GenomicFeatures and *BSgenome*

Patrick Aboyoun

Fred Hutchinson Cancer Research Center

7-9 June, 2010

Outline

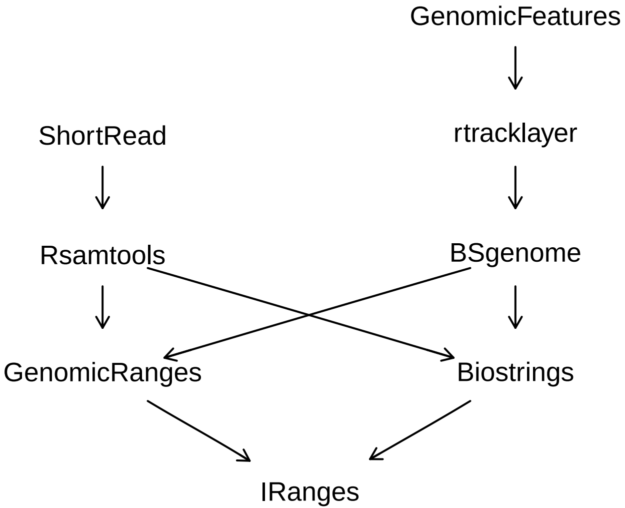
Introduction

GenomicFeatures

BSgenome

Resources

Bioconductor Sequence Packages



Bioconductor Sequence Annotation Packages

GenomicFeatures

- Management of transcript information using *GenomicRanges* infrastructure
- Transcripts stored in separate SQLite databases

BSgenome

- Management of whole genomes using *Biostrings* infrastructure
- Tools for operating on those genomes
- Genomes stored in separate *BSgenome.Organism.Provider.BuildVersion* packages
- Support for pre-build SNP packages for human

Outline

Introduction

GenomicFeatures

BSgenome

Resources

GenomicFeatures transcript sources

Constructors

`makeTranscriptDbFromBiomart`, `makeTranscriptDbFromUCSC`

```
> library(GenomicFeatures)
> nrow(supportedUCSCtables())
```

```
[1] 24
```

```
> head(supportedUCSCtables(), 10)
```

```
                                   track           subtrack
knownGene                     UCSC Genes               <NA>
knownGeneOld3             Old UCSC Genes               <NA>
wgEncodeGencodeManualRel2  Gencode Genes    Genecode Manual
wgEncodeGencodeAutoRel2    Gencode Genes      Genecode Auto
wgEncodeGencodePolyaRel2   Gencode Genes     Genecode PolyA
ccdsGene                   Consensus CDS               <NA>
refGene                     RefSeq Genes               <NA>
xenoRefGene                 Other RefSeq               <NA>
vegaGene                      Vega Genes Vega Protein Genes
vegaPseudoGene                Vega Genes   Vega Pseudogenes
```

TranscriptDb basics

Making a *TranscriptDb* object

```
> mm9KG <-
+   makeTranscriptDbFromUCSC(genome = "mm9",
+                            tablename = "knownGene")
```

Saving and Loading

```
> saveFeatures(mm9KG, file="mm9KG.sqlite")
```

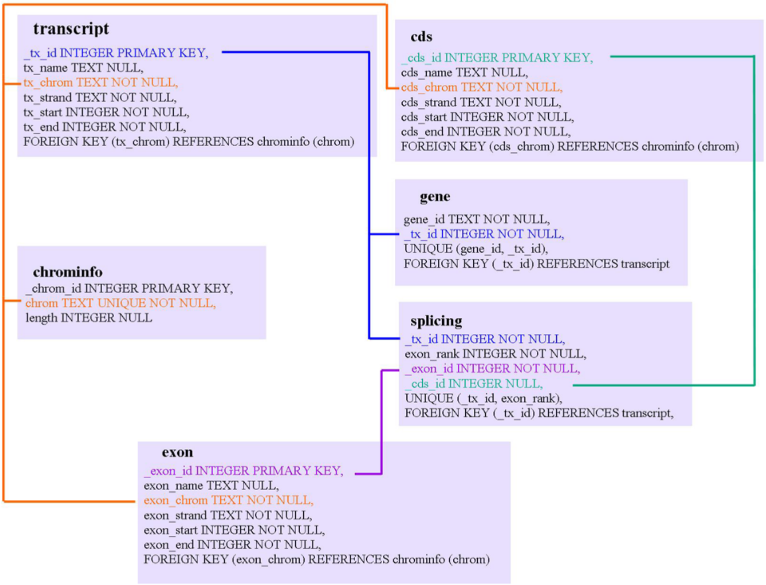
```
> mm9KGChr9 <-
+   loadFeatures(system.file("extdata", "mm9KGChr9.sqlite",
+                            package = "EMBL2010"))
```

TranscriptDb class

```
> mm9KGChr9

TranscriptDb object:
| Db type: TranscriptDb
| Data source: UCSC
| Genome: mm9
| UCSC Table: knownGene
| Type of Gene ID: Entrez Gene ID
| Full dataset: yes
| transcript_nrow: 49409
| exon_nrow: 237551
| cds_nrow: 204831
| Db created by: GenomicFeatures package from Bioconductor
| Creation time: 2010-05-13 17:02:30 -0700 (Thu, 13 May 2010)
| GenomicFeatures version at creation time: 1.1.1
| RSQLite version at creation time: 0.8-3
```

TranscriptDb schema



```
transcript
_tx_id INTEGER PRIMARY KEY,
tx_name TEXT NULL,
tx_chrom TEXT NOT NULL,
tx_strand TEXT NOT NULL,
tx_start INTEGER NOT NULL,
tx_end INTEGER NOT NULL,
FOREIGN KEY (tx_chrom) REFERENCES chrominfo (chrom)
```

```
cds
_cds_id INTEGER PRIMARY KEY,
cds_name TEXT NULL,
cds_chrom TEXT NOT NULL,
cds_strand TEXT NOT NULL,
cds_start INTEGER NOT NULL,
cds_end INTEGER NOT NULL,
FOREIGN KEY (cds_chrom) REFERENCES chrominfo (chrom)
```

```
gene
gene_id TEXT NOT NULL,
_tx_id INTEGER NOT NULL,
UNIQUE (gene_id, _tx_id),
FOREIGN KEY (_tx_id) REFERENCES transcript
```

```
chrominfo
_chrom_id INTEGER PRIMARY KEY,
chrom TEXT UNIQUE NOT NULL,
length INTEGER NULL
```

```
splicing
_tx_id INTEGER NOT NULL,
exon_rank INTEGER NOT NULL,
_exon_id INTEGER NOT NULL,
_cds_id INTEGER NULL,
UNIQUE (_tx_id, exon_rank),
FOREIGN KEY (_tx_id) REFERENCES transcript,
```

```
exon
_exon_id INTEGER PRIMARY KEY,
exon_name TEXT NULL,
exon_chrom TEXT NOT NULL,
exon_strand TEXT NOT NULL,
exon_start INTEGER NOT NULL,
exon_end INTEGER NOT NULL,
FOREIGN KEY (exon_chrom) REFERENCES chrominfo (chrom)
```

Ungrouped transcript-related information

Extractors

`transcripts`, `exons`, `cds`

```
> tx <- transcripts(mm9KGChr9)
> length(tx)
```

```
[1] 2910
```

```
> head(tx, 5)
```

```
GRanges with 5 ranges and 2 elementMetadata values
    seqnames               ranges strand |     tx_id     tx_name
       <Rle>            <IRanges>  <Rle> | <integer> <character>
[1]     chr9 [3215314, 3215339]       + |     24312  uc009oas.1
[2]     chr9 [3335231, 3385846]       + |     24315  uc009oat.1
[3]     chr9 [3335473, 3343608]       + |     24313  uc009oau.1
[4]     chr9 [3335473, 3380423]       + |     24314  uc009oav.1
[5]     chr9 [3335478, 3385846]       + |     24316  uc009oaw.1

seqlengths
         chr1          chr2 ... chrX_random chrY_random
    197195432     181748087 ...     1785075    58682461
```

Grouped transcript-related information

Extractors

transcriptsBy, exonsBy, cdsBy, intronsByTranscript, fiveUTRsByTranscript, threeUTRsByTranscript

```
> txExons <- exonsBy(mm9KGChr9)
> txIntrons <- intronsByTranscript(mm9KGChr9)
> txExons[6]
```

```
GRangesList of length 1
$24313
GRanges with 3 ranges and 3 elementMetadata values
    seqnames             ranges strand |   exon_id   exon_name
       <Rle>          <IRanges>  <Rle> | <integer> <character>
[1]     chr9 [3335473, 3335594]      + |    117005          NA
[2]     chr9 [3338456, 3338591]      + |    117006          NA
[3]     chr9 [3343015, 3343608]      + |    117007          NA
    exon_rank
    <integer>
[1]         1
[2]         2
[3]         3
```

Overlapping with transcripts

Methods

`findOverlaps`, `countOverlaps`, `match`, `%in%`, `subsetByOverlaps`

Usage

```
> findOverlaps(query, subject, maxgap = 0L, minoverlap = 1L,
+              type = c("any", "start", "end"),
+              select = c("all", "first"))
> help("findOverlaps,GRanges,GRangesList-method")
```

```
> grngs <- GRanges("chr9", gaps(ranges(txExons[[6]])), "+")
> countOverlaps(grngs, tx)
```

```
[1] 4 4
```

```
> rbind(countOverlaps(grngs, txExons), countOverlaps(grngs, txIntrons))
```

```
     [,1] [,2]
[1,]    1    0
[2,]    4    4
```

Outline

Introduction

GenomicFeatures

BSgenome

Resources

BSgenome packages

```
> library(BSgenome)
> available.genomes()
 [1] "BSgenome.Amellifera.BeeBase.assembly4"
 [2] "BSgenome.Amellifera.UCSC.apiMel2"
 [3] "BSgenome.Athaliana.TAIR.01222004"
 [4] "BSgenome.Athaliana.TAIR.04232008"
 [5] "BSgenome.Btaurus.UCSC.bosTau3"
 [6] "BSgenome.Btaurus.UCSC.bosTau4"
 [7] "BSgenome.Celegans.UCSC.ce2"
 [8] "BSgenome.Cfamiliaris.UCSC.canFam2"
 [9] "BSgenome.Dmelanogaster.UCSC.dm2"
[10] "BSgenome.Dmelanogaster.UCSC.dm3"
[11] "BSgenome.Drerio.UCSC.danRer5"
[12] "BSgenome.Ecoli.NCBI.20080805"
[13] "BSgenome.Ggallus.UCSC.galGal3"
[14] "BSgenome.Hsapiens.UCSC.hg17"
[15] "BSgenome.Hsapiens.UCSC.hg18"
[16] "BSgenome.Hsapiens.UCSC.hg19"
[17] "BSgenome.Mmusculus.UCSC.mm8"
[18] "BSgenome.Mmusculus.UCSC.mm9"
[19] "BSgenome.Ptroglodytes.UCSC.panTro2"
[20] "BSgenome.Rnorvegicus.UCSC.rn4"
[21] "BSgenome.Scerevisiae.UCSC.sacCer1"
[22] "BSgenome.Scerevisiae.UCSC.sacCer2"
```

BSgenome class decomposition

BSgenome slots

```
> getSlots("BSgenome")
```

```
         source_url            seqnames          seqlengths
        "character"         "character"           "integer"
          mseqnames        seqs_pkgname            seqs_dir
        "character"         "character"         "character"
      nmask_per_seq       masks_pkgname           masks_dir
          "integer"         "character"         "character"
 injectSNPs_handler         .seqs_cache        .link_counts
"InjectSNPsHandler"       "environment"       "environment"
           organism             species            provider
        "character"         "character"         "character"
   provider_version        release_date        release_name
        "character"         "character"         "character"
```

Notes

- `.seqs_cache` and `.link_counts` slots manage memory.
- `seqs_dir` and `masks_dir` slots specify storage location.

BSgenome methods

Sequence selection

`[[`, `$`

Subsequence selection

`getSeq`

Accessors

`length`, `names`/`seqnames`, `mseqnames`, `seqlengths`, `masknames`, `sourceUrl`

Matching

`vmatchPattern`, `vcountPattern`, `vmatchPDict`, `vcountPDict`, `matchPWM`, `countPWM`

SNPs (Human only at this point)

`injectSNPs`, `SNPlocs_pkgname`, `SNPcount`, `SNPlocs`

BSgenome package without masks

```
> library(BSgenome.Scerevisiae.UCSC.sacCer2)
> Scerevisiae

Yeast genome
|
| organism: Saccharomyces cerevisiae (Yeast)
| provider: UCSC
| provider version: sacCer2
| release date: June 2008
| release name: SGD June 2008 sequence
|
| sequences (see '?seqnames'):
|   chrI     chrII    chrIII   chrIV    chrV     chrVI
|   chrVII   chrVIII  chrIX    chrX     chrXI    chrXII
|   chrXIII  chrXIV   chrXV    chrXVI   chrM     2micron
|
| (use the '$' or '[[' operator to access a given sequence)

> Scerevisiae$chrI

  230208-letter "DNAString" instance
seq: CCACACCACACCCACACACCCACACACC...GGTGTGGTGTGGGTGTGGTGTGTGTGGG
```

BSgenome package with masks

```
> library(BSgenome.Hsapiens.UCSC.hg19)
> Hsapiens$chr1

  249250621-letter "MaskedDNAString" instance (# for masking)
seq: ############################...############################
masks:
  maskedwidth maskedratio active names
1    23970000  0.09616827   TRUE AGAPS
2           0  0.00000000   TRUE   AMB
3   114014472  0.45742904  FALSE    RM
4     1581889  0.00634658  FALSE   TRF
                                desc
1                      assembly gaps
2   intra-contig ambiguities (empty)
3                       RepeatMasker
4 Tandem Repeats Finder [period<=12]
all masks together:
  maskedwidth maskedratio
    138071094   0.5539448
all active masks together:
  maskedwidth maskedratio
     23970000  0.09616827
```

Sequence information

Operations that don't load sequences

```
> head(seqnames(Scerevisiae), 6)
```

```
[1] "chrI"   "chrII"  "chrIII" "chrIV"  "chrV"   "chrVI"
```

```
> head(seqlengths(Scerevisiae), 8)
```

```
   chrI   chrII  chrIII   chrIV    chrV   chrVI  chrVII chrVIII
 230208  813178  316617 1531919  576869  270148 1090947  562643
```

Operation that does

```
> sapply(head(seqnames(Scerevisiae), 8), function(i)
+        alphabetFrequency(Scerevisiae[[i]], baseOnly=TRUE))
```

```
       chrI  chrII chrIII  chrIV   chrV chrVI chrVII chrVIII
A     69826 249653  98657 476749 176531 82928 338319  174022
C     44646 157410  62359 289343 109828 52201 207776  109098
G     45765 154397  59639 291356 112313 52435 207451  107488
T     69971 251718  95962 474471 178197 82584 337401  172035
other     0      0      0      0      0     0      0       0
```

Matches for single pattern across genome

```
> exclude <- setdiff(seqnames(Hsapiens), c("chr1", "chr2"))
> vcountPattern("ACYTANCAGT", Hsapiens,
+                fixed = c(pattern = FALSE, subject = TRUE),
+                exclude = exclude)

  seqname strand count
1    chr1      +  1546
2    chr1      -  1545
3    chr2      +  1722
4    chr2      -  1684

> patmatch <-
+ vmatchPattern("ACYTANCAGT", Hsapiens,
+                fixed = c(pattern = FALSE, subject = TRUE),
+                exclude = exclude, asRangedData = FALSE)
> head(patmatch, 3)

GRanges with 3 ranges and 0 elementMetadata values
    seqnames                 ranges strand |
       <Rle>              <IRanges>  <Rle> |
[1]     chr1 [ 361581,  361590]      + |
[2]     chr1 [1738000, 1738009]      + |
[3]     chr1 [1814381, 1814390]      + |
```

Pattern dictionary (Microarray probes)

```
> library("hgu95av2probe")
> probes <- DNAStringSet(hgu95av2probe$sequence[1:100])
> head(probes, 10)
```

```
  A DNAStringSet instance of length 10
     width seq
 [1]    25 TGGCTCCTGCTGAGGTCCCCTTTCC
 [2]    25 GGCTGTGAATTCCTGTACATATTTC
 [3]    25 GCTTCAATTCCATTATGTTTTAATG
 [4]    25 GCCGTTTGACAGAGCATGCTCTGCG
 [5]    25 TGACAGAGCATGCTCTGCGTTGTTG
 [6]    25 CTCTGCGTTGTTGGTTTCACCAGCT
 [7]    25 GGTTTCACCAGCTTCTGCCCTCACA
 [8]    25 TTCTGCCCTCACATGCACAGGGATT
 [9]    25 CCTCACATGCACAGGGATTTAACAA
[10]    25 TCCTTGGTACTCTGCCCTCCTGTCA
```

Count matches for multiple patterns across genome

```
> counts <- vcountPDict(probes, Hsapiens, exclude = exclude)
> head(counts, 5)
```

```
DataFrame with 5 rows and 4 columns
  seqname strand     index count
    <Rle>  <Rle> <integer> <Rle>
1     chr1      +         1     0
2     chr1      +         2     0
3     chr1      +         3     0
4     chr1      +         4     0
5     chr1      +         5     0
```

```
> dim(counts)
```

```
[1] 400   4
```

```
> whichMatch <- seqselect(counts$index, counts$count > 0)
> length(whichMatch)
```

```
[1] 15
```

```
> whichMatch
```

```
 [1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14 16
```

```
> matchedProbes <- probes[whichMatch]
```

Find match locations for probes

```
> matchLocs <- matchPDict(PDict(matchedProbes), Hsapiens$chr2)
> extractAllMatches(Hsapiens$chr2, matchLocs)
```

```
  Views on a 243199373-letter DNAString subject
subject: NNNNNNNNNNNNNNNNNNNNNNNNNN...NNNNNNNNNNNNNNNNNNNNNNNNNN
views:
         start       end width
 [1] 113420812 113420836    25 [TGGCTCCTGCTGAGGTCCCCTTTCC]
 [2] 113420842 113420866    25 [GGCTGTGAATTCCTGTACATATTTC]
 [3] 113420884 113420908    25 [GCTTCAATTCCATTATGTTTTAATG]
 [4] 113420962 113420986    25 [GCCGTTTGACAGAGCATGCTCTGCG]
 [5] 113420968 113420992    25 [TGACAGAGCATGCTCTGCGTTGTTG]
 [6] 113420980 113421004    25 [CTCTGCGTTGTTGGTTTCACCAGCT]
 [7] 113420992 113421016    25 [GGTTTCACCAGCTTCTGCCCTCACA]
 [8] 113421004 113421028    25 [TTCTGCCCTCACATGCACAGGGATT]
 [9] 113421010 113421034    25 [CCTCACATGCACAGGGATTTAACAA]
[10] 113421082 113421106    25 [TCCTTGGTACTCTGCCCTCCTGTCA]
[11] 113421094 113421118    25 [TGCCCTCCTGTCAGTAGTGGCAGGA]
[12] 113421118 113421142    25 [ATCTATTGGCATATTCGGGAGCTTC]
[13] 113421130 113421154    25 [ATTCGGGAGCTTCTTAGAGGGATGA]
[14] 113421274 113421298    25 [AAGATTTCTGGCAGTGTGGGATGGA]
[15] 113421340 113421364    25 [CAGCCTTCCATGTTCATTTGTCTAC]
```

SNP packages

```
> available.SNPs()

[1] "SNPlocs.Hsapiens.dbSNP.20071016"
[2] "SNPlocs.Hsapiens.dbSNP.20080617"
[3] "SNPlocs.Hsapiens.dbSNP.20090506"
[4] "SNPlocs.Hsapiens.dbSNP.20100427"

> SNPlocs_pkgname(Hsapiens)

NULL

> HsWithSNPs <-
+   injectSNPs(Hsapiens, "SNPlocs.Hsapiens.dbSNP.20090506")
> class(HsWithSNPs)

[1] "BSgenome"
attr(,"package")
[1] "BSgenome"

> SNPlocs_pkgname(HsWithSNPs)

[1] "SNPlocs.Hsapiens.dbSNP.20090506"
```

SNP exploration

```
> SNPcount(HsWithSNPs)

  chr1   chr2   chr3   chr4   chr5   chr6   chr7   chr8   chr9
920233 933616 789121 798603 706109 760249 655873 612367 496064
 chr10  chr11  chr12  chr13  chr14  chr15  chr16  chr17  chr18
583240 577300 558759 427010 365742 331501 354239 316396 322866
 chr19  chr20  chr21  chr22   chrX   chrY
268235 323041 160580 187392 391414   6539
```

```
> alphabetFrequency(Hsapiens$chr1)

       A        C        G        T        M        R        W
65570891 47024412 47016562 65668756        0        0        0
       S        Y        K        V        H        D        B
       0        0        0        0        0        0        0
       N        -        +
       0        0        0
```

```
> alphabetFrequency(HsWithSNPs$chr1)

       A        C        G        T        M        R        W
65306157 46833464 46825359 65403357    40477   150327    40710
       S        Y        K        V        H        D        B
   31997   150117    41304   102527   125770   126323   102322
       N        -        +
     410        0        0
```

Outline

Introduction

GenomicFeatures

BSgenome

Resources

Resources

Bioconductor Web site

- ‘*GenomicFeatures*’ and ‘*BSgenome*’ links.
- `http://bioconductor.org`
- ‘Installation’, ‘Software’, and ‘Mailing lists’ links.

Help in *R*

- `help.start()` to view a help browser.
- `help(package = "BSgenome")`
- `?transcriptsBy`
- `browseVignettes("GenomicFeatures")`