listing of corporate code tables/values can be found within the data models in the Ministry's Corporate Oracle Repository and are updated as new codes are identified.

4.2.2 Custodial Code Table

A **custodial code table** is used by multiple datasets but is owned and maintained by a specific business area (business area being the data custodian of the table). The custodial code table exists in one location but can be referenced by many other datasets, provided that the dataset references the schema in which the code table resides.

In the case where the custodial code table does not exist in the same database as the referencing dataset, it will be copied to and referenced from the CORP schema in the database containing the referencing dataset.

If the creation of a new dataset results in the addition of attributes to a custodial code table, either a request can be made to the custodial business area to update their table or a join table of additional attributes can be created (although the latter is preferred).

Overall, a custodial code table has the following standards:

- It may or may not be shared
- It may be used by multiple databases but is owned and maintained by a specific business area
- It is named with a prefix of the application acronym
- It is copied to the CORP schema if the code table is needed by an application outside of its schema
- Any changes to a custodial table must be requested through the custodian

4.2.3 Application/Dataset Level Code

An **Application or dataset specific table** is specific to a single dataset and is only used by that particular application or dataset. It follows standard naming practices and carries the prefix of the application acronym.

- Application specific (not shared)
- Only used by one application

4.3 Code Table Classifications

Each of the code table types (listed in section 4.2) are further classified into one of two options; simple standard or complex. Each of these classifications is determined by whether or not the code table is required to contain a primary key (simple standard) or a surrogate key (complex).

Alternative to the abovementioned classifications there is a third option, which is to maintain code values through a column constraint within the database versus using a code table. However, column constraints are not widely used and this process is under consideration to be phased out.

This section will further explain each of these types as well as the standards required for naming these tables and their attributes.

4.3.1 Simple Standard Code Format

A simple standard code (called CODE) has the code and the description of the code plus audit attributes following the MoE/MAL standard.

Simple Standard Code Table Naming:

A simple standard code;

- On the logical side, is suffixed with CODE, or CD where an abbreviation is needed; when transformed to a table (physical), the table name is suffixed with CODES or CDS
- On the physical side, object names should be a maximum of thirty (30) characters long; therefore, on the logical side the name should be between 24-26 characters.

Simple Standard Code Field Naming:

A simple standard code has the following fields:

- The unique identifier must be the same name as the standard code table that is to be generated, minus the pluralisation (“S”) and application prefix
    - Ex; for an application abbreviated as DSCW, an entity called PARK REFERENCE CODE is created. The plural name property of the entity (which becomes the (physical) table name when transformed) will be DSCW_PARK_REFERENCE_CODES. The unique identifier for the entity (table primary key name) is PARK_REFERENCE_CODE.

- Has an attribute called DESCRIPTION or an attribute ending in NAME, or both. The attribute NAME must be qualified if used.
- May or may not have a attribute called COMMENT
- Must have the audit attributes WHO CREATED, WHO UPDATE, WHEN CREATED and WHEN UPDATED.
- May or may not have the temporal attributes EXPIRY DATE and/or ESTABLISHED DATE. See **Section 7.3** in the **NRS Data Naming Standards** document for standards on use.

Simple Standard Code Attribution Description:

When modelling any code attribute/column, the Ministry requires that all code attributes must have a list of the code values with their meaning, supplied within the attribute description property and comment property of the attribute. If the list is long, a subset of the codes will suffice.

When describing the unique identifier for a Simple Standard Code, use a generic description followed by the meaning of the code then examples of the codes.

The description/comment for a unique identifier as a code will be in the following form:

"The *<unique-code-name>* is a unique code for the object *<class/entity-name>* described as *<meaning taken from the class/entity>.* Examples of codes and descriptions are *<format like S: Salutation, D: Designation, P: Placement>."*

4.3.2 Complex Code Format

A complex code (called TYPE CODE) can be either one of two of the following categories, both having audit attributes.

- The first category has the standard attributes of code and description but has an extra attribute for the NAME of the code.
- The second category is a combination of codes that make up a validation type code.

Both of these categories are based upon a surrogate unique identifier.

Complex Code Table Naming:

A complex code;

- table name is prefixed with the application acronym
- On the logical side, is suffixed with TYPE CODE, or TYPE CD where an abbreviation is needed; when transformed to a table (physical), the table name is suffixed with TYPE_CODES or TYPE_CDS
- On the physical side, object names should be a maximum of thirty (30) characters long; therefore, on the logical side the name should be between 24-26 characters.

Complex Code Field Naming:

A Complex code has the following fields:

- Contains a system generated surrogate unique identifier where its attribute is a NUMBER of data type and ending in _ID. The unique identifier attribute must be the same name as the table name that is to be generated (the plural property of the entity), minus the words CODES or CDS, minus the application prefix, plus the suffix ID
    - Ex; for an application abbreviated as DSCW, an (class/entity) object called PARK ACCESS TYPE CODE is created; the plural name property of the entity (which becomes the table name when transformed) will be DSCW_PARK_ACCESS_TYPE_CODES. The unique identifier for the entity (table primary key name) is PARK ACCESS TYPE ID.
- Contains two or more attributes that have the suffix _CODE, _CD, or _TYPE
- May have an attribute called DESCRIPTION or an attribute ending in _NAME, or both. The attribute NAME must be qualified if used.
- May or may not have an extra indicator attributes CHAR length 1-2 with an _IND suffix
- Contains the audit attributes WHO_CREATED, WHO_UPDATE, WHEN_CREATED, and WHEN_UPDATED
- May or may not have the temporal attributes EXPIRY DATE and/or ESTABLISHED DATE. See **Section 7.3** in the **NRS Data Naming Standards** document for standards on use.

Complex Code Attribution Description:

When modelling any code attribute/column, the Ministry requires that all code attributes must have a list of the code values with their meaning, supplied within the attribute description property and comment property of the attribute. If the list is very long, a subset of the codes will suffice.

For a Complex Code, the code attributes are described in the same manner as the unique identifier for the Simple Standard code object above with the exception of the statement about the unique code for (class/entity) object. So they would be in the following form:

"The *<Code-Name>* is described as *<meaning>*. Examples of codes and descriptions are < *format like code: description, code: description, code: description...>.*"

This type of description above is also standard for all code attributes that reside in business objects, where codes objects are not referenced.

The unique surrogate identifier for a complex code object follows the describing standards mentioned in **Section 6.4 - Descriptions for Unique Identifiers** of the ***Naming and Describing Guidelines for the Natural Resource Sector*** document.

4.3.3 No Code Table – Column Constraint

In cases where a value name is not abstract, is explanatory, and is not used for auditing, a code table is not required and values can be managed through use of a column constraint within the database.

An example of this within the MoE/MAL structure is where values are “winter, spring, summer, and fall” for seasons are used.

NOTE: This process is not widely used and is under consideration to phase out. For more information, please contact Data Architecture.

4.4 Code Table Access and Sharing

Code tables are physically stored in specific schemas within the databases in the Ministry.

Corporate code tables are stored within the CORP schema for operational databases, or within the WHSE_CORP schema for warehouse databases. When using a corporate table, it must be referenced from one of these schemas.

Custodial tables are stored in and referenced from the custodial (application) schema unless the code is needed on a different database, in which case it is copied from the application schema to the CORP schema on the different database and reference from there.

DBA must be notified for the code table to be replicated for use in another database. Corporate code tables will be the same in the production instances and on all database instances.

The procedure for replicating code tables is source to warehouse, warehouse to client database. The staging area or source of replication for these codes objects is the custodial schema, unless it is only available in the CORP schema.

CORP and Custodial codes tables.

For example:

- A table called TA_LAND_DISTRICT is an authoritative custodial table for land districts; it is updated in SDEPROD1 (an operational SDE database).
    - To make it available in the BC Geographic Warehouse (BCGW) is replicated into the BCGW’s WHSE_CORP schema.
    - To make it available on another operational database (ex; ENVPROD1), it is then replicated to the CORP schema in ENVPROD1.

- A table called CORP_ORG_UNIT in the BCGW is a Ministry of Forests, Lands and Natural Resource Operations (FLNR) code table that resides in their “THE” database schema, and is called ORG_UNIT. It is replicated to the WHSE_CORP schema at MoE/MAL and then further copied to any CORP schema on the operational databases.

NOTE: Corporate or Custodial code tables will not always be modelled to standard and will be used in their existing state. This is seen as a grandfathering scenario until the custodial area can bring the object up to Ministry standards.

4.5 Code Table Non-Standard Attributes

If a corporate or custodial code table does not contain all the attributes the business requires, do not add extra non-standard attributes to the original code tables. Create a new application table to contain the extra attributes and relate this new table to the original code table through a one-to-one relationship. For more information on creating and naming new tables, refer to the ***Naming and Describing Guidelines for the Natural Resource Sector*** document.

5 - MEM Codes

A review of the Ministry of Energy and Mines (MEM - formally Ministry of Small Business, Technology & Economic Development) **Programming Standards** document was completed; however, as the code naming and usage standards were informal, a determination was made to not include them within this document.

For further support/questions with regards to MEM code naming and usage standards, please contact Data Architecture.

6 - MARR Codes

There are no current code naming and usage standards for the Ministry of Aboriginal Relations and Reconciliation (MARR).

Please contact Data Architecture for further support/questions for MARR applications.

Appendix A – Organizational Unit

Definition

This standard Organizational Unit code provides a mechanism to identify Programs within Regional and District offices, Timber Sales offices, and Sections within Branch offices, in a short-form code. The code does not direct how to organize reporting relationships in each office.

An Organizational Unit is defined as an element of the ministry's organizational structure which, for supervisory and management purposes, is organized pursuant to the ministry's approved principles, to accomplish certain objectives, directly or indirectly, in accordance with the ministry's Programs and activities. In other words, an Organizational Unit is a distinct office with specific duties that may be further broken down organizationally.

Branch, Regional, District, and Timber Sales offices have in the past used manual and automated systems that had multiple and incompatible code structures that required increasing amounts of staff time both to translate codes and to resolve inconsistencies. With the increasing use of automated systems, making it technically possible to exchange data much more easily than in the past, this problem would have rapidly become unmanageable without a standardized code.

Custodian

Information Management Branch

Scope

Manual systems may use this set of codes immediately (as of approved date). New automated systems, and new releases of existing automated systems that are implemented after the approval date, are required to use this set of codes when representing Organizational Units.

Structure

The full code length is six characters, broken up as follows: **W YY ZZ A**

W - Represents the organizational level of the ministry, and is one of:

D - District Forestry Office

E - Executive

H - Headquarters Branch

R - Regional Forestry Office

T - Timber Sales Office

X - External to Ministry of Forests

YY Represents an office within a specific organizational level, for example, "H FP" is Forest Practices Branch.

ZZ Represents a Section or Program within an office, for example, "H FP FD" is Forest Development Section, Forest Practices Branch.

A Represents a subsection within a Section. Subsections have not yet been coded, except for Nurseries, Provincial Warehouses, Research Stations, and Seed Orchards.

Determining Coded Values

The following are the rules to follow in sequence, to determine new or changed coded values for Organizational Units:

If the office name contains two words (e.g. "Campbell River"), use the **first character** from each word ("Mac"/"Mc" names are treated as two words).

If the office name contains only one word (e.g. "Boundary"), use the **first two characters** from that name (treat "Fort St. James" and "Fort St. John" as one-word names -- ignore "Fort St.").

If there exists a "tie" after the above, change the name that comes later alphabetically, and use the **first and last characters** of the name.

If there still exists a "tie" after the above, change the name that comes later alphabetically, and use whatever makes sense that isn't repetitive.

Exceptions

The following are general exceptions to the coding rules in the Organizational Unit codes:

"Recreation" is referred to as "RN" throughout, to avoid a conflict with Research Branch;

"Timber" is referred to throughout Regions and Districts as "TH", to correspond with the code for Timber Harvesting Branch*;

"Finance" is referred to throughout Regions and Districts as "FS", to correspond with the code for Financial Services Branch*;

"Systems" is referred to throughout Regions as "IS", to correspond with the code for Information Systems Branch*;

Kamloops District is referred to as "KA", and Kalum District is referred to as "KM", to avoid different codes for Kamloops District and Kamloops Region, even though Kalum District comes first alphabetically (following the naming rules exactly would have coded Kalum District "KA" and Kamloops District "KS").

"Generic" Codes

The existing codes include a generic "Staff Manager" at the Region and District level. Regions and Districts may organize their Programs as they require and use these codes as necessary. Specific organizational design is not forced on any office by being inherent in this code structure.

Code List

This code list is provided on the web from the Integrated Data Dictionary (IDD). A printed copy of the codes can also be obtained from there. For further information, please contact Data Architecture staff in Information Management Branch.

* (footnote)

The examples used here were developed at the time the code structure was developed in 1988, and although many office names and corresponding abbreviations have changed (and will change again!), the above examples continue to illustrate the point.

Appendix B – Timber Supply Area

Definition

This standard TSA code provides a mechanism to identify the ministry's Timber Supply Areas throughout the province on manual and automated systems.

In the ten years before 1990, two major sets of codes came into use by different areas of the ministry. Timber Harvesting and Valuation used one set, while Inventory, Protection, Silviculture, and Planning had standardized on another. This was a problem for field staff, but was not such a big problem for the computer systems of the decade, because few of the ministry's applications shared common databases. Increased emphasis on shared data makes standardization on a single code for *TSA number* imperative.

Custodian

Resource Tenures and Engineering Branch

Scope

Most tenures issued by the ministry include a reference to the TSA they are issued within, so any staff involved with tenures will access the TSA code. As of April 30, 1990, all manual and automated systems are required to use this code.

Note that no stored manual files (for example, *Exhibit 'A'* documents) have to be converted. An information memo will have been sent to all clients, where necessary, indicating the simple coding change.

Structure

The full code length is two numeric characters, with no special structure inherent within the code. Current values are between '01' and '43' inclusive, although some numbers are not represented. See the attached list of allowable codes.

Determining Coded Values

TSA numbers (/codes) for new TSAs will be assigned by Resource Tenures and Engineering Branch.

Code List

This code list is provided electronically from the Integrated Data Dictionary (IDD) (search on Code Name: "TSA". For further information, please contact Data Architecture staff.

Appendix C – Tree/Log Species Conversion

Definition

Historically the ministry developed two distinct sets of codes for dealing with tree species: one set follows the scientific naming conventions used in biological areas, and the other was developed for harvested timber (logs), using an abbreviation of the English species name.

Staff resources are required to check and correct errors produced from the use of the two overlapping and different code sets, causing operational difficulties and making ministry staff less effective. This conversion table has been developed to assist staff who are required to convert between the two code sets, so that across the ministry all staff are at least using a single standard for conversion.

Scope

The scaling area of the ministry and industry is the only area that uses the log codes; all others (e.g. cruising, research, silviculture, inventory, etc.) use the biological code. Where conversion between the two sets of codes is necessary, all staff must use the attached table (see Appendix D).

Structure

See conversion table in Appendix D.

Determining Coded Values

Not applicable; see conversion table in Appendix D.

Exceptions

Not applicable; see conversion table in Appendix D.

Code List

A complete list of the conversions between the Tree Species code and the HDBS Log Species code follows (see Appendix D).

This code list is provided electronically from the Integrated Data Dictionary (IDD) (search on Code Name: "tree_species". For further information, please contact Data Architecture staff.

Appendix D - Code Conversion Rules

HDBS Log to Standard Tree Species

The Species level (two characters) should be used as much as possible. Some HDBS codes translate only to the Genus level (one character) in the Ministry standard codes.

Originating HDBS Log Species		*Destination Standard Tree Species*		*Conversion Comment*
AL	**Alder**	DR	red alder	95% of alder in industry is red alder.
AR	**Arbutus**	RA	arbutus	
AS	**Aspen**	AT	trembling aspen	
BA	**Balsam**	B	true fir	
BI	**Birch**	EP	paper birch	95% of birch in industry is paper birch
CE	**Cedar**	CW	western redcedar	
CO	**Cottonwood**	AC	poplar	
CY	**Cypress**	YC	yellow-cedar	
FI	**Fir**	FD	Douglas-fir	
HE	**Hemlock**	H	hemlock	
LA	**Larch**	L	larch	
LO	**Lodge-pine**	PL	lodgepole pine	
MA	**Maple**	MB	bigleaf maple	95% maple in industry is bigleaf maple.
OT	**Other species**			No corresponding Tree Species code.
R	**Rejects**			No corresponding Tree Species code.
SP	**Spruce**	S	spruce	
UU	**Yew**	TW	western yew	
WA	**Waste**			No corresponding Tree Species code.
WB	**White Bark Pine**	PA	whitebark pine	
WH	**White Pine**	PW	western white pine	
WI	**Willow**	W	willow	
YE	**Yellow Pine**	PY	yellow pine	
ZZ	**Rejects**			No corresponding Tree Species code.

Standard Tree Species to HDBS Log

Originating Standard Tree Species		*Destination HDBS Log Species*		*Conversion Comment*
A	**poplar**			No corresponding HDBS code.
AC	**poplar**	CO	Cottonwood	
ACB	**balsam poplar**	CO	Cottonwood	
ACT	**black cottonwood**	CO	Cottonwood	
AT	**trembling aspen**	AS	Aspen	
B	**true fir**	BA	Balsam	
BA	**amabilis fir**	BA	Balsam	
BG	**grand fir**	BA	Balsam	
BL	**subalpine fir**	BA	Balsam	
C	**western redcedar**	CE	Cedar	
CW	**western redcedar**	CE	Cedar	
D	**alder**	AL	Alder	
DR	**red alder**	AL	Alder	
E	**birch**	BI	Birch	
EA	**Alaska paper birch**	BI	Birch	
EP	**paper birch**	BI	Birch	
F	**Douglas-fir**	FI	Fir	
FD	**Douglas-fir**	FI	Fir	
FDC	**Douglas-fir (coast)**	FI	Fir	
FDI	**Douglas-fir (interior)**	FI	Fir	
G	**dogwood**			No corresponding HDBS code.
GP	**western flowering dogwood**			No corresponding HDBS code.
H	**hemlock**	HE	Hemlock	
HM	**mountain hemlock**	HE	Hemlock	
HW	**western hemlock**	HE	Hemlock	
J	**juniper**			No corresponding HDBS code.
JR	**Rocky Mountain juniper**			No corresponding HDBS code.
K	**cascara**			No corresponding HDBS code.
KC	**cascara**			No corresponding HDBS code.
L	**larch**	LA	Larch	
LA	**alpine larch**	LA	Larch	
LT	**tamarack**	LA	Larch	
LW	**western larch**	LA	Larch	
M	**maple**	MA	Maple	
MB	**bigleaf maple**	MA	Maple	
P	**pine**			No corresponding HDBS code.
PA	**whitebark pine**	WB	White Bark Pine	
PF	**limber pine**			No corresponding HDBS code.
PJ	**jack pine**			No corresponding HDBS code.
PL	**lodgepole pine**	LO	Lodge-pine	
PLC	**lodgepole pine (coast)**	LO	Lodge-pine	

PLI	**lodgepole pine (interior)**	LO	Lodge-pine	
PW	**western white pine**	WH	White Pine	
PY	**yellow pine**	YE	Yellow Pine	
Q	**oak**			No corresponding HDBS code.
QG	**Garry oak**			No corresponding HDBS code.
R	**arbutus**	AR	Arbutus	
RA	**arbutus**	AR	Arbutus	
S	**spruce**	SP	Spruce	
SB	**black spruce**	SP	Spruce	
SE	**Engelmann spruce**	SP	Spruce	
SS	**sitka spruce**	SP	Spruce	
SW	**white spruce**	SP	Spruce	
SX	**spruce hybrid**	SP	Spruce	
T	**yew**	UU	Yew	
TW	**western yew**	UU	Yew	
W	**willow**	WI	Willow	
Y	**yellow-cedar**	CY	Cypress	
YC	**yellow-cedar**	CY	Cypress	

*** (footnote)**

From "The Information Credo", *Data Administration and Information Resource Development Seminar Workbook*, by Ronald G. Ross, 21st Edition, August 20, 1991, page D4.