Querying Biological Databases with SQL

Fran Lewitter

11/4/10

Microarrays: a practical example

Typical Excel spreadsheet of microarray data

Affy	Lung	Heart	Gall_bladder	Pancreas	Testis
92632_at	20	20	20	20	20
94246_at	20	71	122	20	20
93645_at	216	249	152	179	226
98132_at	135	236	157	143	145

The Query: Find all of the genes that have at least 2-fold higher expression in the gall bladder compared to the testis, and sort by decreasing RNA abundance in the heart

Today's Goal

Learn

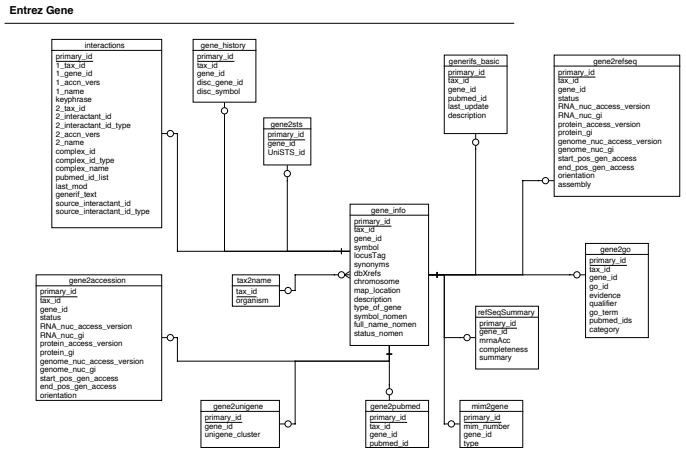
- Why it's useful to use the query language, SQL
- The basics commands of SQL
- About tools to use
- What databases to query

Some Terminology: Flat vs. Relational Database

- Flat file databases use identity tags or delimited formats to describe data and categories without relating data to each other
  - Most biological databases are flat files and require specific parsers and filters
- Relational databases store data in terms of their relationship to each other
  - A simple query language can extract information from any database

The Relational Database

- Data is composed of sets of tables and links (e.g. Key)
- Structured Query Language (SQL) to query the database
- Database management system (DBMS) to manage the data (MySQL)



Some questions in English with Answers in the NCBI Gene Database

- What Is The Gene Symbol and Chromosome of a Specific Gene ID?
- Given a Specific RefSeq Gene ID, Select General Information
- Given a Specific Gene ID, Select GO Terms
- Given a Species and Gene Symbol, Select General Information

Today's Goal

Learn

- Why it's useful to use the query language, SQL
- The basics commands of SQL
  - Examples using:
    - entrez_gene – NCBI Gene Database
    - db4bio – test database
- About tools to use
- What databases to query

Simple Query on One Table

SELECT ***columns***
FROM ***database.table***
WHERE ***expression;***

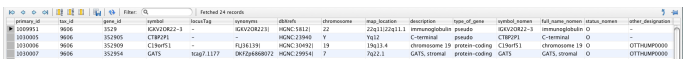
Given a Specific Gene Id, Select Go Terms

- **SELECT gene_id, go_id, go_term FROM gene2go WHERE gene_id = 101;**

```
+---------+------------+--------------------------------+
| gene_id | go_id      | go_term                        |
+---------+------------+--------------------------------+
|     101 | GO:0004222 | metalloendopeptidase activity  |
|     101 | GO:0005515 | protein binding                |
|     101 | GO:0005887 | integral to plasma membrane    |
|     101 | GO:0006508 | proteolysis                    |
|     101 | GO:0008233 | peptidase activity             |
|     101 | GO:0008237 | metallopeptidase activity      |
|     101 | GO:0008270 | zinc ion binding               |
|     101 | GO:0016020 | membrane                       |
|     101 | GO:0046872 | metal ion binding              |
+---------+------------+--------------------------------+
```


Simple Query on One Table

- **SELECT gene_id, symbol, chromosome FROM entrez_gene.gene_info WHERE gene_id LIKE '3529%' AND tax_id = 9606 LIMIT 1;**



```
+---------+---------------+-------------+
| gene_id |  symbol       | chromosome  |
+---------+---------------+-------------+
|  3529   | IGKV2OR22-3   | 22          |
+---------+---------------+-------------+
```


This will provide across-the-board information based on a RefSeq Gene Id

- **SELECT RNA_nuc_access_version, protein_access_version, gene_id FROM entrez_gene.gene2refseq WHERE RNA_nuc_access_version LIKE 'NM_126167%';**

```
+-------------------------+-------------------------+---------+
| RNA_nuc_access_version | protein_access_version | gene_id |
+-------------------------+-------------------------+---------+
|    NM_126167           | NP_178215              |  814629 |
+-------------------------+-------------------------+---------+
```

More Queries of Entrez Gene

- **SELECT gene_id, RNA_nuc_access_version FROM gene2refseq WHERE RNA_nuc_access_version = "NM_001100";**
- **SELECT gene_id, symbol FROM gene_info WHERE gene_id = 58;**
- **SELECT distinct gene_id, symbol, description FROM gene_info WHERE tax_id=10090;**

Knowing Your Data

- Data types
  - Int: 58
  - Float: 9.2
  - Date: 2010-12-31 23:59:59
  - VARCHAR: "apple"
  - Text: "life is good"
- Tak: DESCRIBE database_name.table_name;

More SQL

- **SELECT level, level*2 FROM db4bio.Data LIMIT 5;**
- **WHERE**
  - Restricts queries based on text, numerical value, including inequalities and patterns
  - Not equal: !=
- **SELECT * FROM db4bio.Data WHERE affyId != '1000_at' LIMIT 3;**
- **LIKE, NOT LIKE, %**
- **ORDER BY – DESC, ASC**
- **BETWEEN : BETWEEN 80 AND 100**
- **Operators: AND, OR**

More Advanced SQL

- GROUP BY - group rows by some attribute and get summary info about that group
- HAVING - Sets the conditions for the GROUP BY clause like WHERE sets conditions for SELECT
- **SELECT affyId, SUM(level)/count(level) AS mean_level FROM db4bio.Data GROUP BY affyId HAVING mean_level > 20000;**

More Basics

```
SELECT count(affyId) FROM db4bio.Data WHERE level > 5000;
+------------------+
| count(affyId) |
+------------------+
|789               |
+------------------+
```

```
SELECT count(DISTINCT affyId) FROM db4bio.Data WHERE level > 5000;
+-----------------------------+
| count(distinct affyId) |
+-----------------------------+
|600                          |
+-----------------------------+
```


WHERE And ORDER BY

```
SELECT *
FROM RefSeqs
WHERE linkId BETWEEN 50 AND 100
LIMIT 5;
```

linkId	ntRefSeq	aaRefSeq
50	NM_001098	NP_001089
51	NM_004035	NP_004026
52	NM_004300	NP_004291
53	NM_001610	NP_001601
54	NM_001611	NP_001602

```
SELECT *
FROM RefSeqs
WHERE linkId BETWEEN 50 AND 100
ORDER BY ntRefSeq DESC
LIMIT 5;
```

linkId	ntRefSeq	aaRefSeq
70	NM_005159	NP_005150
81	NM_004924	NP_004915
91	NM_004302	NP_004293
86	NM_004301	NP_004292
52	NM_004300	NP_004291

GROUP BY And HAVING

```
SELECT affyId, MIN(level) AS min,
MAX(level) as max
FROM Data
GROUP BY affyid
HAVING max – min > 5000
LIMIT 5;
```

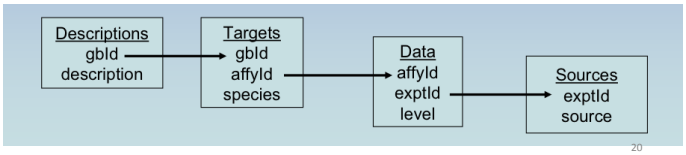
```
SELECT gbId, COUNT(affyId)
AS num_affyIds
FROM Targets
GROUP BY gbId
HAVING COUNT(gbId) > 4
ORDER BY num_affyIds DESC
LIMIT 5;
```

affyId	min	max
100047_at	20	7784
100068_at	414	5883
100069_at	616	6349
100329_at	20	21455
100342_i_at	786	7931

gbId	num_affyIds
J04423	14
AC002397	12
AF109905	9
AF100956	9
AL031228	8

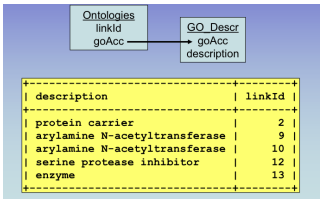
Natural Joins

- Table joining links tables together through their relationships and allows you to traverse your schema/database
- Use SELECT and FROM to join tables
- Join through common attributes with WHERE and AND using operators: =,<,>,!=,>=, <=
- Traverse from descriptions to sources



Binary Join

- **SELECT GO_Descr.description, Ontologies.linkId FROM db4bio.GO_Descr, db4bio.Ontologies WHERE Ontologies.goAcc=GO_Descr.goAcc LIMIT 5;**



description	linkId
protein carrier	2
arylamine N-acetyltransferase	9
arylamine N-acetyltransferase	10
serine protease inhibitor	12
enzyme	13

Today's Goal

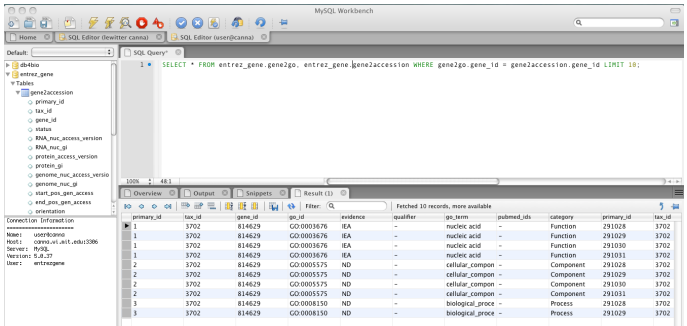
Learn

- Why it's useful to use the query language, SQL
- The basics commands of SQL
- About tools to use
- What databases to query

How to Use MySQL

- For Mac or Windows: MySQL Workbench – http://dev.mysql.com/downloads/workbench
- On tak
  - mysql -h canna entrez_gene -u login –pPassword
  - mysql -h canna -u entrezgene -pwibr entrez_gene
- use db4bio; (practice database)
- use entrez_gene; (NCBI gene database)

MySQL Workbench



Today's Goal

Learn

- Why it's useful to use the query language, SQL
- The basics commands of SQL
- About tools to use
- What databases to query

MySQL Data Resources

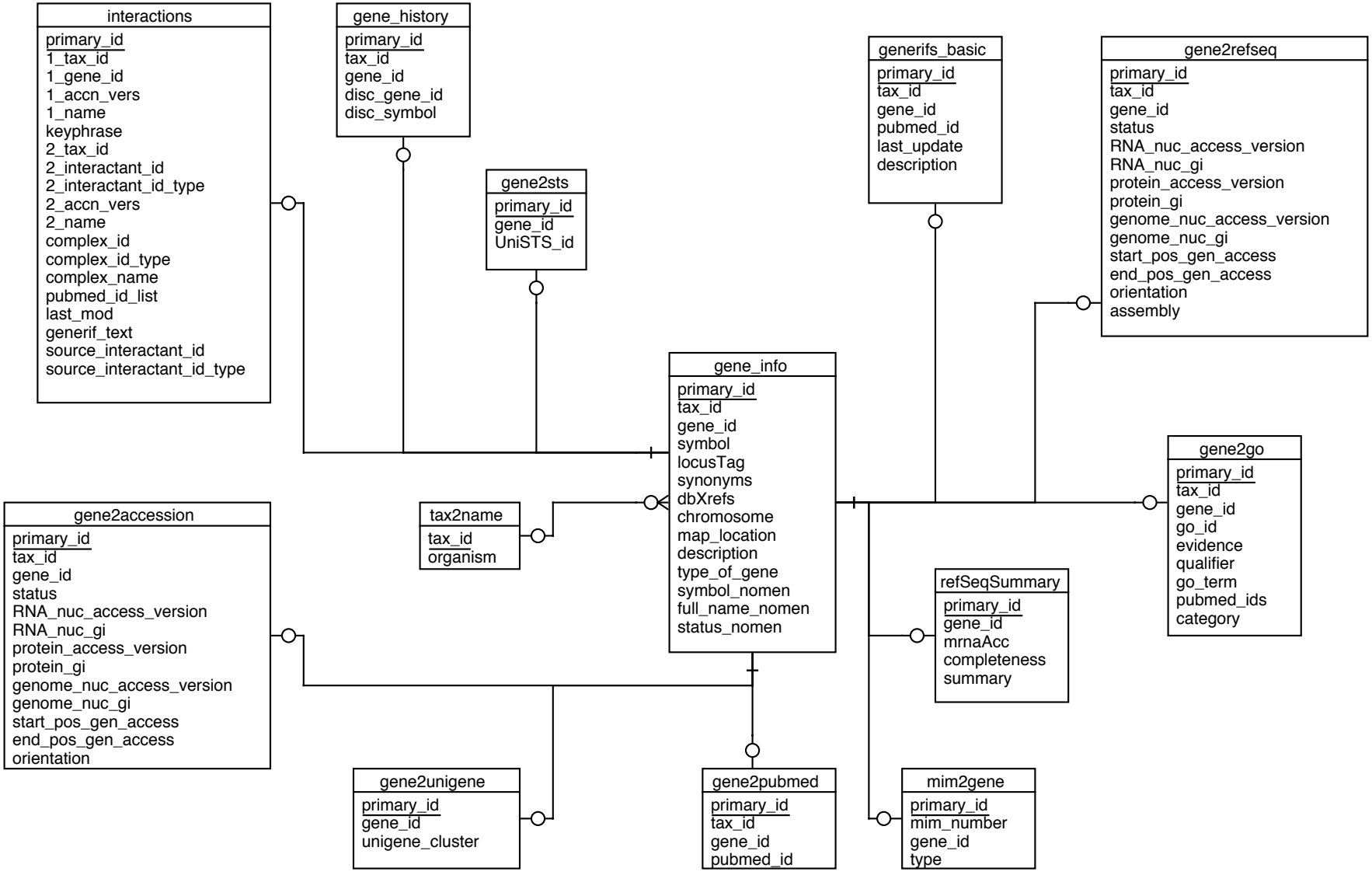
- GO - GOOSE - http://berkeleybop.org/goose
- NCBI taxonomy
- UCSC Bioinformatics http://genome.ucsc.edu/FAQ/FAQdownloads.html#download29
- Ensembl
- Entrez Gene (canna.wi.mit.edu)

Where to go from here

- Consult SQL And MySQL Resources – http://dev.mysql.com/doc/– Tutorial, Reference Manual
- Graphical interfaces to MySQL databases – MySQL Workbench (link above)
- Training materials - http://jura.wi.mit.edu/bio/education/bioinfo2006/db4bio
- Create database; local databases

Demo

- Gene Ontology: http://berkeleybop.org/goose
- MySQL Workbench
  - Add a connection and database to search
    host=canna.wi.mit.edu; user=entrezgene; password=wibr
  - Review database
  - Simple queries
  - Save data
- Tak access
  - **echo "SELECT distinct gene_id, symbol, description FROM gene_info WHERE tax_id=10090" | mysql -h canna entrez_gene -u entrezgene -pwibr > mouse_gene_annotation.txt**;


interactions
primary_id
1_tax_id
1_gene_id
1_accn_vers
1_name
keyphrase
2_tax_id
2_interactant_id
2_interactant_id_type
2_accn_vers
2_name
complex_id
complex_id_type
complex_name
pubmed_id_list
last_mod
generif_text
source_interactant_id
source_interactant_id_type
gene_history
primary_id
tax_id
gene_id
disc_gene_id
disc_symbol
gene2sts
primary_id
gene_id
UniSTS_id
generifs_basic
primary_id
tax_id
gene_id
pubmed_id
last_update
description
gene2refseq
primary_id
tax_id
gene_id
status
RNA_nuc_access_version
RNA_nuc_gi
protein_access_version
protein_gi
genome_nuc_access_version
genome_nuc_gi
start_pos_gen_access
end_pos_gen_access
orientation
assembly
gene_info
primary_id
tax_id
gene_id
symbol
locusTag
synonyms
dbXrefs
chromosome
map_location
description
type_of_gene
symbol_nomen
full_name_nomen
status_nomen
gene2go
primary_id
tax_id
gene_id
go_id
evidence
qualifier
go_term
pubmed_ids
category
gene2accession
primary_id
tax_id
gene_id
status
RNA_nuc_access_version
RNA_nuc_gi
protein_access_version
protein_gi
genome_nuc_access_version
genome_nuc_gi
start_pos_gen_access
end_pos_gen_access
orientation
tax2name
tax_id
organism
refSeqSummary
primary_id
gene_id
mrnaAcc
completeness
summary
gene2unigene
primary_id
gene_id
unigene_cluster
gene2pubmed
primary_id
tax_id
gene_id
pubmed_id
mim2gene
primary_id
mim_number
gene_id
type