A new approach to SNP location metadata

VJ Carey

April 21, 2009

1 Introduction

Versions of GGtools prior to 2.3.x have a complicated approach to SNP location metadata, involving a specially constructed SQLite database. In the current version we will use a structure derived from the SNPlocs.Hsapiens.dbSNP.* package.

One class and two methods are supported.

> require(GGtools)
> getClass("snpLocs")

Class "snpLocs" [package "GGBase"]

Slots:

Name: locEnv offsets organism versions
Class: environment numeric character character

> data(hsSnpLocs)
> hsSnpLocs

snpLocs instance, organism Hs based on:

org.Hs.eg.db SNPlocs.Hsapiens.dbSNP.20071016
"2.2.0" "0.99.1"

The chromosome-specific locations are generated reasonably efficiently:

> snpLocs.Hs(chrnum(20), rsid("rs6060535"))

[,1]
rsid 6060535
loc 33698936
2 Construction of serialized reference container

First, unify the name and location information from the SNPlocs package.

```r
> humanSNPlocs = list()
> library(SNPlocs.Hsapiens.dbSNP.20071016)
> if (file.exists("humanSNPlocs.rda")) load("humanSNPlocs.rda") else {
+   for (i in c(as.character(1:22), "X", "Y")) {
+       curc = getSNPlocs(paste("chr", i, sep = ","))
+       rsid.int = as.integer(curc[, 1])
+       loc.int = as.integer(curc[, 3])
+       humanSNPlocs[[i]] = rbind(rsid = rsid.int, loc = loc.int)
+   }
+ }

Now get offsets for computing the chromosome-wide location values.

```
```r
> require(org.Hs.eg.db)
> chrl = org.Hs.egCHRLENGTHS
> offs = c(0, cumsum(as.double(chrl[1:22])))
```

Now we create the environment-based container instance:

```r
> el = new.env()
> getv = function(x) installed.packages()[x, "Version"]
> for (i in names(humanSNPlocs)) assign(i, humanSNPlocs[[i]], el)
> hsSnpLocs = new("snpLocs", locEnv = el, offsets = offs, organism = "Hs",
+                versions = c(org.Hs.eg.db = getv("org.Hs.eg.db"), SNPlocs.Hsapiens.dbSNP.20071016 = getv("SNPlocs.Hsapiens.dbSNP.20071016"))

This object will be saved in GGBase.
```