# restfulSE – experiments with HDF5 server content wrapped in SummarizedExperiment

*Vincent J. Carey, stvjc at channing.harvard.edu, Shweta Gopaulakrishnan, reshg at channing.harvard.edu*

*May 05, 2017*

## Contents



## 1 Introduction

Extensive human and computational effort is expended on downloading and managing large genomic data at site of analysis. Interoperable formats that are accessible via generic operations like those in RESTful APIs may help to improve cost-effectiveness of genome-scale analyses.

In this report we examine the use of HDF5 server as a back end for assay data, mediated through the RangedSummarizedExperiment API for interactive use.

A modest server configured to deliver HDF5 content via a RESTful API has been prepared and is used in this vignette.

## 2 Executive summary

We want to provide rapid access to array-like data. We'll work with the Banovich 450k data as there is a simple check against an in-memory representation.

```
library(restfulSE)
bigec2 = H5S_source("http://54.174.163.77:5000")
## analyzing groups for their links...
## done
bigec2
## HDF5 server domain:  http://54.174.163.77:5000
```

```
##  There are 10 groups.
##  Use groups(), links(), ..., to probe and access metadata.
##  Use dsmeta() to get information on datasets within groups.
##  Use [[ [dsname] ]]  to get a reference suitable for [i, j] subsetting.
dsmeta(bigec2)[1:2,] # two groups
## DataFrame with 2 rows and 3 columns
##    groupnum                       dsnames
##   <integer>                <CharacterList>
## 1         1 tissues,assays,neurons100k,...
## 2         2
##                               grp.uuid
##                              <character>
## 1 c3ca306c-3020-11e7-806d-123feca22a06
## 2 c3df8476-3020-11e7-806d-123feca22a06
dsmeta(bigec2)[1,2][[1]] # all dataset candidates in group 1
## [1] "tissues"     "assays"      "neurons100k" "neurons400k"
```

We use double-bracket subscripting to grab a reference to a dataset from an H5S source.

```
banref = bigec2[["assays"]] # arbitrary name assigned long ago
banref
## H5S_dataset instance:
##   dsname intl.dim1 intl.dim2              created      type.base
## 1 assays        64    329469 2017-04-05T18:02:37Z H5T_IEEE_F64LE
```

We build a RESTfulSummarizedExperiment by combining an assay-free RangedSummarizedExperiment with this reference.

```
data(banoSEMeta)
rbano = RESTfulSummarizedExperiment(banoSEMeta, banref)
rbano
## class: RESTfulSummarizedExperiment
## dim: 329469 64
## metadata(0):
## assays(1): (served by HDF5Server)
## rownames(329469): cg00000029 cg00000165 ... ch.9.98989607R
##   ch.9.991104F
## rowData names(10): addressA addressB ... probeEnd probeTarget
## colnames(64): NA18498 NA18499 ... NA18489 NA18909
## colData names(35): title geo_accession ... data_row_count naid
```

We can update the SummarizedExperiment metadata as we like through subsetting operations, and then extract the relevant assay data. The data are retrieved from the remote server.

```
rbanoSub = rbano[5:8, 3:9] # currently only trivial subsets retrieved
assay(rbanoSub) # general index processing under construction
##                NA18501    NA18502    NA18516    NA18517    NA18519
## cg00000363  0.325433263  1.3778200  0.5966999 -1.0747716 -0.2686108
## cg00000622  0.003436888 -0.6684993 -1.2106348  0.4990709  0.4555032
## cg00000714 -1.184443665 -1.6540480 -0.1747294 -0.7111111  1.5458591
## cg00000734  0.153831565 -1.2992894  1.9039768  0.8735532 -0.1379805
##               NA18520    NA18855
## cg00000363 1.2819188 -0.4633973
## cg00000622 0.8974313 -0.7700943
## cg00000714 0.5384043  1.0082528
## cg00000734 0.5042480  0.1923326
```

# 3 Background

Banovich et al. published a subset of DNA methylation measures assembled on 64 samples of immortalized B-cells from the YRI HapMap cohort.

```
library(restfulSE)
data(banoSEMeta)
banoSEMeta
## class: RangedSummarizedExperiment
## dim: 329469 64
## metadata(0):
## assays(0):
## rownames(329469): cg00000029 cg00000165 ... ch.9.98989607R
##   ch.9.991104F
## rowData names(10): addressA addressB ... probeEnd probeTarget
## colnames(64): NA18498 NA18499 ... NA18489 NA18909
## colData names(35): title geo_accession ... data_row_count naid
```

The numerical data have been exported using H. Pages' saveHDF5SummarizedExperiment applied to the banovichSE SummarizedExperiment in the yriMulti package. The HDF5 component is simply copied into the server data space on the remote server.

# 4 Hierarchy of server resources

## 4.1 Server

Given the URL of a server running HDF5 server, we create an instance of `H5S_source`:

```
mys = new("H5S_source", serverURL="http://54.163.220.201:5000")
mys
## HDF5 server domain:  http://54.163.220.201:5000
##  There are 0 groups.
##  Use groups(), links(), ..., to probe and access metadata.
##  Use dsmeta() to get information on datasets within groups.
##  Use [[ [dsname] ]]  to get a reference suitable for [i, j] subsetting.
```

## 4.2 Groups

The server identifies a collection of 'groups'. For the server we are working with, only one group, at the root, is of interest.

```
groups(mys)
## DataFrame with 10 rows and 2 columns
##                                    groups    nlinks
##                                <character> <integer>
## 1  8a8c8736-2379-11e7-8c24-12f07892ad80         7
## 2  8a8c873b-2379-11e7-8c24-12f07892ad80         1
## 3  8a8c8739-2379-11e7-8c24-12f07892ad80         1
## 4  8a8c873e-2379-11e7-8c24-12f07892ad80         4
## 5  8a8c8738-2379-11e7-8c24-12f07892ad80         1
## 6  8a8c873f-2379-11e7-8c24-12f07892ad80         1
## 7  8a8c873c-2379-11e7-8c24-12f07892ad80        28
## 8  8a8c873d-2379-11e7-8c24-12f07892ad80         1
```

```
## 9  8a8c873a-2379-11e7-8c24-12f07892ad80          3
## 10 8a8c8737-2379-11e7-8c24-12f07892ad80         88
```

### 4.3 Links for a group

There is a class to hold the link set for any group:

```
lin1 = links(mys,1)
lin1
## HDF5 server link set for group 8a8c8736-2379-11e7-8c24-12f07892ad80
##  There are 7 links.
##  Use targets([linkset]) to extract target URLs.
```

The relevant URLs are

```
restfulSE:::targets(lin1)
## [1] "http://54.163.220.201:5000/?host=tissues.hdfgroup.org"
## [2] "http://54.163.220.201:5000/?host=assays.hdfgroup.org"
## [3] "http://54.163.220.201:5000/groups/8a8c873f-2379-11e7-8c24-12f07892ad80"
## [4] "http://54.163.220.201:5000/?host=assay.hdfgroup.org"
## [5] "http://54.163.220.201:5000/?host=1M_neurons_filtered_gene_bc_matrices_h5.hdfgroup.org"
## [6] "http://54.163.220.201:5000/groups/8a8c8737-2379-11e7-8c24-12f07892ad80"
## [7] "http://54.163.220.201:5000/?host=as.hdfgroup.org"
```

### 4.4 Datasets

Some of these URLs do not resolve directly to data. But the first two do. We obtain some relevant metadata:

```
ds1 = datasetRefs(lin1, 1, drop=3:5)
ds1
```

Here the drop parameter refers to 'host' URLs that will not be investigated.

### 4.5 Acquiring numerical data from a dataset

We use the value/select method directly in the HDF5 row-major orientation.

```
bano = ds1[["assays"]]
bano
bano["0:4:1", "0:6:1"]
```

This matrix is transposed relative to the banovichSE SummarizedExperiment.

## 5 OLDER MATERIAL SUPERSEDED BY THE ABOVE

A wrapper class has been defined in the restfulSE package.

```
banoh5  = banoH5() # default uses EC2
banoh5
```

## 6 Construction

```
restBano = RESTfulSummarizedExperiment(banoSEMeta, banoh5)
restBano
```

## 7 Subsetting and assay extraction

Targeted extraction is possible, but index processing needs considerable work.

```
subr = restBano[1:4,1:6]
subr
assay(subr)
```