



tranSMART

ETL Guide

December 21, 2012
Edition 1

Recombinant
By **Deloitte.**

The bottom of the page features a blue-tinted background image. On the left is a stylized DNA double helix. On the right is a blurred image of a data table with columns for Patient, MRN, DOB, Age, Sex, and Race. The table contains several rows of data, including patient identifiers and dates.

Patient	MRN	DOB	Age	Sex	Race
	100001501	8/17/71	37	F	1
	100001501	1/28/79	33	M	1
	100001501	1/28/79	33	F	1

Copyright © 2011 - 2012 Deloitte Development LLC. All rights reserved.

Recombinant[®] is a registered trademark of Deloitte Development LLC in the United States and other countries.

Other company, product, and service names may be trademarks or service marks of others.

*Any blank pages in this document are intentionally
inserted to allow correct double-sided printing.*

Contents

Chapter 1: ETL Overview	1
The i2b2 Data Tree.....	1
i2b2 Data Tree Structure.....	2
Elements of the Data Tree	3
Defining a Study-Driven Ontology	6
Study-Driven Ontology	6
Transforming Source Files into the Standard Format.....	8
Location of Kettle Files.....	11
Chapter 2: Loading Raw Clinical Data into i2b2	13
Resources	13
Loading Clinical Data Using Kettle	14
Step One: Generate Raw Clinical Data Files	14
Step Two: Create the Column Mapping File.....	14
Step Three: Create the Word Mapping File	18
Step Four: Create the Record Exclusion File.....	19
Step Five: Load the Data.....	20
Loading Clinical Data Using Stored Procedures	22
Standard-Format File – Clinical Data	22
Step One: Edit the Loader Script.....	25
Step Two: Word Count Command.....	26
Step Three: Running the Loader Scripts	26
Step Four: Execute Stored Procedure i2b2_load_clinical_data	26
Chapter 3: Loading SNP Data into DEAPP	29
Resources	29
Prerequisite Tasks	30
Downloads	31
Creating Reference Folders.....	33
Creating a CEL Reference File	34
Creating a Sample Variance File	35
Running Affymetrix Power Tools.....	36
To Measure Genotype	37
To Measure Copy Number Variation	38

Loading the Data.....	41
Step One: Subject-to-Sample Mapping	41
Step Two: Process and Load Annotation Data	44
Step Three: Populate Meta Tables	44
Step Four: Pivot Processed SNP Data	45
Step Five: Load SNP Data into tranSMART.....	47
Step Six: Load SNP Call Data into tranSMART	48
Step Seven: Load SNP Copy Number Data into tranSMART	49
Chapter 4: Loading Gene Expression Data into DEAPP	51
Resources	51
Preparing the Data for Loading	52
Loading Gene Expression Data Using Kettle	57
Kettle Parameters for Loading Gene Expression Data	58
Loading Gene Expression Data Using Stored Procedures.....	59
Step One: Pivot Data	59
Step Two: Edit the Loader Scripts	60
Step Three: Word Count Command	61
Step Four: Run the Loader Scripts.....	61
Step Five: Execute Stored Procedure i2b2_process_mrna_data.....	62
Chapter 5: Loading Study Metadata and Dataset Explorer Search Subjects	65
Input File Format	65
Metadata Loader	66
Kettle Parameters for Loading Study Metadata	67
Chapter 6: Loading Platform Annotation Data	69
Gene Expression Platforms	69
Loading the Annotation Data	70
Kettle Parameters for Loading Gene Expression Annotations	71
SNP Platforms.....	72
Chapter 7: Loading GWAS and eQTL Analysis Data	75
Troubleshooting	75
Manually Staging Analysis Data.....	76
Running the Nightly Processing Job	76
Chapter 8: Study Security and Study Deletion	77
Applying Security Restrictions to a Study	77
Deleting a Study	77

Appendix A: Schemas79

DEAPP Schema 79

 DEAPP Schema Diagram 82

I2B2DEMODATA Schema..... 83

 I2B2DEMODATA Schema Diagram 84

I2B2HIVE Schema 85

 I2B2HIVE Schema Diagram 86

I2B2METADATA Schema 86

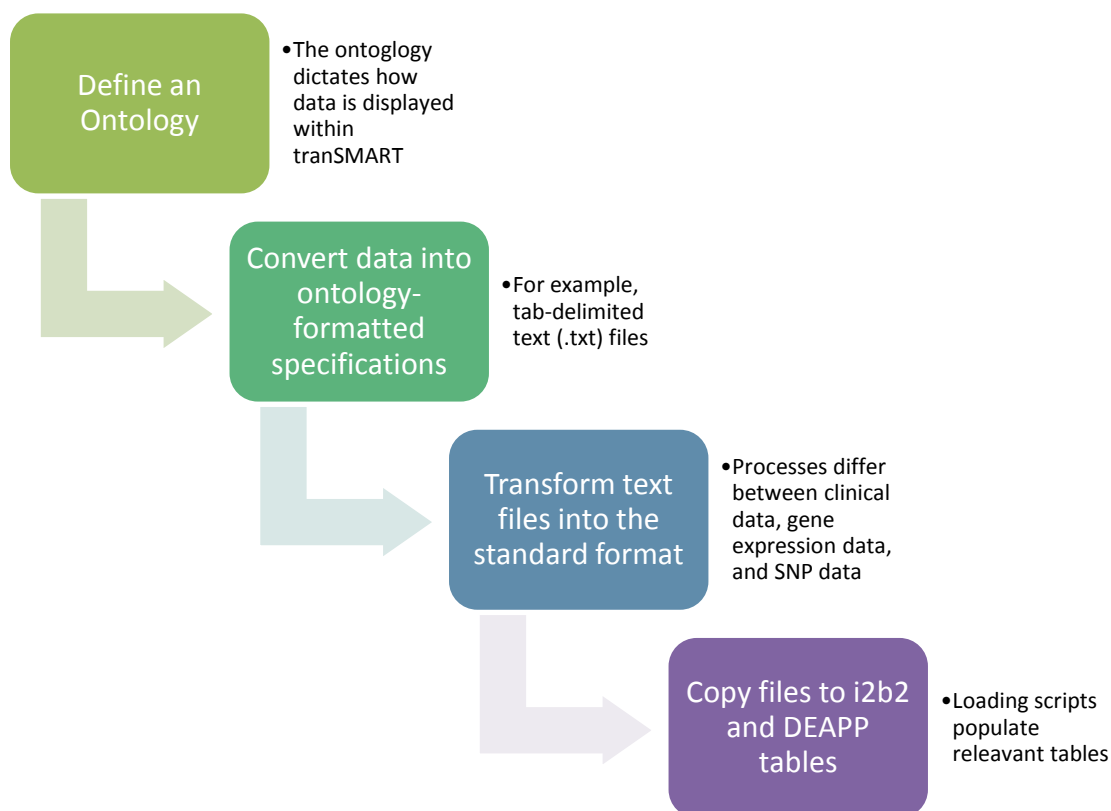
 I2B2METADATA Schema Diagram..... 87

 BIOMART GWAS/EQTL Schema Diagram 88

Chapter 1

ETL Overview

An overview of the ETL (extract, transform, and load) process for loading raw source data is shown below:



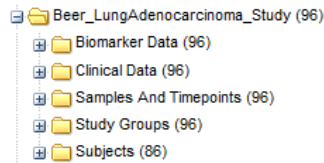
The i2b2 Data Tree

The Dataset Explorer i2b2 Data Tree is a hierarchical representation of your study data. Data values, data labels, and categories of data are specified in columns of a standard-format file that you generate from a source data file.

The following sections provide a conceptual explanation of the tree using clinical data.

i2b2 Data Tree Structure

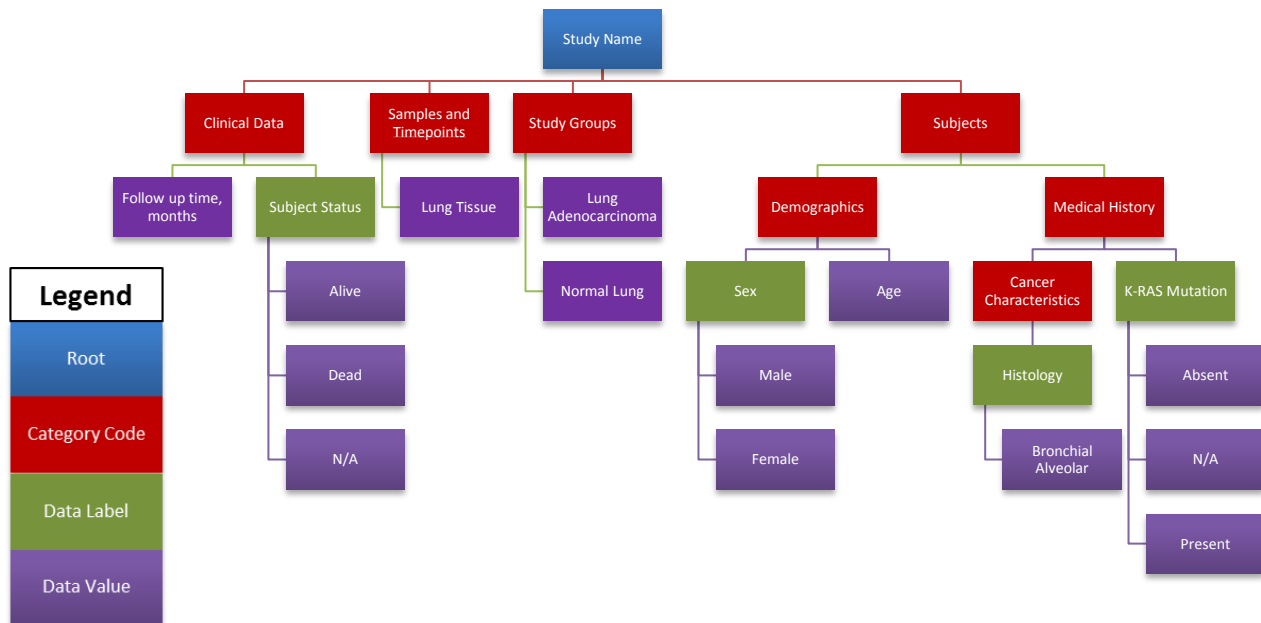
The following example shows a common layout for the upper-level nodes of a study:



The nodes below these upper-level nodes will differ for each study, depending on a variety of factors, such as the type of biomarker data, the samples that were collected, and the demographics and medical history (if any) collected from the subjects.

The data that you load from your raw source files into the columns of the standard-format files determine the organization of the tree – that is, the branches that lead from the study name on down to the data labels and data values.

The illustration below is a conceptual representation of the node structure of study Beer_LungAdenocarcinoma_Study:



Elements of the Data Tree

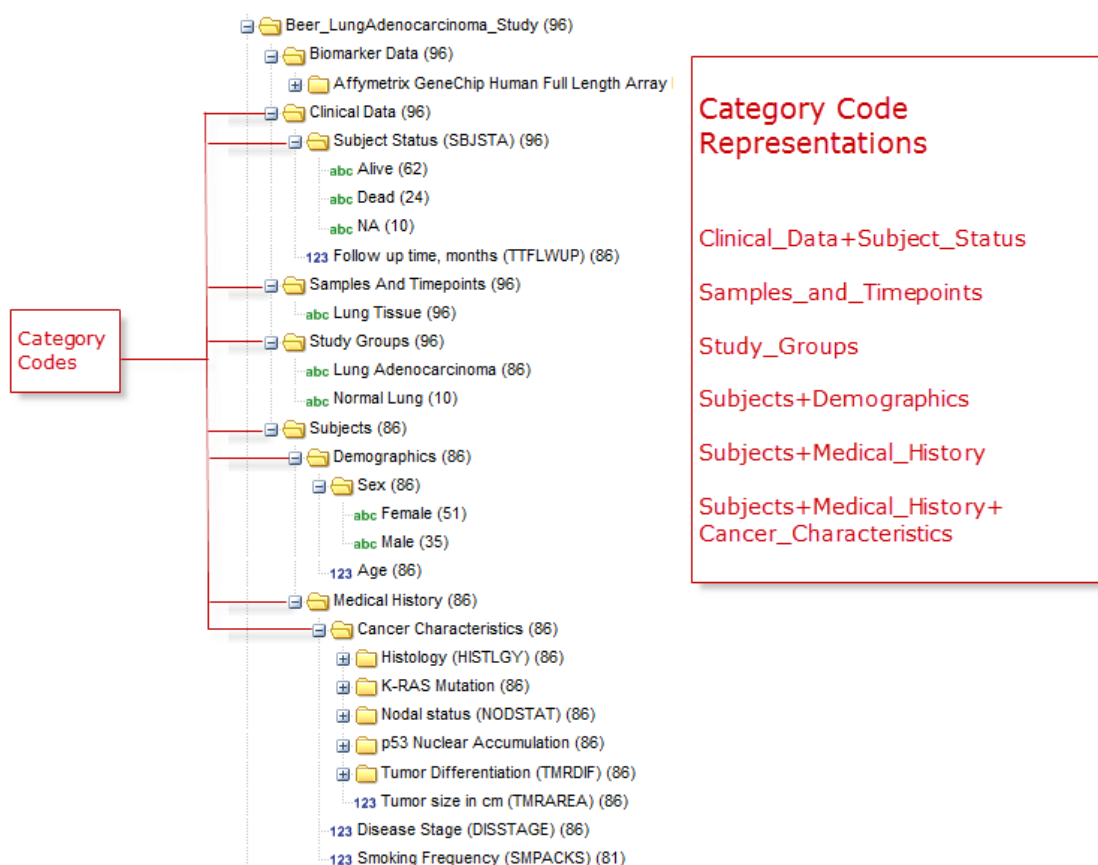
You are responsible for defining the elements of the tree using data from your raw source files. Raw source data is first transformed into a standard-format file that can be loaded into i2b2.

For information on the standard-format for various types of source data, see the following sections:

- [Standard-Format File – Clinical Data](#) on page 22.
- [File Preparation for Gene Expression Files](#) on page 52.
- [File Preparation for Subject-to-Sample Mapping Files](#) on page 54.

Category Codes

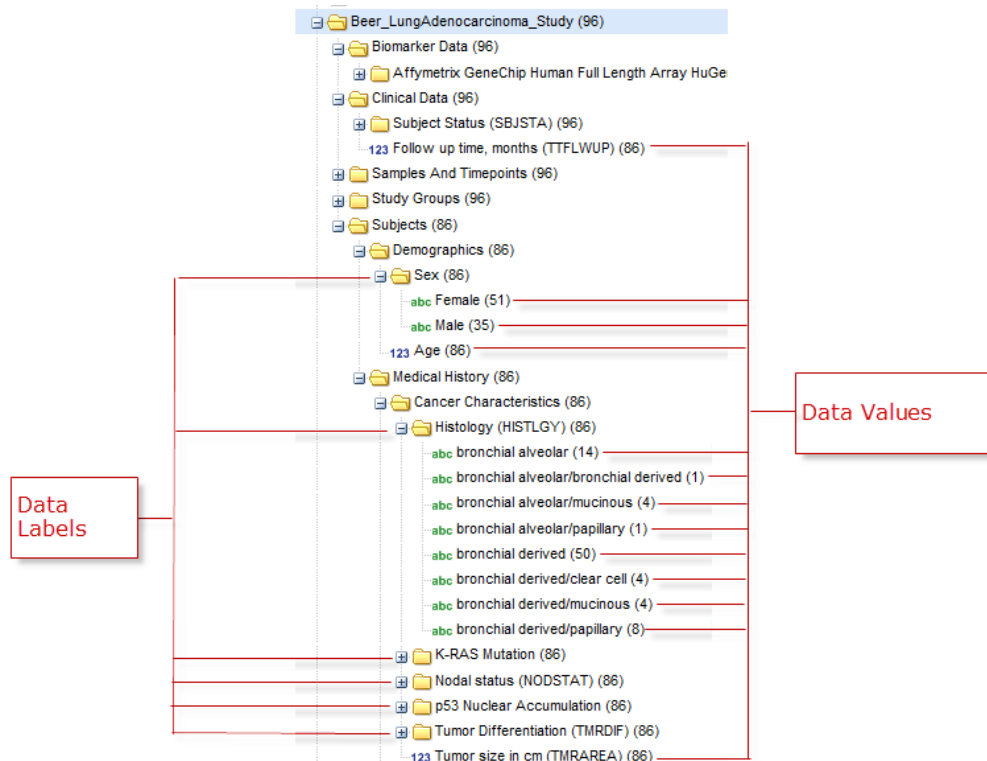
A category code contains the fully-qualified path in the i2b2 tree that leads from the study name down to the data label and data values associated with that particular category code. The graphic below illustrates the folders that combine to construct a category code.



Data Labels & Data Values

A data label specifies the type of data collected from a subject; for example, the subject's age or weight, or the cancer stage of a tissue sample. A data value is the actual measurement collected from the subject or the sample for a given type of data. Hierarchically, data labels appear immediately above data values. Data labels and data values appear at the lowest levels of the i2b2 tree.

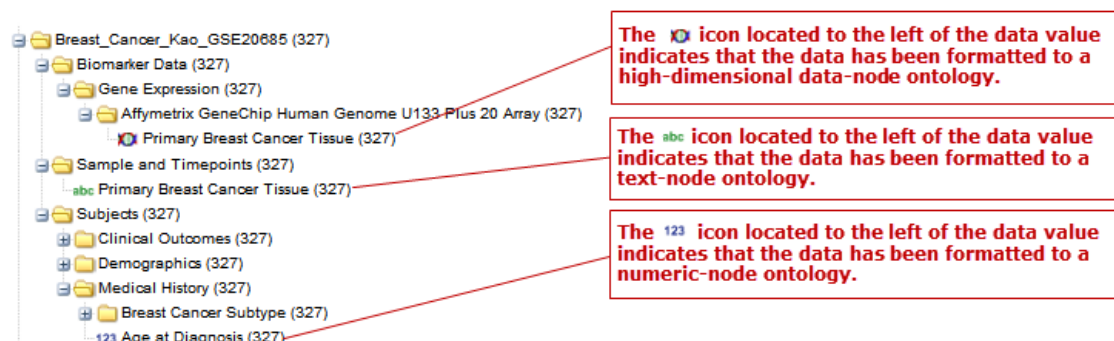
The graphic below illustrates the relationship between data labels and data values, and their position in the i2b2 tree:



Types of Data Values

In tranSMART, there are three types of data values: numeric, text, and high-dimensional data (SNP, gene expression, etc. stored as arrays).

The three types of data value ontologies are illustrated below:



Sample i2b2 Data Tree Organization

The table below illustrates possible ways to format your category codes and how they will appear within Dataset Explorer.

Clinical Data	Category Code	Data Label	Data Value	Example
Subject is female.	Subjects + Demographics	Sex	Female	
Subject is male.	Subjects + Demographics	Sex	Male	
Subject was first diagnosed at age 38.	Subjects + Medical_History	Age at Diagnosis	38	
Subject had sample extracted from lymph nodes.	Samples_and_ Timepoints	Sample Type	Lymph Node	

Clinical Data	Category Code	Data Label	Data Value	Example
Blood sample was taken as a baseline.	Samples_and_Timepoints	Blood	Baseline	

Defining a Study-Driven Ontology

The ontology you define affects how data appears in the Dataset Explorer i2b2 tree.

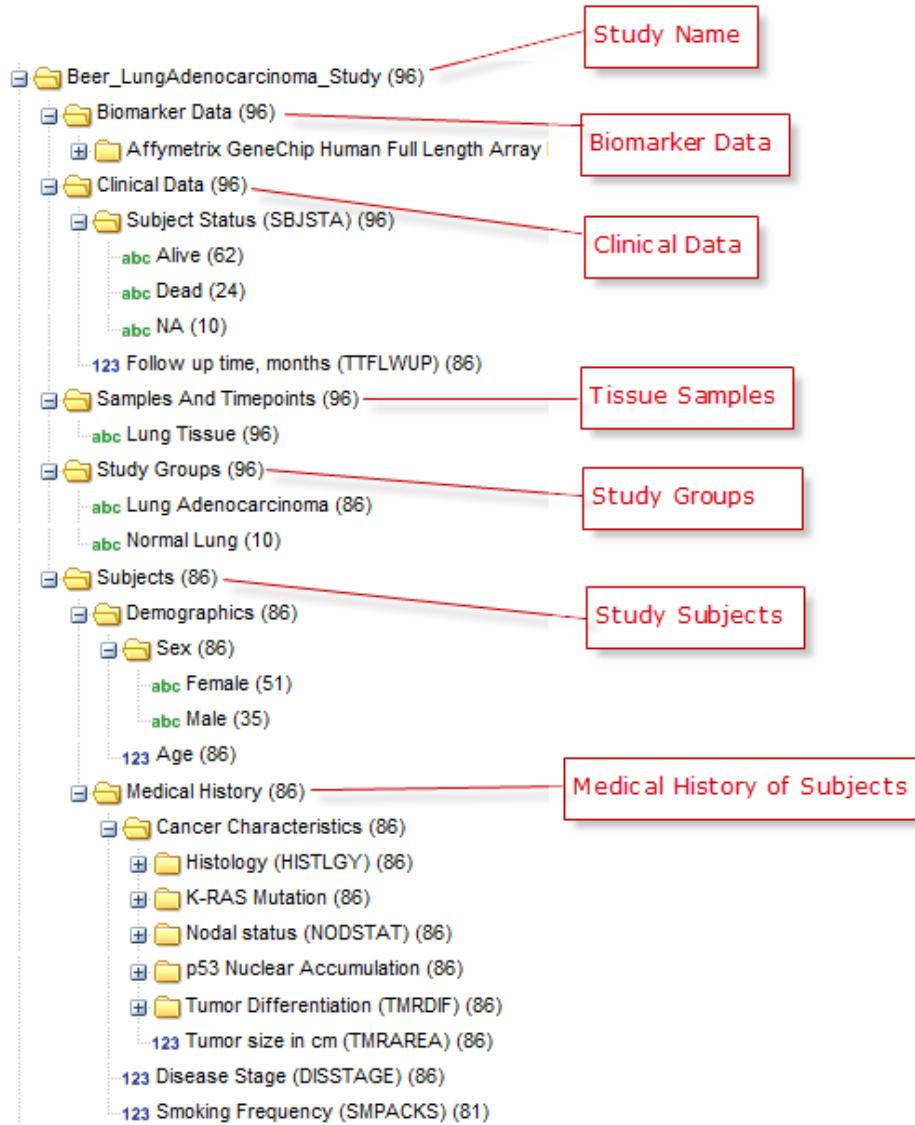
Study-Driven Ontology

A study-driven ontology structure generates category codes dynamically. Study-driven ontologies define the location of information inside the i2b2 Data Tree according to a granular categorization of a study. This structure allows for a flexible use of data and for subtle queries of the data.

In a tranSMART study-driven ontology, queries are driven by SQL code that references fields in a standard-format file (see [Transforming Source Files into the Standard Format](#)). The SQL code creates a table and any relevant sub-tables that mirror the path defined in the standard-format file's category codes, data labels, data values, and any other mapping conventions that best describe the type of data.

Formatting category codes on the study-level allows you to choose how to display data in the way that makes the most sense for that particular study.

In the following example, note how the study containing clinical data has been arranged. Your study may be better represented by a different arrangement:



Transforming Source Files into the Standard Format

Your raw source files are likely to contain a variety of kinds of data and be in a variety of formats. As a result, there is no one procedure that can transform all raw source files into the standard structure required by loader scripts.

You must examine each raw source file to determine how best to transform the data into the standard format. This section includes some considerations to keep in mind when examining the source files, and some tips for curating the data that you are loading into i2b2.



Transformations can be difficult to do correctly and efficiently, and they should be done by an experienced ETL analyst. Recombinant by Deloitte provides data transformation services.

Considerations for Transforming Data

As you examine a raw source file, the following considerations may help guide data curation decisions:

- Does the source file contain study data or biomarker data?
- In which study category (for example, public studies, proprietary studies) does the data belong?
- Is there a control group in the study? If so, who is in the control group and who is in the experimental group?
- Are there multiple experimental groups?
- Is there just one sample per subject or multiple samples per subject?
- If there are multiple samples per subject, is it due to one or more of these reasons:
 - Multiple timepoints during which samples were collected for each subject
 - Multiple sample sources such as Protein or mRNA
 - Multiple platforms such as Affymetrix, Illumina, or custom
 - Multiple types of samples, such as healthy and diseased tissue samples

- Consider the path structure of the i2b2 tree that the source data will be loaded into. The i2b2 tree has the following types of paths:
 - Text data (data values are a mixture of text and numbers):


```
root_node\study_id\category_path\visit_name\data_label\data_value
```
 - Numeric data (all data values are numeric):


```
root_node\study_id\category_path\visit_name\data_label\
```
 - Path with no visit node (text data):


```
root_node\study_id\category_path\data_label\data_value
```
 - Path with no visit node (numeric data):


```
root_node\study_id\category_path\data_label\
```
- With gene expression and SNP data, a subject-to-sample mapping must be created. For information, see [Subject-to-Sample Mapping](#) on page 11.
- Should any data be excluded?

Curation

Curation is the process of standardizing, correcting, and “cleansing” data. Curation is often performed at different stages of the data loading process; for example, it may be performed in the raw source files, during the transformation process, after the data is transformed into the standard format, or in any combination of these stages.

Before running the loader scripts, check the standard-format file carefully to ensure that all text has been properly curated. You should not perform curation in the i2b2 databases.

After you run the loader script, check the results in the i2b2 tree for any curation tasks that you might have missed. If you find any problems, make the fixes in the standard-format file, then run the stored procedure to load the data again.

Here are some examples of data curation:

- Ensure that terminology in the data being loaded is consistent with the terminology already loaded in the i2b2 tree. For example, if the i2b2 tree uses the data label `Medical History`, but the data to be loaded uses the term `History` for the same concept, change `History` in the data to be loaded to `Medical History`.
- Be sure that units of measurement in the source data are consistent with the corresponding data in the tree. For example, if the weights of subjects in the tree are expressed in grams, but are expressed in pounds in the source data, the source-data weights must be converted to grams.

- The following characters are not allowed and must be removed:
 - Backslash (\)
 - Percent (%) – Change to `PCT`
 - Asterisk (*)
 - Ampersand (&) – Change to `and`
- The concepts of age, race, and gender must have the following data labels in order to appear in the Generate Summary Statistics tab of Dataset Explorer:
 - `AGE`
 - `RACE`
 - `GENDER` or `SEX` (either one)
- Are alphabetic characters in the correct case (uppercase, lowercase)? Oracle queries may not find text that uses incorrect case.
- Correct any source data that does not meet the requirements of the standard format. For example, study IDs must contain only characters A-Z and numbers 0-9, must be capitalized, and must contain 25 characters or less.

Scripted Curation

The following table shows some specific corrections that the stored procedures make to the source data in working tables:

External Table Column	Action
DATA_LABEL	Replaces any pipe () character with a dash (-) character.
DATA_LABEL	Removes the value if it duplicates the last part of <code>CATEGORY_CD</code> .
DATA_VALUE	After removing any pipe () characters from the beginning or end of the value, replaces any remaining pipe character with a dash (-) character.
DATA_VALUE	Removes the following parentheses: <ul style="list-style-type: none"> ■ Left parenthesis with no corresponding right parenthesis: (■ Right parenthesis with no corresponding left parenthesis:) ■ Empty parentheses: () ■ Parentheses containing a space: ()
VISIT_NAME	Sets visit name to null if any of the following is true: <ul style="list-style-type: none"> ■ There is a single visit name for <code>CATEGORY_CD</code>. ■ The visit name is the same as <code>DATA_LABEL</code>. ■ The visit name is the same as <code>DATA_VALUE</code>.

Subject-to-Sample Mapping

With studies involving gene expression or SNP data, the subject ID in the clinical data may not be the same as the sample ID associated with the samples in the sample data. If there are multiple samples per subject (due to different timepoints, tissues, sources, etc.), the sample IDs cannot be the same as the subject ID. You must also map the subject IDs in both sets of data in order to properly load the samples and timepoint data.

The following tables in the `TM_CZ` schema are used to map subject IDs in gene expression (mRNA) datasets:

- `LT_SRC_MRNA_DATA` – The sample data, including sample ID.
- `LT_SRC_MRNA_SUBJ_SAMP_MAP` – Maps the sample ID with the subject ID.

Location of Kettle Files

The Kettle jobs and transformations that perform data loading operations are in the following location:

<https://github.com/transmart/tranSMART-ETL/tree/master/Kettle-GPL/Kettle-ETL>

Kettle properties files are located in the folder `conf` at:

<https://github.com/transmart/tranSMART-ETL.git>

Chapter 2

Loading Raw Clinical Data into i2b2

This chapter describes the resources you must have to load data into the tranSMART i2b2 databases from raw source files. The source files can contain either clinical study data or low-dimensional biomarker data.

There are two different methods for loading raw clinical data into i2b2:

- [Loading Clinical Data Using Kettle](#) (page 14): The loading process using Kettle automates several steps in the curation and loading process.
- [Loading Clinical Data Using Stored Procedures](#) (page 22): The loading process using stored procedures requires you to manually execute steps in the loading process. The method assumes your data has been transformed into the standard format.

Resources

The following table summarizes the resources used to load data into i2b2 and specifies the location of the resources:

Resource	Loading Method	Location	Description
Raw source files (can be assigned any name)	Kettle	Any location.	Files containing clinical trial data or low-dimensional biomarker data.
Standard-format file	Stored Procedures	Any location.	Contains source data in the format that the loading scripts require. Each of your raw data files will be transformed into a file of this standard format. See Standard-Format File – Clinical Data on page 22 for details.
<code>create_clinical_data.kjb</code>	Kettle	See Location on page 11.	Loads source data into i2b2. See Loading Clinical Data Using Kettle on page 14 for details.
<code>i2b2_load_clinical_data</code>	Stored Procedures	Stored procedure in the <code>TM_CZ</code> schema.	Loads source data into i2b2 from working tables. See Loading Clinical Data Using Stored Procedures on page 21 for details.

Loading Clinical Data Using Kettle



For information on loading transformed and curated data that is in the standard format using manual steps, see [Loading Clinical Data Using Stored Procedures](#) on page 22.

Prerequisite tasks map and transform raw clinical data into a standard-format file that Kettle can recognize. The high-level tasks are listed below and described in the sections that follow:

[Step One: Generate Raw Clinical Data Files](#)

[Step Two: Create the Column Mapping File](#) (required)

[Step Three: Create the Word Mapping File](#) (optional)

[Step Four: Create the Record Exclusion File](#) (optional)

[Step Five: Load the Data](#)

Step One: Generate Raw Clinical Data Files

Generate one or more files that contain the clinical data you wish to load into the tranSMART i2b2 schema. Each file should be a tab-delimited text file and should contain one header row that identifies the data columns. Additionally every clinical data file must have a column specifying subject IDs that are unique to each subject in the study. A study may have one or more clinical data files associated with it.

Step Two: Create the Column Mapping File

The column mapping file you create instructs the transformation process to treat specified columns as data values, data labels, etc. This step ensures that your data will be displayed in the desired manner within Dataset Explorer.

A column mapping file is a tab-delimited text file with the following columns of data:

Column	Description	Example
Filename	The name of the raw data file you wish to load into i2b2. If your raw data is distributed across multiple files, they can all be referenced with one column mapping file.	cell_line_001.txt

Column	Description	Example
Category Code	The category code you would like to assign the file. For more information, see Category Codes on page 3.	Cell_Line+Subject_Information
Column Number	The column number within the raw clinical data files that should be mapped.	1
Data Label	The data label you wish to assign the record. For more information, see Data Labels & Data Values on page 4.	<p>Kettle will automatically pull the column heading and use it as the data label. If you would like to map the column heading to a new label, enter it here.</p> <p>The reserved words below instruct kettle to perform specific actions. Reserved words must be fully capitalized to be recognized.</p> <ul style="list-style-type: none"> ▪ OMIT: Skip the column. This can be used to explicitly skip a column. Otherwise, to not load a column of data simply do not call its column number. ▪ SUBJ_ID: Assign data in this column to the Subject ID. ▪ SITE_ID: Assign data in this column to the Site ID. ▪ VISIT_NAME: Assign the data in this column to the Visit Name. ▪ VISIT_NAME_2: Assign the data in this column to the Visit Name and append to the value from VISIT_NAME. ▪ DATA_LABEL: Treat the data in this column as a data label for another column. ▪ \: Use when a column has a data label source column defined and there is no other data label for the column. ▪ SEQ_COL: Use the value in this column to generate distinct observations where all other key values (CATEGORY_CD, VISIT_NAME, DATA_LABEL) are the same. ▪ UNITS: append the value in this column to the value in the DATA_LABEL column.

Column	Description	Example
		<ul style="list-style-type: none"> ■ MIN: use the minimum value of the column as the data value. ■ MAX: use the maximum value of the column as the data value. ■ MEDIAN; use the median value of the column as the data value. <p>Note: Each raw data file must include one column that carries the label <code>SUBJ_ID</code>. All other reserved words are optional.</p>
Data Label Source	<p>Use this column if you wish to use the data in a column as a data label for another column.</p> <p>Note: Do not use this column if you wish to use the column header as a data label for another column.</p>	<p>This field specifies the number of the column that should be used as the source for the data label. This column cannot be left blank for any row where the \ is used in the data label field.</p> <p>In rare circumstances where multiple columns must be integrated together, use the following convention:</p> <ul style="list-style-type: none"> ■ Append an A to the column number if you want to have the value from the other column added as a level in the ontology <i>after</i> the data label of the column. ■ Append a B to the column number if you want to have the value from the other column added as a level in the ontology <i>before</i> the data label of the column (for example, <i>4A</i>, <i>6B</i>, etc.). <p>The default is A if nothing else is specified.</p>
Control Vocab Cd	Use this column if you wish to map the record to a controlled terminology (for example, SNOMED or MedDRA).	SNOMED:L-85B02

Column Mapping Examples

The following examples display various uses for the columns of data in the column mapping File.

To omit a column of data:

Filename	Category Code	Column Number	Data Label	Data Label Source	Controlled Vocab Code
cell_line_001		1	OMIT		

This row indicates that the first column in the raw source file *cell_line_001* should be omitted.

To treat a column of data as a Subject ID:

Filename	Category Code	Column Number	Data Label	Data Label Source	Controlled Vocab Code
cell_line_001		2	SUBJ_ID		

This row indicates that the second column in the raw source file *cell_line_001* should be treated as the **SUBJECT_ID** in the standard-format file.

To treat a column of data as a data label and assign a category code:

Filename	Category Code	Column Number	Data Label	Data Label Source	Controlled Vocab Code
cell_line_001	Sample_Factors	3	Cell_Line_Name		

This row indicates that the third column in the raw source file *cell_line_001* should have the Data Label *Cell_Line_Name* and should reside under the *Sample_Factors* node (category code).

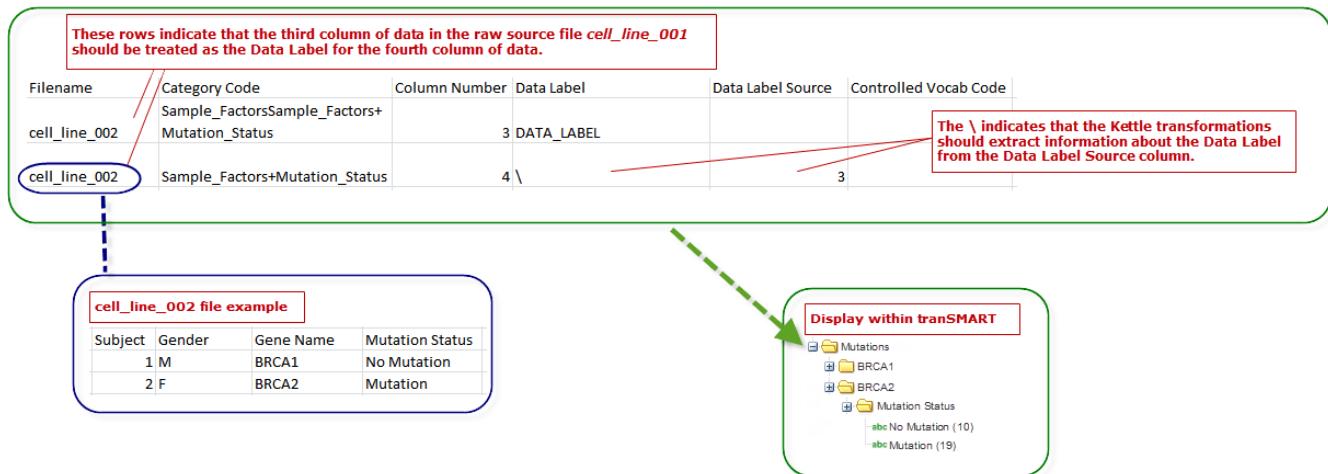
To map a data label to a controlled terminology:

Filename	Category Code	Column Number	Data Label	Data Label Source	Controlled Vocab Code
cell_line_001	Sample_Factors+Demographics	4	Race		L-85B02

This row indicates that the fourth column in the raw source file *cell_line_001* should have the Data Label *Race* and should reside under the *Demographics* node. The Data Label *Race* is mapped to a Controlled Vocabulary Code (SNOMED).

To treat one column of data as a data label for another column of data:

The graphic below illustrates the rows of data within the column mapping file, an example of a raw source file, and how the input of the column mapping file affects the visual display within tranSMART:



Step Three: Create the Word Mapping File

The word mapping file is an optional file that allows a data value for a concept code to be transformed into another data value.

The word mapping file is primarily used to map categorical values to a controlled vocabulary, and also to change unknown and null values into a value that can be displayed in tranSMART. Additionally it can be used to map non-numeric values in numeric fields to nulls so that the data displays correctly.

Word mapping files must have the following characteristics:

- The file name and column number must be defined in the column mapping file (for details, see [Step Two: Create the Column Mapping File](#) on page 14).
- The file must be a tab-delimited text file with the following columns of data:

Column	Description	Example
Filename	The name of the raw data file you wish to load into i2b2 – including the file extension. If the raw data for a study is distributed across multiple files, they can share one word mapping file.	cell_line_001.txt
Column Number	The column number within the raw clinical data files that should be mapped.	1

Column	Description	Example
Original Data Value	The original data value of the record.	carcin0ma
New Data Value	The new data value you wish to display.	Carcinoma



The Kettle job `create_clinical_data.kjb` ignores specific values in a column of data if a period (.) is present. To omit a data value, place a period in the New Data Value column. This is particularly useful for cleansing non-numeric values, such as `unknown value`, out of numeric fields such as `Age`.

Step Four: Create the Record Exclusion File

The record exclusion file is an optional file that allows all data from an input record to be excluded from the data loading based on a value in a column.

Record exclusion files must have the following characteristics:

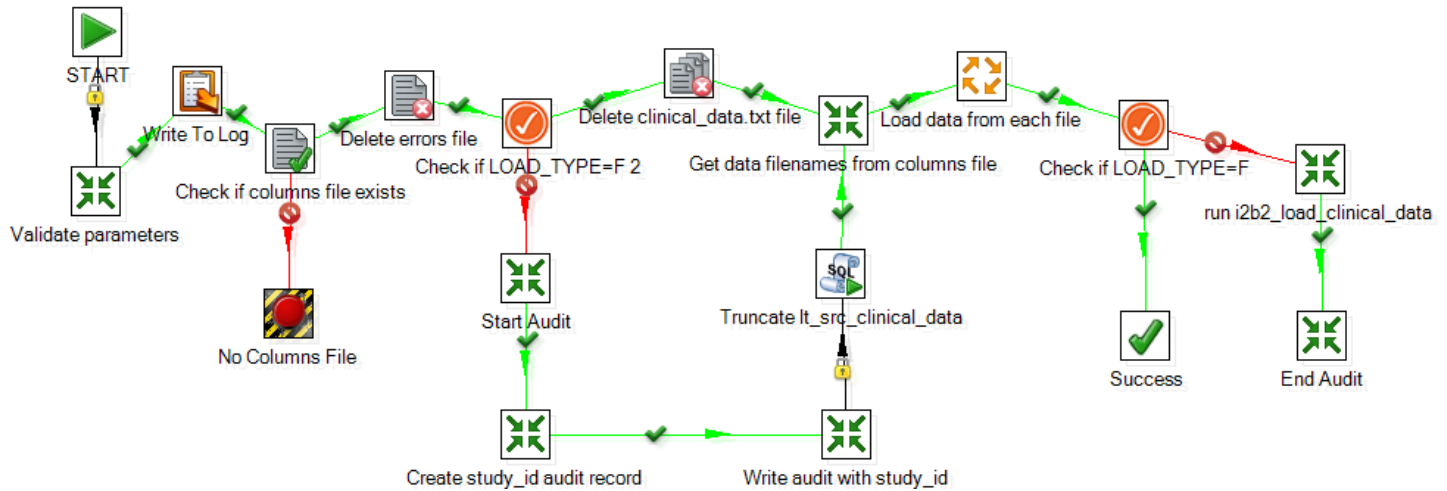
- The file name and column number must be defined in the column mapping file (for details, see [Step Two: Create the Column Mapping File](#) on page 14).
- The file must be a tab-delimited text file with the following columns of data:

Column	Description	Example
Filename	The name of the raw data file that has a column that will be used to exclude all data from that record. If the raw data for a study is distributed across multiple files, they can share one record exclusion file.	cell_line_001.txt
Column Number	The column number within the raw clinical data files that should be mapped.	1
Exclusion Data Value	The data value that will cause all data from the record to be excluded.	Screening

Step Five: Load the Data

The main Kettle job that transforms raw clinical data source files into the standard format is called `create_clinical_data.kjb`.

The Kettle job is illustrated below:



The job is comprised of several sub-jobs and cleansing steps:

High-Level Function	Sub-Jobs
Extract	<ul style="list-style-type: none"> Column mapping file Word mapping file Record exclusion file Raw clinical source data
Transform	<ul style="list-style-type: none"> Drops values of unknowns (.) and omitted columns Sorts data labels by column number and pivots data Inserts backslashes into content Adds a record number to each row Replaces quotation marks (") Merges all files by record number, adding the <code>SUBJID</code> (Subject ID) Imports <code>STUDY_ID</code> from parameters Arranges merged files into the standard format
Load	<ul style="list-style-type: none"> Populates <code>lz_src_clinical_data</code> in <code>TM_LZ</code> schema. Runs the loading script <code>i2b2_load_clinical_data</code>.

Kettle Parameters for Loading Clinical Data

The job `create_clinical_data.kjb` requires you to supply the following parameters for execution:

Parameter	Default Value	Description
COLUMN_MAP_FILENAME	x	Name of column mapping file.
DATA_LOCATION	x	Fully-qualified directory name where files for the study are located.
HIGHLIGHT_STUDY	N	Not used.
LOADER_PATH		Use <code>\$ORACLE_HOME/bin/sqlldr</code> if <code>LOAD_TYPE</code> is L.
LOAD_TYPE	I	<ul style="list-style-type: none"> I = Insert records to <code>lt_src_clinical_data</code> using standard sql statements. L = Insert records to <code>lt_src_clinical_data</code> using <code>sqlldr</code>. F = Create file with standard format records and do not load data. <p>The name of the output file will be <code>STUDY_ID_clinical_data.txt</code>.</p>
RECORD_EXCLUSION_FILE	x	Name of record exclusion file if used, otherwise leave as x.
SECURITY_REQUIRED	N	N enables all users to see the study. If the study requires security, enter Y.
SORT_DIR	<code>%%java.io.tmpdir%%</code>	Default sort directory. Change to a new directory if more space is needed.
STUDY_ID	x	Short name of the study or trial – must be capitalized.
TOP_NODE	x	<p>The string that defines the top nodes of the ontology, including the full name of the study. For example:</p> <pre>\Public Studies\ Breast_Cancer_Kao_GSE20685\</pre>
WORD_MAP_FILE	x	Name of word map file if used, otherwise leave as x.

Loading Clinical Data Using Stored Procedures



For information on loading raw clinical data files using automated steps, see [Loading Clinical Data Using Kettle](#) on page 14.

Loading clinical trial or low-dimensional biomarker data from a raw source file into i2b2 involves the following high-level steps:

1. Transform the data in a raw source file into the standard format described below.
2. Modify the control scripts associated with the loading scripts (`load_clinical_data.ctl` and `load_clinical_data.sh`) to reference the name of the standard-format file you just created from a raw source file.
3. Run the stored procedure `i2b2_load_clinical_data`.

Perform these steps for every raw source file that needs to be loaded into i2b2.

Standard-Format File – Clinical Data

The script `i2b2_load_clinical_data` loads data into the i2b2 databases from files that are in the standard format.

As long as the data files conform to the standard format, no modification of this stored procedure is needed.

The required characteristics of the standard-format file are as follows:

- Tab-delimited text file (hex 09).
- Seven columns of data with no column headings.
- The columns must appear in the same order as the columns in the associated external table.
- With some columns, values are optional. If a record omits a value (value is null) in an optional column, the record must nevertheless have eight distinct columns. Indicate a null value with two consecutive tab characters.

For example, in the following record, a null value is indicated for column 4:

1	2	3	4	5	6	7	8
aa	bb	cc		ee	ff	gg	hh

Columns in the Standard-Format File

The following table lists the columns in the standard-format file. The table lists the columns in the order in which they must appear in the file.

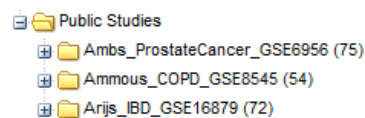
The name shown for each column is the name of the column in the corresponding working tables. The standard-format file itself does not specify column names.

Column Name	Required	Description								
STUDY_ID	Yes	Unique study ID; for example, GSE12345. Must not contain spaces. The study ID is referred to informally as the “short name” of the study. The full name is the name that appears in the i2b2 tree. For information on constructing the full study name, see the note that follows this table.								
SITE_ID	No	Unique ID of the site where the data was collected. This value is for your internal use only. It is not exposed in the i2b2 UI.								
SUBJECT_ID	Yes	ID that was assigned to a study participant. Note: Subject IDs that are unique across tranSMART are constructed by the loader script <code>i2b2_load_clinical_data</code> , and are assigned to <code>SUBJID</code> . The format is: <code>study_id:site_id:subject_id</code>								
VISIT_NAME	No	A free-text field describing a timepoint in the study when data was collected (for example, <code>Baseline</code> or <code>Week 010</code>).								
DATA_LABEL	No	A type of data value collected from the subjects in the study. Labels appear as nodes in the i2b2 tree. Examples: <ul style="list-style-type: none">■ Age■ Baseline Weight (kg)■ Frequency at Home Note: <code>DATA_LABEL</code> is not always required, but is used frequently.								
DATA_VALUE	Yes	A value of a given data type collected from a given subject. Examples: <table><tr><td><u>Label</u></td><td><u>Value</u></td></tr><tr><td>Age</td><td>25</td></tr><tr><td>Baseline Weight (kg)</td><td>68</td></tr><tr><td>Frequency at Home</td><td>Once a week</td></tr></table>	<u>Label</u>	<u>Value</u>	Age	25	Baseline Weight (kg)	68	Frequency at Home	Once a week
<u>Label</u>	<u>Value</u>									
Age	25									
Baseline Weight (kg)	68									
Frequency at Home	Once a week									

Column Name	Required	Description
CATEGORY_CD	Yes	<p>The category code of a given type of data within a given study.</p> <p>For detailed information, see Category Codes on page 3.</p> <p>Note: The script <code>i2b2_load_clinical_data</code> makes the following character transformations when loading data from the external table:</p> <ul style="list-style-type: none"> Changes the plus character (+) to a backslash (\). Changes the underscore character (_) to a space (). <p>The following example shows the category code <code>Subjects+Demographics</code> for the study <code>Ammous_COPD_GSE8545</code>:</p>



Studies are sorted in the i2b2 tree alphabetically by the full study name. For example, the study names in the illustration below are constructed from the primary researcher's last name, the disease name, and the study ID, in that order, separated by underscore characters (_):



If you prefer that studies be grouped by disease name, you would construct the full study name with the disease at the beginning, rather than the researcher's name.

Step One: Edit the Loader Script

The ETL processes for loading clinical data rely on the SQL script `i2b2_load_clinical_data`. You will see an associated control (`.ctl`) and run (`.sh`) file.

The control files (`.ctl`) associated with the loading process must be edited to point to the files you wish to load.

Edit the scripts using `vi` – a modal text editor used with Unix systems. For more information or to download `vi`, visit www.vim.org.

To edit load_clinical_data.cti:

1. Open `load_clinical_data.ctl` using vi.
2. Type the location of your clinical data to the right of **infile**:

```

load data
infile GSE20685_clinical_data.txt
into table lt_src_clinical_data
truncate
fields terminated by X'09'
trailing nullcols
(study_id char(200)
,site_id char(200)
,subject_id char(200)
,visit_name char(200)
,data_label char(200)
,data_value char(200)
,category_cd char(200))

```

"load_clinical_data.ct1" 14L, 298C 1,1 All

3. Save the file.

Step Two: Word Count Command

Execute a word count command prior to running the file loading scripts on both clinical data and subject-to-sample mapping files.

To execute a word count command:

1. Type `$wc -l [filename]`.

For example: `$ wc -l`

2. Click **Execute**.
3. The number of records within your source data is calculated.

Subtract one number from the word count to account for the column header. This number is the total expected or actual record count.

Step Three: Running the Loader Scripts

To execute `load_clinical_data.sh`, type `./load_clinical_data.sh`.

`lt_src_clinical_data` is populated in the `TM_LZ` schema.

The loading process displays a count of the records loaded. Compare this count with the number of expected records in [Step Two](#). If you notice discrepancies, check that you followed the correct procedure starting with [Step One](#).

Step Four: Execute Stored Procedure `i2b2_load_clinical_data`

To execute the stored procedure `i2b2_load_clinical_data`:

1. Open SQL Developer as `tm_cz` user.
2. Run the following command:

```
begin
i2b2_load_clinical_data (STUDYID, TOPNODE, SECURE_STUDY,
HIGHLIGHT_STUDY);
end;
```

The following table describes the fields in the command above:

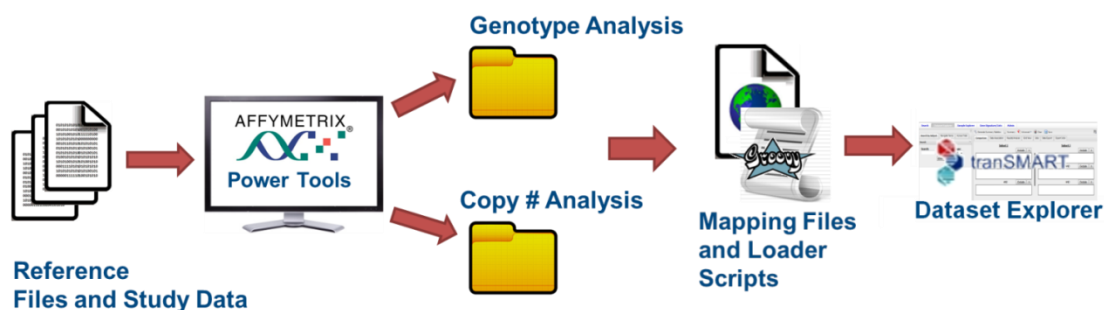
Field	Description	Example
STUDYID	The short name of the study enclosed by single quotes.	'GSE12345'

Field	Description	Example
TOPNODE	The fully-qualified path to the top level of the study including leading and ending backslashes, all enclosed by single quotes.	'\Public Studies\Lung_Cancer_Smith_GSE12345'
SECURE_STUDY	N allows all users to see the study. If the study requires security, enter Y.	
HIGHLIGHT_STUDY	'N'	

Chapter 3

Loading SNP Data into DEAPP

The process for loading SNP data into the DEAPP schema involves extracting genotype and copy number variation from raw data files. This chapter describes the loading process for loading Affymetrix platforms using Affymetrix Power Tools. The graphic below illustrates the high-level processes associated with loading SNP data into tranSMART:



Resources

The following table summarizes the resources involved in loading SNP data into i2b2 and specifies the location of the resource:

Resource	Location	Description
Software (such as Affymetrix Power Tools) to transform raw SNP data into normalized text files.	Any location that does not require administrative privileges to write files. For example: C:\Users\TestUser\Desktop\	Software that analyzes raw SNP data and transforms the analysis into required reference files. For more information on creating reference files, see Running Affymetrix Power Tools on page 36.

Resource	Location	Description
Standard-format file (<code>sample_variance.txt</code>)	Within the folder that contains your copy number variation data.	Contains source data in the format that the loading scripts require. Each of your raw data files will be transformed into a file of this standard format. See Creating a Sample Variance File on page 35 for details.
Loader Scripts	<code>/transmart/etl/pipeline</code>	Loads source data into i2b2 and populates DEAPP tables. See Loading the Data on page 41 for details.

Prerequisite Tasks

Prerequisite tasks involve the following high-level steps:

- Downloading, extracting, and installing Affymetrix Power Tools and other reference data
- Modifying the PATH environment variable
- Creating reference files and directories



You will need to register for a free account with Affymetrix prior to downloading tools and reference files.

Downloads

You will need to download and install the following files before SNP analysis can be conducted:



The installation and configuration of the following tools should be run from an account with administrative privileges.

Program or File Name	Description	Download Link	Extract To
Affymetrix Power Tools (APT)	<p>Command line tool used for calculating both the genotype and copy number for SNP data.</p> <p>For information on supported platforms, see http://media.affymetrix.com/support/developer/powertools/changelog/PLATFORMS.html</p>	http://www.affymetrix.com/partners_programs/programs/developer/tools/devnettools.affx	<p>Extract to a similar location:</p> <p>C:\Users\TestUser\Desktop\Affymetrix Power Tools</p>
Copy Number Files	<p>Copy number data for the following mapping arrays:</p> <ul style="list-style-type: none"> Mapping500K Mapping100K Mapping10K 	http://www.affymetrix.com/Author/support/developer/downloads/Tools/cn-1.5.6_v3.2-release-executables.tar.bz2	<p>Extract to the bin folder within the Affymetrix Power Tools directory:</p> <p>...\Affymetrix Power Tools\apt-1.14.3.1\bin</p>
Library Files	<p>Library files for the following mapping arrays:</p> <ul style="list-style-type: none"> 500K 100K 10K <p>Note: The 500K library file contains parameter files needed for other types of analyses. Download the file regardless of whether you intend to use 500K library files.</p>	<p>Library files can be found in the following locations:</p> <ul style="list-style-type: none"> 500K: http://www.affymetrix.com/Author/support/developer/downloads/Tools/cn-1.5.6_v3.2-release-lib.tar.bz2 100K: http://www.affymetrix.com/Author/support/developer/downloads/Tools/cn-1.5.6_v3.2-release-100K.tar.bz2 10K: http://www.affymetrix.com/Author/support/developer/downloads/Tools/cn-1.5.6_v3.2-release-10K.tar.bz2 	<p>Extract to a similar folder located within the Affymetrix Power Tools directory:</p> <p>...\Affymetrix Power Tools\apt-1.14.3.1\Library Files</p>

Program or File Name	Description	Download Link	Extract To
CEL files	Download CEL files from the source – in this example, Public Study GSE14860 is used (GSE14860_RAW.tar).	To follow the example in the chapter, Public Study GSE14860 can be downloaded here: http://www.ncbi.nlm.nih.gov/projects/geo/query/acc.cgi?acc=GSE14860	Extract to a similar location: ...\\Affymetrix Power Tools\\apt-1.14.3.1\\GSE14860

Setting the PATH Environment Variable

The PATH environment variable in the Windows Operating System lets you specify a set of directories where executable programs and other important files can be found.



It is not required that you set the PATH variable for the Affymetrix Power Tools directory to process SNP data. Doing so allows you to execute analyses via the command line without providing a complete path. It is advised that the PATH variable be set if you intend to process multiple studies.

Add the Affymetrix Power Tools bin folder

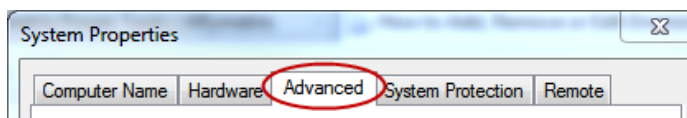
(C:\\Users\\TestUser\\Desktop\\Affymetrix Power Tools\\apt-1.14.3.1\\bin) to the PATH variable by modifying the Windows environment variable settings.

To add the Affymetrix Power Tools bin folder to the PATH:

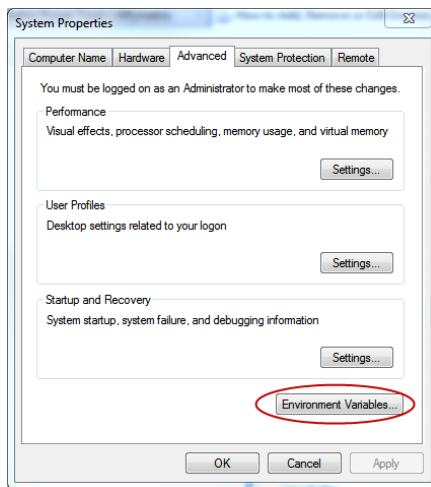
1. Right-click the **Computer** icon located on your desktop (alternatively, you may access it via the **Start** menu).
2. Select the **Properties** option.
3. Click **Advanced system settings**:



4. Select the **Advanced** tab:



- Click the **Environment Variables** button:



- Under **System Variables** select the **Path** variable, then click **Edit**.

The Edit System Variable dialog box appears.

- Under **Variable value**, type the location of the `bin` file you specified when you installed Affymetrix Power Tools.
- Click **OK**.
- Click **Apply**, then click **OK** to close the dialog box.

Creating Reference Folders

Reference folders contain the sample IDs in CEL (.CEL) file formats. You will need to create a folder for each type of platform used in the experiment ([Mapping50K_Hind240], [Mapping50K_Xba240], Agilent-011521, etc.).

In the example provided, the two created folders would have the following paths:

- C:\Users\TestUser\Desktop\Affymetrix Power Tools\apt-1.14.3.1\GSE14860\Hind
- C:\Users\TestUser\Desktop\Affymetrix Power Tools\apt-1.14.3.1\GSE14860\Xba

Creating a CEL Reference File

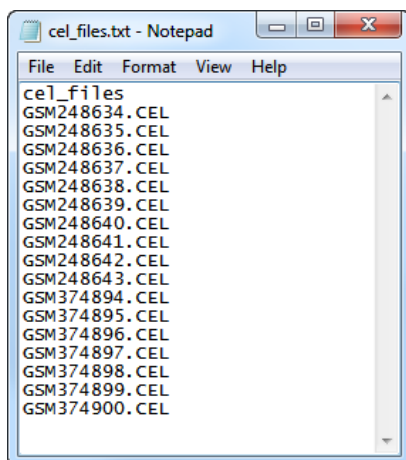
After you have downloaded study data from the source (.CEL files), they will need to be separated based on the type of platform (for example, Mapping50K_Hind240 and Mapping50K_Xba240).



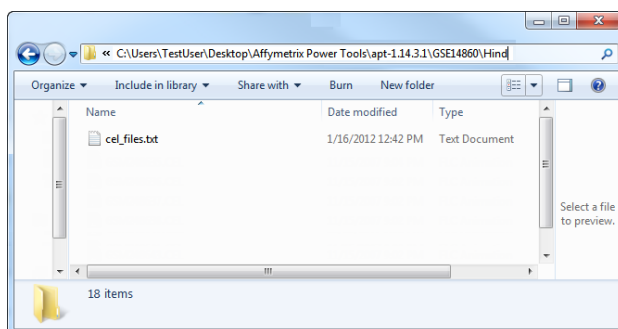
In cases where you are loading Public Study data, the Sample ID of the applicable test group is provided within the Gene Expression Omnibus (GEO) website. If you are using private data, you will need to parse the Sample IDs by platform manually or by using an internal tool.

To create a .CEL reference file:

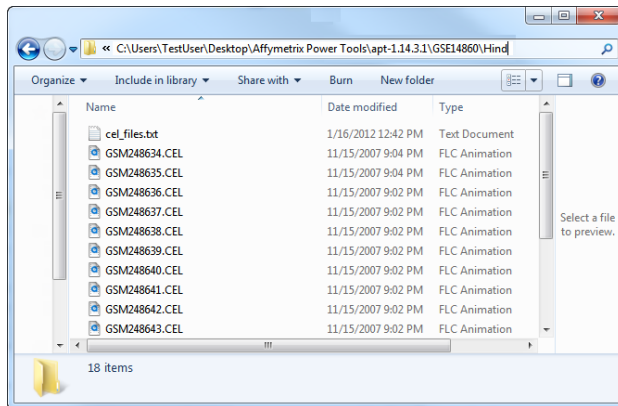
1. Open **Notepad** or a similar tool (click **Start > All Programs > Accessories > Notepad**).
2. Type **cel_files** in the header of the .txt file.
3. List each sample ID for the relevant platform – one per row:



4. Save the file as **cel_files.txt** in the relevant platform folder created in the previous section, [Creating Reference Folders](#):



5. Move all applicable CEL (.CEL) files you listed in the newly-created **cel_file.txt** into the applicable folder:



Creating a Sample Variance File

Creating a sample variance file is required for running copy number variation. The file should be a tab-delimited text file.



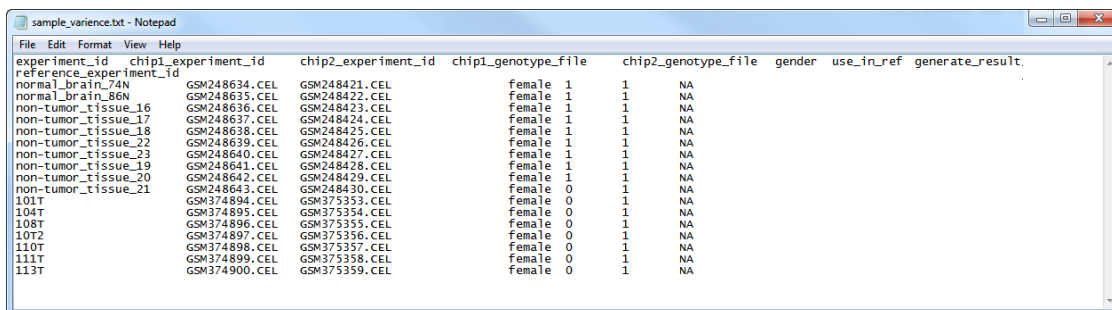
Verify that the .CEL file names have the same capitalization as the original file.

The table below describes the columns of data in `sample_variance.txt`:

Column	Description
experiment_id	The subject ID.
chip1_experiment_id	Sample ID (.CEL file) from the first chip. This field is case-sensitive.
chip2_experiment_id	Sample ID (.CEL file) from the second chip. This field is case-sensitive.
chip1_genotype_file	.chp file if you are running a paired comparison.
chip2_genotype_file	.chp file if you are running a paired comparison.
gender	The gender of the subject.
use_in_ref	Indicates whether the sample is considered normal (reference data) or not. <ul style="list-style-type: none"> ▪ 1 indicates that the sample is normal. ▪ 0 indicates that the sample is not normal. Note: There is a minimum requirement of 10 normal samples to provide reference information.

Column	Description
<code>generate_result</code>	Indicates whether to generate a result for the row. You would use this in cases where, for example, you are only interested in generating results for the non-normal samples. <ul style="list-style-type: none"> ▪ 1 indicates that a result should be generated. ▪ 0 indicates that a result should not be generated.
<code>reference_experiment_id</code>	Used when a paired experiment is run. The ID refers to the corresponding sample in the reference experiment. <p>Note: The examples in this chapter do not detail the processes for a paired experiment.</p>

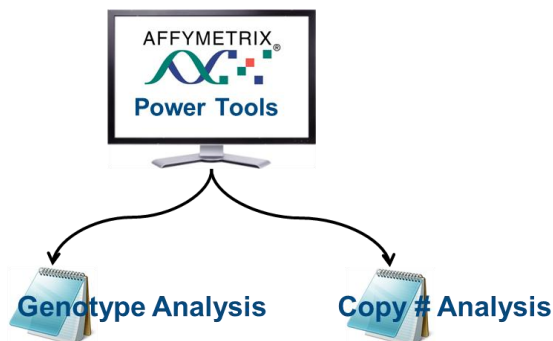
The graphic below illustrates an example of the `sample_variance.txt` file created for Public Study GSE14860:



experiment_id	chip1_experiment_id	chip2_experiment_id	chip1_genotype_file	chip2_genotype_file	gender	use_in_ref	generate_result
normal_brain_74N	GSM248634.CEL	GSM248421.CEL	female 1	1	NA		
normal_brain_86N	GSM248635.CEL	GSM248422.CEL	female 1	1	NA		
non-tumor_tissue_16	GSM248636.CEL	GSM248423.CEL	female 1	1	NA		
non-tumor_tissue_17	GSM248637.CEL	GSM248424.CEL	female 1	1	NA		
non-tumor_tissue_18	GSM248638.CEL	GSM248425.CEL	female 1	1	NA		
non-tumor_tissue_22	GSM248639.CEL	GSM248426.CEL	female 1	1	NA		
non-tumor_tissue_23	GSM248640.CEL	GSM248427.CEL	female 1	1	NA		
non-tumor_tissue_19	GSM248641.CEL	GSM248428.CEL	female 1	1	NA		
non-tumor_tissue_20	GSM248642.CEL	GSM248429.CEL	female 1	1	NA		
non-tumor_tissue_21	GSM248643.CEL	GSM248430.CEL	female 0	1	NA		
101T	GSM374894.CEL	GSM375353.CEL	female 0	1	NA		
104T	GSM374895.CEL	GSM375354.CEL	female 0	1	NA		
108T	GSM374896.CEL	GSM375355.CEL	female 0	1	NA		
1072	GSM374897.CEL	GSM375356.CEL	female 0	1	NA		
110T	GSM374898.CEL	GSM375357.CEL	female 0	1	NA		
111T	GSM374899.CEL	GSM375358.CEL	female 0	1	NA		
113T	GSM374900.CEL	GSM375359.CEL	female 0	1	NA		

Running Affymetrix Power Tools

Affymetrix Power Tools calculates two types of measurements from raw SNP data:



The analyses result in files that can be processed by the associated loader scripts into DEAPP.

To Measure Genotype

Processing genotype data allows end users of tranSMART to access the *genotype call* for a particular SNP. For more information on SNP's, see:

http://www.ornl.gov/sci/techresources/Human_Genome/faq/snps.shtml.



The genotype calculations should be processed before copy number variation. The output of the genotype calculation is required to process copy number variation.



In this example, you will see commands utilize the full path to a desired file. This will be different if you have modified the PATH environment variable. See [Setting the PATH Environment Variable](#) on page 32 for details.

apt-probeset-genotype

The `apt-probeset-genotype` command generates genotype calls from Affymetrix SNP microarrays. The command will be used once for each type of platform specified in [Creating Reference Folders](#) on page 33. In the example provided, the command should be run twice. – once for each of the chip types (Hind and Xba).

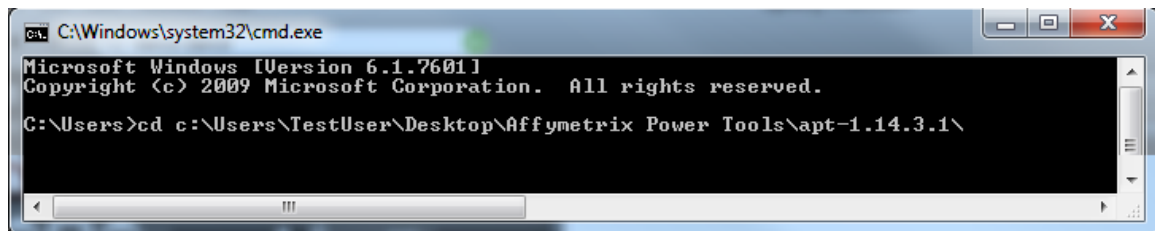
The command requires the following arguments:

Command Name	Description	Example
<code>-o</code>	The path to the location of output files.	<code>"C:\Users\TestUser\Desktop\Results\Hind"</code>
<code>-c</code>	The path to the Chromosome Definition File (.cdf) that defines probe sets.	<code>"C:\Users\TestUser\Desktop\Affymetrix Power Tools\APT-1.14.3.1\Library Files\Mapping50K_Hind240.cdf"</code>
<code>--chrX-snps</code>	The path to the file containing SNPs on chrX.	<code>"C:\Users\TestUser\Desktop\Affymetrix Power Tools\APT-1.14.3.1\Library Files\Mapping50K_Hind240.chrx"</code>
<code>--summaries</code>	Instructs the program to output the summary values from the quantification method for each allele. Note: This argument creates the file needed to process copy number variation (<code>brlmm.summary.txt</code>)	<code>-- summaries</code>

Command Name	Description	Example
--cel-files	The path to the CEL reference file created in Creating a CEL Reference File on page 34.	"C:\Users\TestUser\Desktop\Affymetrix Power Tools\APT-1.14.3.1\GSE14860\Hind\cel_files.txt"

To execute the apt-probeset-genotype command:

1. Click **Start > All Programs > Accessories > Command Prompt** to open a command prompt.
2. Change the directory to the Affymetrix Power Tools location you specified earlier:



3. Run the **apt-probeset-genotype** command by typing the following:



Enter the following command on one line only.

```
apt-probeset-genotype -c "path_to_location" --chrX-snps
"path_to_location" -o "path_to_location" --summaries --cel-files
"path_to_location"
```

4. Press **Enter**.
5. Repeat steps 1 through 4 for each type of platform used (in this example, Hind and Xba).

To Measure Copy Number Variation

Processing copy number data allows end users of tranSMART to find copy number changes on a per-sample basis with respect to a reference set of samples.



The copy number calculations should be processed after genotype. The output of the genotype calculation is required to process copy number variation.



In this example, commands specify the full path to a desired file. Specifying a full path is unnecessary if you have added the PATH environment variable. See [Setting the PATH Environment Variable](#) on page 32 for details.

copynumber-pipeline

The `copynumber-pipeline` command produces copy number variation data from Affymetrix SNP microarrays.

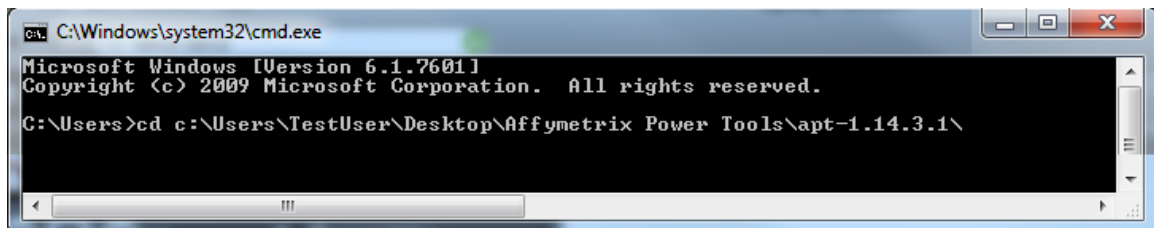
The command takes the following arguments:

Argument	Description	Example
<code>--workflow unpaired- copy-number</code>	A setting that alerts the program to run an unpaired copy number calculation.	<code>--workflow unpaired-copy-number</code>
<code>--chip1- summ</code>	The path to the summary file generated for Chip1.	"C:\Users\TestUser\Desktop\ Results\GT\Hind\brlmm.summary.txt"
<code>--chip1- summ</code>	The path to the summary file generated for Chip2.	"C:\Users\TestUser\Desktop\ Results\GT\Xba\brlmm.summary.txt"
<code>--chip1- snpinfo</code>	The path to the processed .tsv file for the first platform used.	"C:\Users\TestUser\Desktop\Affymetrix Power Tools\APT-1.14.3.1\Library Files\Mapping50K_Hind240_ncbi35.tsv"
<code>--chip2- snpinfo</code>	The path to the processed .tsv file for the second platform used.	"C:\Users\TestUser\Desktop\Affymetrix Power Tools\APT-1.14.3.1\Library Files\Mapping50K_Xba240_ncbi35.tsv"
<code>--out-dir</code>	The path to the location of output files.	"C:\Users\TestUser\Desktop\Results\cnv"
<code>--chip1- out-dir</code>	The path to the location of output files for Chip1.	"C:\Users\TestUser\Desktop\Results\cnv\Hind"
<code>--chip2- out-dir</code>	The path to the location of output files for Chip2.	"C:\Users\TestUser\Desktop\Results\cnv\Xba"

Argument	Description	Example
--sample-info	The path to the Sample Variance file created earlier. For details, see Creating a Sample Variance File on page 35.	"C:\Users\TestUser\Desktop\Results\cnv\sample_variance.txt"
--cn-hmm-paramf	The path to the library files you extracted earlier. For details, see Downloads on page 31.	"C:\Users\TestUser\Desktop\Affymetrix Power Tools\apt-1.14.3.1\Library Files\lib\parameters\cn_hmm_input_sumLog100k.tsv"

To execute the copynumber-pipeline command:

1. Click **Start > All Programs > Accessories > Command Prompt** to open a command prompt.
2. Change the directory to the Affymetrix Power Tools location you specified earlier:



3. Run the **copynumber-pipeline** command by typing the following:



Enter the following command on one line only.

```
copynumber-pipeline --workflow unpaired-copy-number --chip1-summ
"<path to location>" --chip2-summ "<path to location>" --chip1-
snpinf "<path to location>" --chip2-snpinfo "<path to location>" -
out-dir "<path to location>" --chip1-out-dir "<path to location>" --
chip2-out-dir "<path to location>" --sample-info "<path to location>"
--cn-hmm-paramf "<path to location>"
```

4. Press **Enter**.

Loading the Data

The command `loader.jar`, in combination with several configuration files, loads SNP data from processed source files into DEAPP. The data loading workflow involves the following high-level steps:

- Creating a subject-to-sample mapping file.
- Processing annotation data.
- Populating meta tables.
- Pivoting processed SNP data.
- Loading SNP data, including call and copy number data.

The loading process uses binary PLINK files to arrange and distribute data easily.



PLINK is an open source genome association analysis tool. For more information on the functions and features of PLINK, visit <http://pngu.mgh.harvard.edu/~purcell/plink>.

Step One: Subject-to-Sample Mapping

Processed SNP data requires a subject-to-sample mapping file to be loaded into DEAPP.

The required characteristics of the standard-format file are as follows:

- Tab-delimited text file (hex 09).
- Eight columns of data with no column headings.
- With some columns, values are optional. If a record omits a value (value is null) in an optional column, the record must nevertheless have eight distinct columns. Indicate a null value with two consecutive tab characters.

For example, in the following record, a null value is indicated for column 4:

1	2	3	4	5	6	7	8
aa	bb	cc		ee	ff	gg	hh

The following table lists the columns in the standard-format file used to map subjects to samples, as well as provides necessary platform and sample metadata. The table lists the columns in the order in which they must appear in the file. The standard-format file itself does not specify column names.

Column Name	Required	Description
STUDY_ID	Yes	Unique study ID; for example, GSE12345. Must not contain spaces. The study ID is referred to informally as the “short name” of the study. The full name is the name that appears in the i2b2 tree.
SITE_ID	No	Unique ID of the site where the data was collected. This value is for your internal use only. It is not exposed in the i2b2 UI.
SUBJECT_ID	Yes	ID that was assigned to the study participant. Note: Subject IDs that are unique across tranSMART are constructed by the loader script <code>i2b2_process_mRNA_data</code> , and are assigned to <code>USUBJID</code> : <code>study_id:site_id:subject_id</code>
SAMPLE_ID	Yes	ID that was assigned to the sample for the study. If there is only one sample for each patient, the <code>SUBJECT_ID</code> can be used as the <code>SAMPLE_ID</code> .
PLATFORM	Yes	The unique GEO accession number (GPLxxx) used to identify the array or sequencer. Information regarding GPL numbers can be found here: http://www.ncbi.nlm.nih.gov/geo/browse/?view=platforms For custom platforms, a unique identifier may be selected. Scripts responsible for loading data into i2b2 must be edited to match the custom platform’s annotation files.
TISSUETYPE	Yes	A free-text field providing information about the type of tissue; for example, Lung Tissue.
ATTR1	No	A free-text field providing information about an attribute of the sample or data.
ATTR2	No	A free-text field providing additional information about an attribute of the sample or data.

Column Name	Required	Description
CATEGORY_CD	No	<p>The category code of a given type of data within a given study.</p> <p>CATEGORY_CD is a combination of variables and optional text that is expressed as a path in the i2b2 tree – specifically, it is the path between (and not including) the study name and the data label (or the sample type).</p> <p>Optional text and variables are delimited by the + sign. If the optional text contains multiple levels, the text for each level is delimited by a +, and any spaces in the text is replaced by an underscore (_). During the loading process, + signs are replaced by backslashes (\) and underscores are replaced by spaces.</p> <p>The following are <i>reserved</i> variables. The reserved variables must be fully capitalized to be recognized by the loading script. The script will then replace the variable with the value in the applicable field (for example, PLATFORM). Any other text string will be reproduced as is, including strings such as Platform and platform.</p> <ul style="list-style-type: none"> ■ PLATFORM ■ TISSUETYPE ■ ATTR1 ■ ATTR2 <p>For example, if the category code is given as</p> <p>Biomarker_data+GeneExpression+PLATFORM+TISSUETYPE</p> <p>and the reserved category codes have the following values:</p> <p>PLATFORM "GPL96"</p> <p>TISSUETYPE "Vastus Lateralis Muscle"</p> <p>the path displayed in the i2b2 tree will be:</p> <p>Data+[HG-U133A] Affymetrix Human Genome U133A Array + Vastus Lateralis Muscle</p>

Step Two: Process and Load Annotation Data

Annotation data associates biological data to genetic sequences. In this step, annotation data is processed and loaded into DEAPP.

For information on loading SNP annotation data, see [SNP Platforms](#) on page 72.

Step Three: Populate Meta Tables

In this step, the subject-to-sample mapping file created in [Step One](#) is used to populate tables in I2B2METADATA, I2B2DEMOTDATA, and DEAPP.

To populate meta tables:

1. Edit the `loader.properties` configuration file to point to the correct study information (as well as platform). The configuration file is shown below:

```
driver_class=oracle.jdbc.driver.OracleDriver
url=jdbc:oracle:thin:@localhost:1521:orcl

deapp_username=deapp
deapp_password=deapp

biomart_username=biomart
biomart_password=biomart

searchapp_username=searchapp
searchapp_password=searchapp

i2b2demodata_username=i2b2demodata
i2b2demodata_password=i2b2demodata

i2b2metadata_username=i2b2metadata
i2b2metadata_password=i2b2metadata

# STUDY \\top_level_folder\\Study_Full_Name\\Biomarker_Data\\SNP_Data
source_system_prefix=STUDY
study_name=STUDY
platform=Affymetrix Human Mapping 50K Hind240 and 50K Xba240 SNP
Array

# hardcoded for de_gpl_info for now
#tile -> platform
organism=Homo sapiens
marker_type=SNP
gpl_platform=GPL2004_2005

#source_directory=C:/
snp_base_node=/Top Level
Folder/Study_Full_Name/Biomarker_Data/SNP_Data
source_directory=C:/
subject_sample_mapping=STUDY_subject_sample_mapping.txt
```

```
# loading flags
#skip_i2b2=yes
#skip_i2b2_secure=yes
#skip_patient_dimension=yes
#skip_concept_dimension=yes
#skip_concept_counts=yes
#skip_observation_fact=yes
#skip_de_subject_sample_mapping=yes
#skip_de_gpl_info=yes
```

2. Execute the following command

```
/transmart/share/jdk1.6.0_26/bin/java -cp loader.jar
    com.recomdata.pipeline.loader.loader
```

The command will populate the following tables:

- ☐ I2B2 in I2B2METADATA
- ☐ I2B2_SECURE in I2B2METADATA
- ☐ PATIENT_DIMENSION in I2B2DEMODATA
- ☐ CONCEPT_DIMENSION in I2B2DEMODATA
- ☐ CONCEPT_COUNTS in I2B2DEMODATA
- ☐ OBSERVATION_FACT in I2B2DEMODATA
- ☐ DE_SUBJECT_SAMPLE_MAPPING in DEAPP

Step Four: Pivot Processed SNP Data

This step involves executing scripts that transform and map analyzed SNP data. Data is transformed and arranged into binary PLINK files used by tranSMART to generate visualizations. The commands accomplish the following steps:

- Maps sample_ID to patient_number in I2B2DEMODATA.PATIENT_DIMENSION.
- Transforms analyzed SNP data into long format PLINK files, then pivots the data into multiple PLINK format files by chromosome.
- Transforms SNP call data for DEAPP.DE_SNP_CALLS_BY_GSM.
- Transforms copy number data for DEAPP.DE_SNP_COPY_NUMBER.

To pivot processed SNP data:

1. Edit the configuration file `converter.properties` to point to the correct study information. The configuration file is displayed below:

```

plink=C:/software/plink/plink.exe
cut=C:/Program Files (x86)/GnuWin32/bin/cut.exe

# STUDY: \\top_level_folder\\Study_Full_Name\\Biomarker_Data\\SNP_Data
i2b2_node_prefix==\Top Level Folder\Study_Full_Name\Biomarker_Data\SNP_Data

// used in patient_dimension's SOURCESYSTEM_CD
source_system_prefix=STUDY

study_name=STUDY
source_directory=C:/
output_directory=C:/

# convert CN filename to Sample number (GSM number)
expt_to_gsm=sampleCovariate.csv

# GSM# mapping between SNP and gene expression plink
//gsm_to_gsm=GSM_Mapping.csv
sample_info=STUDY_subject_sample_mapping.txt

# location for Copy Number files
cn_directory=CN
# location for Genotyping files
gt_directory=GT
source_gt_file_pattern=brlmm.calls.txt

driver_class=oracle.jdbc.driver.OracleDriver
url=jdbc:oracle:thin:@localhost:1521:orcl

deapp_username=deapp
deapp_password=deapp

i2b2demodata_username=i2b2demodata
i2b2demodata_password=i2b2demodata
# processing control flag
skip_copy_number_process=no
skip_lgen_file_creation=yes
skip_plink_file_creation=yes

```

2. Execute the following command to pivot data:

```

/transmart/share/jdk1.6.0_26/bin/java -cp loader.jar
com.recomdata.pipeline.converter.Converter

```

Step Five: Load SNP Data into tranSMART

Transformed SNP data, combined with copy number information is loaded into tranSMART by chromosome and by probe.

To load SNP data into tranSMART:

1. Edit the configuration file `PLINK.properties` to point to the correct study information. The configuration file is displayed below:

```
// location of PED and MAP files
//file_location=C:/
//file_location=C:/
file_location=C:/
// GPL2004: [Mapping50K_Hind240] Affymetrix Human Mapping 50K Hind240 SNP Array
// GPL2005: [Mapping50K_Xba240] Affymetrix Human Mapping 50K Xba240 SNP Array
platform=GPL2004_2005
study_name=STUDY
chromosome_prefix=chr
// used in patient_dimension's SOURCESYSTEM_CD
source_system_prefix=STUDY

// JDBC driver
driver_class=oracle.jdbc.driver.OracleDriver
url=jdbc:oracle:thin:@localhost:1521:orcl

deapp_username=deapp
deapp_password=deapp

i2b2demodata_username=i2b2demodata
i2b2demodata_password=i2b2demodata

fam_file_name=STUDY.fam
cn_file_name=STUDY.cn

start_chr=1
end_chr=24

// loading flag
//skip_patient_dimension=yes
//skip_snp_dataset=yes
//skip_snp_sorted_def=yes
//skip_snp_data_by_patient=yes
//skip_snp_data_by_probe=yes
```

2. Execute the following command:

```
/transmart/share/jdk1.6.0_26/bin/java -cp loader.jar
    com.recomdata.pipeline.plink.PLINK.loader
```

Step Six: Load SNP Call Data into tranSMART

This step uses Oracle SQL*Loader to move call data from external files into specified tables.

To load SNP call data into tranSMART:

1. Edit the SNP call data control file (`gsm.ctl`) to reference the correct study data. The content of `gsm.ctl` is displayed below:

```
load data
infile "GSE14860.lgen.gsm"
into table de_snp_calls_by_gsm1
truncate
fields terminated by '\t'
trailing nullcols
(
  GSM_NUM          char(200)
  ,PATIENT_NUM      char(200)
  ,SNP_NAME         char(200)
  ,SNP_CALLS        char(200)
)
```

2. Using Oracle SQL*Loader, execute the following command:

```
sqlldr deapp/<password> control=gsm.ctl direct=yes
```

3. Copy data from the temporary table `DE_SNP_CALLS_BY_GSM1` to the destination table `DE_SNP_CALLS_BY_GSM` using the following SQL script:

```
insert into de_snp_calls_by_gsm nologging
(
  trial_name
  ,gsm_num
  ,patient_num
  ,snp_name
  ,snp_calls
)
select 'GSE14860',
  ,gsm_num
  ,patient_num
  ,snp_name
  ,snp_calls
from de_snp_calls_by_gsm1;
```

Step Seven: Load SNP Copy Number Data into tranSMART

This step uses Oracle SQL*Loader to move call data from external files into specified tables.

To load SNP copy number data into tranSMART:

1. Edit the copy number control file (cn.ctl) to reference the correct study data.
The content of cn.ctl is displayed below:

```
load data
infile "GSE14860.cn"
into table de_snp_copy_number1
truncate
fields terminated by '\t'
trailing nullcols
(
    PATIENT_NUM    char(200)
    ,SNP_NAME      char(200)
    ,CHROM char(200)
    ,CHROM_POS     char(200)
    ,COPY_NUMBER   char(200)
)
```

2. Using Oracle SQL*Loader, execute the following command:

```
sqlldr deapp/<password> control=cn.ctl direct=yes
```

3. Copy data from the temporary table DE_SNP_COPY_NUMBER1 to the destination table DE_SNP_COPY_NUMBER using the following SQL script:

```
insert into de_snp_copy_number nologging
(
    trial_name
    ,patient_num
    ,snp_name
    ,chrom
    ,chrom_pos
    ,copy_number
)
select 'GSE14860',
    ,patient_num
    ,snp_name
    ,chrom
    ,chrom_pos
    ,copy_number
from de_snp_copy_number;
```


Chapter 4

Loading Gene Expression Data into DEAPP

This chapter describes the resources and workflow for loading data into the DEAPP tables from raw source files containing gene expression data.

There are two different methods for loading gene expression data into i2b2:

- [Loading Gene Expression Data Using Kettle](#) (page 57): Using Kettle automates several steps in the curation and loading process.
- [Loading Gene Expression Data Using Stored Procedures](#) (page 59): Using stored procedures requires you to manually execute steps in the loading process. The method assumes your data has been transformed into the standard format.

Before you begin the data loading process using either of the above methods, prepare the data for loading by following the instructions in [Preparing the Data for Loading](#) on page 52.

Resources

The following table summarizes the resources involved in loading data into i2b2 and specifies the location of the resources:

Resource	Location	Description
Software to transform raw expression data into normalized text files. We illustrate an example of how you can treat gene expression data using Affymetrix® Expression Console	Any location.	Software that transforms raw binary files containing gene expression data into tab-delimited text (.txt) files.
FilePivot.jar	Any directory or folder.	Pivots gene expression data to produce files in the standard format.

Resource	Location	Description
Standard-format file (can be assigned any name)	Any location. The applicable control file is updated to indicate the location and name of the standard-format file.	Contains source data in the format that the script <code>i2b2_process_mRNA_data</code> requires. Each of your raw data files will be transformed into a file of this standard format. See Preparing the Data for Loading on page 52 for details.
SQL Loader Files	The <code>sqlldr</code> (SQL Loader) control files and script that run the <code>sqlldr</code> command can be placed in any directory.	Control and command files used for loading files in the standard format.
<code>i2b2_process_mRNA_data</code>	Stored procedure in the <code>TM_CZ</code> schema.	Loads source data into <code>i2b2</code> from <code>lt_src_mrna_data</code> and <code>lt_src_mrna_subj_samp_map</code> . See Step Five: Execute Stored Procedure <code>i2b2_process_mrna_data</code> on page 62 for details.

Preparing the Data for Loading



Use the instructions in this section whether you will load data using Kettle or stored procedures.

File preparation involves putting your source files in the standard format required by the packaged tranSMART ETL processes. There are similar, yet slightly different steps for preparing gene expression files and subject-to-sample mapping files:

- For gene expression files, see [File Preparation for Gene Expression Files](#) below.
- For subject-to-sample mapping files, see [File Preparation for Subject-to-Sample Mapping Files](#) on page 54.

File Preparation for Gene Expression Files

Gene expression files must undergo several transformations to be recognized by the file loader mentioned later in the process. The end result of file preparation is normalized gene expression data by probe and subject. Depending on the platform being used and the normalization required, these steps may differ.



This example illustrates how to process Affymetrix gene expression data with a simple RMA normalization.

Converting gene expression data into the standard format involves the following prerequisite steps:

1. Download and install Affymetrix Gene Expression Console.
2. Download library files in Affymetrix Gene Expression Console that correspond to the platform(s) intended for analysis.
3. Download annotation files in Affymetrix Gene Expression Console that correspond to the platform(s) intended for analysis.

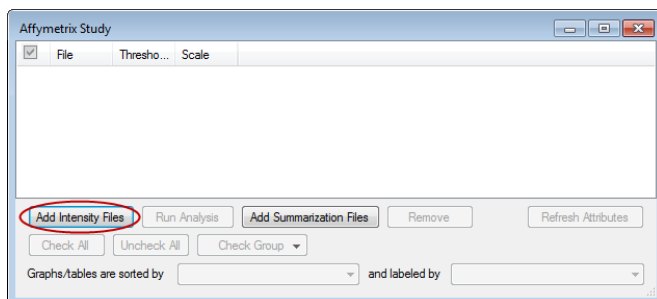
Once you have downloaded and installed all necessary files, you may run analyses within Affymetrix Gene Expression Console to produce un-pivoted text (.txt) files. You will then run a batch file (.bat) that pivots the data into the standard format.

To convert gene expression data into an un-pivoted text file:

1. Open Affymetrix Gene Expression Console.
2. In the **Toolbox**, click **Create New Study**.

The Affymetrix Study dialog box appears.

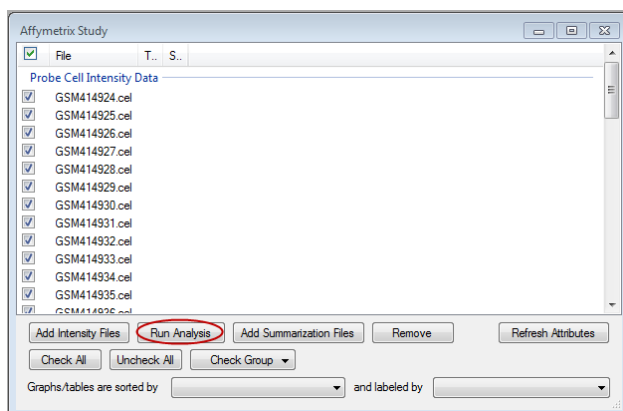
3. Click **Add Intensity Files**:



4. Select the **Probe Cell Intensity File(s)** requiring analysis, then click **Open**.

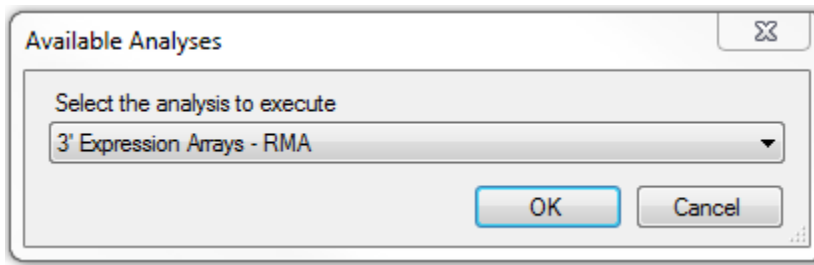
It will take a few minutes to add selected files.

5. Click **Run Analysis**:



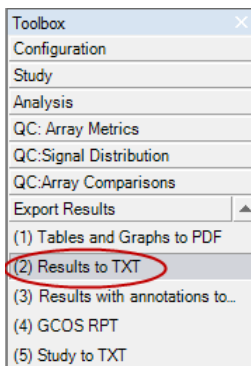
The Available Analyses dialog box appears.

6. Select **3' Expression Arrays - RMA** from the dropdown menu:



If you would like the output of the file to have a suffix, you may add it here, otherwise click **OK** to proceed.

7. After the analysis is complete, click **Export Results**, then select **Results to TXT**:



8. Type the location of the file's destination, then click **Save**.

File Preparation for Subject-to-Sample Mapping Files

Subject-to-sample mapping files provide information on the platform used to generate the gene expression data as well as other attributes of the sample.

Mapping files must be in the standard format described in this section in order for the script `i2b2_process_mRNA_data` to load the contents of the files into i2b2 databases.

The required characteristics of the standard-format file are as follows:

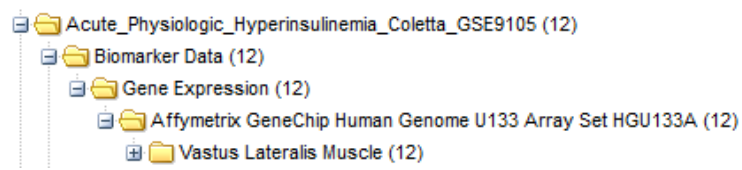
- Tab-delimited text file (hex 09).
- Eight columns of data with no column headings.
- With some columns, values are optional. If a record omits a value (value is null) in an optional column, the record must nevertheless have eight distinct columns. Indicate a null value with two consecutive tab characters.

For example, in the following record, a null value is indicated for column 4:

1	2	3	4	5	6	7	8
aa	bb	cc		ee	ff	gg	hh

The following table lists the columns in the standard-format file used to map subjects to samples, as well as provides necessary platform and sample metadata. The table lists the columns in the order in which they must appear in the file. The standard-format file itself does not specify column names.

Column Name	Required	Description
STUDY_ID	Yes	Unique study ID; for example, GSE12345. Must not contain spaces. The study ID is referred to informally as the “short name” of the study. The full name is the name that appears in the i2b2 tree.
SITE_ID	No	Unique ID of the site where the data was collected. This value is for your internal use only. It is not exposed in the i2b2 UI.
SUBJECT_ID	Yes	ID that was assigned to the study participant. Note: Subject IDs that are unique across tranSMART are constructed by the loader script <code>i2b2_process_mRNA_data</code> , and are assigned to <code>USUBJID</code> : <code>study_id:site_id:subject_id</code>
SAMPLE_ID	Yes	ID that was assigned to the sample for the study. If there is only one sample for each patient, the <code>SUBJECT_ID</code> can be used as the <code>SAMPLE_ID</code> .
PLATFORM	Yes	The unique GEO accession number (GPLxxx) used to identify the array or sequencer. Information regarding GPL numbers can be found here: http://www.ncbi.nlm.nih.gov/geo/browse/?view=platforms For custom platforms, a unique identifier may be selected. Scripts responsible for loading data into i2b2 must be edited to match the custom platform’s annotation files.
TISSUETYPE	Yes	A free-text field providing information about the type of tissue; for example, Lung Tissue.
ATTR1	No	A free-text field providing information about an attribute of the sample or data.
ATTR2	No	A free-text field providing additional information about an attribute of the sample or data.

Column Name	Required	Description
CATEGORY_CD	No	<p>The category code of a given type of data within a given study.</p> <p>CATEGORY_CD is a combination of variables and optional text that is expressed as a path in the i2b2 tree – specifically, it is the path between (and not including) the study name and the data label (or the sample type).</p> <p>Optional text and variables are delimited by the + sign. If the optional text contains multiple levels, the text for each level is delimited by a +, and any spaces in the text is replaced by an underscore (_). During the loading process, + signs are replaced by backslashes (\) and underscores are replaced by spaces.</p> <p>The following are <i>reserved</i> variables. The reserved variables must be fully capitalized to be recognized by the loading script. The script will then replace the variable with the value in the applicable field (for example, PLATFORM). Any other text string will be reproduced as is, including strings such as Platform and platform.</p> <ul style="list-style-type: none"> ■ PLATFORM ■ TISSUETYPE ■ ATTR1 ■ ATTR2 <p>For example, if the category code is given as</p> <p>Biomarker_data+GeneExpression+PLATFORM+TISSUETYPE</p> <p>and the reserved category codes have the following values:</p> <p>PLATFORM "GPL96"</p> <p>TISSUETYPE "Vastus Lateralis Muscle"</p> <p>the path displayed in the i2b2 tree will be:</p> <p>Data+[HG-U133A] Affymetrix Human Genome U133A Array + Vastus Lateralis Muscle</p> 
SOURCE_CD	No	<p>The default is 'STD'. This column is used only if the gene expression data has been normalized using multiple methods and the code is used to differentiate between the sets of data.</p>

Loading Gene Expression Data Using Kettle

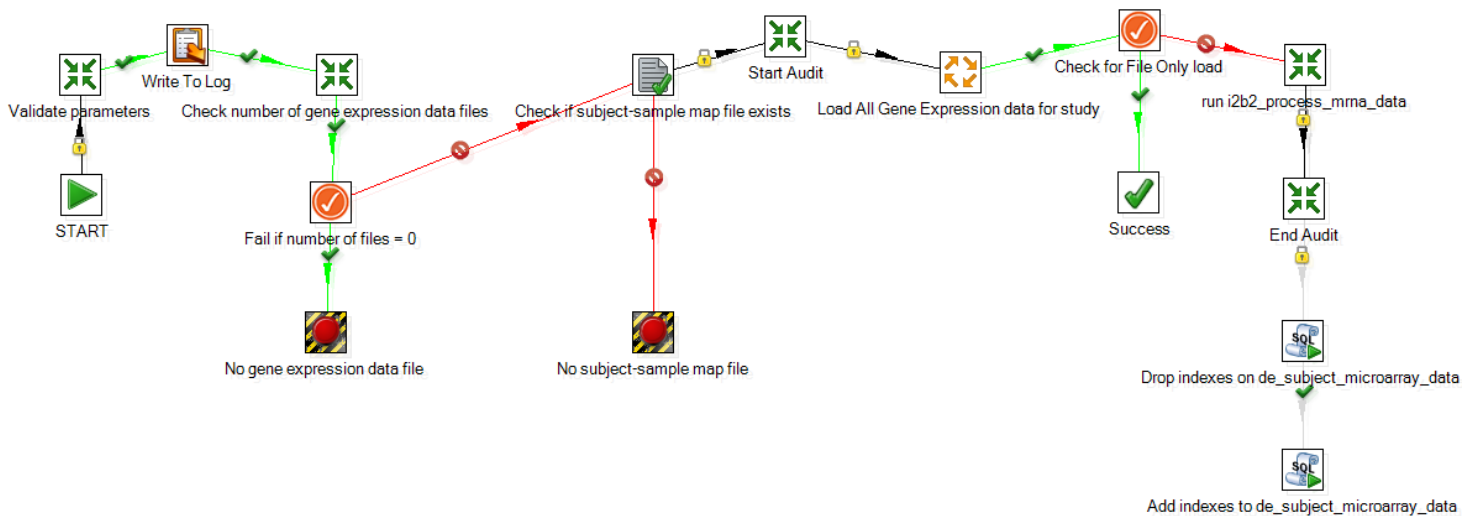


For information on loading gene expression data and subject-to-sample maps using manual steps, see [Loading Gene Expression Data Using Stored Procedures](#) on page 59.

Before you run the Kettle job described in this section, your gene expression data and subject-to-sample mapping files must be in a format that can be recognized by Kettle. For information, see [Preparing the Data for Loading](#) on page 52.

The main Kettle job that loads the prepared gene expression data file and subject-to-sample mapping file into the `TM_LZ` schema is called `load_gene_expressiondata.kjb`.

The Kettle job is illustrated below:



The job contains several sub-jobs and cleansing steps:

High-Level Function	Sub-Jobs
Pivots Gene Expression data	Transforms the gene expression data from a spreadsheet-like format (probe sets as rows, samples as columns) to one row for each probe set/sample combination.
Transforms	Reformats sample codes by removing the suffix string, or re-maps sample codes to easily readable names.
Loads	<ul style="list-style-type: none"> Populates <code>lt_src_mrna_data</code> table in <code>TM_LZ</code> schema. Populated <code>lt_src_mrna_subj_samp_map</code> table in <code>TM_LZ</code> schema. Runs the loading script <code>i2b2_load_clinical_data</code>.

Kettle Parameters for Loading Gene Expression Data

The job `load_gene_expression_data.kjb` requires that you supply the following parameters for execution:

Parameter	Default Value	Description
DATA_FILE_PREFIX	x	The substring that can identify all the filenames that have gene expression data.
DATA_LOCATION	x	Fully-qualified directory name where files for the study are located.
DATA_TYPE	L	<ul style="list-style-type: none"> ▪ R = Raw data. ▪ L = Log transformed data. ▪ T = Fully transformed data.
FilePivot_LOCATION		The name of the full path to the directory or folder where the <code>FilePivot.jar</code> file is located
LOAD_TYPE	I	<ul style="list-style-type: none"> ▪ I = Insert records to <code>lt_src_clinical_data</code> using standard sql statements. ▪ L = Insert records to <code>lt_src_clinical_data</code> using <code>sqlldr</code>. ▪ F = Create file with standard format records and do not load data. <p>The name of the output file will be <code>STUDY_ID_clinical_data.txt</code>.</p>
LOG_BASE	2	The base of log-transformed data.
MAP_FILENAME	N	The name of the subject-to-sample mapping file.
SAMPLE_REMAP_FILENAME	NOSAMPLEREMAP	File containing a list of sample_cd's in gene expression data to be re-mapped. Also, use if sample_cd's are to be removed from the dataset.
SAMPLE_SUFFIX		A string that will be removed from all samples; for example, <code>.CEL</code> , <code>.CEL.gz</code> .
SECURITY_REQUIRED	N	N enables all users to see the study. If the study requires security, enter Y.
SORT_DIR	%%java.io.tmpdir%%	Default sort directory. Change to a new directory if more space is needed.
SOURCE_CD	STD	Value that will be entered in <code>SOURCE_CD</code> column of <code>lt_src_mrna_subj_samp_map</code> .

Parameter	Default Value	Description
STUDY_ID	x	Short name of the study or trial. Must be capitalized.
TOP_NODE	x	The string that defines the top nodes of the ontology, including the full name of the study. For example: \Public Studies\ BreastCancer_Kao_GSE20685\

Loading Gene Expression Data Using Stored Procedures



For information on loading gene expression data and subject-to-sample maps using automated steps, see [Loading Gene Expression Data Using Kettle](#) on page 57.

Before you perform the steps described in this section, your gene expression data and subject-to-sample mapping files must be in a format that can be recognized by the packaged stored procedures. For information, see [Preparing the Data for Loading](#) on page 52.

Step One: Pivot Data

Pivoting data involves running a batch file (.bat) that arranges gene expression data into the standard format. You must edit the batch file to point towards your unpivoted gene expression data.

To pivot Affymetrix gene expression data:

1. Open `Run_FilePivot.bat` using Notepad.
2. Edit the following lines to point to the location of your gene expression data:
 - ☐ `Inputfile=`
 - ☐ `Outputfile=`
3. Run the batch file by double-clicking `Run_FilePivot.bat`.

The output is sent to the destination stated in `outputfile=`.

Step Two: Edit the Loader Scripts

The ETL processes for loading gene expression data rely on the following scripts:

- `load_mRNA_data`
- `load_mRNA_sample_map`

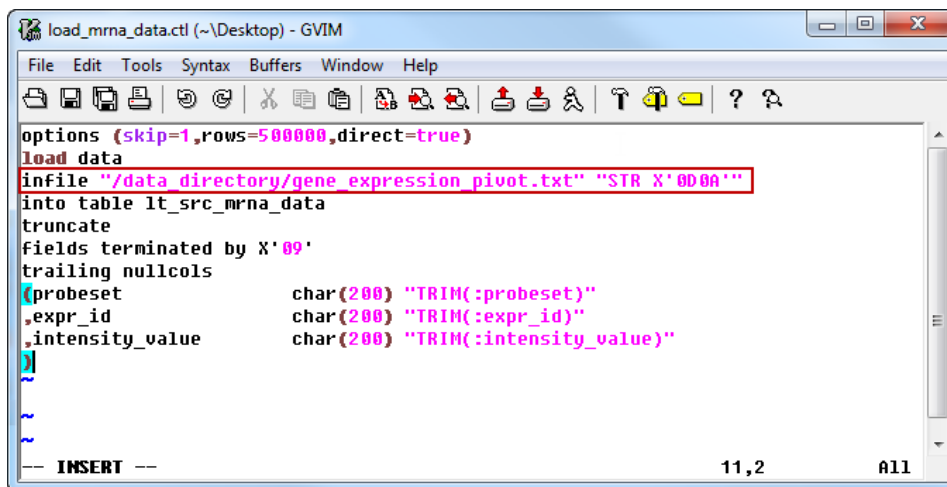
For each data type, you will see an associated control (.ctl) and run (.sh) file.

The control files associated with the loading process must be edited to point to the files you wish to load.

Edit the scripts using `vi` – a modal text editor used with Unix systems. For more information or to download `vi`, visit www.vim.org.

To edit `load_mRNA_data.ctl`:

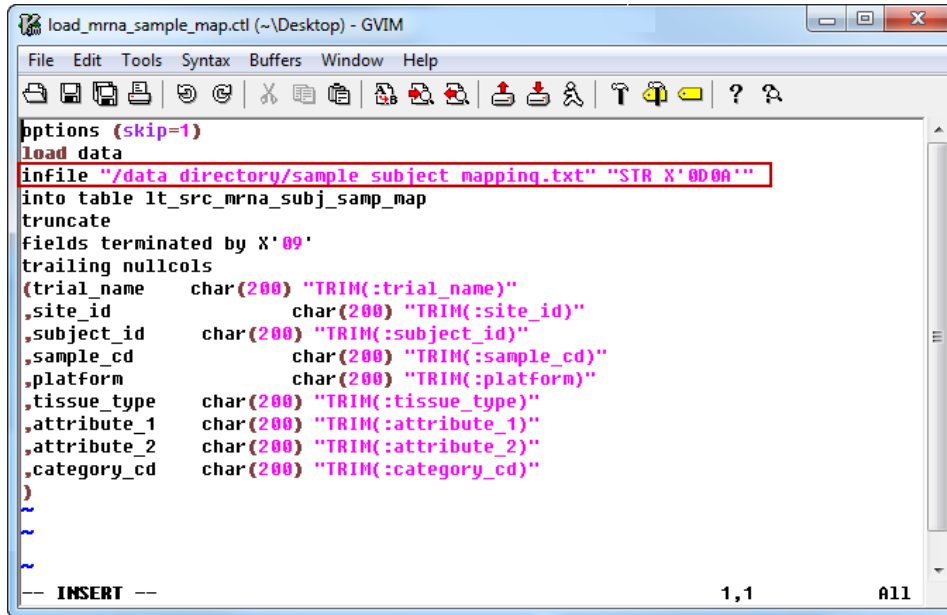
1. Open `load_mRNA_data.ctl` using `vi`.
2. Type the location of your gene expression data to the right of **infile**:



3. Save the file.

To edit load_mRNA_sample_map.ctl:

1. Open load_mRNA_sample_map.ctl using vi.
2. Type the location of your gene expression data to the right of **infile**:



```

load_mrna_sample_map.ctl (~\Desktop) - GVIM
File Edit Tools Syntax Buffers Window Help
options (skip=1)
load data
infile "/data directory/sample subject mapping.txt" "STR X'0000'"
into table lt_src_mrna_subj_samp_map
truncate
fields terminated by X'09'
trailing nullcols
(trial_name      char(200) "TRIM(:trial_name)"
,site_id        char(200) "TRIM(:site_id)"
,subject_id     char(200) "TRIM(:subject_id)"
,sample_cd      char(200) "TRIM(:sample_cd)"
,platform       char(200) "TRIM(:platform)"
,tissue_type    char(200) "TRIM(:tissue_type)"
,attribute_1    char(200) "TRIM(:attribute_1)"
,attribute_2    char(200) "TRIM(:attribute_2)"
,category_cd    char(200) "TRIM(:category_cd)"
)
~
~
~
-- INSERT --
1,1      All

```

3. Save the file.

Step Three: Word Count Command

Execute a word count command prior to running the file loading scripts on both gene expression and subject-to-sample mapping files.

To execute a word count command:

1. Type `$wc -l [filename]`.

For example: `$ wc -l`

The number of records within your source data is calculated.

2. Subtract one number from the word count to account for the column header. This number is the total expected or actual record count.

Step Four: Run the Loader Scripts

The loader scripts can be found in the `TM_CZ` schema. The scripts load data in the `i2b2` location you specified in [Step Two](#).

You should not need to modify the loader script if the standard-format file is organized as described in [Preparing the Data for Loading](#) on page 52

Gene Expression Loader Script

To execute gene expression loader script:

To execute `load_mRNA_data.sh`, type `./load_mRNA_data.sh`.

`lt_src_mrna_data` is populated in the `TM_LZ` schema.

The loading process displays a count of the records loaded.

Compare this count with the number of expected records in [Step Three](#). If you notice discrepancies, check that you followed the correct procedure starting with [Preparing the Data for Loading](#) on page 52.

Subject-to-Sample Mapping Script

To execute `load_mRNA_sample_map.sh`, type `./load_mRNA_sample_map.sh`.

`lt_src_mrna_subj_samp_map` is populated in the `TM_LZ` schema.

The loading process displays a count of the records loaded. Compare this count with the number of expected records in [Step Three](#). If you notice discrepancies, check that you followed the correct procedure starting with [Preparing the Data for Loading](#) on page 52.

Step Five: Execute Stored Procedure `i2b2_process_mrna_data`

To execute the stored procedure `i2b2_process_mrna_data`:

1. Open SQL Developer as `tm_cz` user.
2. Run the following command:

```
declare
  rtn_code    int;
begin
  i2b2_process_mrna_data (STUDYID, TOPNODE, DATATYPE, SOURCE_CD,
    LOGBASE, SECURE_STUDY, null, rtn_code);
end;
```

The following table describes the fields in the command:

Field	Description	Example
STUDYID	The short name of the study enclosed by single quotes.	'GSE12345'
TOPNODE	The fully-qualified path to the top level of the study including leading and ending backslashes, all enclosed by single quotes.	'\Public Studies\Lung_Cancer_Smith_GSE12345'

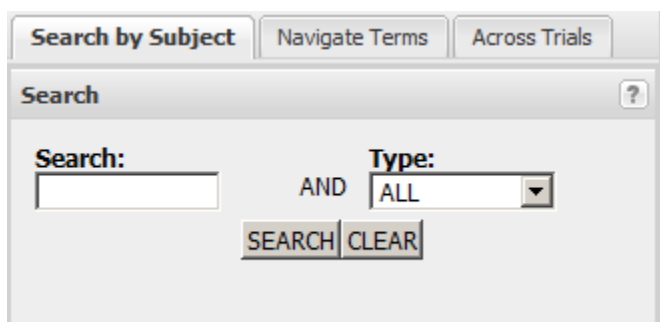
Field	Description	Example
DATATYPE	<p>Possible values:</p> <ul style="list-style-type: none"> ▪ 'R' – Data is raw intensity, no transformation has occurred. Data is log-2 transformed, a z-score is calculated, and the z-score is trimmed to -2.5 to 2.5. ▪ 'L' – data has been log transformed. A z-score is calculated, and the raw intensity is derived using LOGBASE. The z-score is trimmed to -2.5 to 2.5. ▪ 'T' – data is to be loaded "as-is." The value of T is used as the z-score and is trimmed to -2.5 to 2.5. 	
SOURCE_CD	Value that will be entered in the SOURCE_CD column of lt_src_mrna_subj_samp_map.	
LOGBASE	The base of log-transformed data.	
SECURE_STUDY	N enables all users to see the study. If the study requires security, enter Y.	

Chapter 5

Loading Study Metadata and Dataset Explorer Search Subjects

Study metadata is information about the study, such as accession number, title, description, the study's primary investigator, and the institution that conducted the study. The raw source files for study metadata are Microsoft Excel spreadsheets.

The spreadsheet that contains study metadata also contains filters that allow Dataset Explorer users to search for specific types of data (such as compounds or diseases) in the Search by Subject tab:



Input File Format

Study metadata must be in a tab-delimited text file with the following columns of data:

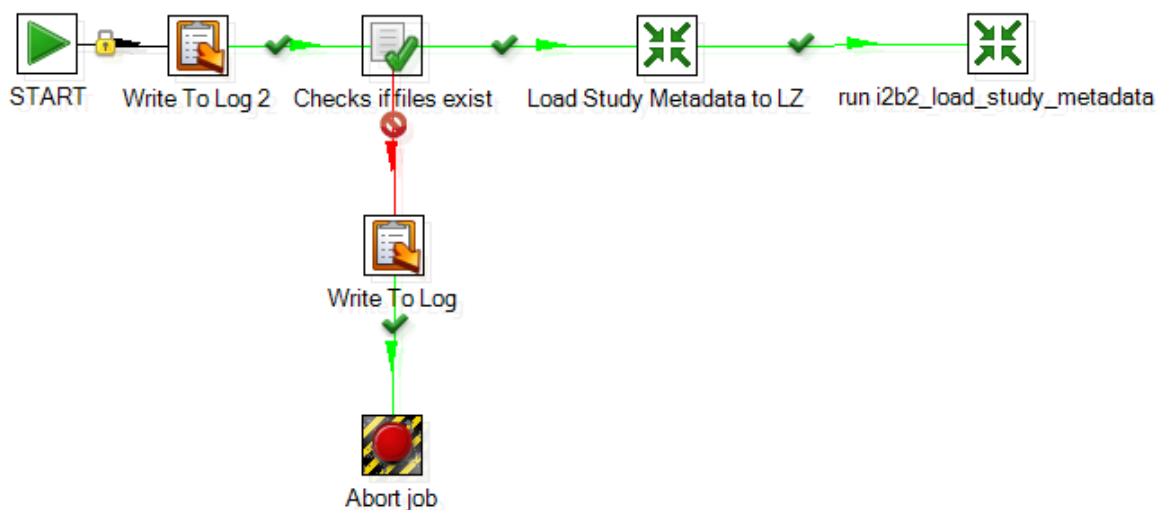
Column	Description	Example
Study Id	The short name or ID of the study. This will be repeated for every row of data that pertains to the study. There may be data for multiple studies in one file.	GSE1234, ADS001

Column	Description	Example
Metadata Type	The metadata that is being described by the data in the Text column.	<p>For standard metadata and Dataset Explorer search subjects, the Metadata Type is one of the following:</p> <ul style="list-style-type: none"> For metadata: DESCRIPTION, TITLE, PRIMARY_INVESTIGATOR, INSTITUTION, PUBMED For search subjects: COMPOUND, DISEASE <p>The name of the Metadata Type for standard metadata fields must be entered exactly as shown above.</p> <p>For ad-hoc metadata, the Metadata Type is user-supplied text.</p>
Text	<p>The text that describes the metadata in Metadata Type.</p> <p>For metadata type of <code>PUBMED</code>, the text is a semi-colon delimited list of PUBMED IDs. A link to the PUBMED article will be available when the study metadata is displayed.</p> <p>For <code>COMPOUND</code> and <code>DISEASE</code> metadata types, the text can be a single compound or disease or a list separated by semi-colons. Each compound or disease will be added to the Search by Subject dropdown in Dataset Explorer.</p>	<p>RNA binding activity of the recessive parkinsonism protein DJ-1</p>

Metadata Loader

The main Kettle job that loads the contents of the study metadata file into the `TM_LZ` schema is called `load_study_metadata.kjb`.

The Kettle job is illustrated below:



`load_study_metadata.kjb` contains sub-jobs that perform the following tasks:

High-Level Function	Sub-Jobs
Transforms	<ul style="list-style-type: none"> Reformats the input file into a single record for study metadata and multiple records for ad-hoc metadata.
Loads	<ul style="list-style-type: none"> Populates table <code>lt_src_study_metadata</code> in <code>TM_LZ</code> schema. Populates table <code>lt_src_study_metadata_ad_hoc</code> in <code>TM_LZ</code> schema. Runs the loading script <code>i2b2_load_study_metadata</code>.

Kettle Parameters for Loading Study Metadata

`load_study_metadata.kjb` requires you to supply the following parameters for execution:

Parameter	Default Value	Description
<code>METADATA_FILENAME</code>	x	Name of the study metadata file.
<code>METADATA_LOCATION</code>	x	Name of the fully-qualified directory where the study metadata file is located.
<code>SORT_DIR</code>		Default sort directory. Change to a new directory if more space is needed.

Loading Platform Annotation Data

This chapter describes how to load annotation data for gene expression and SNP platforms.

Platform annotation data can be downloaded from the platform manufacturer or the NCBI web site. The annotations provided by the platform manufacturer are the most current. You may need to register with the platform manufacturer to be able to download the annotation. For annotations found on the NCBI website, add the platform's GEO accession number to the following URL:

<http://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=>

For example, to access the page for Affymetrix Human Genome U133 Plus 2.0 Array (GPL570), use the following URL:

<http://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GPL570>

Gene Expression Platforms

The main Kettle job for loading gene expression annotation data into the `TM_LZ` schema is `load_GEX_annotation.kjb`. Before you perform the steps below, be sure to supply the required parameters shown in section [Kettle Parameters for Loading Gene Expression Annotations](#) on page 71.

If a preformatted annotation file is used, the columns must be as follows:

Populate this column...	With this data...
gpl_id	GEO accession number for the platform. For example, GPL570 or any other unique identifier for annotation.
probe_id	The probe ID.
gene_symbol	The gene symbol.
gene_id	The gene id.
organism	Scientific name for the species.

Loading the Annotation Data

To load annotation data for a gene expression microarray platform:

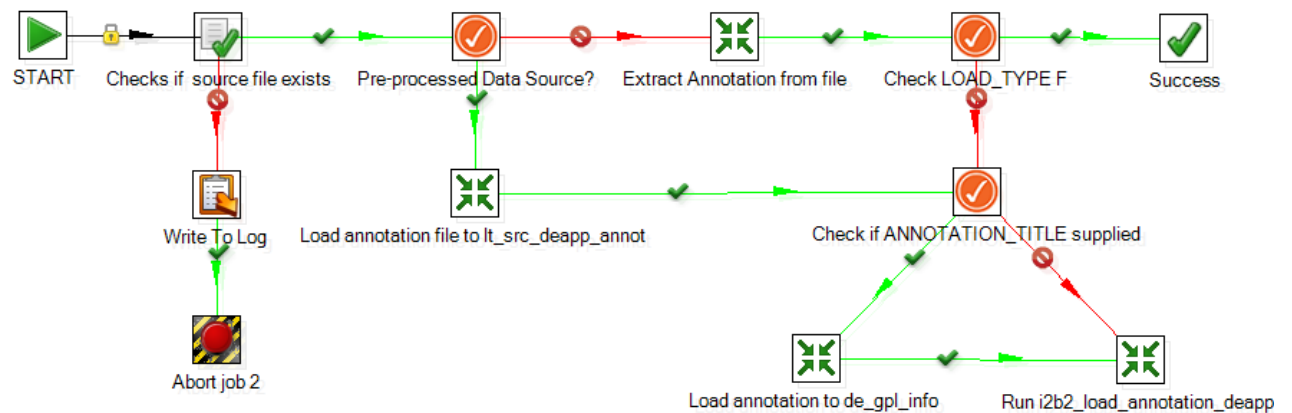
1. Download the data file from the platform manufacturer's website or the appropriate NCBI page.

For example to download data for GPL570 from NCBI, go to the following URL and follow the download instructions:

<http://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GPL570>

2. Convert the downloaded file to a tab-delimited file if it is not already in that format.
3. Run the Kettle job `load_GEX_annotation.kjb`.

The Kettle job is illustrated below:



The job includes several sub-jobs and cleansing steps:

High-Level Function	Sub-Jobs
Transform	Reformats annotation data into a standard input format
Loads	<ul style="list-style-type: none"> Populates table <code>lt_src_deapp_annotation</code> in the <code>TM_LZ</code> schema. Populates table <code>lt_src_mrna_subj_samp_map</code> in the <code>TM_LZ</code> schema. Runs the loading script <code>i2b2_load_clinical_data</code>.

Kettle Parameters for Loading Gene Expression Annotations

The job `load_GEX_annotation.kjb` requires that you to supply the following parameters for execution:

Parameter	Default Value	Description
ANNOTATION_DATE		The annotation date in the format <code>YYYY/MM/DD</code> . Optional.
ANNOTATION_RELEASE		The release number of the annotation. Optional.
ANNOTATION_TITLE	NOTITLE	The name of the annotation. This name will be used as a folder name when the gene expression data is loaded.
DATA_LOCATION		The full path name to the directory or folder where the annotation file is located.
DATA_SOURCE	A	<ul style="list-style-type: none"> ■ A - Data file is an annotation file. ■ P - Data is in a standard format file.
EMBEDDED_GENE_TABLE	N	The gene ID and gene symbol data are in a table that is embedded in one of the columns of the annotation file.
GENETAB_DELIM	//	Field delimiter for an embedded gene table.
GENETAB_ID_COL	-1	Column number of <code>gene_id</code> in the embedded gene table.
GENETAB_REC_DELIM	///	Record delimiter for the embedded gene table.
GENETAB_SYMBOL_COL	-1	Record delimiter for the embedded gene table.
GENE_ID_COL	19	Column number of <code>gene_id</code> .
GENE_SYMBOL_COL	15	Column number of gene symbol.
GPL_ID	GPLXXX	Identifying string for annotation. This is usually <code>GPLXXX</code> , it but can be any unique string.
LOAD_TYPE	I	I - Insert data to <code>TM_LZ.lt_src_deapp_annotation</code> .
ORGANISM_COL	3	Column number of organism. Use -1 if the annotation does not have an organism column.
PROBE_COL	1	Column number of probe ID.

Parameter	Default Value	Description
SKIP_ROWS	1	Number of header rows to skip in the annotation file.
SOURCE_FILENAME	x	Name of the annotation file.

SNP Platforms

SNP annotation data is processed and loaded into `DEAPP`.

To process and load annotation data:

1. Download the platform-associated annotation file.

Following the example used in [Prerequisite Tasks](#) on page 30, you would download Hind240 and Xba240 annotation files from the following Affymetrix site:

<http://www.affymetrix.com/support/technical/byproduct.affx?product=100k>

2. Create a mapping file for PLINK. Each line of the map file should describe a single `.ped` file and must contain the following columns of information:

Column	Description
Chromosome	<ul style="list-style-type: none"> ▪ 1-22 ▪ X ▪ Y ▪ 0 (if unplaced)
rs# or SNP identifier	
Genetic distance	Genetic distance (in morgans).
Base-pair position	Base-pair position (in bp units).

3. Edit the `annotation.properties` configuration file to point to the correct `input_file`. The configuration file is shown below:

```
# STUDY
source_directory=C:/SNP/GPL
input_file=GPL3718-44346.txt;GPL3720-22610.txt
destination_directory=C:/SNP/STUDY

organism=Homo sapiens
output_file=annotation.tsv
snp_mapping_file=mapping.tsv
map_file=STUDY.map
```

```
driver_class=oracle.jdbc.driver.OracleDriver  
url=jdbc:oracle:thin:@localhost:1521:orcl  
username=deapp  
password=deapp
```

4. Execute the following command to load annotation file:

```
/transmart/share/jdk1.6.0_26/bin/java -cp loader.jar  
com.recomdata.pipeline.annotation.AnnotationLoader
```

The command will populate the following tables in DEAPP:

- ☐ DE_SNP_INFO
- ☐ DE_SNP_PROBE
- ☐ DE_SNP_GENE_MAP
- ☐ DE_GPL_INFO

Loading GWAS and eQTL Analysis Data

Analysis data is uploaded through the tranSMART UI, as described in the chapter “Analysis Data Upload” in the *tranSMART User’s Guide*.

The following section describes troubleshooting steps to help diagnose problems that may occur with the analysis upload. It also includes a procedure for manually uploading analysis data.

Troubleshooting

If a problem occurs with an attempt to upload analysis data via the tranSMART UI, and the analysis does not appear in tranSMART, check the value of `STATUS` in `TM_LZ.LZ_SRC_ANALYSIS_METADATA` for the `ANALYSIS_NAME` that failed.

■ If `STATUS=PRODUCTION`

The upload worked successfully. There may be another reason why the analysis is not appearing in the application.

■ If `STATUS=STAGED`

The data was successfully loaded into the staging tables but has not been moved into production. Wait for the nightly processing job to run or run it manually. See [Running the Nightly Processing Job](#) on page 76.

■ If `STATUS=ERROR`

- Check the log file from the nightly processing job. The log files can usually be found in the `transmart/ETL/logs` directory. Frequently, the most recent log file will provide enough detail to fix the error in the data file.
- Fix any errors in the data file. The filename can be found in the `TM_LZ.LZ_SRC_ANALYSIS_METADATA` record, and the corresponding file can be found in the `transmart/uploads` directory.

After you fix all errors, do the following:

1. Update the corresponding `TM_LZ.LZ_SRC_ANALYSIS_METADATA` record to `STATUS=PENDING`.
2. Load the file into staging manually. See [Manually Staging Analysis Data](#) on page 76.
3. Wait for the nightly processing job to run or run it manually. See [Running the Nightly Processing Job](#) on page 76.

Manually Staging Analysis Data

To manually load analysis data into staging:

1. Run the file `load_analysis_stage.sh`, typically located in `transmart/ETL`.
2. Check that the value of `STATUS` in the `TM_LZ.LZ_SRC_ANALYSIS_METADATA` record is set to `STAGED`.

Running the Nightly Processing Job

To run the nightly processing job:

1. Run the file `nightly_processing.sh`, typically located in `transmart/ETL`.
2. Check that the value of `STATUS` in the `TM_LZ.LZ_SRC_ANALYSIS_METADATA` record is set to `PRODUCTION`.
3. Reload the SOLR indexes by running `reload_solr.sh`, typically located in `/solr-rwg/conf`.

Study Security and Study Deletion

Security can be assigned to proprietary studies so that only authorized users can view the studies in Dataset Explorer. Studies that a user does not have permission to access will be grayed out in Dataset Explorer. Users will be able to view the description of a grayed-out study, but will not be able to view the study data.

Public studies should not have any security restrictions applied to them.

Applying Security Restrictions to a Study

Granting access to a secured study is managed through the tranSMART Administrator tab. It is an entirely UI-driven process.

A study is secured when the `SECURITY_REQUIRED` parameter is set to `Y` during the process of loading clinical or gene expression data.

For information on applying security restrictions to a study, see the *tranSMART User Administrator's Guide*.

Deleting a Study

Deleting a study is achieved by editing and running the `i2b2_backout_trial` script available within the tranSMART `TM_CZ` schema.

To remove a study from tranSMART:

1. Open SQL Developer.
2. Open a SQL worksheet for the `TM_CZ` schema.
3. Enter the following three lines on the SQL worksheet:

```
begin
i2b2_backout_trial('study_id','top_node');
end;
```

4. Edit the `study_id` and `top_node` fields within single quotations (`'`) to reference the study data that you wish to delete. The contents of the file are described in the graphic below:

study_id

The **short name** of the study that was used during the loading process (for example, **GSE4475**)

top_node

The **top node** of the study in the ontology (for example, **\Clinical Studies\Lymphoma\Lymphoma_Hummel_GSE4475**)



The leading and trailing slashes associated with the top node of a study must be present within the `top_node` field.

5. Execute `i2b2_backout_trial`.

Appendix A

Schemas

Dataset Explorer requires the following database schemas:

- DEAPP
- I2B2DEMODATA
- I2B2HIVE
- I2B2METADATA

DEAPP Schema

The DEAPP schema contains the following tables:

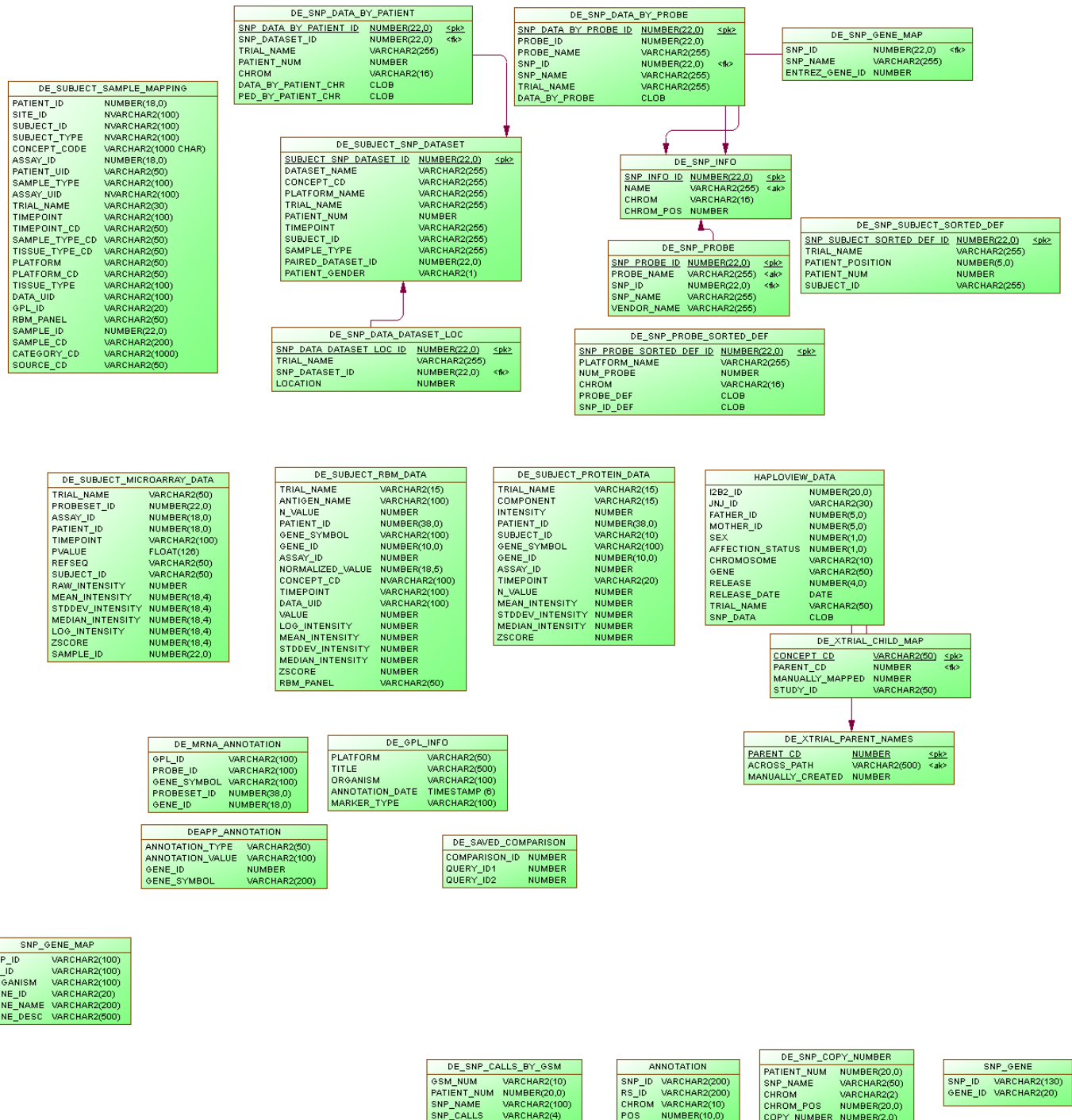
Table	Description
ANNOTATION	Annotation information for SNP data such as SNP ID, RS ID, chromosome, and position on the loci.
DEAPP_ANNOTATION	Contains annotation information by platform.
DE_GPL_INFO	GPL platform IDs and other information (title, organism) related to microarray organizations such as Affymetrix and Illumina.
DE_MRNA_ANNOTATION	GPL platform ID, probe IDs and gene IDs related to gene expression data in table <code>DE_SUBJECT_MICROARRAY_DATA</code> .
DE_SAVED_COMPARISON	Contains the IDs of the subset definitions saved with the Dataset Explorer Save button.
DE_SNP_CALLS_BY_GSM	SNP call data.
DE_SNP_COPY_NUMBER	SNP copy number data.
DE_SNP_DATA_BY_PATIENT	SNP data collected for specific patients in specific studies. For each patient, SNP data is concatenated as a blob in <code>DATA_BY_PATIENT_CHR</code> . Primary key: <code>SNP_DATA_BY_PATIENT_ID</code>

Table	Description
DE_SNP_DATA_BY_PROBE	<p>SNP data associated with specific probe IDs in specific studies.</p> <p>For each probe, SNP data is concatenated as a blob in DATA_BY_PROBE.</p> <p>Primary key: SNP_DATA_BY_PROBE_ID</p>
DE_SNP_DATA_DATASET_LOC	<p>Specifies the location of a particular SNP in a SNP blob.</p> <p>Each SNP is represented as a fixed, 7-character segment of the blob (six characters plus a space).</p> <p>Primary key: SNP_DATA_DATASET_LOC_ID</p>
DE_SNP_GENE_MAP	<p>Association of SNP IDs, SNP names, and Entrez gene IDs.</p> <p>Primary key: SNP_ID</p>
DE_SNP_INFO	<p>Chromosome number and position associated with SNP data.</p> <p>Primary key: SNP_INFO_ID</p>
DE_SNP_PROBE	<p>SNP probe name and ID associated with a SNP ID, SNP name, and platform.</p> <p>Primary key: SNP_PROBE_ID</p>
DE_SNP_PROBE_SORTED_DEF	<p>SNP IDs associated with a platform, probe, probe definition and chromosome (or all chromosomes).</p> <p>For each probe definition, SNP IDs are concatenated as a blob in SNP_ID_DEF.</p> <p>Primary key: SNP_PROBE_SORTED_DEF_ID</p>
DE_SNP_SUBJECT_SORTED_DEF	<p>SNP IDs associated with a trial name, patient number, etc.</p> <p>Primary key: SNP_SUBJECT_SORTED_DEF_ID</p>
DE_SUBJECT_MICROARRAY_DATA	<p>Gene expression data. Includes patient, probe set, and assay IDs, and values for raw intensity, log intensity, and z-scores.</p>
DE_SUBJECT_PROTEIN_DATA	<p>Protein data, including patient, subject, assay, and gene IDs, and gene symbol, timepoint, and intensity values.</p>
DE_SUBJECT_RBM_DATA	<p>RBM data. Includes patient, assay, and gene IDs, gene symbol and antigen names, and intensity values.</p>
DE_SUBJECT_SAMPLE_MAPPING	<p>Table that maps subjects with SNP and RBM and microarray sample data.</p>

Table	Description
DE_SUBJECT_SNP_DATASET	Table that maps subjects with SNP sample data. Primary key: SUBJECT_SNP_DATASET_ID
DE_XTRIAL_CHILD_MAP	Not currently used.
DE_XTRIAL_PARENT_NAMES	Not currently used.
HAPLOVIEW_DATA	SNP data for haploview visualizations.
SNP_GENE	Mapping table used in SNP_GENE_MAP.
SNP_GENE_MAP	Mapping table between SNP data and gene data.

DEAPP Schema Diagram

Click the image to display it in your default graphics viewer:

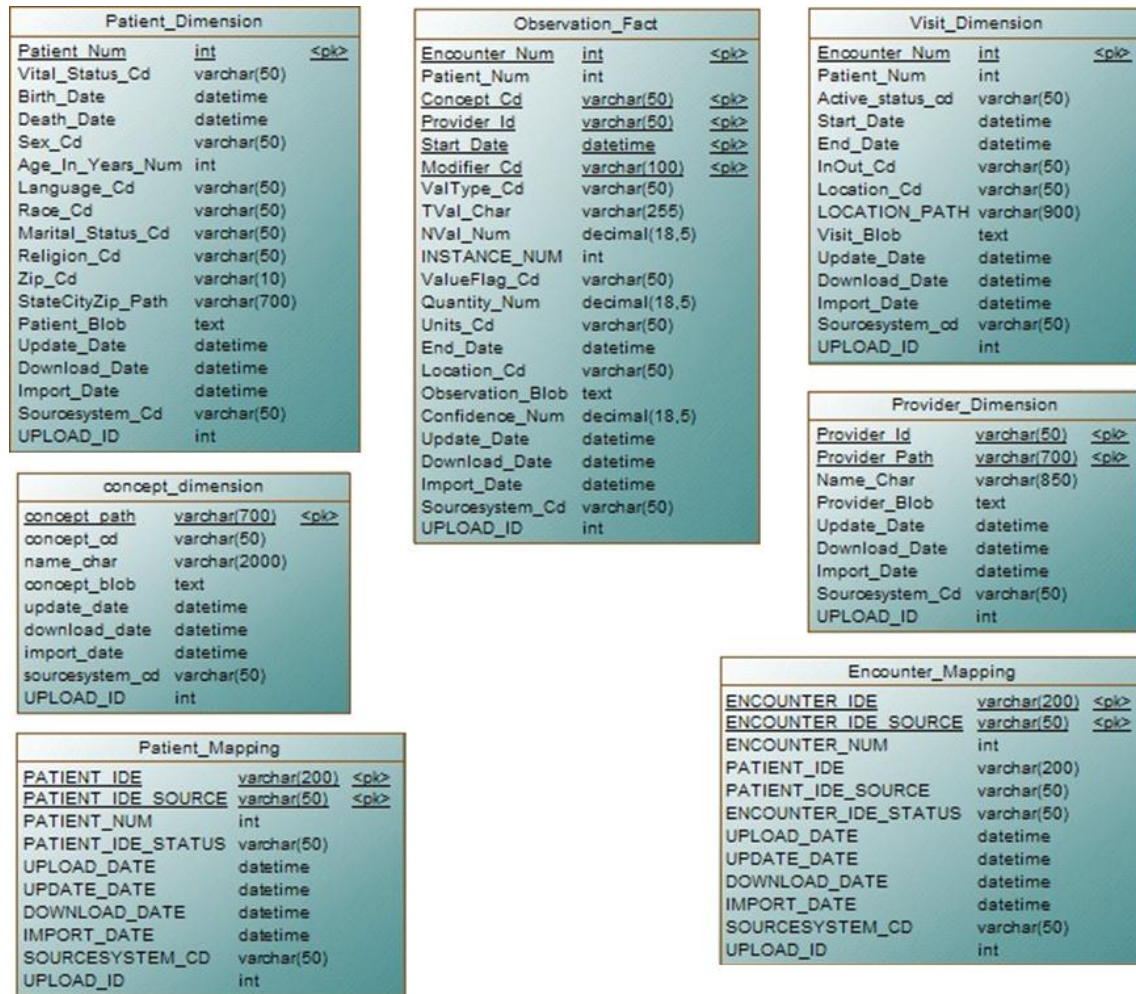


I2B2DEMODATA Schema

The I2B2DEMODATA schema contains the following tables:

Table	Description
CONCEPT_DIMENSION	Path in the i2b2 tree for a particular concept code. Primary key: CONCEPT_PATH
ENCOUNTER_MAPPING	Maps an encounter number to the encounter encrypted number (ENCOUNTER_IDE) from the source system. Includes the dates of encounter data uploads and the source of the uploaded data. Primary keys: ENCOUNTER_IDE ENCOUNTER_IDE_SOURCE
OBSERVATION_FACT	Measurements collected from patients during the trial. Primary keys: ENCOUNTER_NUM CONCEPT_CD PROVIDER_ID START_DATE MODIFIER_CD
PATIENT_DIMENSION	Patient demographic data. Primary key: PATIENT_NUM
PATIENT_MAPPING	Maps a patient number to the patient encrypted number (PATIENT_IDE) from the source system. Includes the dates of patient data uploads and the source of the uploaded data. Primary keys: PATIENT_IDE PATIENT_IDE_SOURCE
PROVIDER_DIMENSION	Provider information such as name, title, and ID. Primary key: SNP_PROBE_ID
VISIT_DIMENSION	Data relating to an encounter, such as encounter number, patient number, and dates. Primary keys: PROVIDER_ID PROVIDER_PATH

I2B2DEMODATA Schema Diagram



I2B2HIVE Schema

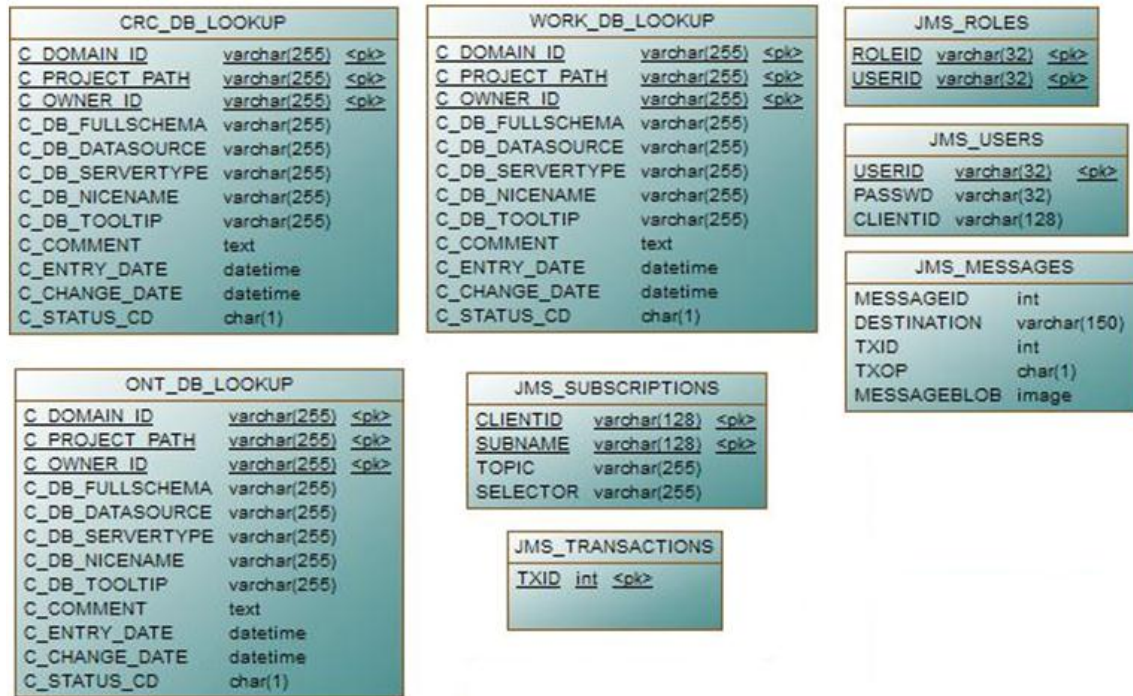
See the i2b2 web site for information about the i2b2 hive:

<https://www.i2b2.org/>

The I2B2HIVE schema contains the following tables:

Table	Description
CRC_DB_LOOKUP	Schema information for the i2b2 Data Repository Cell (also called the Clinical Research Chart, or CRC). Schema I2B2DEMOTADATA. Primary keys: C_DOMAIN_ID C_PROJECT_PATH C_OWNER_ID
JMS_TRANSACTIONS	JMS transaction IDs. Primary key: TXID
JMS_MESSAGES	JMS message IDs and destinations.
JMS_ROLES	JMS users and associated roles. Primary keys: ROLEDID USERID
JMS_SUBSCRIPTIONS	JMS Subscriptions. Primary keys: CLIENTID SUBNAME
JMS_USERS	JMS usernames and passwords. Primary key: USERID
ONT_DB_LOOKUP	Ontology schema information. Schema I2B2METADATA. Primary keys: C_DOMAIN_ID C_PROJECT_PATH C_OWNER_ID
WORK_DB_LOOKUP	Workplace schema information. Schema I2B2WORKDATA. Primary keys: C_DOMAIN_ID C_PROJECT_PATH C_OWNER_ID For table details, see https://www.i2b2.org/software/projects/workplace/Workplace_Design_15.pdf .

I2B2HIVE Schema Diagram

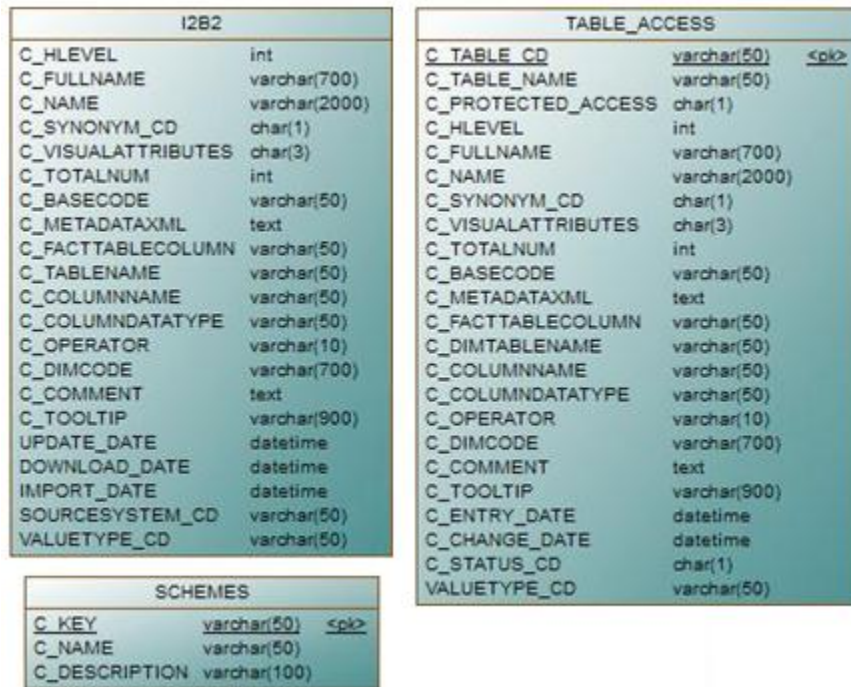


I2B2METADATA Schema

The I2B2METADATA schema contains the following tables:

Table	Description
I2B2	Detailed information about the i2b2 tree. For table details, see https://www.i2b2.org/software/projects/ontologymgmt/Ontology_Design_15.pdf .
SCHEMES	Standard terminologies such as NDC, ICD9, and LOINC.
TABLE_ACCESS	Categories of studies (root nodes such as Public Studies) in the i2b2 tree. Primary key: C_TABLE_CD

I2B2METADATA Schema Diagram



BIOMART GWAS/EQTL Schema Diagram

