Nomenclature Load

Author: Lori Corbani

Created: May 23, 2003

Last Modified: December 30, 2003 10:52

1   Purpose of Document

To describe the Nomenclature Load product.

2   Overview

The purpose of the Nomenclature Load is to:

1. To load new gene records into MGI Nomenclature tables and set these attributes:
   - Marker Type
   - Symbol
   - Name
   - Chromosome
   - Status
   - Primary Reference
   - Synonyms (optional)
   - Accession IDs (optional)
   - Nomenclature Notes
   - Submitter
   - Event = assigned
2. To assign MGI Accession IDs to the new genes.
3. To set these nomenclature attributes to null (no value):
   - Human Symbol
   - Broadcast By
   - Broadcast Date
   - Gene Family
   - Editor's Notes
   - Nomenclature Coordinator Notes

3   Inputs

A tab-delimited file in the following format (no column headings):

Table 1: Input File

Field	Name	Example	Type	Max Length
1*	Marker Type	Gene (see MRK_Types)	string	255
2*	Symbol	Olfr1000	string	25
3*	Name	olfactory receptor 1000	string	255
4*	Chromosome	UN	string	2
5*	Marker Status	In Progress Reserved Approved	string	255
6*	J: (with the J:)	J:82104	integer	none
7	List of Synonyms separated by "|"	syn1|syn2|	string	none
8	List of Accession IDS with Logical DB identifier	Sequence DB:AY073866|	string	none
9	Nomenclature Notes	this gene is good	string	255
10*	Submitter	bobs	string	255

Fields with "*" require a non-null value. Other fields must be present but can be blank.

Field 4:

Any mouse chromosome

Field 6:

The Reference (J:) is used as:

- the primary reference
- the reference for each synonym (field 7)
- the reference for each accession ID (field 8)

Field 7:

A list of synonyms separated by "\|":

```
Syn1|Syn2|Syn3|
```

Field 8:

A list of accession IDs with the Logical DB name (see ACC_LogicalDB.name) of that accession ID. Most often this is a Nucleotide Sequence ID:

```
Sequence DB:AY073866|Sequence DB:AY073027|
```

Field 10:

The Submitter. Use your SYBASE login.

4   Outputs

1. 5 BCP files (inputfile name is prepended to each bcp file name):
   - NOM_Marker.bcp
   - MGI_Reference_Assoc.bcp
   - NOM_Synonym.bcp
   - ACC_Accession.bcp
   - ACC_AccessionReference.bcp
2. Diagnostics file (inputfile.MMDDYYYY.diagnostics) that includes all SQL commands issued by the load.
3. Error file (inputfile.MMDDYYYY.error) that includes any data validation errors.
4. Input file with MGI Accession IDs as the last column.

5   Processing

1. Verify Mode.
   - if mode = load: process records
   - if mode = preview: set "DEBUG" to True
2. For each line in the input file:
   1. Verify the Marker Type is valid.
      If the verification fails, write the error to the error file and skip the record.
   2. Verify the Marker Status is valid.
      If the verification fails, write the error to the error file and skip the record.
   3. Verify the Logical DB is valid.
      If the verification fails, write the error to the error file and skip the record.
   4. Verify the J:is valid.
      If the verification fails, write the error to the error file and skip the record.
   5. Verify the Submitted By is valid (not null).

If the verification fails, write the error to the error file and skip the record.

6. Create the Nomen bcp records (NOM_Marker,bcp, MGI_Reference_Assoc.bcp).

7. Create the MGI Accession ID for the Marker (ACC_Accession.bcp, ACC_AccessionReference.bcp).

8. If Synonyms are provded, then process them (NOM_Synonym.bcp).

9. If Accession ids are provided, then process them (ACC_Accession.bcp, ACC_AccessionReference.bcp).

3. Load the bcp files into the database.

6   Usage

To execute:

Python Program: nomenload.py

Parameters:

- -S server
- -D mgd database
- -U user
- -P password file
- -M mode (*load* or *preview*)
- -I input file