Preparing Genome Assembly Build Files for SeqFetch and BLAT

Author: sc

Created: November 12, 2004

Last Modified: December 13, 2006 10:49

1 Purpose of Document

To document the process by which we obtain, format, and store the mouse and human genome assemblies for use by the SeqFetch tool and BLAT.

2 Introduction

Initially the MGS group, namely Ben King then Bob Sinclair, was responsible for obtaining, formatting and storing the mouse and human genome assembly builds for their BLAT servers. As of the 3.1 Assembly Release (TR6265) the SeqFetch tool (used by the WI to fetch actual assembly sequence using the UCSC BLAT tool nibFrag) will be using the formatted assembly files to fetch assembly coordinate sequences.

Bob Sinclair and Sharon agreed on standards for where and when to obtain and store the assembly FASTA files, which species to obtain, how to create and where to store the 'nib' files needed by SeqFetch and BLAT so that all products may have a consistent interface to the nib files. This document is based on the information in TR6251 'Processing Assembly sequence for use by SeqFetch and BLAT'

3 Where to get the current Assembly Build FASTA files

We are currently interested in the mouse and human assemblies. The latter for BLAT only. The NCBI creates the assembly, but the assembly chromosome files are not in the format expected by the seqfetch tool and BLAT; one sequence per chromosome file.

Both UCSC (University of CA, Santa Cruz) and Ensembl provide the single sequence chromosome files needed by these programs. We are currently getting the assembly files from Ensembl because they have a consistent ftp path to the most current version of the assembly. The mirror_ftp product ftp.ensembl.org pacakage file has a commented out package for ad hoc mirroring. As of Build 33 Ensembl does not yet have a Mitochondrial chromosome; we are using the UCSC Mitochondrial file.

Note that upon release of a new assembly version by the NCBI, one or the other (Ensembl or UCSC) may make their single chromosome files available first. For this reason we document UCSC ftp paths here.

3.1 Ensembl

- mouse ftp address

  ftp://ftp.ensembl.org/pub/current_mouse/data/fasta/dna/
- human ftp address

  ftp://ftp.ensembl.org/pub/current_human/data/fasta/dna/
- mouse or human filename expression - this gets the known chromosome files; no random files

  dna.chromosome..fa.gz

3.2 UCSC

- mouse ftp address

  ftp://hgdownload.cse.ucsc.edu/goldenPath/mm5/chromosomes/
- human ftp address

  ftp://hgdownload.cse.ucsc.edu/goldenPath/hg17/chromosomes/
- mouse or human filename expression - this gets the known chromosome files; no random files

  chr*.fa.gz

4 When to get the current Assembly Build FASTA files

The mirror_ftp product is set up to mirror the current assembly from Ensembl. One only needs to comment out the human and mouse DNA packages in the package file ftp.ensembl.org. When to do this will be driven by the MGS group and is dependent on when they are ready to analyze a new Assembly build.

5 Where to store the Assembly Build FASTA files

Before download to the following directories mv the old data in the directory to a directory named for the build number

- mouse fasta files

  hobbiton:/data/downloads/ftp.ensembl.org/dna/mouse_build_current
- human fasta files

  hobbiton:/data/downloads/ftp.ensembl.org/dna/human_build_current

Example:

Before we download mouse build 34 we want to archive the current data (build 33):

1. mkdir /data/downloads/ftp.ensembl.org/dna/mouse_build_33
2. mv /data/downloads/ftp.ensembl.org/dna/mouse_build_current/* /data/downloads/ftp.ensembl.org/dna/mouse_build_33

6 How to create Assembly NIB files from the Assembly FASTA files

Run assemblyToNib product /bin/assemblyToNib.sh. This script is a wrapper over the UCSC BLAT tool faToNib and is designed to handle both UCSC and Ensembl filenames for both human and rat, but must be configured to do so. See Configuration file and edit as necessary.

Note that we use the nib file naming convention, chr#.nib, because it is required by the UCSC Genome Browser which the MGS group uses extensively and is also displayed on the BLAT output and should be succinct and meaningful.

What the script does:

1. Sources Configuration
2. Creates a log files
3. Removes all the files from the output directories - ${FA_OUTPUTDIR} and ${NIB_OUTPUTDIR}
4. Unzip the files in ${INPUTDIR} determined by ${FILENAME_PATTERN} to ${FA_OUTPUTDIR}
5. rename the FASTA files in ${FA_OUTPUTDIR} to follow the pattern chr#.fa
6. run faToNib on the files in ${FA_OUTPUTDIR} sending output to ${NIB_OUTPUTDIR}

Example:

- gunzip-c /data/downloads/ftp.ensembl.org/dna/human_build_current/Homo_sapiens.NCBI35.nov.dna.chromosome.1.fa.gz > /data/research/dna/build_fa/Homo_sapiens.NCBI35.nov.dna.chromosome.1.fa
- rename /data/research/dna/build_fa/ Homo_sapiens.NCBI35.nov.dna.chromosome.1.fa to /data/research/dna/build_fa/ chr1.fa
- faToNib /data/research/dna/build_fa/chr.1.fa /data/research/dna/build_nib/chr1.nib

Next:

Upon release (or development testing) manually create directories for that each organism and build number for example:

- Run assemblyToNib configured for mouse
- mkdir /data/research/dna/mouse_build_37_fa

- mv /data/research/dna/build_fa/* /data/research/dna/mouse_build_37_fa - this is the directory for which BLAT and seqfetch should be configured.

7 Notes:

1. As of mouse Build 33, Ensembl does not yet have a Mitochondrial chromosome file. We are currently using the ChrM.fa file fetched from UCSC
2. The human Ensembl Mitochondrial chromsome FASTA is named *MT* and is not handled by assemblyToNib.sh properly. The assembly group does not currently need the human mitochondrial nib file.
3. Neither the MGS group (BLAT) or the seqfetch tool currently need nib files for Unknown or Random chromsome sequence.