NDHL:
Planning a shared stack and data repository

The Royal Danish Library

Per Møldrup-Dalum
specialkonsulent

Humanisten, Göteborg
december 2019

DET KGL.
BIBLIOTEK
Royal Danish Library

Data—or The Collections

This lists what we have, not what is available to everyone

- Radio: ~1.5e6 hours (from 1989 –)
- TV: ~1e6 hours (from 1989 –)
- The Danish Web Archive: ~30e9 “documents” from 2005 –
- The Danish Newspaper Collection: ~35e6 (~2e6) pages from 1666 –
- Limited amounts of eBooks
- Old image scans, an old map collections, postcards, ...
- Holberg, Grundtvig, Kierkegaard, Brandes, Andersen, ...
- ...

Also open data e.g. 17TB from The British Library and LOAR

- OCR text from Danish newspapers from 1660 to 1877 in csv files
- Word2vec dictionaries of danish newspapers and Gutenberg E-books
- Ruben Recordings: The oldest Danish audio recordings from the 1890's

Platform—hardware for computation

Local cloud for system management and project specific machines

- Two DELL servers
- 512 GB RAM
- Two Intel 18 core CPU
- Four 4TB spinning hard drives
- Four 10Gbit network interfaces

Computation ”on the metal” – 9 DELL servers

- Each with two Intel 18 core CPU
- 256GB RAM
- Eight 4TB spinning discs
- Four 10Gbit network interfaces

One IBM Power System AC922

- 256GM RAM
- Two Power9 CPU with 32 cores
- Two NVIDIA Tesla V100 SXM2 16GB (10.240 GPU cores, 1280 TPU cores)

Cost: ~~150.000€
2.8e6 core*hours/year

~1 FTE for management, support, and development

NFS server with 47TB (primary for the Power System)

Platform — Software

***Big Data* platform — also for personally identifiable data/GDPR**

- Linux — RedHat, CentOS, and Springdale
- FreeIPA for user management. Kerberos for everything. OpenVPN for access. More OSS stuff
- oVirt for virtualization for the local cloud

- Logging of data access

- Hortonworks Data Platform
  - Ambari, MapReduce, HDFS, Hive
- Apache Spark
  - sparklyr for R and pySpark for Python

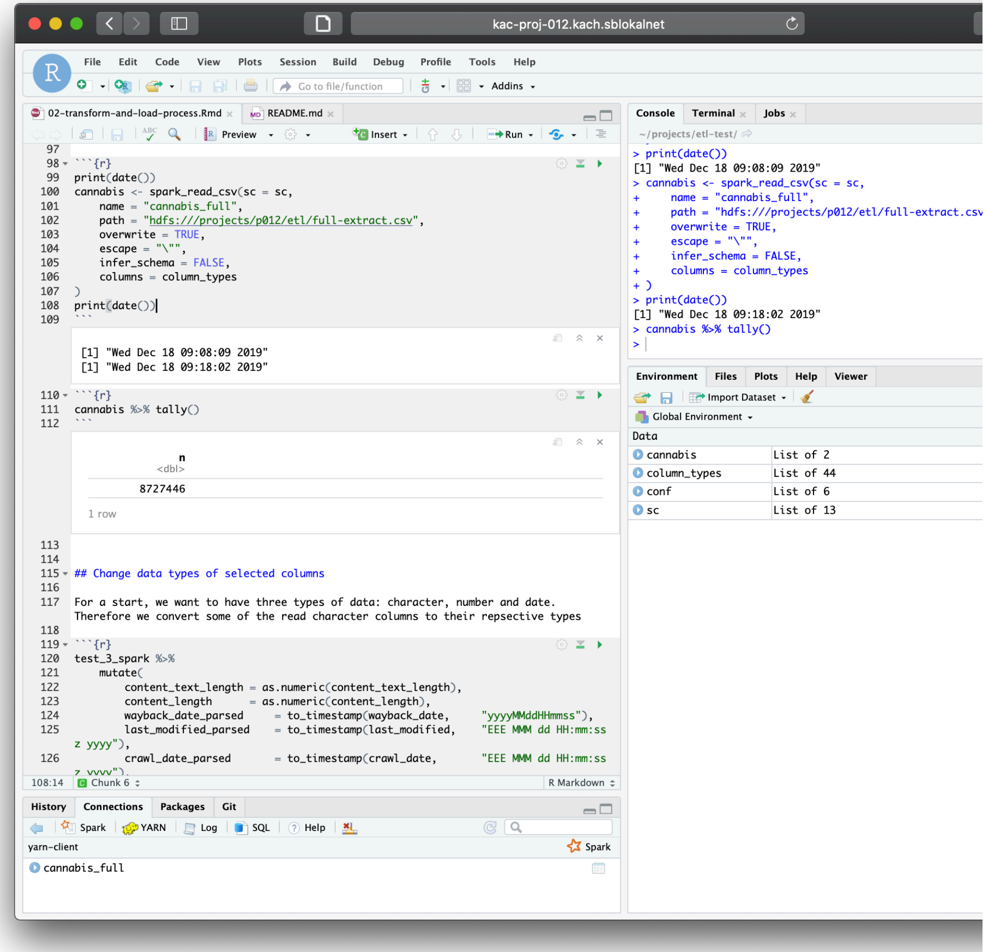
Compute — For the end user(*)

- Only access through VPN
- THEN ssh and browser based access and applications

- Data on 288TB/96TB HDFS storage (replication set to three)

Primary applications

- R
  - RStudio Server and Jupyter
  - sparklyr
  - Keras + tensorflow
- Python
  - Jupyter
  - pySpark
  - keras + tensorflow



~~~
```{r}
print(date())
cannabis <- spark_read_csv(sc = sc,
    name = "cannabis_full",
    path = "hdfs:///projects/p012/etl/full-extract.csv",
    overwrite = TRUE,
    escape = "\"",
    infer_schema = FALSE,
    columns = column_types
)
print(date())
```

[1] "Wed Dec 18 09:08:09 2019"
[1] "Wed Dec 18 09:18:02 2019"

```{r}
cannabis %>% tally()
```

         n
     <dbl>
   8727446

1 row


## Change data types of selected columns

For a start, we want to have three types of data: character, number and date.
Therefore we convert some of the read character columns to their repsective types

```{r}
test_3_spark %>%
    mutate(
        content_text_length = as.numeric(content_text_length),
        content_length      = as.numeric(content_length),
        wayback_date_parsed     = to_timestamp(wayback_date,      "yyyyMMddHHmmss"),
        last_modified_parsed    = to_timestamp(last_modified,     "EEE MMM dd HH:mm:ss
z yyyy"),
        crawl_date_parsed       = to_timestamp(crawl_date,        "EEE MMM dd HH:mm:ss
z yyyy"),
~~~

```
> print(date())
[1] "Wed Dec 18 09:08:09 2019"
> cannabis <- spark_read_csv(sc = sc,
+     name = "cannabis_full",
+     path = "hdfs:///projects/p012/etl/full-extract.csv
+     overwrite = TRUE,
+     escape = "\"",
+     infer_schema = FALSE,
+     columns = column_types
+ )
> print(date())
[1] "Wed Dec 18 09:18:02 2019"
> cannabis %>% tally()
> 
```

Data	
cannabis	List of 2
column_types	List of 44
conf	List of 6
sc	List of 13

cannabis_full

Consultancy and project support

- Project management
- Legal
- Data understanding (for KB collections)
- ETL
- Data Engineering
- Analysis

Project examples

- Study of the approval of medicinal cannabis
- Temporality on the frontpage of a national newspaper
- Historical development of the tracking of users on the internet
- Historical development of Danish morning radio
- The Danish co-creative innovation culture