On-line supplement to manuscript "Galaxy for collaborative analysis of ENCODE data: Making large-scale analyses biologist-friendly"

Daniel Blankenberg, James Taylor, Ian Schenck, Jianbin He, Yi Zhang, Matthew Ghent, Narayanan Veeraraghavan, Istvan Albert, Webb Miller, Kateryna Makova, Ross C. Hardison, and Anton Nekrutenko

Supplement to the Materials section

Upload and processing of ENCODE data

At http://encode-upload.g2.bx.psu.edu, ENCODE members can upload a dataset in either Browser Extendible Format (BED) or Gene Feature Format (GFF). Users are required to: (1) provide a short feature name (e.g., Affymetrix_transfrag), (2) select an ENCODE analysis group (from a list of Chromatin and Chromosomes, Genes and Transcripts, Multi-species Sequences Analysis, or Transcription Regulation), and (3) paste an informative description of the dataset. Once data is submitted, this information is used to generate a filename in standard ENCODE format: `group.feature.date.format` (the `group` and `feature` parts are taken from the user input; `data` is added automatically, and the `format` is either BED or GFF. The Galaxy2$^{\text{ENCODE}}$ Upload tool ensures that both formats are available: if the user uploads data in BED format, Galaxy2$^{\text{ENCODE}}$ automatically generates GFF and vice versa). Next, the upload tool automatically partitions data against the standard set of protein-coding genes provided by the GENCODE group. In order of precedence, the categories of partitions are: (1) coding exons, (2) 5'UTR exons, (3) 3'UTR exons, (4) intronic proximal regions (within 5kb of an exon), (5) intergenic proximal regions (between genes and within 5kb of an exon), (6) intronic distal regions (greater than 5kb from an exon), and (7) intergenic distal regions (between genes but greater than 5kb from an exon). Finally, deposited data are immediately available through Galaxy2^{ENCODE} (http://encode.g2.bx.psu.edu) or directly by ftp (ftp://encode:encode@g2.bx.psu.edu).

Accessing the ENCODE data from within Galaxy2^{ENCODE} is intuitive (Screencast 3). The `ENCODE Data` tool lists all ENCODE analysis groups (Chromatin and Chromosomes, Genes and Transcripts, Multi-species Sequence Analysis, Transcription Regulation, and a set of combined latest datasets). After selecting a group of interest, the user is presented with a listing of all available datasets for that particular group. The user is then able to select any number of the datasets at a time and add them to his/her history pane on the right side of Galaxy2^{ENCODE} screen. Note that Galaxy2^{ENCODE} stores all versions of the datasets, with the latest ones highlighted in bold font.

Finding Non-Gencode ESTs (Figure 3; Screencast 15)

First, we upload the coordinates of human ESTs and Gencode genes that fall within ENCODE regions from the UCSC Table Browser. To simplify the analysis, redundant ESTs (that appear more than once in the downloaded table) are removed from this dataset using a combination of Count, Filter, and Compare tools. Both downloaded tables are gene oriented: each line of the file represents a single EST of a single gene encoded together with all exons. By definition, a Non-Gencode ESTs must not have any exons that overlap Gencode exons. To find EST that satisfy this criterion we must first convert both tables into exon-based tables, where each line represents an exon rather than a gene. We use the Gene-BED-to-exon-BED tool to achieve this goal. Once gene tables are converted into exon-based tables, we use the Subtract tool to remove all EST exons that overlap Gencode exons (Figure 3C). We can use this information to locate those ESTs where none of the exons overlap Gencode exons in two steps. First, we count the number of exons per EST before subtraction and after subtraction. Next, we compare these two numbers. ESTs where the number of exons stays the same before and after subtraction are Non-Gencode ESTs (Figure 3D). This is done using a combination of count, join, filter, and compare tools. Once Non-Gencode ESTs are identified it is necessary to split them into three categories: intronic, intergenic, and intertwined. To find intergenic ESTs we can remove all Non-Gencode ESTs that overlap Gencode genes (Figure 3E) using the Subtract tool. To identify intertwined ESTs, we find the overlap between Non-Gencode ESTs and Gencode exons using the Intersect tool. Because we already have intergenic and intertwined ESTs we can easily define intronic ESTs by subtracting these two categories from the superset, identified in Figure 3D, using the Compare tool. The three sets can be downloaded from the http://www.bx.psu.edu/cgi-bin/trac.cgi/wiki/GenomeResearchSupp2006).

Implementation

Galaxy2$^{\mathrm{ENCODE}}$ is a completely new compact implementation that combines the latest open-source technologies with ideas previously developed by our group (Giardine et al. 2005).

LANGUAGE**.** The entire system is written in Python, a modern dynamic language that fully supports the object-oriented paradigm and is widely regarded as an optimal tool for fast and efficient software development. Thanks to the work of hundreds of developers worldwide, programs written in Python can be run unaltered on most platforms, from all variants of Unix to Mac OSX or Windows. Python also has one of the most extensive default libraries of any programming language.

APPLICATION SERVER**.** The application server we use for Galaxy2$^{\mathrm{ENCODE}}$ development is called Paste. It is a simple and fully object-oriented web development framework that allows the creation of "CGI-like" programs with ease. This application server has been in development for over three years and has a number of production deployments around the world. The Paste framework provides us with a lightweight, portable, multithreaded web server that can be started with minimal setup and allows for a very flexible development environment. We believe that the web server is sufficiently robust to handle any load that we can support. The fact that our services are primarily CPU-bound imposes a limit on how many processes can be queued or run; this number will always be much smaller than the number of simultaneous users the Paste framework can possibly serve. At the same time, the simplicity of the server setup will be tremendously useful in allowing us to replicate the entire functionality of the service with ease. We can easily start copies of the server on any system that supports Python. This way we can distribute the computation load on different computers by assigning users to different servers. The Paste server can be easily integrated with any other web server, including Apache. If a need for load balancing, virtual hosts, or secure HTTP arises, it can be easily accomplished by running Paste "behind" Apache and customizing the front web server to control the HTTP traffic as desired.

DATA STORAGE. GALAXY2$^{\text{ENCODE}}$ deals with two kinds of data: (1) data from external sources and (2) its own annotations attached to each of the results. Both types of data vary in size and type and are represented as files on the file system. This is desirable since most tools operate on files and the file system is the most efficient way of storing large objects. Data can have a number of annotations associated with them, such as filename, display name, selection information, genome build, which user it belongs to, etc. The database functionality is very loosely coupled to the actual storage mechanism and, therefore, we could easily switch databases if the need arises (e.g., to MySQL or PostgreSQL; in fact, the current production version of Galaxy uses PostgreSQL). Upon starting, GALAXY2$^{\text{ENCODE}}$ creates the database-related storage if it does not already exist.

EXTERNAL DATA SOURCES. GALAXY2$^{\text{ENCODE}}$ connects to external sources using a simple data-collection API. There are two main categories for data sources: (1) sources allowing synchronous access (where a data source can start streaming back data as soon as the user has finished his/her parameter selection) and (2) sources with asynchronous access that break off the connection and then later provide

GALAXY2$^{\text{ENCODE}}$ with a URL pointing to the data. We have commitments from the developers of several data sources including UCSC, Ensembl (BioMart), and ENCODEdb to support our data access protocol. GALAXY2$^{\text{ENCODE}}$'s external data source protocols support HTTP calls and remote procedural calls via XML-RPC (for the exact protocols see the on-line documentation on the GALAXY2$^{\text{ENCODE}}$ wiki page).

TESTING**.** In addition to manually testing each new feature that is put into GALAXY2$^{\text{ENCODE}}$, we have developed an automated test suite. This serves two purposes: it checks that the older parts of GALAXY2$^{\text{ENCODE}}$ keep working when new changes are made, and it enables us to simulate many simultaneous users so we can examine GALAXY2$^{\text{ENCODE}}$'s performance under load. The suite is designed to be run constantly (on a nightly basis) and to automatically warn the developers of any errors. The suite is implemented as a Python script that submits a series of requests to GALAXY2$^{\text{ENCODE}}$ and monitors the responses. It contains tests for all of GALAXY2$^{\text{ENCODE}}$'s features: Table Browser queries, Featured Datasets, Tools, and History retrievals. Needed improvements to this facility include a more comprehensive set of feature tests, the ability to check the accuracy of the output, the ability to test the functioning of the user interface in addition to the GALAXY2$^{\text{ENCODE}}$ core, and the ability to determine if a request is taking too long.

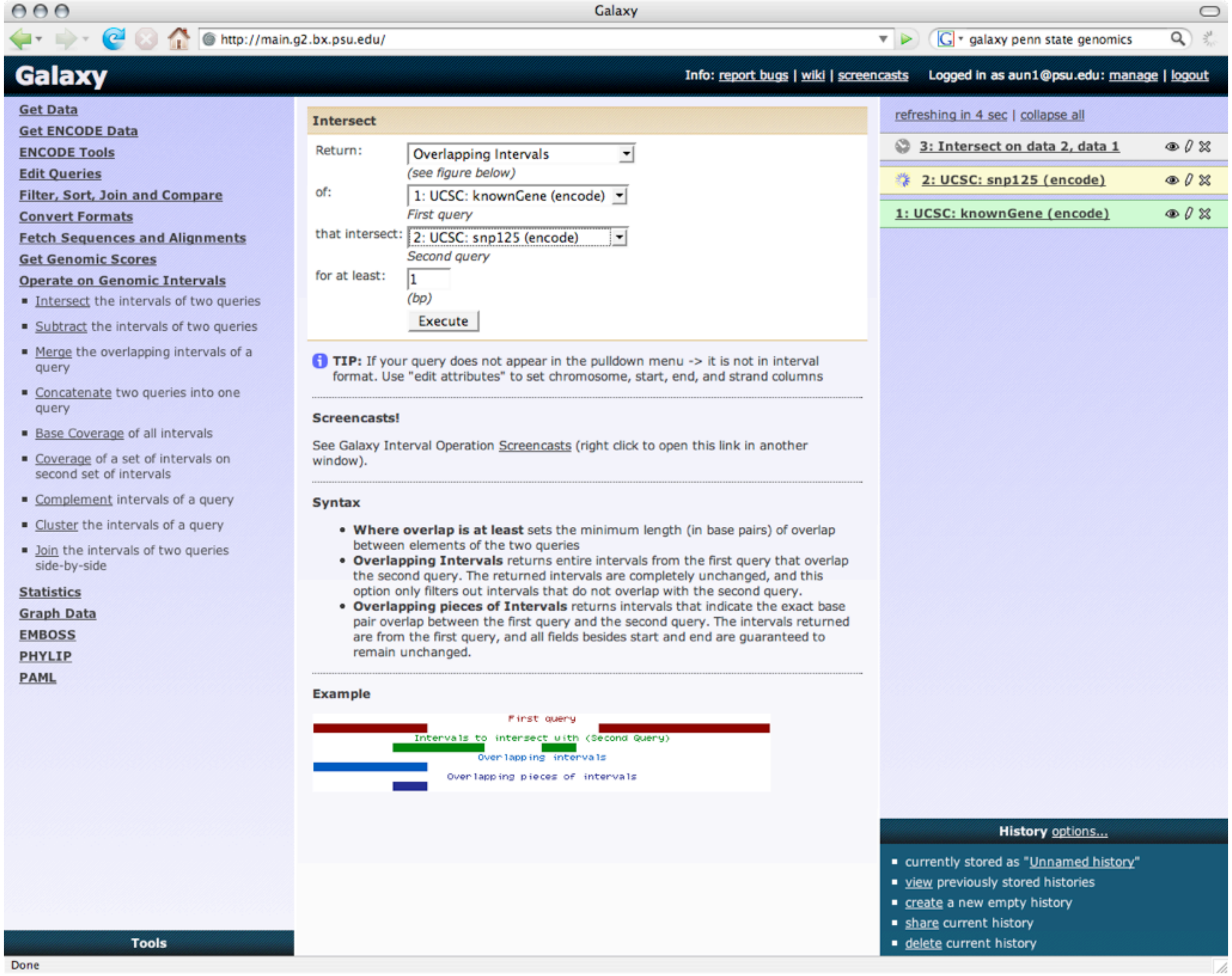
Legends for Supplementary Figures

Figure S1. Galaxy interface contains four areas: the upper bar, tool frame (left column), detail frame (middle column), and history frame (right column). The upper bar contains user account controls as well as help and contact links. The left frame lists the analysis tools and data sources available to the user. The middle frame displays interfaces for tools selected by the user. The right frame (the history frame) shows data and the results of analyses performed by the user. Pictured here are three history items in different stages of completion: Green background = ready; Yellow background with rotating hourglass = computation (in this case upload) in progress; Gray with clock icon = queued (in this case it waits to be executed until history item 2 is finished uploading. This is because history item 3 will contain results of intersection between item 1 and 2). Every action by the user generates a new history item, which can then be used in subsequent analyses, downloaded, or visualized. The Galaxy history page can display results from multiple genome builds, and a single user can have multiple histories.

Figure S2. Genomic vicinity of intergenic Non-Gencode EST DR731323 (highlighted with red arrow). Exons of this EST overlap Exonify predictions and are well conserved in all mammals included in the conservation track of the UCSC Genome Browser.

Figure S3. Genomic vicinity of intergenic EST DB275065. Exons of this EST do not overlap with any protein-coding regions (including experimentally verified or computationally predicted) but coincide with region of high conservation.

Figure S1.


Galaxy
http://main.g2.bx.psu.edu/
galaxy penn state genomics
Galaxy
Info: report bugs | wiki | screencasts
Logged in as aun1@psu.edu: manage | logout
Get Data
Get ENCODE Data
ENCODE Tools
Edit Queries
Filter, Sort, Join and Compare
Convert Formats
Fetch Sequences and Alignments
Get Genomic Scores
Operate on Genomic Intervals
Intersect the intervals of two queries
Subtract the intervals of two queries
Merge the overlapping intervals of a query
Concatenate two queries into one query
Base Coverage of all intervals
Coverage of a set of intervals on second set of intervals
Complement intervals of a query
Cluster the intervals of a query
Join the intervals of two queries side-by-side
Statistics
Graph Data
EMBOSS
PHYLIP
PAML
Tools
Done
refreshing in 4 sec | collapse all
3: Intersect on data 2, data 1
2: UCSC: snp125 (encode)
1: UCSC: knownGene (encode)
History options...
currently stored as "Unnamed history"
view previously stored histories
create a new empty history
share current history
delete current history

Intersect

Return: Overlapping Intervals
(see figure below)

of: 1: UCSC: knownGene (encode)
First query

that intersect: 2: UCSC: snp125 (encode)
Second query

for at least: 1
(bp)

Execute

TIP: If your query does not appear in the pulldown menu -> it is not in interval format. Use "edit attributes" to set chromosome, start, end, and strand columns

Screencasts!

See Galaxy Interval Operation Screencasts (right click to open this link in another window).

Syntax

- **Where overlap is at least** sets the minimum length (in base pairs) of overlap between elements of the two queries
- **Overlapping Intervals** returns entire intervals from the first query that overlap the second query. The returned intervals are completely unchanged, and this option only filters out intervals that do not overlap with the second query.
- **Overlapping pieces of Intervals** returns intervals that indicate the exact base pair overlap between the first query and the second query. The intervals returned are from the first query, and all fields besides start and end are guaranteed to remain unchanged.

Example

First query
Intervals to intersect with (Second Query)
Overlapping intervals
Overlapping pieces of intervals

Fig. S2.

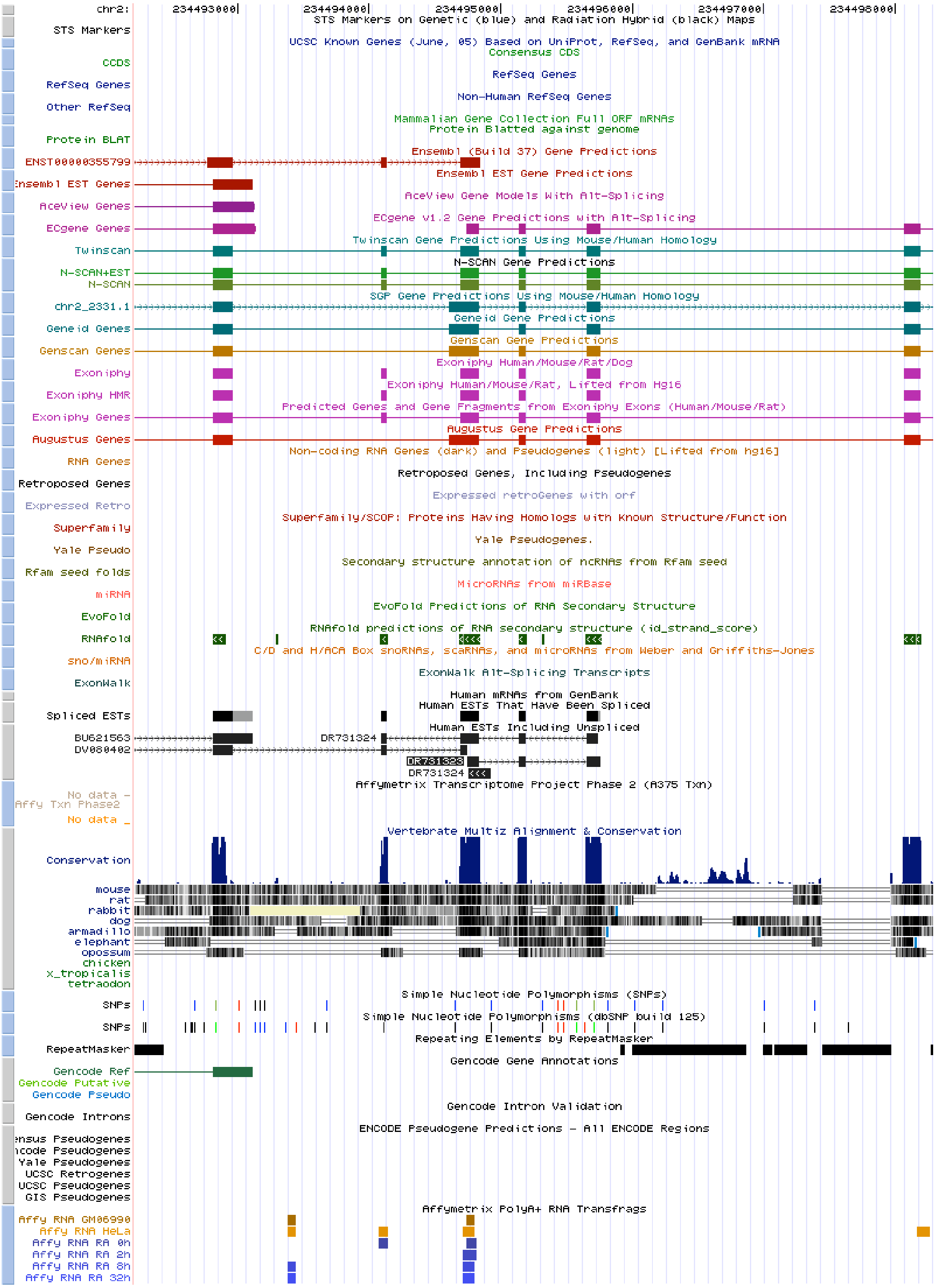
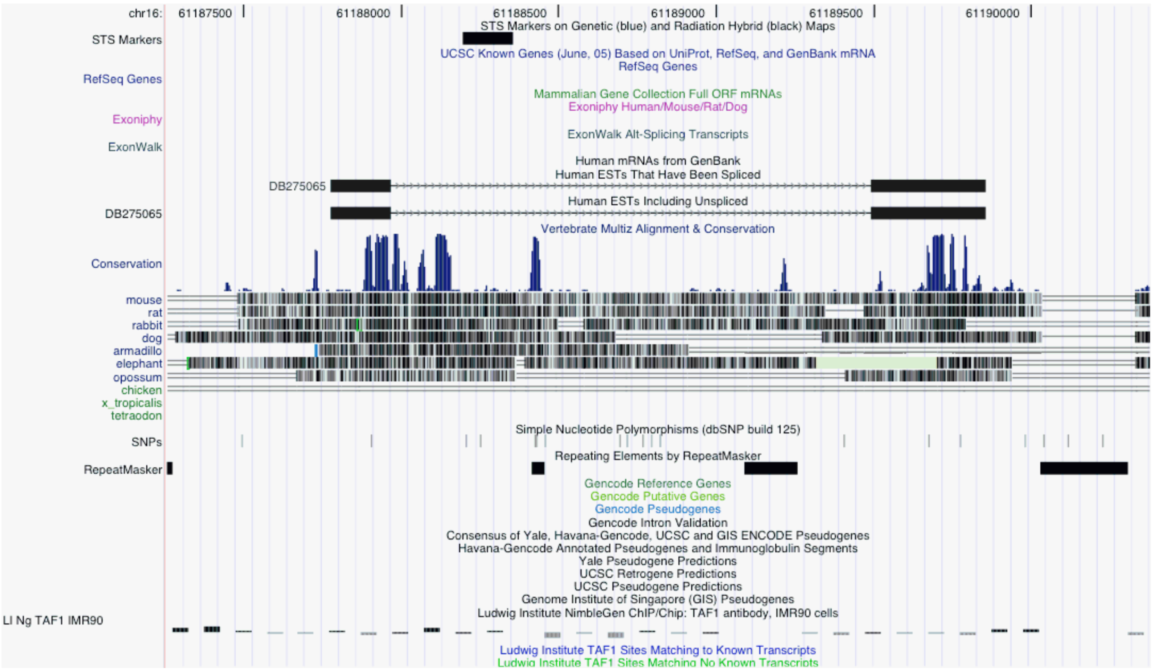

chr2: 234493000| 234494000| 234495000| 234496000| 234497000| 234498000|
STS Markers
STS Markers on Genetic (blue) and Radiation Hybrid (black) Maps
UCSC Known Genes (June, 05) Based on UniProt, RefSeq, and GenBank mRNA
CCDS
Consensus CDS
RefSeq Genes
RefSeq Genes
Other RefSeq
Non-Human RefSeq Genes
Mammalian Gene Collection Full ORF mRNAs
Protein BLAT
Protein Blatted against genome
Ensembl (Build 37) Gene Predictions
ENST00000355799
Ensembl EST Genes
Ensembl EST Gene Predictions
AceView Genes
AceView Gene Models With Alt-Splicing
ECgene Genes
ECgene v1.2 Gene Predictions with Alt-Splicing
Twinscan
Twinscan Gene Predictions Using Mouse/Human Homology
N-SCAN Gene Predictions
N-SCAN+EST
N-SCAN
SGP Gene Predictions Using Mouse/Human Homology
chr2_2331.1
Geneid Genes
Geneid Gene Predictions
Genscan Genes
Genscan Gene Predictions
Exoniphy
Exoniphy Human/Mouse/Rat/Dog
Exoniphy HMR
Exoniphy Human/Mouse/Rat, Lifted from Hg16
Exoniphy Genes
Predicted Genes and Gene Fragments from Exoniphy Exons (Human/Mouse/Rat)
Augustus Genes
Augustus Gene Predictions
RNA Genes
Non-coding RNA Genes (dark) and Pseudogenes (light) [Lifted from hg16]
Retroposed Genes
Retroposed Genes, Including Pseudogenes
Expressed Retro
Expressed retroGenes with orf
Superfamily
Superfamily/SCOP: Proteins Having Homologs with Known Structure/Function
Yale Pseudo
Yale Pseudogenes.
Rfam seed folds
Secondary structure annotation of ncRNAs from Rfam seed
miRNA
MicroRNAs from miRBase
EvoFold
EvoFold Predictions of RNA Secondary Structure
RNAfold
RNAfold predictions of RNA secondary structure (id_strand_score)
sno/miRNA
C/D and H/ACA Box snoRNAs, scaRNAs, and microRNAs from Weber and Griffiths-Jones
ExonWalk
ExonWalk Alt-Splicing Transcripts
Human mRNAs from GenBank
Human ESTs That Have Been Spliced
Spliced ESTs
Human ESTs Including Unspliced
BU621563
DV080402
DR731324
DR731323
DR731324
Affymetrix Transcriptome Project Phase 2 (A375 Txn)
No data -
Affy Txn Phase2
No data _
Vertebrate Multiz Alignment & Conservation
Conservation
mouse
rat
rabbit
dog
armadillo
elephant
opossum
chicken
x_tropicalis
tetraodon
Simple Nucleotide Polymorphisms (SNPs)
SNPs
Simple Nucleotide Polymorphisms (dbSNP build 125)
SNPs
Repeating Elements by RepeatMasker
RepeatMasker
Gencode Gene Annotations
Gencode Ref
Gencode Putative
Gencode Pseudo
Gencode Intron Validation
Gencode Introns
ENCODE Pseudogene Predictions - All ENCODE Regions
nsus Pseudogenes
ncode Pseudogenes
Yale Pseudogenes
UCSC Retrogenes
UCSC Pseudogenes
GIS Pseudogenes
Affymetrix PolyA+ RNA Transfrags
Affy RNA GM06990
Affy RNA HeLa
Affy RNA RA 0h
Affy RNA RA 2h
Affy RNA RA 8h
Affy RNA RA 32h

Figure S3.


chr16:
61187500
61188000
61188500
61189000
61189500
61190000
STS Markers on Genetic (blue) and Radiation Hybrid (black) Maps
STS Markers
UCSC Known Genes (June, 05) Based on UniProt, RefSeq, and GenBank mRNA
RefSeq Genes
RefSeq Genes
Mammalian Gene Collection Full ORF mRNAs
Exoniphy Human/Mouse/Rat/Dog
Exoniphy
ExonWalk
ExonWalk Alt-Splicing Transcripts
Human mRNAs from GenBank
Human ESTs That Have Been Spliced
DB275065
Human ESTs Including Unspliced
DB275065
Vertebrate Multiz Alignment & Conservation
Conservation
mouse
rat
rabbit
dog
armadillo
elephant
opossum
chicken
x_tropicalis
tetraodon
Simple Nucleotide Polymorphisms (dbSNP build 125)
SNPs
Repeating Elements by RepeatMasker
RepeatMasker
Gencode Reference Genes
Gencode Putative Genes
Gencode Pseudogenes
Gencode Intron Validation
Consensus of Yale, Havana-Gencode, UCSC and GIS ENCODE Pseudogenes
Havana-Gencode Annotated Pseudogenes and Immunoglobulin Segments
Yale Pseudogene Predictions
UCSC Retrogene Predictions
UCSC Pseudogene Predictions
Genome Institute of Singapore (GIS) Pseudogenes
Ludwig Institute NimbleGen ChIP/Chip: TAF1 antibody, IMR90 cells
LI Ng TAF1 IMR90
Ludwig Institute TAF1 Sites Matching to Known Transcripts
Ludwig Institute TAF1 Sites Matching No Known Transcripts