

# GS01 0163

## Analysis of Microarray Data

Keith Baggerly and Kevin Coombes

Department of Bioinformatics and Computational Biology  
UT M. D. Anderson Cancer Center

[kabagg@mdanderson.org](mailto:kabagg@mdanderson.org)

[kcoombes@mdanderson.org](mailto:kcoombes@mdanderson.org)

6 November 2007

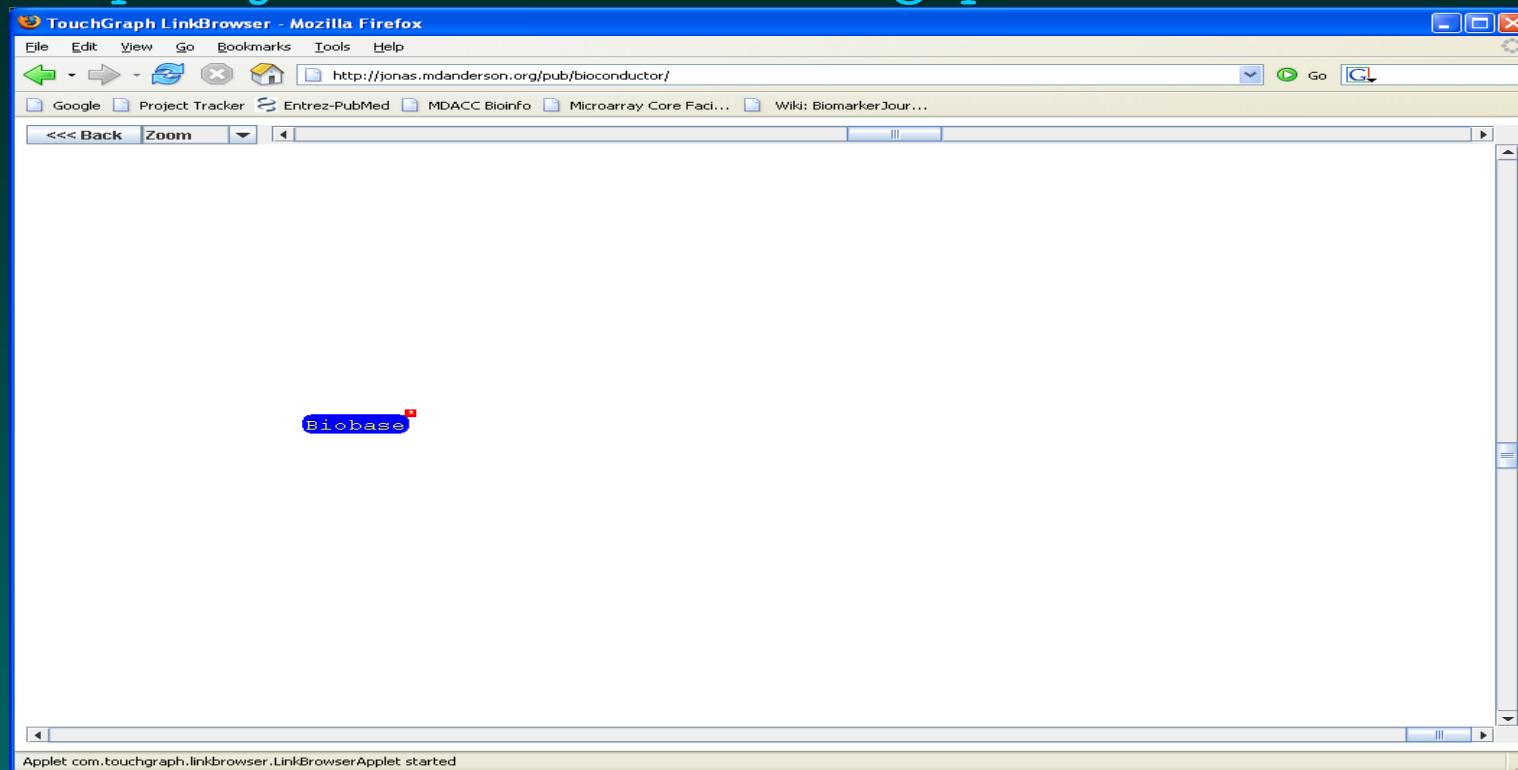
# Lecture 20: Genome Browsing

- Learning What BioConductor Contains
- Annotation Environments in R
- AnnBuilder: Rolling Your Own Annotations
- The UCSC Genome Browser
- Chromosome Locations
- Building a Custom Track
- Viewing Your Custom Track

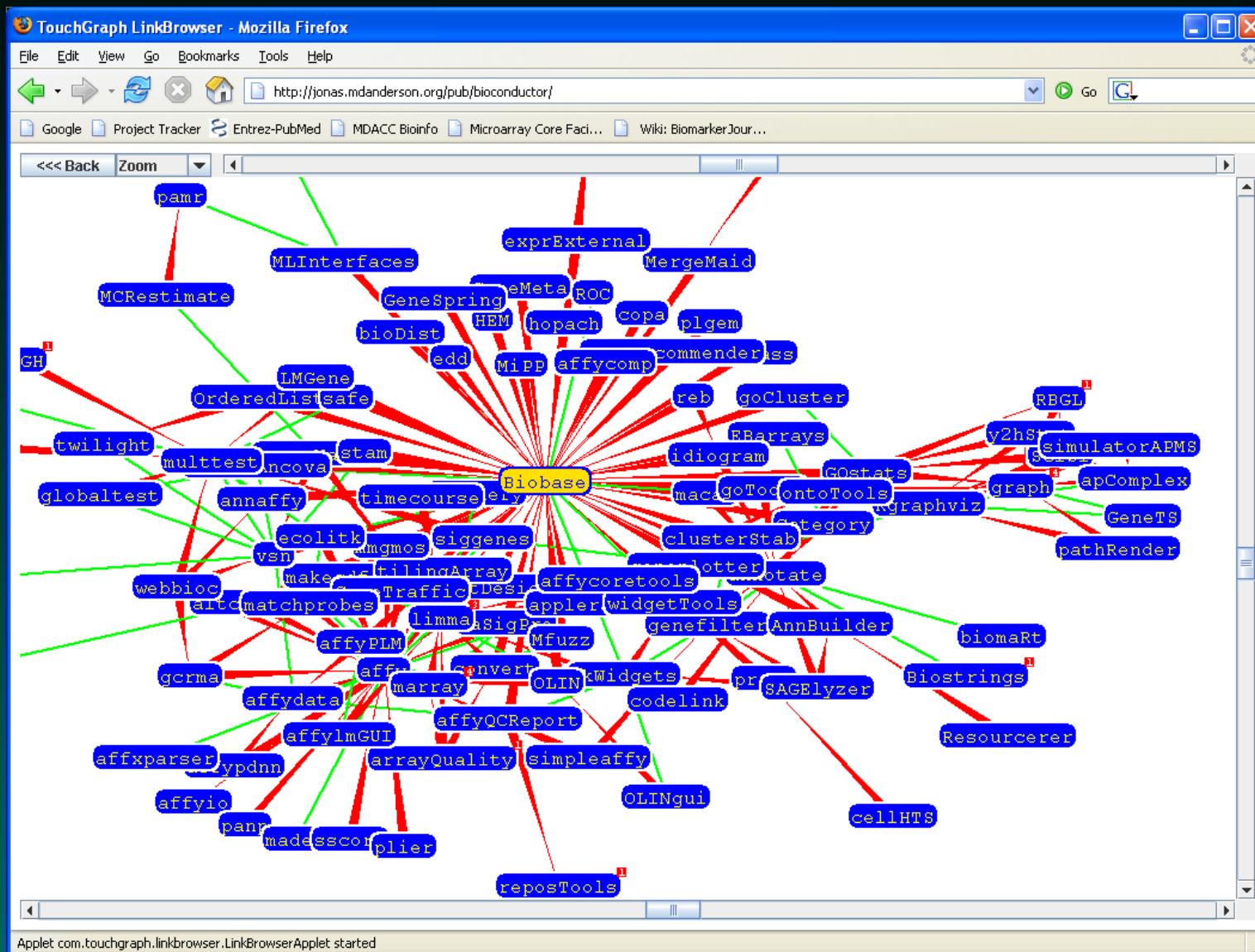
# Learning What BioConductor Contains

We are developing (i.e., it is not completed, so may behave strangely at times) a graphical tool to browse through the BioConductor documentation.

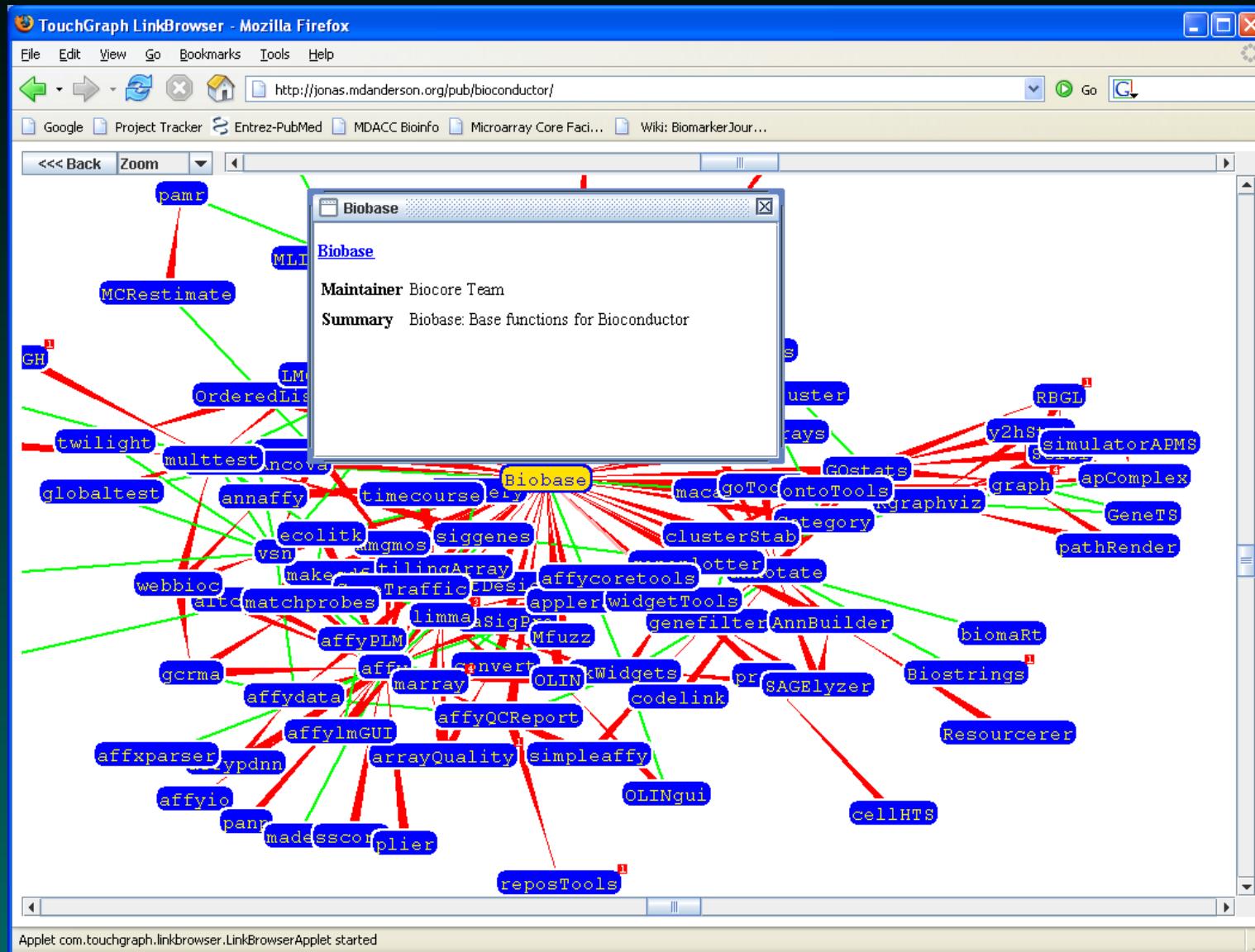
<http://jonas.mdanderson.org/pub/bioconductor/>



# The Documentation Graph



# Hovering the Mouse Gives a Summary



# Left-click Takes You to the Documentation

The screenshot shows a Mozilla Firefox browser window with the title bar "Biobase - Mozilla Firefox". The address bar displays the URL <http://bioconductor.org/packages/1.8/bioc/html/Biobase.html>. The page content is as follows:

## Biobase

### Biobase: Base functions for Bioconductor

Functions that are needed by many other packages or which replace R functions. Suggests: [widgetTools](#), [tkWidgets](#)

**Author** R. Gentleman, V. Carey, M. Morgan, S. Falcon  
**Maintainer** Biocore Team

**Vignettes (Documentation)**

- [Biobase.pdf](#)
- [Bioconductor.pdf](#)
- [esApply.pdf](#)
- [HowTo.pdf](#)

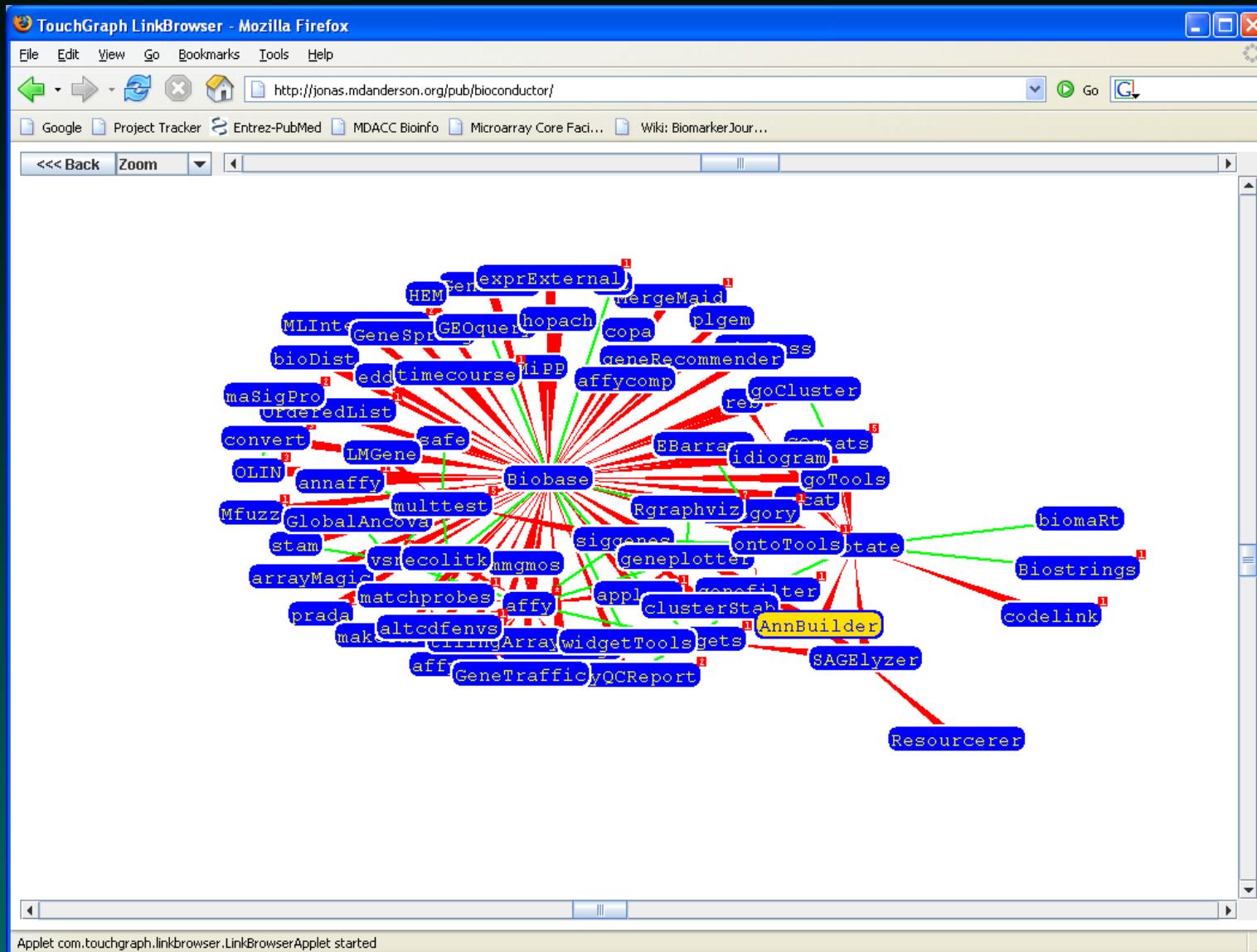
**Package Downloads**

Source		<a href="#">Biobase 1.10.1.tar.gz</a>
Windows binary		<a href="#">Biobase 1.10.1.zip</a>
OS X binary		<a href="#">Biobase 1.10.1.tgz</a>

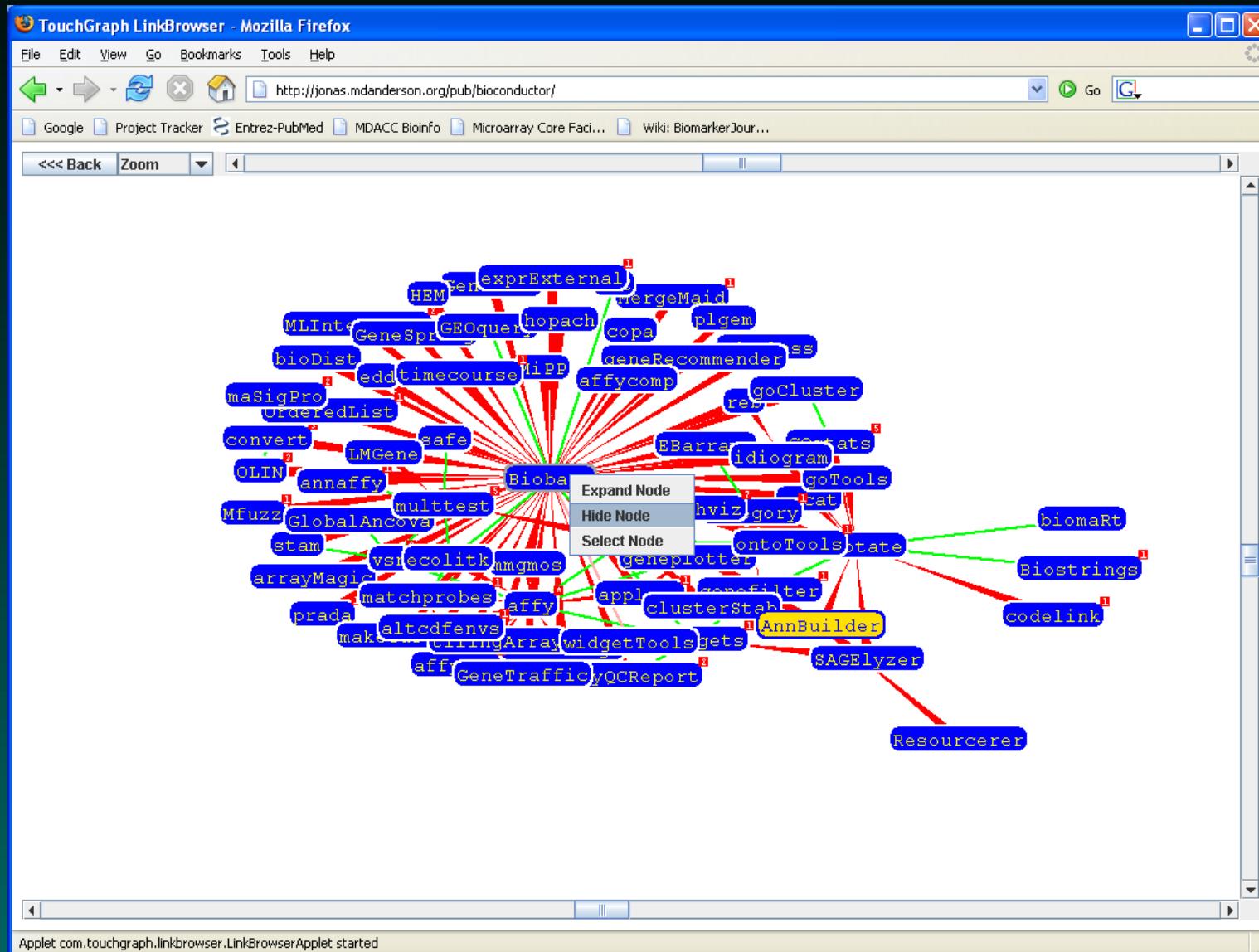
**Details**

biocViews	<a href="#">Infrastructure</a> , <a href="#">Statistics</a>
Depends	R, tools, methods
Suggests	
Imports	
SystemRequirements	

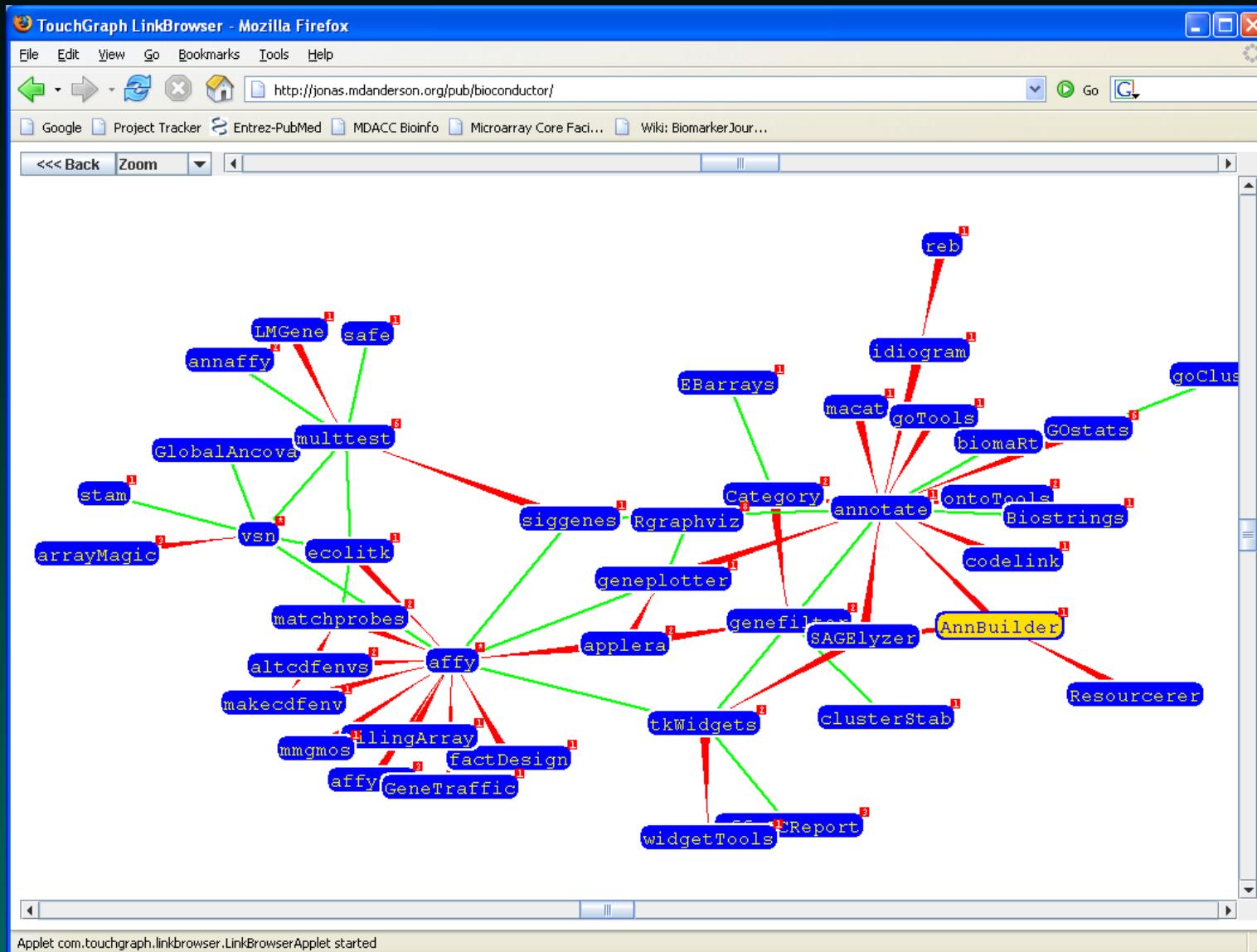
# Left-click Also Recenters on a New Selection



# Right-click Lets You Hide Part of the Graph



# Hiding BioBase Often Clarifies the Structure



# Hubs in the Documentation Graph Are Probably Important

We talked about the `annotate` package previously. It is clear from the graph that this is a central “hub” upon which many of the annotation-related packages depend. (We can also see that `affy` is another hub, defining the basic tools for Affymetrix arrays, and that the `multtest` package for multiple testing is another hub.)

One of the annotation tools that is worth exploring is `biomaRt`, but we are going to leave that for another time. If you want to find out more about the BioMart project, go to <http://www.biomart.org>.

Right now, we want to look at the `AnnBuilder` package.

# Documentation for the AnnBuilder Package

The screenshot shows a Mozilla Firefox browser window with the title bar "AnnBuilder - Mozilla Firefox". The address bar contains the URL "http://bioconductor.org/packages/1.8/bioc/html/AnnBuilder.html". The page content is as follows:

## AnnBuilder

### Bioconductor annotation data package builder

Processing annotation date from public data repositories and building annoation data packages or XML data documents using the source data.

Author J. Zhang  
Maintainer J. Zhang

**Vignettes**  
(Documentation)

[ABPrimer.pdf](#)  
[AnnBuilder.pdf](#)

**Package Downloads**

Source	<a href="#">AnnBuilder 1.10.5.tar.gz</a>
Windows binary	<a href="#">AnnBuilder 1.10.5.zip</a>
OS X binary	<a href="#">AnnBuilder 1.10.5.tgz</a>

**Details**

biocViews	<a href="#">Annotation , Microarray</a>
Depends	R, methods, Biobase, XML, annotate, utils, RSQLite
Suggests	
Imports	
SystemRequirements	

Done

## Annotation Environments in R

For most Affymetrix arrays, annotation packages are available directly (and automatically) from BioConductor whenever you need them. These packages were built using [AnnBuilder](#).

You can load one of these packages as follows:

```
> require(hgu95av2)
```

```
[1] TRUE
```

To see what is in an annotation package, use its name as a function:

```
> hgu95av2()
```

Quality control information for hgu95av2

Date built: Created: Mon Apr 23 12:21:36 2007

Number of probes: 12625

Probe number mismatch: None

Probe missmatch: None

Mappings found for probe based rda files:

hgu95av2ACCNUM found 12625 of 12625

hgu95av2CHR found 12149 of 12625

hgu95av2CHRLLOC found 11730 of 12625

hgu95av2ENZYME found 1861 of 12625

hgu95av2ENTREZID found 12225 of 12625

hgu95av2GENENAME found 12161 of 12625

hgu95av2GO found 11421 of 12625

hgu95av2MAP found 12121 of 12625

hgu95av2MIM found 10157 of 12625

hgu95av2PATH found 4322 of 12625

hgu95av2PFAM found 12046 of 12625

hgu95av2PMID found 12120 of 12625  
hgu95av2PROSITE found 12046 of 12625  
hgu95av2REFSEQ found 12004 of 12625  
hgu95av2SYMBOL found 12161 of 12625  
hgu95av2UNIGENE found 11973 of 12625

Mappings found for non-probe based rda files:

hgu95av2CHRLENGTHS found 25  
hgu95av2ENZYME2PROBE found 677  
hgu95av2G02ALLPROBES found 7501  
hgu95av2G02PROBE found 5339  
hgu95av2PATH2PROBE found 189  
hgu95av2PMID2PROBE found 127350

## Getting Annotations From Environments

Each of the items in the package is an **environment**, which computer scientists may recognize better if we tell them it is a hash table. The key into the probe-based hash table environments is the manufacturers identifier (i.e., an Affymetrix probe set id such as 1854\_at).

```
> get("1854_at", hgu95av2ACCNUM)
```

```
[1] "X13293"
```

```
> get("1854_at", hgu95av2UNIGENE)
```

```
[1] "Hs.179718"
```

```
> get("1854_at", hgu95av2CHR)
```

```
[1] "20"
```

```
> get("1854_at", hgu95av2MAP)
```

```
[1] "20q13.1"
```

```
> get("1854_at", hgu95av2CHRLLOC)
```

```
20  
41729122
```

```
> get("1854_at", hgu95av2SYMBOL)
```

```
[1] "MYBL2"
```

```
> get("1854_at", hgu95av2GENENAME)
```

```
[1] "v-myb myeloblastosis viral oncogene homolog (avian)-like
```

```
> get("1854_at", hgu95av2ENTREZID)
```

```
[1] 4605
```

We have also talked previously about how to find the probe set ids if you start with a gene symbol or a UniGene cluster id.

## AnnBuilder: Rolling Your Own Annotations

We recently had to analyze some data from an Agilent 44K two-color glass microarray. The corresponding annotation package was not available, so we had to build our own. Finding the manufacturers basic annotations was a nontrivial task. We started at the web site (<http://www.agilent.com>), then followed the link under “Products and Services” for “Life Sciences” to get to the “DNA Microarrays” page.

# Follow the Link for “Whole Human Genome”

The screenshot shows a Mozilla Firefox browser window displaying the Agilent Technologies website for DNA Microarrays. The page title is "Agilent | DNA Microarrays - Mozilla Firefox". The URL in the address bar is <http://www.chem.agilent.com/Scripts/PCol.asp?Page=494>. The page header includes the Agilent Technologies logo, the text "Life Sciences & Chemical Analysis", and links for "Select a Country or Area" and "Contact Us". The main navigation menu at the top has links for "Products & Services", "Technical Support", "Industries", "Buy", "About Agilent", and a search bar. Below the menu, there are links for "Home", "Register", and "Login". The main content area features a section titled "DNA Microarrays" with a sub-section "Optimize your experimental design" showing a photograph of a microarray chip. To the right of this is a "Buy" sidebar with links for "Request a quote", "Where to buy", and "Store Home". The bottom left contains a "Related Information" sidebar with links for "Literature Library", "Applications", "Technical Notes", "Brochures", "Posters", "Scientific Publications", "Manuals", "more...", "Technical Support", "Frequently Asked Questions", "Design with eArray", and "more...". The bottom right contains an "Announcements" sidebar for "Agilent Multi-Pack Gene Expression Microarrays" featuring "Human", "Mouse", and "Rat" options, along with links for "Product Announcement", "New 1.3 Version - ChIP Analytics Software", "Support", and "Feature Extraction Software 9.1.3 patch".

# Follow the Link for “Download Gene Lists”

The screenshot shows a Mozilla Firefox browser window displaying the Agilent Technologies website for DNA Microarrays. The page has a blue header bar with the Agilent logo and the title "Agilent | DNA Microarrays - Mozilla Firefox". Below the header is a navigation menu with links for File, Edit, View, Go, Bookmarks, Tools, Help, and a search bar. The main content area features the Agilent Technologies logo and the heading "Life Sciences & Chemical Analysis". A "Select a Country or Area" dropdown and "Contact Us" link are also present. The main content area is titled "DNA Microarrays" and includes a photograph of a microarray slide. To the right of the slide, a box titled "Optimize your experimental design" describes Agilent Printed Microarray Solutions as an integrated, flexible approach for gene expression analysis. A "Buy" sidebar on the right contains links for Request a quote, Where to buy, and Store Home. The bottom left corner of the page shows a "Related Information" sidebar with links to Literature Library, Applications, Technical Notes, Brochures, Posters, Scientific Publications, Manuals, and more. The central content area lists various DNA microarray platforms under "DNA Microarrays", including Arabidopsis 2 (V2), Arabidopsis 3, C. elegans, Canine, Human 1A (V2), M. grisea 2.0, Mouse (V2), Mouse Development 44K, Rat (V2), Rhesus Monkey, Rice, Yeast (V2), Whole Human Genome, Whole Mouse Genome, Whole Rat Genome, Xenopus laevis, Zebrafish, and Custom Gene Expression.

## Reading the Feature Info

In any event, we finally obtained a pair of files that contained the mappings from spots to genomic material. (In addition to the “download gene lists”, you can also follow the link to “Download design files”, but this will only work if you know one of the barcodes on the slides.) We used the `read.table` command to get this file into R:

```
> featureInfo <- read.table("012391_D_DNAFront_BCBottom_2005  
+ header = TRUE, row.names = NULL, sep = "\t",  
+ quote = "", comment.char = "")
```

## Looking at the Feature Info

Here is part of the file:

```
> colnames(featureInfo)
```

```
[1] "Column"          "Row"           "Name"          "ID"  
[5] "RefNumber"       "ControlType"    "GeneName"      "TopHit"  
[9] "Description"
```

```
> featureInfo[1:5, 1:4]
```

	Column	Row	Name	ID
3	103	426	NM_001003689	A_23_P80353
4	103	424	NM_005503	A_23_P158231
5	103	422	NM_004672	A_32_P223017

```
6    103 420 NM_001008727 A_24_P935782
8    103 416      NM_020630 A_24_P343695
```

The critical information is given by the columns that contain the manufacturers identifier (ID) and the GenBank or RefSeq accession number (Name). The function we are going to use to build annotations requires only these two columns (in the reverse order) to be present in a file. So we make them available:

```
> temp <- featureInfo[, c(4, 3)]
> write.table(temp, "agilentGenes.tsv", sep = "\t",
+   quote = FALSE, col.names = NA)
```

# Setting Up the Annotation Package

```
> library(AnnBuilder)
> baseName <- "agilentGenes.tsv"
> baseType <- "gb"
> srcUrls <- getSrcUrl("all", organism = "Homo sapiens")
> myDir <- getwd()
```

## Building the Annotation Package

The next command takes a **very** long time, since it makes calls to databases all over the internet for every one of the 44,000 probes on the array. Be prepared to go get lunch while it executes.

```
ABPkgBuilder(baseName = baseName, srcUrls = srcUrls,  
             baseMapType = baseType, pkgName = "Agilent44K",  
             pkgPath = myDir, organism = "Homo sapiens",  
             version = "1.0", author = list(authors = "krc@mdacc.tmc.  
                                           maintainer = "krc@mdacc.tmc.edu"), fromWeb = TRUE)
```

## Producing the Final Package

This command produces the **source** for a package, which must still be compiled and zipped into a binary package that can be installed easily. This task is most easily accomplished on a UNIX based machine:

```
helios% R CMD build Agilent44K  
helios% R CMD build --binary Agilent44K
```

You can then convert the resulting .tar.gz file to a .zip file, which is the preferred form for distributing a Windows package.

You can check out the results by getting the annotation package from our course web site.

## The Agilent 44K Annotations

```
> library(Agilent44K)
> Agilent44K()
```

Quality control information for Agilent44K  
Date built: Created: Sun Sep 03 07:50:38 2006

Number of probes: 41001

Probe number missmatch: None

Probe mismatch: None

Mappings found for probe based rda files:

Agilent44KACCNUM found 41001 of 41001

Agilent44KCHR found 31185 of 41001

Agilent44KCHRLOC found 28795 of 41001

Agilent44KENZYME found 3056 of 41001

Agilent44KGENENAME found 27824 of 41001  
Agilent44KG0 found 23644 of 41001  
Agilent44KLOCUSID found 31224 of 41001  
Agilent44KMAP found 30939 of 41001  
Agilent44KOMIM found 17942 of 41001  
Agilent44KPATH found 6715 of 41001  
Agilent44KPMID found 30361 of 41001  
Agilent44KREFSEQ found 30057 of 41001  
Agilent44KSUMFUNC found 0 of 41001  
Agilent44KSYMBOL found 31217 of 41001  
Agilent44KUNIGENE found 31010 of 41001

Mappings found for non-probe based rda files:

Agilent44KCHRENGTHS found 25  
Agilent44KENZYME2PROBE found 794  
Agilent44KG02ALLPROBES found 6883  
Agilent44KG02PROBE found 5117

Agilent44KORGANISM found 1  
Agilent44KPATH2PROBE found 183  
Agilent44KPFAM found 21902  
Agilent44KPMID2PROBE found 131104  
Agilent44KPROSITE found 15055

# The UCSC Genome Browser

We are going to shift gears slightly:

<http://genome.ucsc.edu/>

A screenshot of a Mozilla Firefox browser window displaying the UCSC Genome Bioinformatics website. The title bar reads "UCSC Genome Browser Home - Mozilla Firefox". The address bar shows the URL "http://genome.ucsc.edu/". The page content includes a sidebar with links to various tools: Genome Browser, ENCODE, Blat, Table Browser, Gene Sorter, In Silico PCR, VisiGene, Proteome Browser, Utilities, Downloads, and Release Log. The main content area features a section titled "About the UCSC Genome Bioinformatics Site" which describes the site's purpose and tools. Below this is a "News" section with a link to "News Archives". Under "News", there is an announcement about the "Upgraded Custom Tracks Tool" released on October 6, 2006.

**About the UCSC Genome Bioinformatics Site**

This site contains the reference sequence and working draft assemblies for a large collection of genomes. It also provides a portal to the ENCODE project.

We encourage you to explore these sequences with our tools. The Genome Browser zooms and scrolls over chromosomes, showing the work of annotators worldwide. The Gene Sorter shows expression, homology and other information on groups of genes that can be related in many ways. Blat quickly maps your sequence to the genome. The Table Browser provides convenient access to the underlying database. VisiGene lets you browse through a large collection of *in situ* mouse and frog images to examine expression patterns.

**News**

To receive announcements of new genome assembly releases, new software features, updates and training seminars by email, subscribe to the [genome-announce](#) mailing list.

**6 October 2006 – Announcing Upgraded Custom Tracks Tool:**

We are pleased to announce the release of an upgraded software tool in the Genome Browser collection — the Custom Tracks [tool](#).

The new Custom Tracks Tool provides more flexibility and a more user-friendly interface for creating and managing your custom tracks than the tool it replaces.

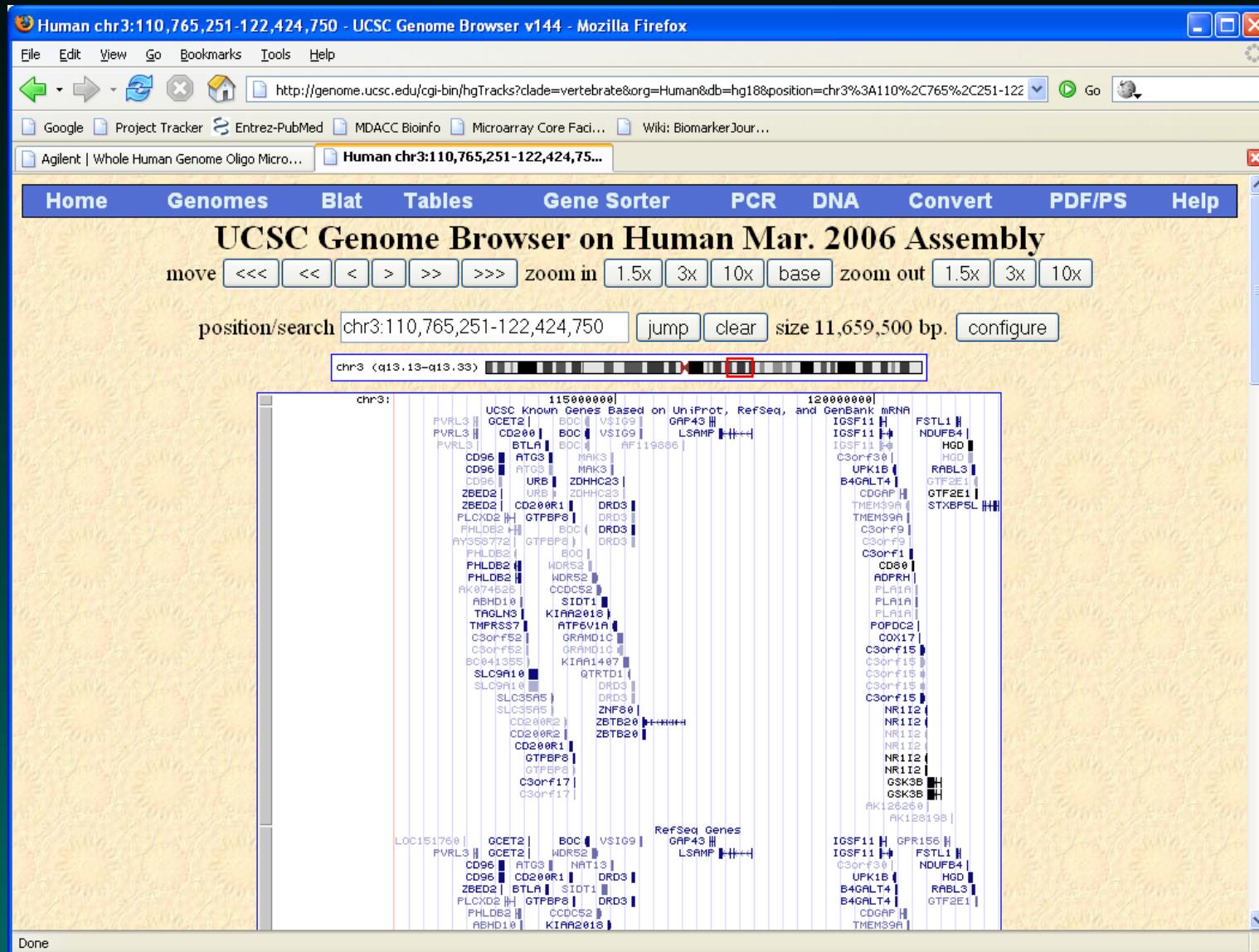
# Follow the Link to “Genome Browser”

The screenshot shows a Mozilla Firefox browser window displaying the UCSC Genome Browser. The title bar reads "Human (Homo sapiens) Genome Browser Gateway - Mozilla Firefox". The address bar shows the URL <http://genome.ucsc.edu/cgi-bin/hgGateway>. The toolbar includes standard buttons for back, forward, search, and refresh. The menu bar has options like File, Edit, View, Go, Bookmarks, Tools, and Help. The window title is "Human (Homo sapiens) Genome Bro...". Below the title bar is a navigation menu with links: Home, Genomes, Blat, Tables, Gene Sorter, PCR, FAQ, and Help. The main content area is titled "Human (*Homo sapiens*) Genome Browser Gateway". It contains a message from the UCSC Genome Bioinformatics Group of UC Santa Cruz, noting the software's copyright by the Regents of the University of California. A search form is present, with dropdown menus for "clade" (Vertebrate), "genome" (Human), "assembly" (Mar. 2006), and "position or search term" (chr3:110,765,251-122,424,750). An "image width" input field is set to 620, and a "submit" button is nearby. Below the search form is a link to reset settings. At the bottom of the search area are three buttons: "add custom tracks", "configure tracks and display", and "clear position". The main content area also includes a section titled "About the Human Mar. 2006 (hg18) assembly (sequences)" which provides information about the March 2006 human reference sequence. It lists the International Human Genome Sequencing Consortium as the producer. A "Sample position queries" section follows, explaining how genome positions can be specified. It includes a table showing examples of valid position queries:

Request:	Genome Browser Response:
chr7	Displays all of chromosome 7

A "Done" button is located at the bottom left of the main content area.

# Press “Submit” to Start Browsing



## About the Genome Browser

The genome browser lets you see a great deal of information laid out along the latest completed build of the human genome. The most obvious thing to look at are the known genes, which are typically displayed in such a way that you can see the individual introns and exons (provided you zoom in closely).

For our purposes (as people who analyze microarray data), an extremely interesting feature of the Genome Browser is that it lets you add your own “Custom Tracks”, which is their name for a set of annotations you can define.

# Custom Tracks

To learn about the genome (custom) tracks, go to the FAQ.

The screenshot shows a Mozilla Firefox browser window with the title bar "UCSC Genome Bioinformatics: FAQ - Mozilla Firefox". The address bar contains the URL "http://genome.ucsc.edu/FAQ/". The main content area displays the "FAQ Table of Contents" for the UCSC Genome Bioinformatics website. The page lists various topics under the heading "FAQ Table of Contents", including "Display Problems", "Assembly Releases and Versions", "Data and Downloads", "Genome Browser Tracks", "Custom Annotation Tracks", "Data File Formats", "Blat", "Citing the Genome Browser", "Linking to the Genome Browser", "Mirroring or Licensing the Genome Browser", and "Posting a question to the mailing list". Below the table of contents are two search boxes: "Search the Genome Browser FAQ:" and "Search the entire Genome Browser website:", each with a "Submit" button. The browser interface includes standard navigation buttons (back, forward, search, etc.) and a menu bar with options like File, Edit, View, Go, Bookmarks, Tools, and Help.

# BED Format

The screenshot shows a Mozilla Firefox browser window displaying the UCSC Genome Browser User's Guide. The title bar reads "UCSC Genome Browser: User's Guide - Mozilla Firefox". The address bar shows the URL <http://genome.ucsc.edu/goldenPath/help/hgTracksHelp.html#BED>. The page content is titled "BED Lines" and describes the BED format, its required fields (chrom, chromStart, chromEnd), and optional fields (name, score, strand, thickStart, thickEnd, itemRgb). The text is presented in a standard web page layout with headings and numbered lists.

**BED Lines**

BED format provides a flexible way to define the data lines that are displayed in an annotation track. BED lines have three required fields and nine additional optional fields. The number of fields per line must be consistent throughout any single set of data in an annotation track.

The first three required BED fields are:

1. **chrom** - The name of the chromosome (e.g. chr3, chrY, chr2\_random) or scaffold (e.g. scaffold10671).
2. **chromStart** - The starting position of the feature in the chromosome or scaffold. The first base in a chromosome is numbered 0.
3. **chromEnd** - The ending position of the feature in the chromosome or scaffold. The *chromEnd* base is not included in the display of the feature. For example, the first 100 bases of a chromosome are defined as *chromStart*=0, *chromEnd*=100, and span the bases numbered 0-99.

The 9 additional optional BED fields are:

4. **name** - Defines the name of the BED line. This label is displayed to the left of the BED line in the Genome Browser window when the track is open to full display mode or directly to the left of the item in pack mode.
5. **score** - A score between 0 and 1000. If the track line *useScore* attribute is set to 1 for this annotation data set, the *score* value will determine the level of gray in which this feature is displayed (higher numbers = darker gray).
6. **strand** - Defines the strand - either '+' or '-'.
7. **thickStart** - The starting position at which the feature is drawn thickly (for example, the start codon in gene displays).
8. **thickEnd** - The ending position at which the feature is drawn thickly (for example, the stop codon in gene displays).
9. **itemRgb** - An RGB value of the form R,G,B (e.g. 255,0,0). If the track line *itemRgb* attribute is set to "On", this RGB value

## Chromosome Locations

You can read more of the custom track documentation on your own; here, we are going to focus on how to build a custom track in R. The first thing we want to point out is that we need to know both the starting base location and the ending base location in order to build a custom track. Thus, the CHRLOC annotations that the *AnnBuilder* BioConductor package constructs are not adequate.

Fortunately, we can get start and end points directly from the folks at the UCSC Genome Browser. Go back to the main page, then follow the link for “Downloads”.

# UCSC Download Page

The screenshot shows a Mozilla Firefox browser window displaying the UCSC Genome Bioinformatics Downloads page. The title bar reads "UCSC Genome Browser: Downloads - Mozilla Firefox". The address bar shows the URL "http://hgdownload.cse.ucsc.edu/downloads.html". The page content is titled "Sequence and Annotation Downloads". It contains text about the availability of sequence and annotation data downloads for various genome assemblies, mentioning the FTP server and the "current genomes" directory. It also provides instructions for viewing table descriptions and credits. A list of genome species links is provided at the bottom:

- [Human](#)
- [Chimpanzee](#)
- [Rhesus](#)
- [Dog](#)
- [Cow](#)
- [Mouse](#)
- [Rat](#)

# Follow the link for “Human”

The screenshot shows a Mozilla Firefox browser window with the title bar "UCSC Genome Browser: Downloads - Mozilla Firefox". The address bar displays the URL "http://hgdownload.cse.ucsc.edu/downloads.html#human". The main content area is titled "Human Genome" and "Mar. 2006 (hg18)". Below this, a bulleted list provides links to various pairwise alignments:

- [Full data set](#)
- [Data set by chromosome](#)
- [Annotation database](#)
- [Human self alignments](#)
- [Human/Chimp \(panTro2\) pairwise alignments](#)
- [Human/Chimp \(panTro1\) pairwise alignments](#)
- [Human/Rhesus \(rheMac2\) pairwise alignments](#)
- [Human/Cow \(bosTau2\) pairwise alignments](#)
- [Human/Dog \(canFam2\) pairwise alignments](#)
- [Human/Mouse \(mm8\) pairwise alignments](#)
- [Human/Mouse \(mm7\) pairwise alignments](#)
- [Human/Rat \(rn4\) pairwise alignments](#)
- [Human/Opossum \(monDom4\) pairwise alignments](#)
- [Human/Chicken \(galGal3\) pairwise alignments](#)
- [Human/Chicken \(galGal2\) pairwise alignments](#)
- [Human/Zebrafish \(danRer4\) pairwise alignments](#)

# In “Annotation Database”, Scroll To “refGene”

Index of /goldenPath/hg18/database			
<a href="#">productName.txt.gz</a>	08-Oct-2006	13:14	4.4M
<a href="#">recombRate.sql</a>	13-Apr-2006	12:46	877
<a href="#">recombRate.txt.gz</a>	13-Apr-2006	12:46	88K
<a href="#">refFlat.sql</a>	08-Oct-2006	13:26	1.6K
<a href="#">refFlat.txt.gz</a>	08-Oct-2006	13:26	2.0M
<a href="#">refGene.sql</a>	08-Oct-2006	12:54	1.9K
<a href="#">refGene.txt.gz</a>	08-Oct-2006	12:54	2.2M
<a href="#">refLink.sql</a>	08-Oct-2006	13:26	1.6K
<a href="#">refLink.txt.gz</a>	08-Oct-2006	13:26	4.9M
<a href="#">refSeqAli.sql</a>	08-Oct-2006	12:45	2.1K
<a href="#">refSeqAli.txt.gz</a>	08-Oct-2006	12:45	2.4M
<a href="#">refSeqStatus.sql</a>	08-Oct-2006	13:30	1.2K
<a href="#">refSeqstatus.txt.gz</a>	08-Oct-2006	13:30	681K
<a href="#">refSeqSummary.sql</a>	08-Oct-2006	13:30	1.3K
<a href="#">refSeqSummary.txt.gz</a>	08-Oct-2006	13:30	2.0M
<a href="#">reqPotential7X.sql</a>	24-Jun-2006	07:48	1.7K
<a href="#">reqPotential7X.txt.gz</a>	24-Jun-2006	07:49	59M
<a href="#">rnBlastTab.sql</a>	22-Jul-2006	11:29	1.6K
<a href="#">rnBlastTab.txt.gz</a>	22-Jul-2006	11:30	586K
<a href="#">scBlastTab.sql</a>	13-Apr-2006	12:46	797
<a href="#">scBlastTab.txt.gz</a>	13-Apr-2006	12:46	350K
<a href="#">seq.sql</a>	13-Apr-2006	10:13	586
<a href="#">seq.txt.gz</a>	13-Apr-2006	10:14	29M
<a href="#">sov.sql</a>	08-Oct-2006	12:45	1.2K

## Using the RefGene locations in R

Load the file.

```
> refgene <- read.table("refGene.txt", header = FALSE,  
+ sep = "\t", comment.char = "", quote = "")
```

Add the column names, which are not included.

```
> colnames(refgene) <- c("bin", "name", "chrom",  
+ "strand", "txStart", "txEnd", "cdsStart",  
+ "cdsEnd", "exonCount", "exonStarts", "exonEnds",  
+ "id", "name2", "cdsStartStat", "cdsEndStat",  
+ "exonFrames")
```

We are going to ignore the intron and exon boundaries. We are also going to remove duplicate entries, which seem for some reason to exist;

the search to identify these is time consuming.

```
> temprg <- refgene[, c(1:9, 13:15)]
> omit <- unlist(lapply(levels(temprg$name), function(x,
+ n) {
+   which(n == x)[1]
+ }), as.character(temprg$name)))
> summary(omit)
> refgene <- temprg[omit, ]
> rownames(refgene) <- as.character(refgene[, "name"])
```

Finally, we save this as a binary object that we can load later.

```
> save(refgene, file = "refgene.rda")
```

## Linking the Agilent Array to RefGene locations

First, convert the environment in the AnnBuilder package for the Agilent 44K arrays to a list.

```
> temp2 <- as.list(Agilent44KREFSEQ)
```

Next, we produce a list that maps the annotations to the spots. This code works because the ID column of the `featureInfo` object contains RefSeq IDs (primarily), which are the names of the rows in the `temp2` object we just created.

```
> ag.annoList <- temp2[as.character(featureInfo[,
+           "ID"])]
```

# Alternative Splicing

```
> ag.annoList[1]
```

```
$A_23_P80353
```

```
[1] "NM_001003689" "NP_001003689" "NM_031488"  
[4] "NP_113676"
```

Notice that some probes are associated with more than one RefSeq gene; this happens because different isoforms (produced by alternative splicing) of the same gene have different RefSeq identifiers. That is, the same piece of DNA can give rise to different mRNA molecules. So, we now search through and select just the first annotation for each spot.

```
> agilent.lc <- unlist(lapply(ag.annoList, length))  
> agilentREFSEQ <- unlist(lapply(ag.annoList, function(x) {
```

```
+     if (length(x) == 0) {
+         return(NA)
+
+     }
+
+     if (length(x) == 1) {
+         return(x)
+
+     }
+
+     idx <- 1
+
+     while (idx <= length(x)) {
+
+         if (x[[idx]] == "") {
+
+             idx <- idx + 1
+
+             next
+
+         }
+
+         return(x[[idx]])
+
+     }
+
+     return(NA)
+
+ }))
```

```
> agilentREFSEQ[agilentREFSEQ == ""] <- NA
```

```
> length(agilentREFSEQ)
```

```
[1] 41675
```

```
> sum(!is.na(agilentREFSEQ))
```

```
[1] 30612
```

Finally, we use the updated RefSeqs (that we just constructed in the `agilentREFSEQ` object) as indices into the `refgene` chromosome locations above. This computation is also slow, since it uses a search in a list instead of in a hash.

```
> agilent2refgene <- refgene[agilentREFSEQ, ]
```

```
> agilent2refgene[1:3, ]
```

	bin		name	chrom	strand	txStart	
NM_001003689	889	NM_001003689		chr22		+	39931258
NM_005503	98	NM_005503		chr15		+	27001144
NM_004672	795	NM_004672		chr1		-	27554256
		txEnd	cdsStart	cdsEnd	exonCount		name2
NM_001003689	39957220	39931312	39953547		18	L3MBTL2	
NM_005503	27197806	27133379	27196628		14	APBA2	
NM_004672	27565924	27554468	27565675		29	MAP3K6	
	cdsStartStat	cdsEndStat					
NM_001003689		cmpl		cmpl			
NM_005503		cmpl		cmpl			
NM_004672		cmpl		cmpl			

## Building a Custom Track

We analyzed the Agilent 44K microarray data using a linear model. The results are contained in an object called `ourResults`:

```
> summary(ourResults)
```

UntreatedMeanLog	Beta	PValue
Min. : 4.870	Min. :-3.15530	Min. : 2.024e-09
1st Qu.: 6.907	1st Qu.:-0.19572	1st Qu.: 8.142e-02
Median : 8.058	Median :-0.05431	Median : 2.749e-01
Mean : 8.742	Mean :-0.04300	Mean : 3.511e-01
3rd Qu.: 9.982	3rd Qu.: 0.10075	3rd Qu.: 5.823e-01
Max. :16.523	Max. : 3.27672	Max. : 1.000e+00

## Computing a Displayable Score

We are going to use the p-values to decide which genes to display, and we are going to use the coefficient (Beta) to compute a score that shows the amount of differential expression. The allowed scores for a custom track range from 0 to 1000. Since the true values of Beta range between  $-3$  and  $+3$  (more or less), we are going to multiply by 300 to get a useful score.

```
> score <- 300 * ourResults[, "Beta"]  
> score[score > 1000] <- 1000  
> score[score < -1000] <- -1000  
> score <- abs(score)
```

## A Track Data Frame

Now we build a data frame that includes the information we need for a custom track in the desired order:

```
> temp <- data.frame(agilent2refgene[, c("chrom",
+      "txStart", "txEnd", "name2")], score = score,
+      strand = agilent2refgene[, "strand"])
> temp[1:3, 1:5]
```

	chrom	txStart	txEnd	name2	score
NM_001003689	chr22	39931258	39957220	L3MBTL2	96.902254
NM_005503	chr15	27001144	27197806	APBA2	74.415391
NM_004672	chr1	27554256	27565924	MAP3K6	2.281971

## Significant Overexpressed Genes

We built this data frame for all genes; now we are going to select the ones that are significant ( $p\text{-value} < 0.02$ ) and are overexpressed in response to the treatment ( $\beta > 0$ ). We further restrict to those genes that we are able to map to the genome.

```
> trackInfo <- temp[!is.na(temp[, "chrom"]) & ourResults[,  
+   "PValue"] < 0.02 & ourResults[, "Beta"] >  
+   0, ]
```

We also have to create a header line that tells the browser to make use of the scores.

```
> trackheader <- paste("track name=upNormal",  
+   "description=\"Increased in Normal Cells\"",  
+   "useScore=1 color=0,60,120")
```

## Writing the Track Info to a File

We can now write the header line followed by the track data:

```
> write(trackheader, file = "upNormalRNA.tsv",
+       append = FALSE)
> write.table(trackInfo, file = "upNormalRNA.tsv",
+              append = TRUE, quote = FALSE, sep = "\t",
+              row.names = FALSE, col.names = FALSE)
```

Finally, we do the same thing for the genes that are underexpressed.

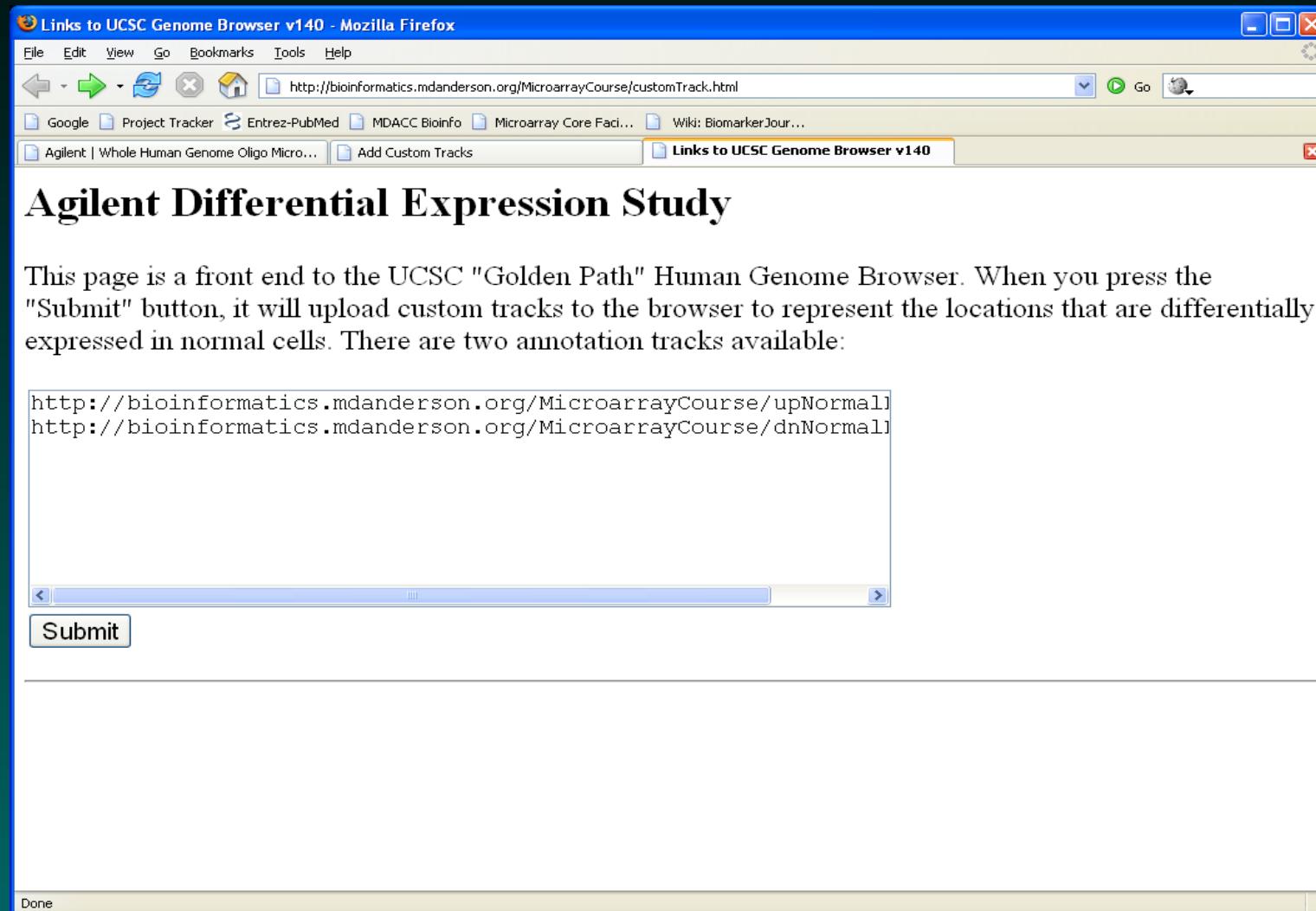
```
> trackInfo <- temp[!is.na(temp[, "chrom"]) & ourResults[,  
+      "PValue"] < 0.02 & ourResults[, "Beta"] <  
+      0, ]  
  
> trackheader <- paste("track name=downNormal",  
+      "description=\"Decreased in Normal Cells\"",  
+      "useScore=1 color=100,50,0")  
  
> write(trackheader, file = "dnNormalRNA.tsv",  
+      append = FALSE)  
> write.table(trackInfo, file = "dnNormalRNA.tsv",  
+      append = TRUE, quote = FALSE, sep = "\t",  
+      row.names = FALSE, col.names = FALSE)
```

# Viewing Your Custom Track

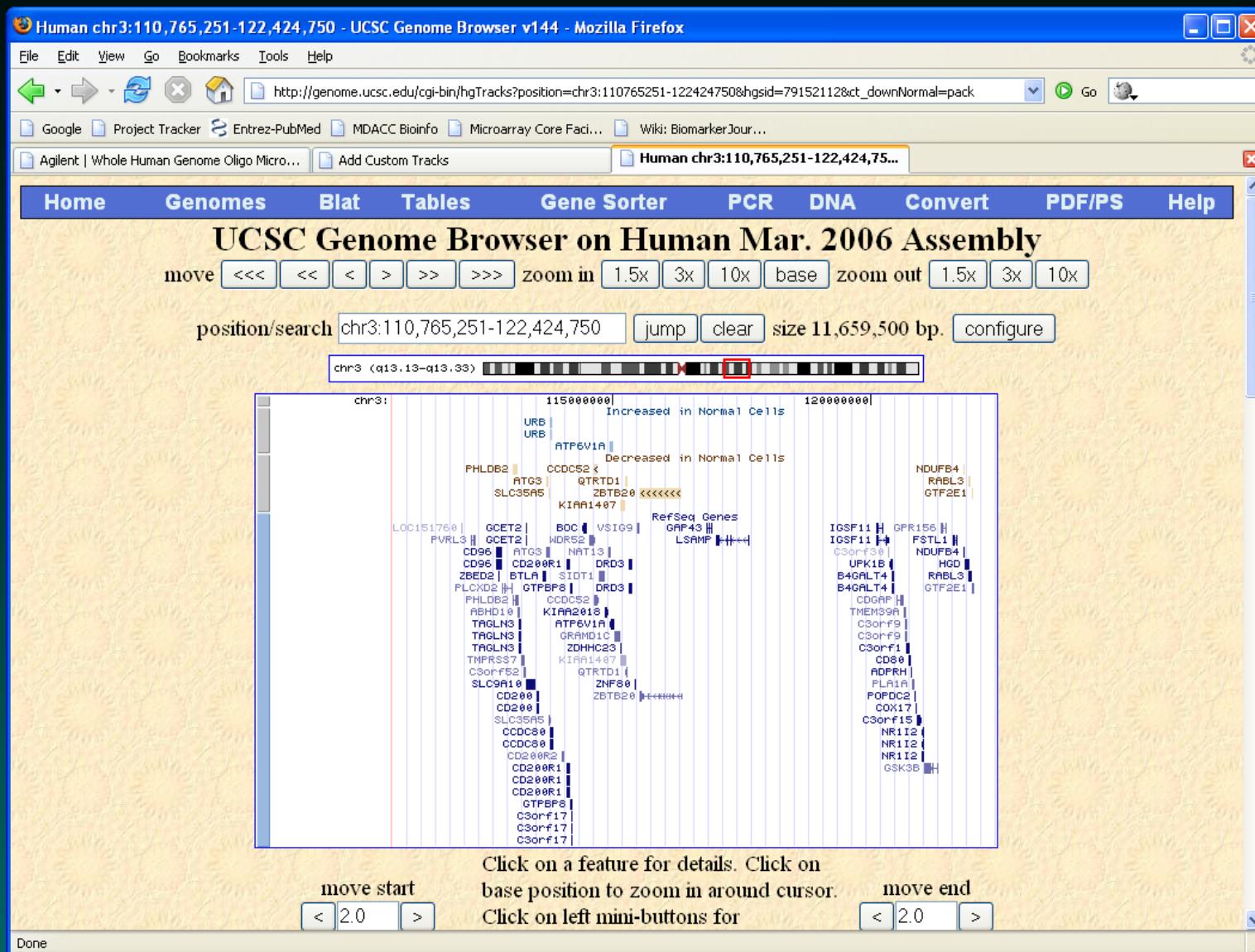
Now we can return to the genome browser and look at our custom tracks. Unfortunately, their web page only lets you attach one at a time unless you can make them available from a web site:

The screenshot shows a Mozilla Firefox browser window with the title "Add Custom Tracks - Mozilla Firefox". The address bar displays the URL <http://genome.ucsc.edu/cgi-bin/hgCustom?hgSID=79151647>. The page content is titled "Add Custom Tracks" and contains instructions for adding custom annotation tracks in BED, GFF, WIG, or PSL formats. It features two main input fields: "Paste URLs or data:" and "Optional track documentation:", each with a corresponding "Browse..." button and a "Clear" button. On the right side of the form are "Submit" and "Cancel" buttons.

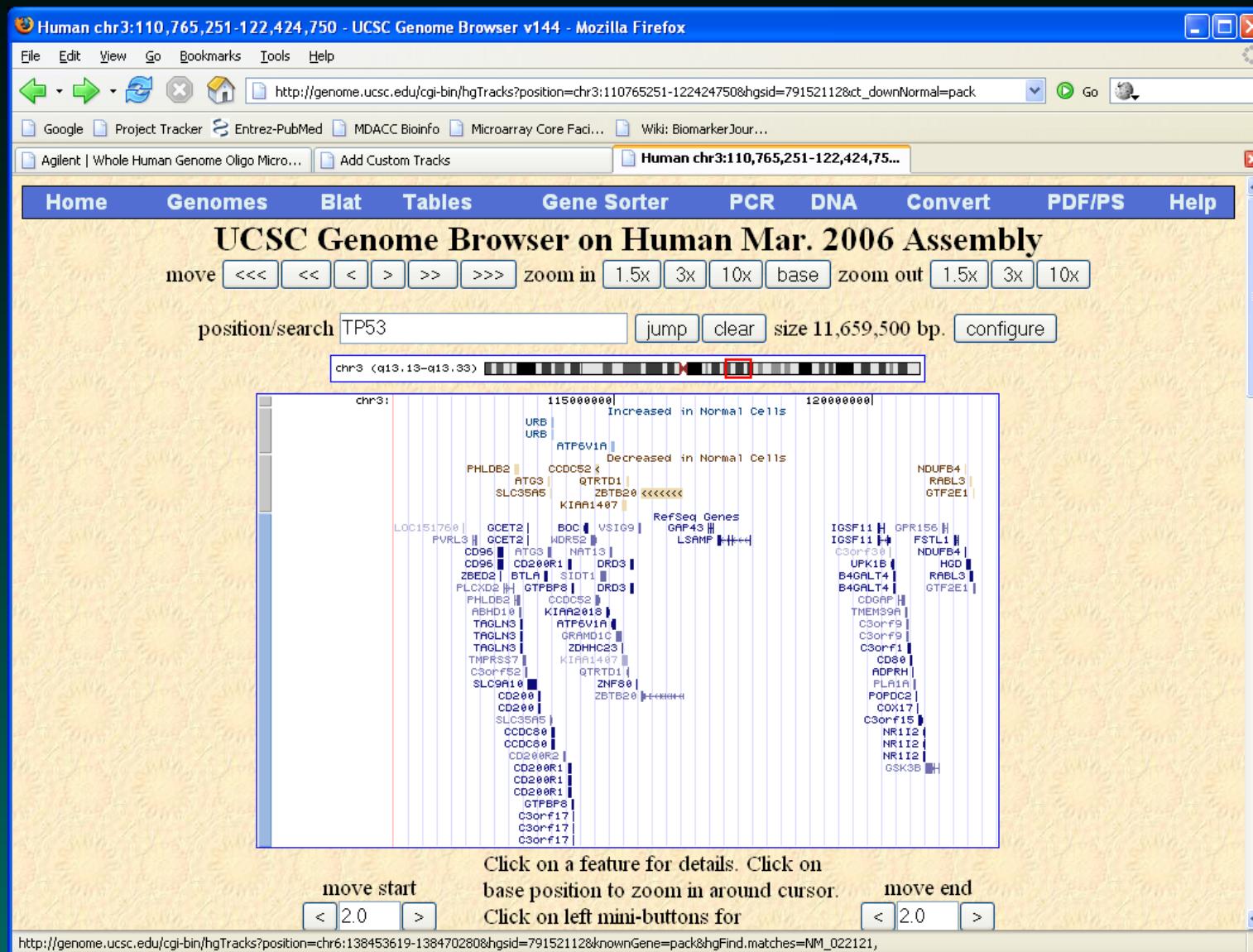
[http://bioinformatics.mdanderson.org/  
MicroarrayCourse/customTrack.html](http://bioinformatics.mdanderson.org/MicroarrayCourse/customTrack.html)



# Displaying Our Tracks



# Searching for a Gene

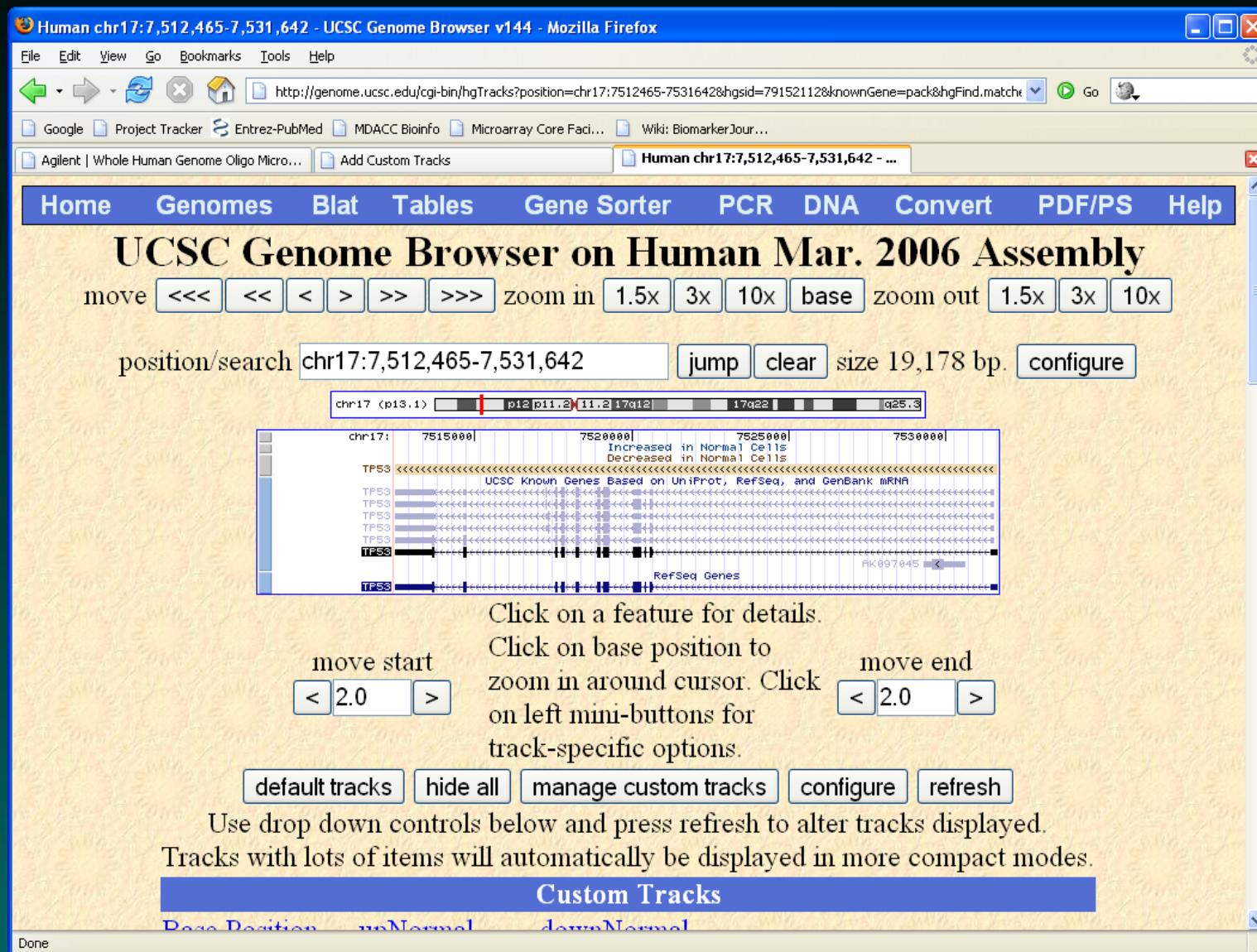


# Searching for a Gene

The screenshot shows a Mozilla Firefox browser window with the title "Human TP53 - UCSC Genome Browser v144 - Mozilla Firefox". The address bar displays the URL <http://genome.ucsc.edu/cgi-bin/hgTracks>. The main content area is titled "Known Genes" and lists numerous genes and their descriptions. The genes listed include TP53, C20orf10, TP53AP1, TP53RK, PERP, RPRM, RRM2B, TP53 (BC003596), VRK1, TP53INP1, UBE2L6, PPP1R13B, TP53BP2, ENC1, PPM1D, TP53BP2 (NM 001031685), GNL3, ING5, PDRG1, and ING4. Each entry provides the gene name, NM identifier, chromosomal location, and a brief description.

Gene	Location	Description
TP53 (NM 000546)	chr17:7512465-7531642	tumor protein p53
TP53 (DQ186649)	chr17:7512447-7531524	Dell133 p53 gamma isoform.
TP53 (DQ186649)	chr17:7512447-7531524	Dell133 p53 gamma isoform.
TP53 (DQ186648)	chr17:7512447-7531524	Dell133 p53 beta isoform.
TP53 (DQ186648)	chr17:7512447-7531524	Dell133 p53 beta isoform.
C20orf10 (NM 014477)	chr20:43435935-43440371	TP53-target gene 5 protein
TP53AP1 (NM 007233)	chr7:86792477-86812767	TP53 activated protein 1
TP53RK (BC019621)	chr20:44747581-44751486	TP53 regulating kinase.
PERP (NM 022121)	chr6:138453619-138470280	PERP, TP53 apoptosis effector
RPRM (NM 019845)	chr2:154042098-154043568	represso, TP53 dependant G2 arrest mediator
RRM2B (NM 015713)	chr8:103285907-103320522	ribonucleotide reductase M2 B (TP53 inducible
TP53 (BC003596)	chr17:7512465-7531511	Dell133 p53 isoform.
VRK1 (BC103761)	chr14:96333459-96417696	vaccinia related kinase 1
TP53INP1 (NM 033285)	chr8:96007377-96030767	tumor protein p53 inducible nuclear protein
TP53INP1 (AF409114)	chr8:96007377-96030767	tumor protein p53 inducible nuclear protein 1
UBE2L6 (BC032491)	chr11:57075712-57091756	ubiquitin-conjugating enzyme E2L 6
PPP1R13B (NM 015316)	chr14:103269842-103383555	protein phosphatase 1, regulatory (inhibitory) subunit 1B
TP53BP2 (BC058918)	chr1:222034413-222100255	tumor protein p53 binding protein, 2
ENC1 (NM 003633)	chr5:73958991-73972273	ectodermal-neural cortex (with BTB-like domain)
PPM1D (BC042418)	chr17:56032412-56096616	protein phosphatase 1D magnesium-dependent, del
TP53BP2 (NM 001031685)	chr1:222034413-222100255	tumor protein p53 binding protein, 2 isoform 1
GNL3 (NM 206825)	chr3:52694976-52703548	guanine nucleotide binding protein-like 3
ING5 (NM 032329)	chr2:242290129-242317563	inhibitor of growth family, member 5
PDRG1 (NM 030815)	chr20:29996420-30003544	p53 and DNA damage-regulated protein
ING4 (NM 016162)	chr12:6629707-6642565	inhibitor of growth family, member 4 isoform 1

# Searching for a Gene



# Searching for a Gene

