

The QA Release Process

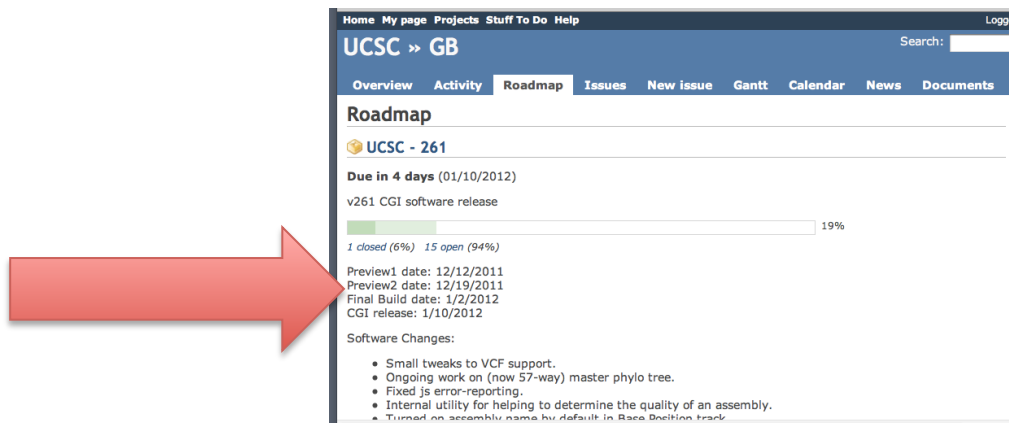
what gets pushed, and when

Overview

- This is not about how we *should* release things
- This is about our current release processes:
 - CGIs
 - Tracks, trackDb, and how to update existing tracks
 - Static docs
 - Table descriptions
- Slides include several links and will be up on genomewiki

Most changes

- Most code changes to the kent source tree go out with the CGI release every 3 weeks
- The release schedule has an iCalendar:
http://www.google.com/calendar/ical/soe.ucsc.edu_r3m0u3g5o4glfer53i9vbe596s%40group.calendar.google.com/public/basic.ics
- You can also find the scheduled release dates in Redmine: click on Roadmap



The screenshot shows the Redmine interface for project UCSC - 261. The 'Roadmap' tab is selected in the navigation menu. The main content area displays the project name 'UCSC - 261', a progress bar for 'v261 CGI software release' at 19% completion, and a list of dates: Preview1 date: 12/12/2011, Preview2 date: 12/19/2011, Final Build date: 1/2/2012, and CGI release: 1/10/2012. Below this, there is a section for 'Software Changes' with a bulleted list of updates.

Home My page Projects Stuff To Do Help Logged

UCSC >> GB Search:

Overview Activity Roadmap Issues New issue Gantt Calendar News Documents

Roadmap

UCSC - 261

Due in 4 days (01/10/2012)

v261 CGI software release

19%

1 closed (6%) 15 open (94%)

Preview1 date: 12/12/2011
Preview2 date: 12/19/2011
Final Build date: 1/2/2012
CGI release: 1/10/2012

Software Changes:

- Small tweaks to VCF support.
- Ongoing work on (now 57-way) master phylo tree.
- Fixed js error-reporting.
- Internal utility for helping to determine the quality of an assembly.
- Turned on assembly name by default in Bacc. Definition track.

New features and big code changes

- The best time to commit changes:
 - Self-contained new module: the week of the “final build” (so that it gets the most visibility on hgwdev)
 - Changes to many places in the code: immediately after a successful CGI release (to keep from interfering with build patches)
- If changes need more testing time, use a demo sandbox:
http://genomewiki.ucsc.edu/index.php/Demo_sandbox
- Please be available on push day if you have changes going out



CGIs: what is tested?

- **hgTracks** and **hgTables** are the only CGIs that are tested every release cycle
- Other CGIs tested only if they have changed
- If you change something, let us know!
 - add the appropriate QA person as a watcher to the Redmine ticket (the sooner the better):
http://genomewiki.ucsc.edu/genecats/index.php/CGI_testing_responsibilities
 - fill out the CGI field in Redmine
 - **add a target version in Redmine** (this makes it show up on the Roadmap page, where QA sees it)

CGI

Target version

CGIs: what is pushed?

- We ask for a specific subset of CGIs and files to be pushed:
http://genomewiki.ucsc.edu/genecats/index.php/CGI_Build_Process#Push_to_hgw0_only
- If you add a new file or CGI, **TELL US!**
- Many files, such as images, are only pushed once – alert the CGI push shepherd to get them pushed



Build Patches

- QAers test CGIs on hgwbeta during the week of the final build
- To get a fix into the release candidate build, fill out a **Build Patch** ticket

The screenshot shows the 'New issue' form in Redmine. The 'Tracker' is set to 'Build Patch'. The 'Subject' field is empty. The 'Description' field is a large text area with a rich text editor toolbar. Below the description, there are several dropdown menus: 'Status' (set to 'New'), 'Priority' (set to 'Medium'), 'Assignee', 'Target version', 'Developer', and 'Commit ID'. There are also input fields for 'Parent task', 'Start date' (2012-01-08), 'Due date', 'Estimated time' (Hours), and '% Done' (0%). A 'Suggested Code Reviewer' dropdown is at the bottom, currently showing '--- Please select ---'. A 'Private' checkbox is also visible.

The screenshot shows the 'Build Patch Process' wiki page. The page title is 'Build Patch Process'. Below the title, there is a list of steps:

1. Someone finds a bug/problem in the release candidate on hgwbeta
2. A Developer fixes it and completes a "Build Patch" issue in redmine (see below for details)
3. Another Developer reviews the code change
4. A QAer tests the fix and determines if it will be patched or not

Below the list, there is a section titled 'How to Create a Build Patch Issue (Developer)'. It states: 'After the problem has been fixed on hgwdev, the Developer completes the initial issue report like so:'

1. Description: include a brief description of what the problem is and why it needs to be solved with a patch
2. Status: New
3. Priority: Urgent
4. Assignee: leave blank
5. Target Version: choose the current release number from the list
6. Developer: choose, from the list, the person who caused this bug to happen
7. Commit ID: the git commit ID from your fix on hgwdev
8. Files Changed: a list of the source code files that were changed with your commit (include full path)
9. CGIs to retest: a list of all of the CGIs that could potentially be affected by your commit
10. Test Case: a detailed, step-by-step list of how to reproduce the failure **and** the fix
11. Suggested Code Reviewer: nominate someone to review your code (most often, this will be the same person who did the initial code review, as they should already be familiar with this code)
12. Watchers: **always add this watcher: "QA Team"** (all of QA will receive email notification), as well as the build-meister

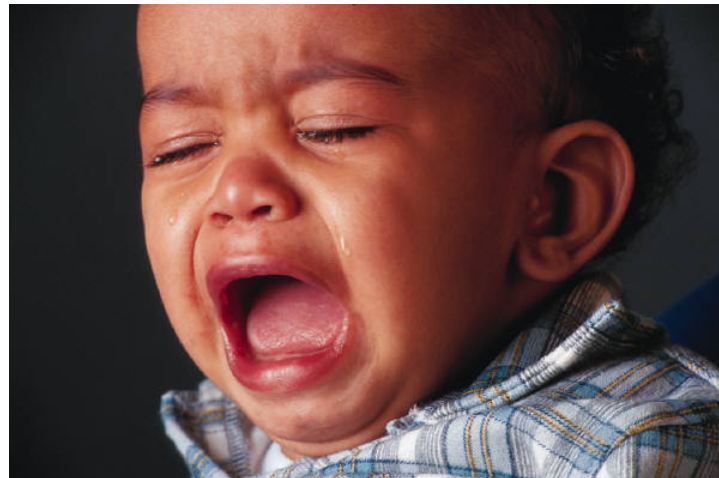
Below the list, there is a section titled 'What to do if you have been asked to review a code change'.

1. You will find your name in the "Suggested Code Reviewer" field of the Build Patch issue
2. Review the change listed in the "Files Changed" section of the issue (see also "Commit ID")
3. If the code is fine, simply choose **Approved** from the "Code Review Status" drop-down list
4. If you find a problem in the code, update the Build Patch issue with the problem and choose **Feedback** from the "Code Review Status"

Some things don't follow the 3 week
release cycle . . .

Changes here can go out at ANY time:

- trackDb and friends:
kent/src/hg/makeDb/trackDb/*
- Static docs:
kent/src/hg/htdocs/*, htdocsExtras/*, hgdownload/*
- Table descriptions:
mostly in kent/src/hg/lib/*.as



Tracks

- Track pushes are mostly decoupled from the CGI pushes, but if your track depends on a specific CGI version, note it in Redmine
- Make Redmine tickets and push queue entries as complete as possible:
 - give background info
 - suggest tracks to compare to
 - mention use of release tags
 - include ALL tables and files (extFile and seq tables are often forgotten, for example)
- Instructions:
[http://genomewiki.ucsc.edu/genecats/index.php/Dev to QA Handoff](http://genomewiki.ucsc.edu/genecats/index.php/Dev_to_QA_Handoff)
- Redmine will replace push queue in the future

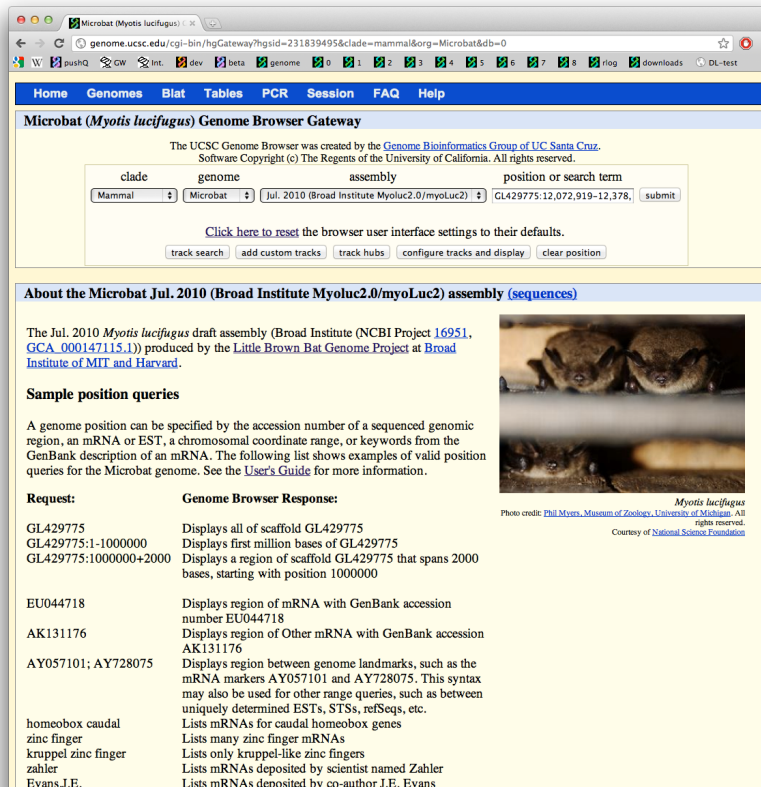
trackDb pushes: what

- It's actually "trackDb and friends":
http://genomewiki.ucsc.edu/genecats/index.php/Pushing_trackDb
- The changes in kent/src/hg/makeDb/trackDb/* get to the RR via 3 tables:
 - trackDb
 - hgFindSpec
 - metaDb
 - and 3 files: trackDb.ix, trackDb.ixx, cv.ra
- All except trackDb and hgFindSpec tables support ENCODE and Track Search

A trackDb exception: description.html

- Though description.html files reside in kent/src/hg/makeDb/trackDb/, they never get pushed after an initial release unless specifically requested

Example: trackDb/microbat/myoLuc2/description.html



The screenshot shows a web browser window displaying the Microbat Genome Browser Gateway. The page title is "Microbat (*Myotis lucifugus*) Genome Browser Gateway". The main content area shows search results for the query "GL429775:12,072,919-12,378,". The results are displayed in a table with columns for "Request:" and "Genome Browser Response:". The response for the query is "Displays a region of scaffold GL429775 that spans 2000 bases, starting with position 1000000".

Request:	Genome Browser Response:
GL429775	Displays all of scaffold GL429775
GL429775:1-1000000	Displays first million bases of GL429775
GL429775:1000000+2000	Displays a region of scaffold GL429775 that spans 2000 bases, starting with position 1000000
EU044718	Displays region of mRNA with GenBank accession number EU044718
AK131176	Displays region of Other mRNA with GenBank accession AK131176
AY057101; AY728075	Displays region between genome landmarks, such as the mRNA markers AY057101 and AY728075. This syntax may also be used for other range queries, such as between uniquely determined ESTs, STSs, refSeqs, etc.
homeobox caudal	Lists mRNAs for caudal homeobox genes
zinc finger	Lists many zinc finger mRNAs
Kruppel zinc finger	Lists only Kruppel-like zinc fingers
Zahler	Lists mRNAs deposited by scientist named Zahler
Evans.JE.	Lists mRNAs deposited by co-author J.E. Evans

(These get pushed from /gbdb on hgwbdev to /gbdb on hgnfs1, which serves hgwbeta and the RR.)

trackDb pushes: when

- Usually pushed when a track is released or settings adjusted
- hg19, hg18, and mm9 go out all the time
- Others may not go out unless requested:
 - Email browser-qa if you need something minor pushed (e.g., you fixed a typo or link)
 - Make a Redmine ticket and email browser-qa if it needs more review
- We also periodically check for changes and push them

Updating existing tracks



Updating existing tracks: trackDb

- If your changes to trackDb.ra files get pushed to the RR but the matching tables are not there, no problem
- If you are changing an existing track, you probably need to use trackDb “release tags”:
<http://genomewiki.ucsc.edu/index.php/ThreeStateTrackDb>
(also documented in kent/src/hg/makeDb/trackDb/README)

Updating existing tracks: trackDb

- A release tag determines whether a version of a track will be included in the trackDb table on a particular machine during a make (from genomewiki page):
 - **make alpha** run on hgwdev loads all tracks with **release alpha** into trackDb and hgFindSpec regardless of the table existing
 - **make beta** run on hgwbeta loads all tracks with **release beta** into trackDb and hgFindSpec only if the table exists
 - **make public** run on hgwbeta loads all tracks with **release public** into trackDb_public and hgFindSpec_public only if the table exists
 - **Having no release tag is equivalent to having all 3 release tags present.**
- (trackDb_public is pushed to the RR and renamed trackDb)

Release tag example

Start:

```
track someRnas  
shortLabel Mediocre RNAs  
visibility hide
```

Developer adds:

```
track someRnas  
release alpha  
shortLabel Great RNAs  
visibility pack  
html someRnasNew.html
```

```
track someRnas  
release beta,public  
shortLabel Mediocre RNAs  
visibility hide
```

QA eventually changes to:

```
track someRnas  
shortLabel Great RNAs  
visibility pack
```

Updating existing tracks: CGI changes

- **Never, never, never** make code changes such that your track will only work with a new version of tables
- . . . But if you do, **coordinate with QA** to get tables pushed at the same time as a CGI release
- Allow ample time for QA!



Static docs: what

- Static docs are mostly in `kent/src/hg/htdocs/*`, but there are two other source trees: `htdocsExtras` and `hgdownload`
- They all wind up in `/usr/local/apache/htdocs/*`

Sequence and Annotation Downloads

This page contains links to sequence and annotation data downloads for the genome assembly Genome Browser. Table downloads are also available via the Genome Browser [FTP server](#) most recent assembly of each genome, see the [current genomes](#) directory. This directory may with automated scripts that must always reference the most recent assembly.

To view the current descriptions and formats of the tables in the annotation database, use the button in the Table Browser. The [Description of the annotation database](#) page (no longer descriptions of selected tables in the database.

All tables in the Genome Browser are freely usable for any purpose except as indicated in the download directories. To view and review the README text, see the [credits](#) page. Please acknowledge.

Genome Browser User Guide

Table of Contents:

- [What does the Genome Browser do?](#)
- [Getting started: Genome Browser gateways](#)
- [Fine-tuning the Genome Browser display](#)
- [Annotation track descriptions](#)
- [Using BLAT alignments](#)
- [Getting started on the Table Browser](#)
- [Getting started using Sessions](#)
- [Getting started using Sessions](#)
- [Using t](#)
- [DNA t](#)
- [Conver](#)
- [Downl](#)
- [Creati](#)


Neanderthal Genome Analysis Consortium Tracks at UCSC

Neanderthals are the closest extinct relatives of humans. They lived from several hundred thousand years ago until their disappearance approximately 30,000 years ago. The Neanderthal genome sequence (published by Green *et al.* in *Science* May 2010) consists of short sequence fragments, usually about 50 base pairs long, mapped to the human reference genome. The DNA was extracted largely from three Neanderthal bones, each about 40,000 years old, from the Vindija Cave in Croatia: V133.16, V133.25, and V133.26. The bulk sequencing was carried out on the Illumina GAI platform. Neanderthal DNA was identified from among the background of microbial sequences in the bone by similarity to the human or chimpanzee genomes.

This portal provides access to the sequence data and alignments to the reference human genome (NCBI Build 36/hg18, GRCh37/hg19) as well as the reference chimpanzee genome (CGSC 2.1/pauTro2) and several associated analyses (see [Downloads](#) and [References](#)).

The following annotations are available on the human hg18 and hg19 genome assemblies, except as noted:

- [Neanderthal Alleles in Human](#)
Displays Neanderthal alleles in orangutan as the outgroup
- [Selective Sweep Scan \(SS\)](#) of positive selection in hum the human reference genom
- [Selective Sweep Scan \(SS\): 5](#) derived alleles (regions from in early humans.
- [SNPs Used for Selective S](#)
- [Candidate Regions for Gen](#) there is considerably more h evidence for gene flow from
- [Neanderthal Sequence Conti](#) passed mapping and base q
- [Neanderthal Sequence Reads](#)
- [Alignments of Sequence R](#) These data were used in the
- [Neanderthal Mitochondrial S](#) *et al.*, 2008) to a modern hu



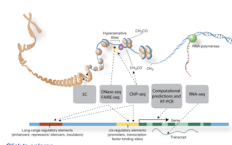
Encyclopedia of DNA Elements

About ENCODE Data

The [Encyclopedia of DNA Elements \(ENCODE\)](#) Consortium is an international col groups funded by the National Human Genome Research Institute (NHGRI). The comprehensive parts list of functional elements in the human genome, including protein and RNA levels, and regulatory elements that control cells and circumstan active.

ENCODE data are now available genome. **All ENCODE data are immediate use via :**

- [Search for displayable trac](#)
- [Download of data files](#)
- [Visualization in the UCSC \(ENCODE data marked w](#)
- [Data mining with the UCSC UCSC Genome Bioinform](#)



About the UCSC Genome Bioinformatics Site

Welcome to the UCSC Genome Browser website. This site contains the reference sequence and working draft assemblies for a large collection of genomes. It also provides portals to the [ENCODE](#) and [Neanderthal](#) projects.

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. [VizGene](#) lets you browse through a large collection of *in situ* mouse and frog images to examine expression patterns. [Genome Graphs](#) allows you to upload and display genome-wide data sets.

The UCSC Genome Browser is developed and maintained by the Genome Bioinformatics Group, a cross-departmental team within the Center for Biomolecular Science and Engineering (CBSE) at the University of California Santa Cruz (UCSC). If you have feedback or questions concerning the tools or data on this website, feel free to contact us on our [public mailing list](#).

News

3 January 2012 - Roadmap Epigenomics Now Available through Data Hub at Washington University

We are pleased to announce the release of the Roadmap Epigenomics data on the UCSC Genome Browser through our Data Hub function. The Roadmap Epigenomics Project is part of the The NIH's [Common Fund's Epigenomics Program](#). It was launched with the goal of producing a public resource of human epigenomic data to catalyze basic biology and disease-oriented research. The Consortium leverages experimental pipelines built around next-generation sequencing technologies to map DNA methylation, histone modifications, chromatin accessibility and small RNA transcripts in stem cells and primary ex vivo tissues selected to represent the normal counterparts of tissues and organ systems frequently involved in human disease. The Consortium expects to deliver a collection of normal epigenomes that will provide a framework or reference for comparison and integration within a broad array of future studies.

All data were produced and processed by the [Roadmap Epigenomics Mapping Consortium](#), and will be periodically updated. Genome Browser tracks were constructed and hosted by [YizHub](#) at Washington University in St. Louis. Tracks are available at UCSC Genome Browser via the Data Hub function, or follow this [link](#). The Roadmap Epigenomics Mapping Consortium is responsible for the quality of the data.

What does the Genome Browser do?

As vertebrate genome sequences near completion and research re-focuses on their analysis, the issue becomes critical: it is not helpful to have 3 billion letters of genomic DNA shown as plain text! As Browser provides a rapid and reliable display of any requested portion of genomes at any scale, to annotation tracks (Known genes, predicted genes, ESTs, mRNAs, CpG islands, assembly gaps and mouse homologues, and more). Half of the annotation tracks are computed at UCSC from publicly remaining tracks are provided by collaborators worldwide. Users can also add their own custom track or research purposes.

VERTEBRATES - Complete s

- Human
- Cat
- Chicken
- Chimpanzee
- Cow
- Dog
- Elephant
- Fugu
- Gibbon
- Gorilla
- Guinea pig

VERTEBRATES - Sequence d

- Armadillo
- Bushbaby
- European hedgehog
- Shrew

DEUTEROSTOMES

- C. intestinalis

Static docs: when

- Static docs can be pushed at any time, but
- Static docs are not ever pushed automatically (so if QA doesn't know about a change, it might not be pushed for a long time)
- It is safe to add changes as comments until QA can look at them

NOTE: this comment style messes up the syntax highlighting in some text editors:

```
<!-- ----- comment goes here ----- -->
```

These are fine:

```
<!-- ===== comment goes here ===== -->
```

```
<!-- *----- comment goes here -----* -->
```

```
<!-- /----- comment goes here -----/ -->
```

```
<!-- _____ comment goes here _____ -->
```

Table Descriptions

Schema for UCSC Genes - UCSC Genes Based on RefSeq, UniProt, GenBank, CCDS and Comparative Genomics

Database: hg19 Primary Table: knownGene Row Count: 77,614

Format description: Genes based on RefSeq, GenBank, and UniProt.

field	example	SQL type	info	description
name	uc001aaa.3	varchar(255)	values	Name of gene
chrom	chr1	varchar(255)	values	Reference sequence chromosome or scaffold
strand	+	char(1)	values	+ or - for strand
txStart	11873	int(10) unsigned	range	Transcription start position
txEnd	14409	int(10) unsigned	range	Transcription end position
cdsStart	11873	int(10) unsigned	range	Coding region start
cdsEnd	11873	int(10) unsigned	range	Coding region end
exonCount	3	int(10) unsigned	range	Number of exons
exonStarts	11873,12612,13220	longblob		Exon start positions
exonEnds	12227,12721,14409	longblob		Exon end positions
proteinID		varchar(40)	values	UniProt display ID for Known Genes, UniProt accession or RefSeq protein ID for UCSC Genes
alignID	uc001aaa.3	varchar(255)	values	Unique identifier for each (known gene, alignment position) pair

these

- tableDescriptions tables for assembly databases are built nightly on hgwdev and mysqlbeta
<http://genomewiki.ucsc.edu/index.php/TableDescriptions>
- The tableDescriptions tables are currently pushed from mysqlbeta to mysqlrr once a week, on Fridays
- If you change an existing .as file, make sure it applies to all of the current tables on the RR

In summary

- The current processes are not perfect, but they are what we have right now
- If you have ever had your changes go out at an unexpected time, **you are not alone**
- QA accepts apology chocolate, cookies, cake, lunch, etc.
- QA also accepts suggestions for improvement ;-)

Thank you!

- Questions?
- Reach all QAers at browser-qa@soe.ucsc.edu



Links from previous slides

- http://www.google.com/calendar/ical/soe.ucsc.edu_r3m0u3g5o4glfer53i9vbe596s%40group.calendar.google.com/public/basic.ics
- http://genomewiki.ucsc.edu/index.php/Demo_sandbox
- http://genomewiki.ucsc.edu/genecats/index.php/CGI_testing_responsibilities
- http://genomewiki.ucsc.edu/genecats/index.php/CGI_Build_Process#Push_to_hgw0_only
- http://redmine.soe.ucsc.edu/projects/genomebrowser/wiki/Build_Patch
- http://genomewiki.ucsc.edu/genecats/index.php/Dev_to_QA_Handoff
- http://genomewiki.ucsc.edu/genecats/index.php/Pushing_trackDb
- <http://genomewiki.ucsc.edu/index.php/ThreeStateTrackDb>
- <http://genomewiki.ucsc.edu/index.php/TableDescriptions>