Metadata and Annotation with Bioconductor

What is “Metadata” ?

„Data about data“

Helps in understanding the data under study.

For example,
annotation of genomic regions (exons, transcription start sites, regulatory regions)

annotation of gene products (transcript and protein IDs, gene ontology annotation, protein classifications, associated phenotypes)

The label „meta-“ is in the eye of the beholder.

We will also want to integrate other ‚primary‘ datasets.

Managing Annotation

Static Annotation

- Bioconductor packages containing annotation information that are installed locally on a computer
- standardized and well-documented structure
- support reproducible analyses
- no need for network connection

Dynamic Annotation

- stored in a database accessed via the internet
- more frequent updates → possibly different result when repeating analyses
- more information
- higher diversity of data formats and conventions

BioMart is usually employed as “dynamic”, but could also be made “static” when installed locally on your computer

Metadata - examples

- **EntrezGene**
  is a catalog of genetic loci that connects curated sequence information to official nomenclature. It replaced LocusLink.

- **RefSeq**
  is a non-redundant set of transcripts and proteins of known genes for many species, including human, mouse and rat.

- **Enzyme Commission (EC)**
  numbers are assigned to different enzymes and linked to genes through EntrezGene.

Metadata - examples

- **Gene Ontology (GO)**
  is a structured vocabulary of terms describing gene products according to molecular function, biological process, or cellular component

- **PubMed**
  is a service of the U.S. National Library of Medicine. PubMed provides a rich resource of data and tools for papers in journals related to medicine and health. While large, it is not comprehensive, not all papers have been abstracted, and only abstracts are searched.

- **Microarray Data Archives**
  The NCBI coordinates the Gene Expression Omnibus (GEO); TIGR provides the Resourcerer database, and the EBI runs ArrayExpress.
  (**`GeoQuery`** and **`ArrayExpress`** packages)

Metadata - examples

- **OMIM**
  Online Mendelian Inheritance in Man is a catalog of human genes and genetic disorders.

- **KEGG**
  Kyoto Encyclopedia of Genes and Genomes; a collection of data resources including a rich collection of pathway data.

- **IntAct**
  Protein Interaction data, mainly derived from experiments.

- **Pfam**
  Pfam is a large collection of multiple sequence alignments and hidden Markov models covering many common protein domains and families.

Bioconductor Annotation Packages

```
> ls("package:org.Hs.eg.db")
 [1] "org.Hs.eg"                 "org.Hs.egACCNUM"
 [3] "org.Hs.egACCNUM2EG"        "org.Hs.egALIAS2EG"
 [5] "org.Hs.egCHR"              "org.Hs.egCHRLENGTHS"
 [7] "org.Hs.egCHRLOC"           "org.Hs.egCHRLOCEND"
 [9] "org.Hs.eg_dbconn"          "org.Hs.eg_dbfile"
[11] "org.Hs.eg_dbInfo"          "org.Hs.eg_dbschema"
[13] "org.Hs.egENSEMBL"          "org.Hs.egENSEMBL2EG"
[15] "org.Hs.egENSEMBLPROT"      "org.Hs.egENSEMBLPROT2EG"
[17] "org.Hs.egENSEMBLTRANS"     "org.Hs.egENSEMBLTRANS2EG"
[19] "org.Hs.egENZYME"           "org.Hs.egENZYME2EG"
[21] "org.Hs.egGENENAME"         "org.Hs.egGO"
[23] "org.Hs.egGO2ALLEGS"        "org.Hs.egGO2EG"
[25] "org.Hs.egMAP"              "org.Hs.egMAP2EG"
[27] "org.Hs.egMAPCOUNTS"        "org.Hs.egOMIM"
[29] "org.Hs.egOMIM2EG"          "org.Hs.egORGANISM"
[31] "org.Hs.egPATH"             "org.Hs.egPATH2EG"
[33] "org.Hs.egPFAM"             "org.Hs.egPMID"
[35] "org.Hs.egPMID2EG"          "org.Hs.egPROSITE"
[37] "org.Hs.egREFSEQ"           "org.Hs.egREFSEQ2EG"
[39] "org.Hs.egSYMBOL"           "org.Hs.egSYMBOL2EG"
[41] "org.Hs.egUNIGENE"          "org.Hs.egUNIGENE2EG"
[43] "org.Hs.egUNIPROT"
```

Each of these objects represents a (SQL-like) database table

```
> head(toTable(org.Hs.egOMIM))
  gene_id omim_id
1       1  138670
2       2  103950
3       2  104300
4       9  108345
5      10  243400
6      10  612182

> head(toTable(org.Hs.egPFAM))
  gene_id      ipi_id  PfamId
1       1 IPI00022895 PF00047
2       1 IPI00644018 PF00047
3       1 IPI00646799 PF00047
4       2 IPI00478003 PF00207
5       2 IPI00478003 PF01835
6       2 IPI00478003 PF07677
```

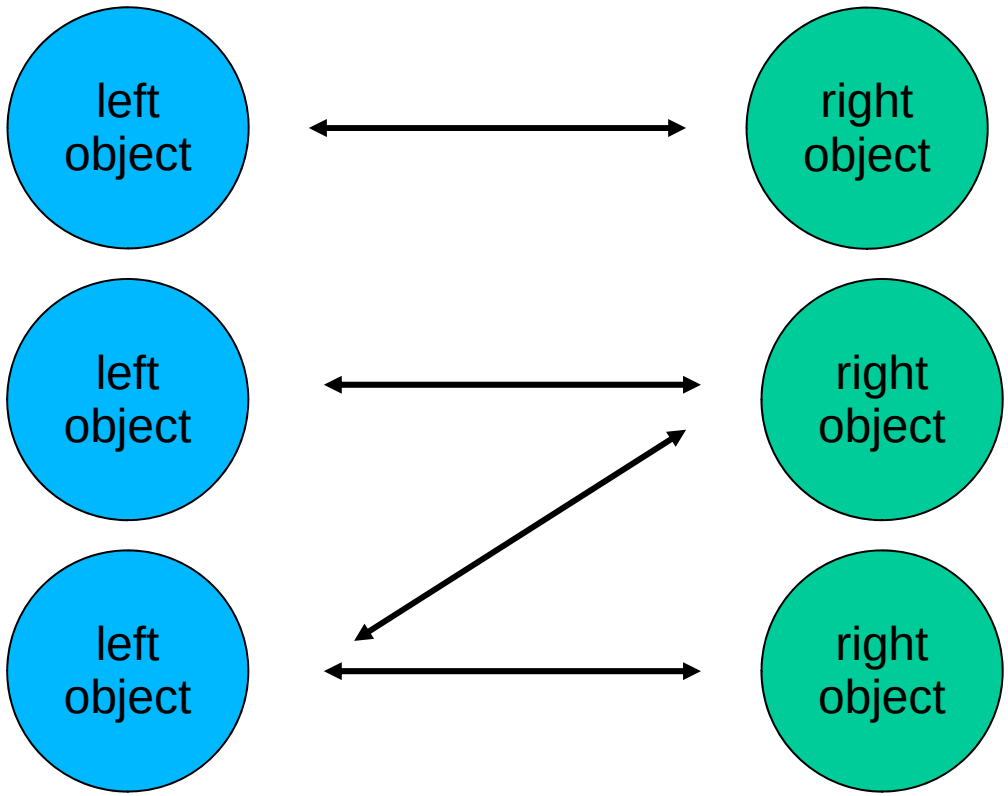
gene_id is the Entrez Gene ID and is used as a key to link the information together

Direct SQL querying of these tables is possible, but there are also some R functions for efficient querying and joining.

Bimap objects

Data in the annotation packages are stored in an SQLite database

Conceptually, the tables are Bimaps (bi-directional maps), i.e. n:m mappings between two sets („left“ and „right“ set)



Lkeys, Rkeys: Get left and right keys of a Bimap object

```
> head(Lkeys(org.Hs.egOMIM))
[1]    "1"      "10"         "100"
[3] "1000"   "10000"   "100008586"
```

gene_id

```
> head(Rkeys(org.Hs.egOMIM))
[1] "100300" "100640" "100650"
[3] "100660" "100670" "100678“
```

omim_id

Querying across multiple tables

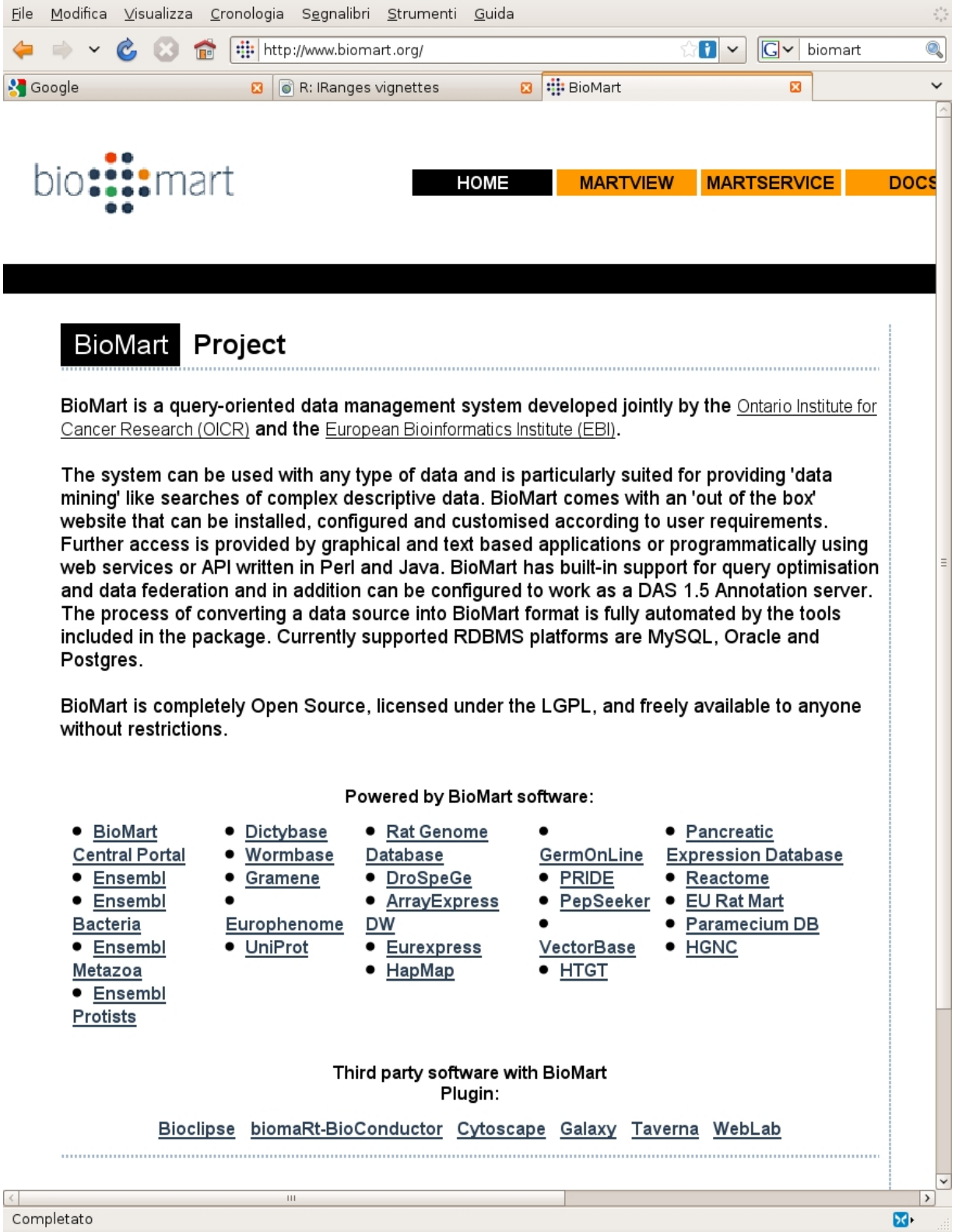
```
> # Find all PfamIDs associated with OMIM code `125853`
> # DIABETES MELLITUS, NONINSULIN-DEPENDENT; NIDDM

> t1 = toTable(revmap(org.Hs.egOMIM)["125853"])

> t2 = toTable(org.Hs.egPFAM[t1$gene_id])

> tt = merge(t1, t2)

> head(tt)
  gene_id omim_id      ipi_id  PfamId
1   10644  125853 IPI00179713 PF00076
2   10644  125853 IPI00179713 PF00013
3   10644  125853 IPI00180983 PF00076
4   10644  125853 IPI00922167 PF00013
5   10644  125853 IPI00790840    <NA>
6   10644  125853 IPI00180983 PF00013
```


File Modifica Visualizza Cronologia Segnalibri Strumenti Guida
http://www.biomart.org/
biomart
Google
R: IRanges vignettes
BioMart
HOME
MARTVIEW
MARTSERVICE
DOCS
BioMart Project
BioMart is a query-oriented data management system developed jointly by the Ontario Institute for Cancer Research (OICR) and the European Bioinformatics Institute (EBI).
The system can be used with any type of data and is particularly suited for providing 'data mining' like searches of complex descriptive data. BioMart comes with an 'out of the box' website that can be installed, configured and customised according to user requirements. Further access is provided by graphical and text based applications or programmatically using web services or API written in Perl and Java. BioMart has built-in support for query optimisation and data federation and in addition can be configured to work as a DAS 1.5 Annotation server. The process of converting a data source into BioMart format is fully automated by the tools included in the package. Currently supported RDBMS platforms are MySQL, Oracle and Postgres.
BioMart is completely Open Source, licensed under the LGPL, and freely available to anyone without restrictions.
Powered by BioMart software:
BioMart Central Portal
Ensembl
Ensembl Bacteria
Ensembl Metazoa
Ensembl Protists
Dictybase
Wormbase
Gramene
Europhenome
UniProt
Rat Genome Database
DroSpeGe
ArrayExpress DW
Eurexpress
HapMap
GermOnLine
PRIDE
PepSeeker
VectorBase
HTGT
Pancreatic Expression Database
Reactome
EU Rat Mart
Paramecium DB
HGNC
Third party software with BioMart Plugin:
Bioclipse
biomaRt-BioConductor
Cytoscape
Galaxy
Taverna
WebLab
Completato

Ensembl

Joint project between EMBL-EBI and the Sanger Institute

Produces and maintains genome databases for vertebrates and other eukaryotic species.

http://www.ensembl.org

Find a Species

Ensembl Species


Alpaca
Vicugna pacos


Anole Lizard
Anolis carolinensis


Anopheles
Anopheles gambiae


Armadillo
Dasypus novemcinctus


Bushbaby
Otolemur garnettii


Caenorhabditis elegans


Ciona intestinalis


Ciona savignyi


Cat
Felis catus


Chicken
Gallus gallus


Chimpanzee
Pan troglodytes


Cow
Bos taurus


Dog
Canis familiaris


Dolphin
Tursiops truncatus


Elephant


Fugu
Takifugu rubripes


Gorilla
Gorilla gorilla


Guinea Pig
Cavia porcellus


Hedgehog
Erinaceus europaeus


Horse
Equus caballus


Human
Homo sapiens


Hyrax
Procavia capensis


Kangaroo rat
Dipodomys ordii


Lamprey (preview - assembly only)
Petromyzon marinus


Lesser hedgehog tenrec
Echinops telfairi


Macaque
Macaca mulatta


Medaka
Oryzias latipes


Megabat
Pteropus vampyrus


Microbat
Myotis lucifugus


Mouse


Orangutan
Pongo pygmaeus


Pig (preview - assembly only)
Sus scrofa


Pika
Ochotona princeps


Platypus
Ornithorhynchus anatinus


Rabbit
Oryctolagus cuniculus


Rat
Rattus norvegicus


Saccharomyces cerevisiae


Shrew
Sorex araneus


Sloth
Choloepus hoffmanni


Squirrel
Spermophilus tridecemlineatus

Stickleback
Gasterosteus aculeatus

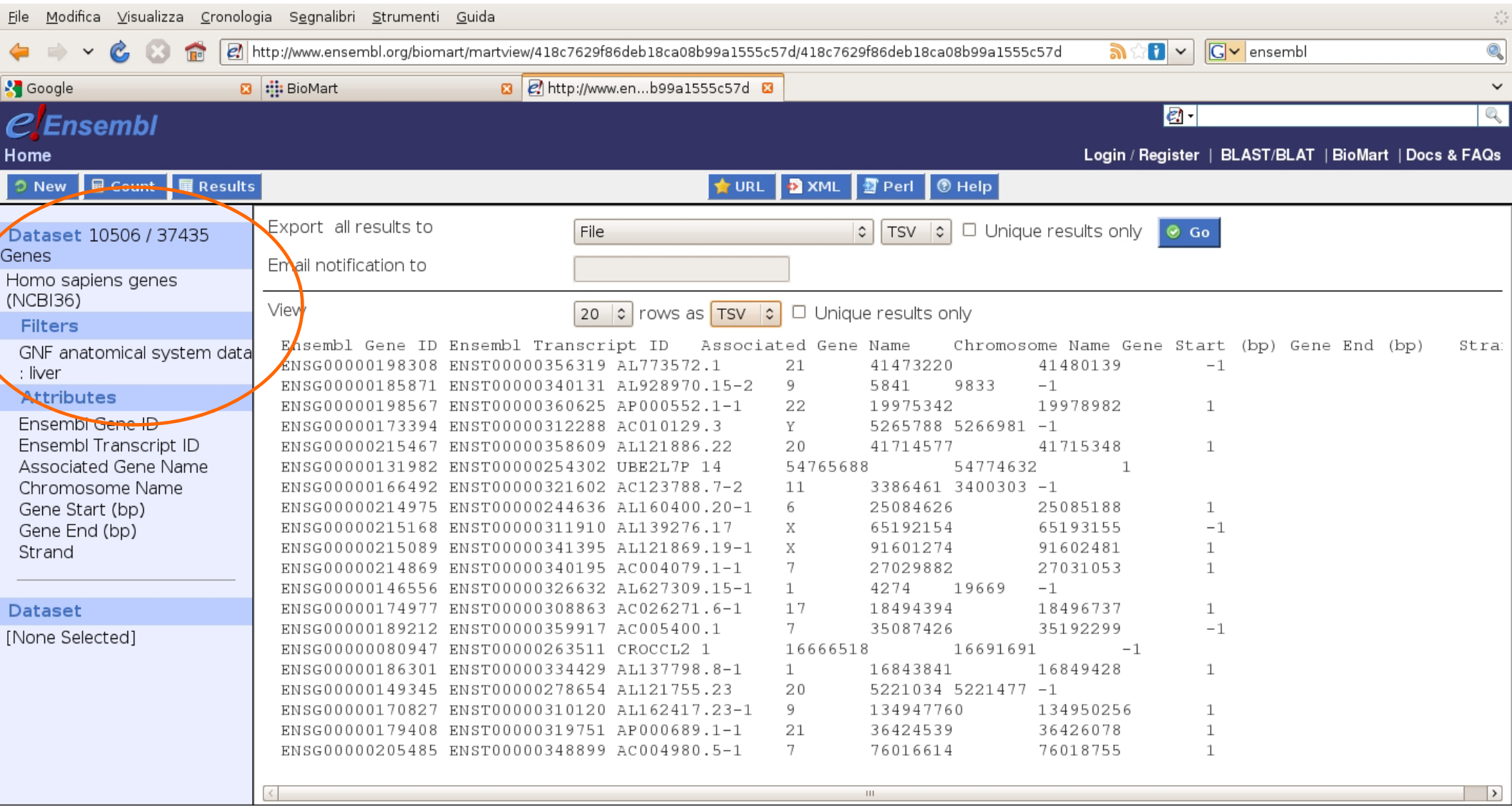
Tarsier
Tarsius syrichta

Tetraodon
Tetraodon nigroviridis

Tree Shrew
Tupaia belangeri

Xenopus tropicalis

Ensembl martview



VEGA

The Vertebrate Genome Annotation (VEGA) database is a central repository for high quality, frequently updated, manual annotation of vertebrate finished genome sequence.

Current release:
- Human
- Mouse
- Zebrafish
- Dog

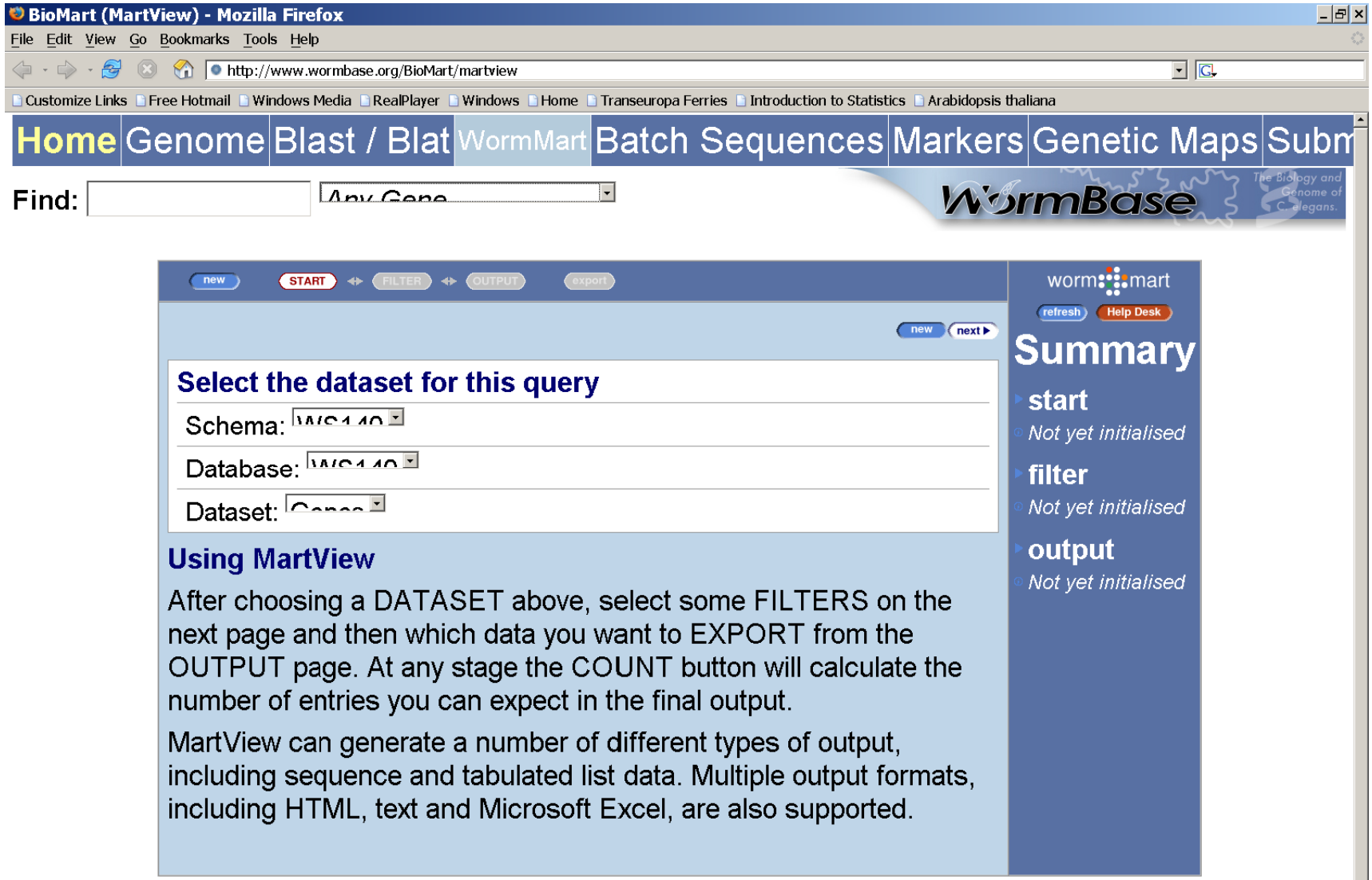
http://vega.sanger.ac.uk

WormBase

WormBase is the repository of mapping, sequencing and phenotypic information for *C. elegans* (and some other nematodes).

http://www.wormbase.org

WormMart

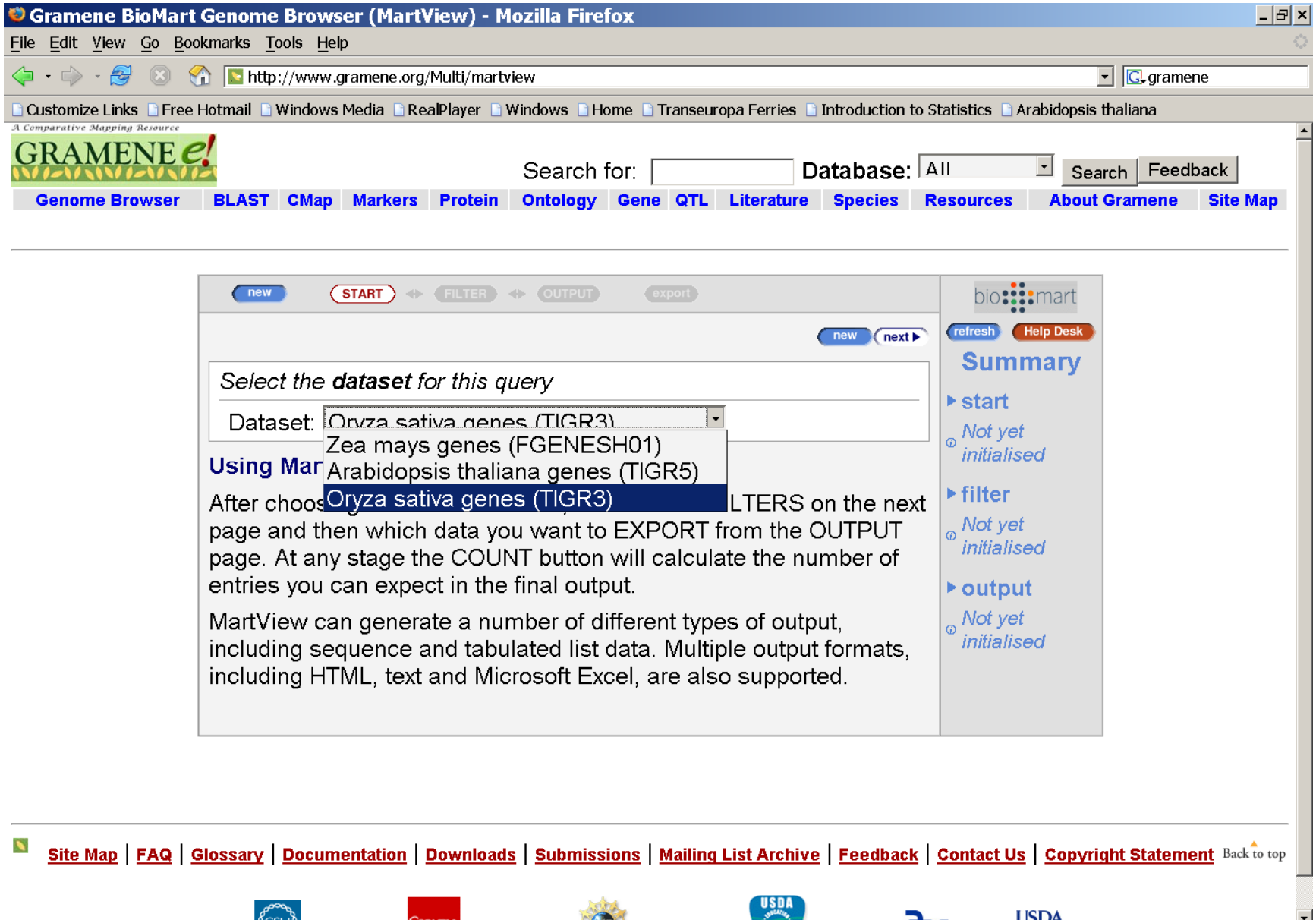

BioMart (MartView) - Mozilla Firefox
File Edit View Go Bookmarks Tools Help
http://www.wormbase.org/BioMart/martview
Customize Links Free Hotmail Windows Media RealPlayer Windows Home Transeuropa Ferries Introduction to Statistics Arabidopsis thaliana
Home Genome Blast / Blat WormMart Batch Sequences Markers Genetic Maps Subm
Find:
Any Gene
WormBase
The Biology and Genome of C. elegans
new START FILTER OUTPUT export
new next
Select the dataset for this query
Schema:
Database:
Dataset:
Using MartView
After choosing a DATASET above, select some FILTERS on the next page and then which data you want to EXPORT from the OUTPUT page. At any stage the COUNT button will calculate the number of entries you can expect in the final output.
MartView can generate a number of different types of output, including sequence and tabulated list data. Multiple output formats, including HTML, text and Microsoft Excel, are also supported.
wormmart
refresh Help Desk
Summary
start
Not yet initialised
filter
Not yet initialised
output
Not yet initialised

GrameneMart

Gramene: A Comparative Mapping Resource for Grains

Gramene is a curated, open-source, Web-accessible data resource for comparative genome analysis in the grasses.

http://www.gramene.org


Gramene BioMart Genome Browser (MartView) - Mozilla Firefox
File Edit View Go Bookmarks Tools Help
http://www.gramene.org/Multi/martview
gramene
Customize Links Free Hotmail Windows Media RealPlayer Windows Home Transeuropa Ferries Introduction to Statistics Arabidopsis thaliana
A Comparative Mapping Resource
GRAMENE e!
Search for:
Database: All
Search
Feedback
Genome Browser BLAST CMap Markers Protein Ontology Gene QTL Literature Species Resources About Gramene Site Map
new START FILTER OUTPUT export
new next ▶
Select the dataset for this query
Dataset: Oryza sativa genes (TIGR3)
Zea mays genes (FGENESH01)
Arabidopsis thaliana genes (TIGR5)
Oryza sativa genes (TIGR3)
Using Mar
After choos LTERS on the next page and then which data you want to EXPORT from the OUTPUT page. At any stage the COUNT button will calculate the number of entries you can expect in the final output.
MartView can generate a number of different types of output, including sequence and tabulated list data. Multiple output formats, including HTML, text and Microsoft Excel, are also supported.
biomart
refresh Help Desk
Summary
start
Not yet initialised
filter
Not yet initialised
output
Not yet initialised
Site Map | FAQ | Glossary | Documentation | Downloads | Submissions | Mailing List Archive | Feedback | Contact Us | Copyright Statement Back to top

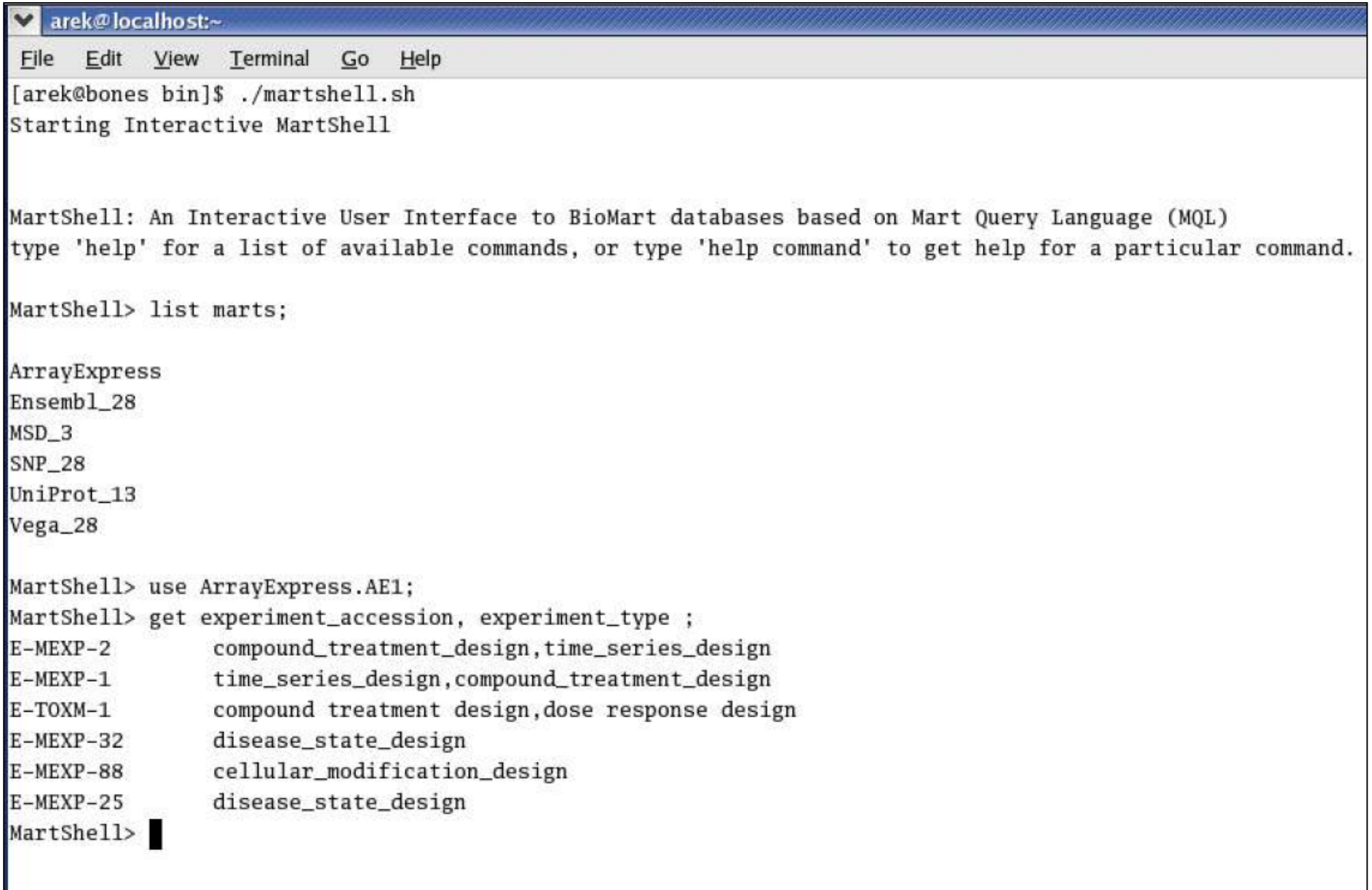
Other databases with BioMart interfaces

- **dbSNP (via Ensembl)**
- **HapMap**
- **Sequence Mart: Ensembl genome sequences**

BioMart user interfaces

MartShell

MartShell is a command line BioMart user interface based on a structured query language: Mart Query Language (MQL)



BioMart user interfaces

Martview Web based user interface for BioMart, provides functionality for remote users to query all databases hosted by the EBI's public BioMart server.

MartExplorer

Perl and Java **libraries**

biomaRt interface to R/Bioconductor

The biomaRt package

Developed by Steffen Durinck (started Feb 2005)

Two main sets of functions:

1. **Tailored towards Ensembl, shortcuts for FAQs (frequently asked queries): getGene, getGO, getOMIM...**

2. **Generic queries, modeled after MQL (Mart query language), can be used with any BioMart dataset, usually require some pre- and postprocessing in R.**

Getting started

```
> library(biomaRt)
> listMarts()

$biomart
[1] "dicty"  "ensembl"  "snp"  "vega"  "uniprot"  "msd"  "wormbase"

$version
[1] "DICTYBASE (NORTHWESTERN)" "ENSEMBL 38 (SANGER)"
[3] "SNP 38 (SANGER)"          "VEGA 38 (SANGER)"
[5] "UNIPROT 4-5 (EBI)"        "MSD 4 (EBI)"
[7] "WORMBASE CURRENT (CSHL)"

$host
[1] "www.dictybase.org" "www.biomart.org"   "www.biomart.org"
[4] "www.biomart.org"   "www.biomart.org"   "www.biomart.org"
[7] "www.biomart.org"

$path
[1] ""                      "/biomart/martservice" "/biomart/martservice"
[4] "/biomart/martservice" "/biomart/martservice" "/biomart/martservice"
[7] "/biomart/martservice"
```

Gene annotation

The function getGene allows you to get gene annotation for many types of identifiers

Supported identifiers are:

- **RefSeq**
- **Entrez-Gene**
- **EMBL**
- **HUGO**
- **Ensembl**
- **Affymetrix Genechip Probeset ID**

getGene

```
> mart <- useMart("ensembl", dataset = "hsapiens_gene_ensembl")
> myProbes <- c("210708_x_at", "202763_at", "211464_x_at")
> z <- getGene(id = myProbes, array = "affy_hg_u133_plus_2", mart = mart)

           ID symbol
1   202763_at  CASP3
2 210708_x_at CASP10
7 211464_x_at  CASP6


description
1 Caspase-3 precursor (EC 3.4.22.-) (CASP-3) (Apopain) ...
2 Caspase-10 precursor (EC 3.4.22.-) (CASP-10) (ICE-like apoptotic pro..
7 Caspase-6 precursor (EC 3.4.22.-) (CASP-6) (Apoptotic protease Mch-2)...


  chromosome  band strand chromosome_start chromosome_end ensembl_gene_id
1          4 q35.1     -1        185785845      185807623 ENSG00000164305
2          2 q33.1      1        201756100      201802372 ENSG00000003400
7          4   q25     -1        110829234      110844078 ENSG00000138794


  ensembl_transcript_id
1       ENST00000308394
2       ENST00000272879
7       ENST00000265164
```

getGene

returns a dataframe

- **Gene symbol**
- **Description**
- **Chromosome name**
- **Band**
- **Start position**
- **End position**
- **BioMartID**

Other functions

- getGO**: GO id, GO term, evidence code**
- getOMIM **(Online Mendelian Inheritance in Man, a catalogue of human genes and genetic disorders): OMIM id, Disease, BioMart id**
- getINTERPRO **(an integrated resource of protein families, domains and functional sites): Interpro id, description**
- getSequence
- getSNP
- getHomolog

getSequence

```
> seq <- getSequence(species="hsapiens", chromosome = 19, start =
18357968, end = 18360987, mart = mart)

chromosome
[1] "19"
start
[1] 18357968
end
[1] 18360987
sequence
"AGTCCCAGCTCAGAGCCGCAACCTGCACAGCCATGCCCGGGCAAGAACTCAGGACGGTGAATGGCTCTCAG
ATGCTCCTGGTGTTGCTGGTGCTCTCGTGGCTGCCGCATGGGGGCGCCCTGTCTCTGGCCGAGGCGAGCCGC
GCAAGTTTCCCGGGACCCTCAGAGTTGCACTCCGAAGACTCCAGATTCCGAGAGTTGCGGAAACGCTACGAG
GACCTGCTAACCAGGCTGCGGGCCAACCAGAGCTGGGAAGATTCGAACACCGACCTCGTCCCGGCCCCTGCA
GTCCGGATACTCACGCCAGAAGGTAAGTGAAATCTTAGAGATCCCCTCCCACCCCCCAAGCAGCCCCCATAT
CTAATCAGGGATTCCTCATCTTGAAAAGCCCAGACCTACCTGCGTATCTCTCGGGCCGCCCTTCCCGAGGGG
CTCCCCGAGGCCTCCCGCCTTCACCGGGCTCTGTTCCGGCTGTCCCCGACGGCGTCAAGGTCGTGGGACGTG
ACACGACCGCTGCGGCGTCAGCTCAGCCTTGCAAGACCCCAGGCGCCCGCGCTGCACCTGCGACTGTCGCCG
CCGCCGTCGCAGTCGGACCAACTGCTGGCAGAATCTTCGTCCGCACGGCCCCAGCTGGAGTTGCACTTGCGG
CCGCAAGCCGCCAGGGGGCGCCGCAGAGCGCGTGCGCGCAACGGGGACCACTGTCCGCTCGGGCCCGGGCGT
TGCTGCCGTCTGCACACGGTCCGCGCGTCGCTGGAAGACCTGGGCTGGGCCGATTGGGTGCTGTCGCCACGG
GAGGTGCAAGTGACCATGTGCATCGGCGCGTGCCCGAGCCAGTTCCGGGCGGCAAACATG....
```

SNPs

- **Single Nucleotide Polymorphisms (SNPs) are one type of common DNA sequence variations between individuals.**

  e.g.

  AAGGCTAA
  ATGGCTAA

- **biomaRt uses the SNP mart of Ensembl which is obtained from dbSNP**

getSNP

```
> getSNP(chromosome = 8, start = 148350, end = 148612, mart = mart)

          tsc  refsnp_id allele chrom_start chrom_strand
1  TSC1723456  rs3969741    C/A      148394            1
2  TSC1421398  rs4046274    C/A      148394            1
3  TSC1421399  rs4046275    A/G      148411            1
4                rs13291    C/T      148462            1
5  TSC1421400  rs4046276    C/T      148462            1
6               rs4483971    C/T      148462            1
7              rs17355217    C/T      148462            1
8              rs12019378    T/G      148471            1
9  TSC1421401  rs4046277    G/A      148499            1
10             rs11136408    G/A      148525            1
11 TSC1421402  rs4046278    G/A      148533            1
12             rs17419210    C/T      148533           -1
13             rs28735600    G/A      148533            1
14 TSC1737607  rs3965587    C/T      148535            1
15              rs4378731    G/A      148601            1
```

Homology mapping

The **getHomolog** function enables mapping of many types of identifiers from one species to the same or another type of identifier in another species.

getHomolog

```
> from.mart = useMart("ensembl", dataset = "hsapiens_gene_ensembl")
> to.mart = useMart("ensembl", dataset = "mmusculus_gene_ensembl")

> getHomolog(id = 2, from.type = "entrezgene", to.type = "refseq",
+     from.mart = from.mart, to.mart = to.mart)
```

	V1	V2	V3
1	ENSMUSG00000030111	ENSMUST00000032203	NM_175628
2	ENSMUSG00000059908	ENSMUST00000032228	NM_008645
3	ENSMUSG00000030131	ENSMUST00000081777	NM_008646
4	ENSMUSG00000071204	ENSMUST00000078431	NM_001013775
5	ENSMUSG00000030113	ENSMUST00000032206	
6	ENSMUSG00000030359	ENSMUST00000032510	NM_007376

getFeature

Select all RefSeq id's involved in diabetes mellitus:

```
>getFeature(   OMIM="diabetes mellitus",
          type="refseq",
          species="hsapiens",
          mart=mart)
```

The generic interface: the getBM function

Selecting a dataset: the useDataset function

```
> library(biomaRt)
> mart <- useMart("ensembl")
> listDatasets(mart)
                      dataset    version
1    rnorvegicus_gene_ensembl    RGSC3.4
2    scerevisiae_gene_ensembl       SGD1
3       celegans_gene_ensembl     CEL150
4  cintestinalis_gene_ensembl       JGI2
5   ptroglodytes_gene_ensembl    CHIMP1A
6      frubripes_gene_ensembl      FUGU4
7       agambiae_gene_ensembl     AgamP3
8       hsapiens_gene_ensembl     NCBI36
9        ggallus_gene_ensembl    WASHUC1
10   xtropicalis_gene_ensembl     JGI4.1
11        drerio_gene_ensembl     ZFISH5
....(more)...
> mart <- useDataset(dataset = "hsapiens_gene_ensembl", mart = mart)
```

getBM

```
> getBM(attributes = c("affy_hg_u95av2", "hgnc_symbol"),
        filter = "affy_hg_u95av2",
        values = c("1939_at", "1000_at"),
        mart = mart)

  affy_hg_u95av2 hgnc_symbol
1        1000_at       MAPK3
3        1939_at        TP53
```

mart – an object describing the database connection and the dataset

attributes – the name of the data you want to obtain

filter – the name of the data by which you want to filter from the dataset

values – values to filter on

Locally installed BioMarts

- **Main use case currently is to use biomaRt to query public BioMart servers over the internet**
- **But you can also install BioMart server locally, populated with a copy of a public dataset (particular version), or populated with your own data**
- **Versioning is supported by naming convention**

Read more

Mapping Identifiers for the Integration of Genomic Datasets with the R/Bioconductor package biomaRt. S. Durinck, E. Birney, P. Spellmann, W. Huber, Nature Protocols 2009 (to appear)