rhdf5: HDF5 in the Bioconductor ecosystem

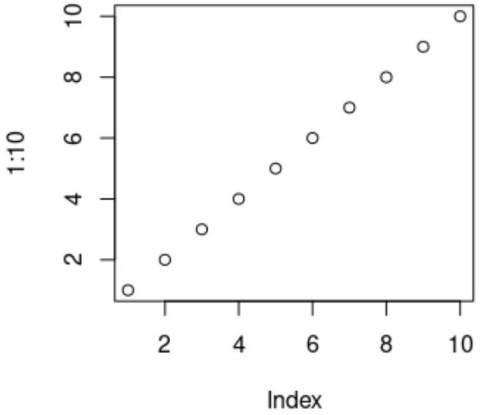
Mike L. Smith
@grimbough

- Statistical programming language & environment
- Great for interactive data exploration & rapid prototyping

```
> plot(1:10)

> mean(1:10)

[1] 5.5
```



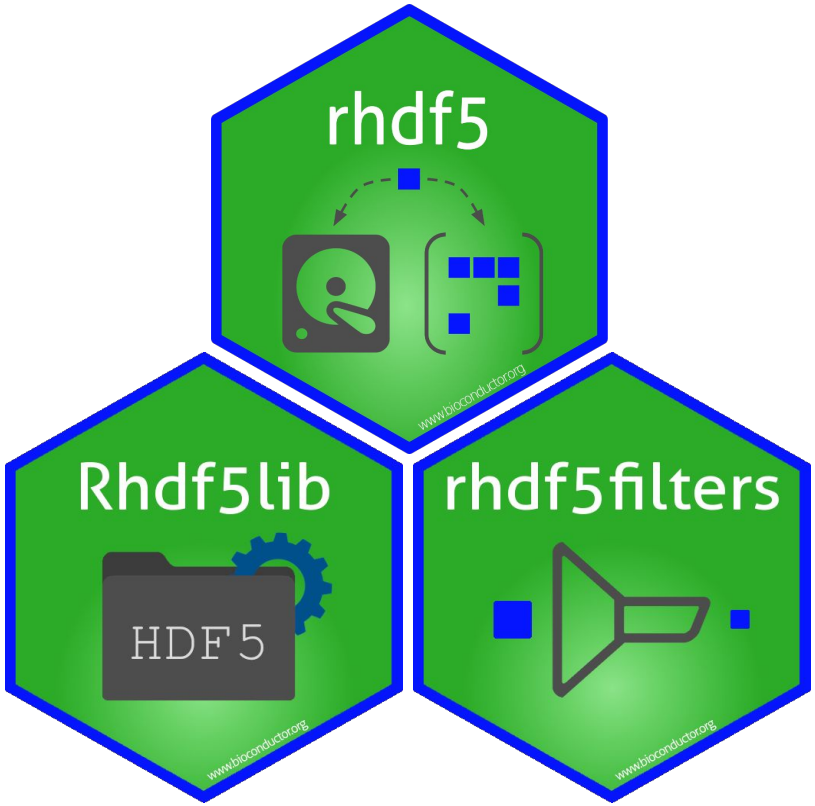
- 10,000s of addon “packages”
  - CRAN, Github, etc
  - Cover a huge range of topics and application areas
  - Easy to install (most of the time)

- Additional R package repository with specific focus on biological research
- Has more / different rules than CRAN!
  - intention is to make better software & improve user experience
  - package review, minimum documentation requirements, daily CI testing, ...
- Strong emphasis on code reuse and modularisation within the ecosystem
  - Core infrastructure implemented once and used by everyone
  - e.g. reading specific file types, classes representing common data types
- HDF5 falls into this category

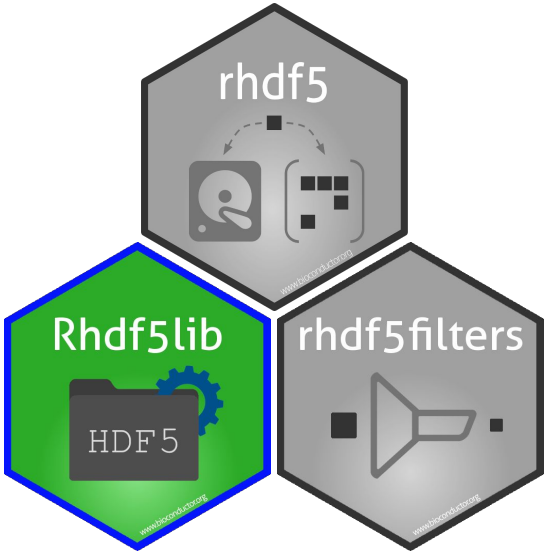
rhdf5 package

Bernd Fischer

Three connected packages

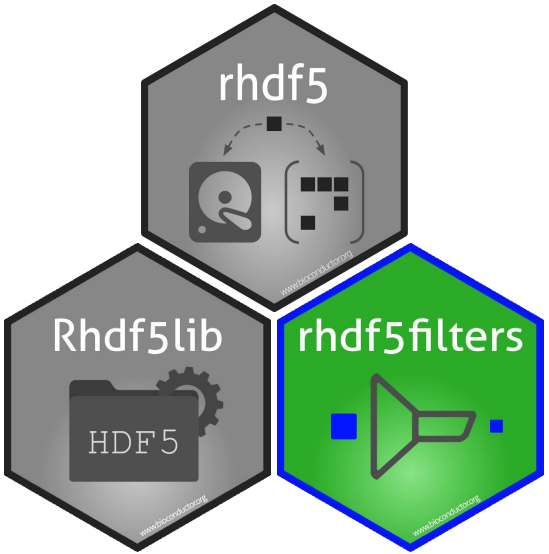


Rhdf5lib



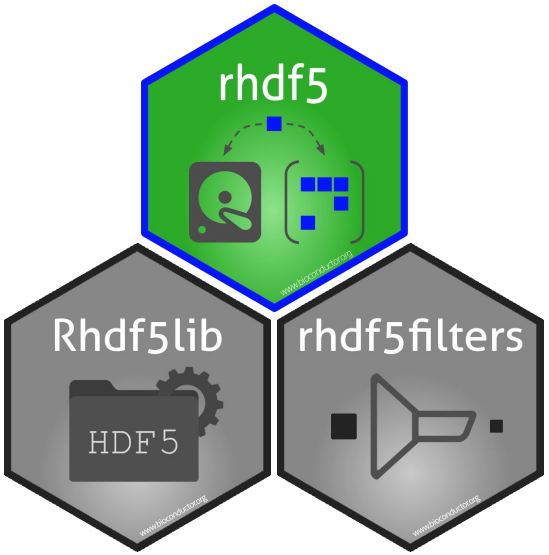
- Distributes static HDF5 library (currently 1.10.6)
- Ensures consistent version for users
- Ensures consistent installation instructions and toolchain
- Compiles on Linux and Mac, pre-compiled for Windows

rhdf5filters



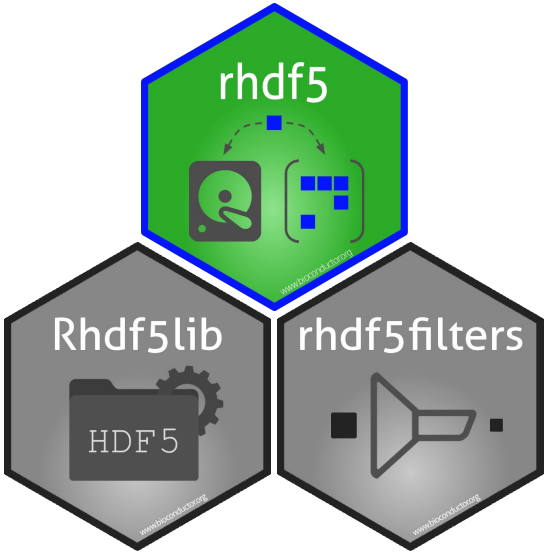
- Distributes several dynamic filters
  - bzip2
  - lzf
  - blosc
    - blosclz, lz4, lz4hc, snappy, zstd, zlib
- Sets `HDF5_PLUGIN_PATH` environment variable in R session
- Can be used by external programs too

rhdf5



- Provides “high” and “low” level interfaces with C-API
- Reasonable coverage at “low level” with `H5X()` functions
  - Mapping to C interface
- “High level” functions for common operations - `h5x()`
  - Wrappers with default choices made

rhdf5 - C-API mapping

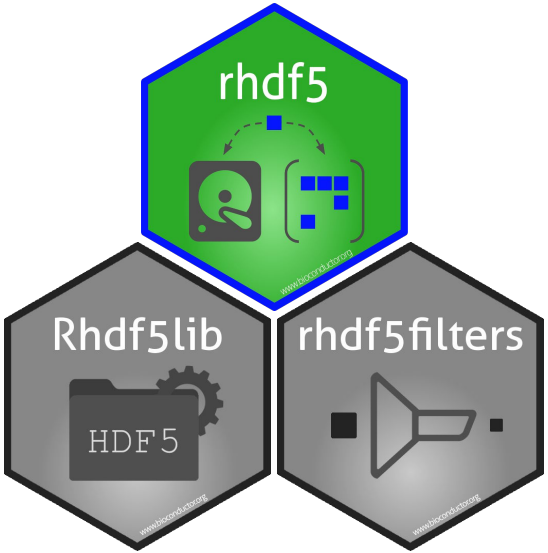



```
fid <- H5Fcreate( name = "/my/special/file.h5" )
sid <- H5Screate_simple( c(2,1) )
did <- H5Dcreate( fid, "A", "H5T_STD_I32LE", sid )

H5Dwrite(did, 1L:2L, h5spaceMem = sid, h5spaceFile = sid)

H5Dclose( did )
H5Sclose( sid )
H5Fclose( fid )
```

rhdf5 - wrapper functions



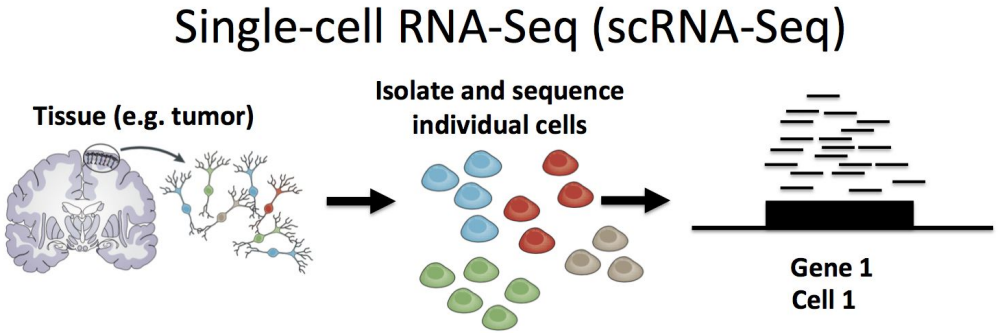

```
h5createFile( file = "/my/special/file.h5" )

h5write( file = "/my/special/file.h5",
         obj  = 1L:2L,
         name = "A" )

h5read(  file = "/my/special/file.h5",
         name = "A" )
## [1] 1 2
```

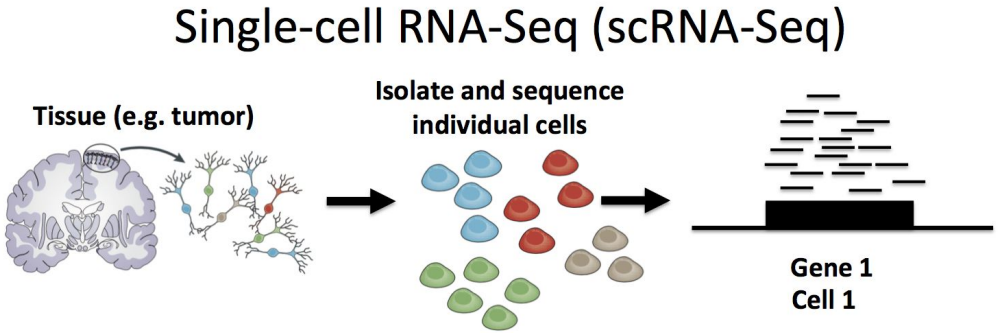
Example use case: Single-cell sequencing

It’s all about a counts matrix



Stephanie Hicks - Welcome to the World of Single-Cell RNA-Sequencing
https://speakerdeck.com/stephaniehicks/welcome-to-the-world-of-single-cell-rna-sequencing?slide=3

It’s all about a counts matrix

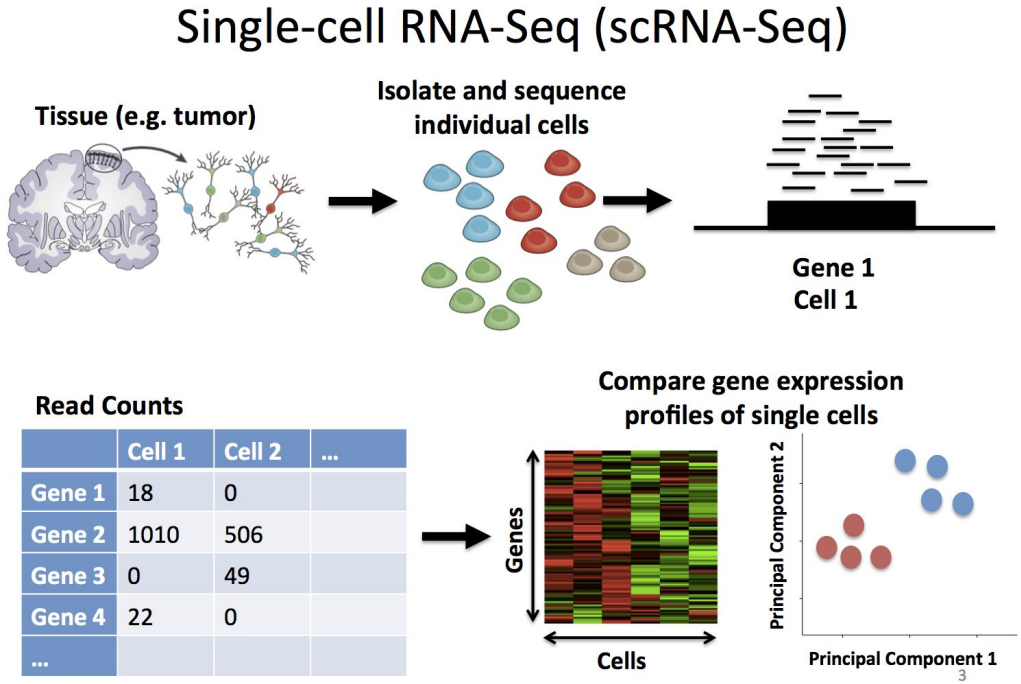


Read Counts

	Cell 1	Cell 2	…
Gene 1	18	0	
Gene 2	1010	506	
Gene 3	0	49	
Gene 4	22	0	
…			

Stephanie Hicks - Welcome to the World of Single-Cell RNA-Sequencing
https://speakerdeck.com/stephaniehicks/welcome-to-the-world-of-single-cell-rna-sequencing?slide=3

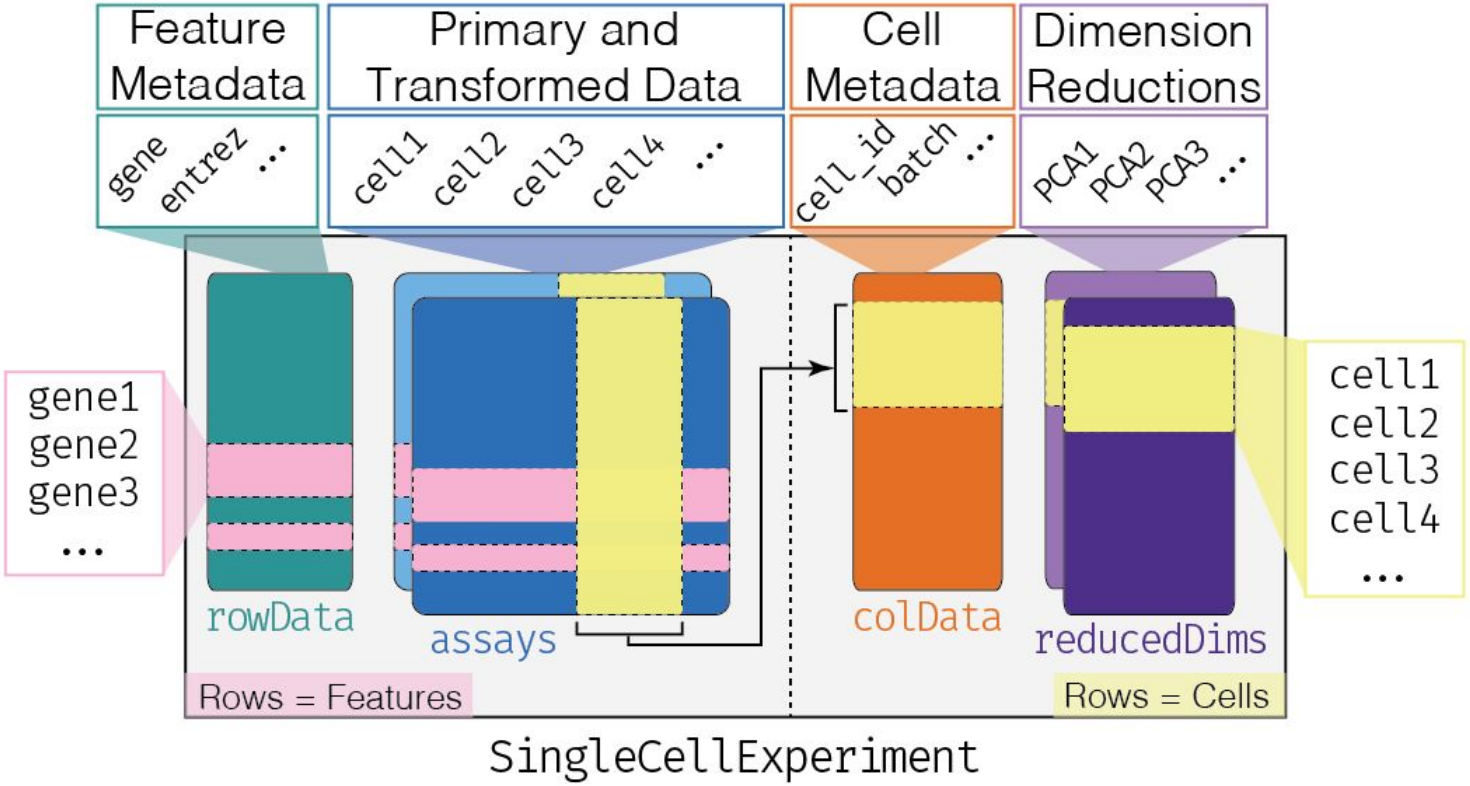
It’s all about a counts matrix



	Cell 1	Cell 2	...
Gene 1	18	0	
Gene 2	1010	506	
Gene 3	0	49	
Gene 4	22	0	
...			

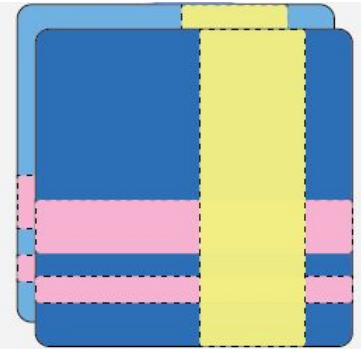
Stephanie Hicks - Welcome to the World of Single-Cell RNA-Sequencing
https://speakerdeck.com/stephaniehicks/welcome-to-the-world-of-single-cell-rna-sequencing?slide=3

Bioconductor defines a common class for this data



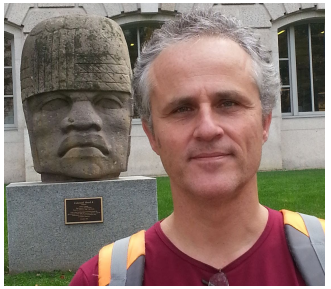
Counts matrices

- Data are typically sparse (> 90% zeros)
- Number of genes & cells vary a lot
- Small datasets can be represented in memory
  - Either dense or sparse representations
- Large datasets (30,000 genes, > 1,000,000 cells) need another solution
  - HDF5 backed on-disk arrays



HDF5Array package provides familiar R interface to on-disk arrays

Hervé Pagès



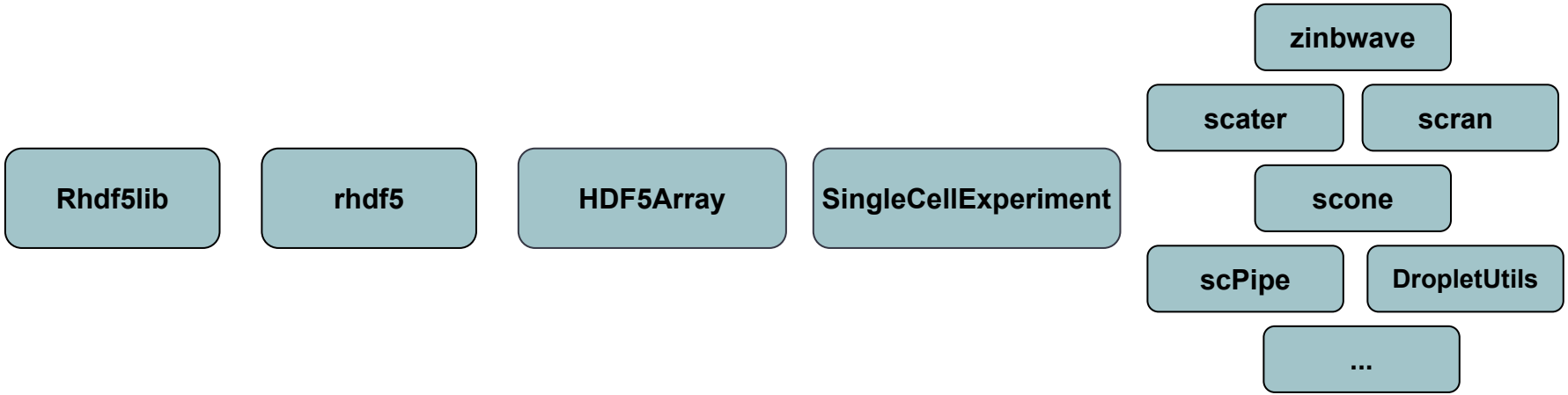
- Drop-in replacement for in-memory arrays
- Points to a single HDF5 dataset
- Upstream analysis packages don't (necessarily) care
- In practice algorithms probably need to be optimised - many are

```
M1 <- HDF5Array(
      file = "/my/special/file.h5",
      name = "counts" )

M1[1:10, ]

mean( M1 )
```

On-disk single-cell software stack



C / C++ Library → R Interface → Counts Matrix → Complete SC Dataset → Analysis Tools

Thanks to EMBL Huber Lab & BioC community!

`https://bioconductor.org/packages/rhdf5`

@grimbough