Traffic Loop Data

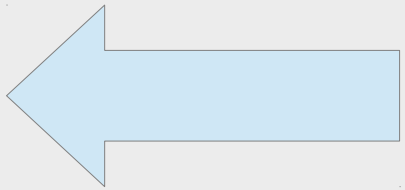
A case study in RHadoop

Marco Puts, B. Meindl, L. Ma, P. Del Rey

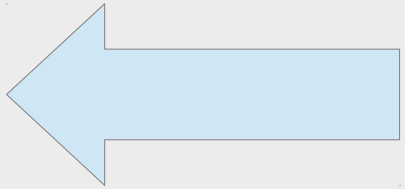
#NTTS2015

RHadoop

```
library("rmr2")
library("stringr")
mapfunction <- function(k,v)
{
  text = gsub("\\n|[[:punct:]]",  " ", v)
  k = str_split(string = tolower(text), pattern = " ")[[1]]
  v = rep(1, length(k));
  return (keyval(k,v))
}


reducefunction <- function(k,v)
{

  key = unique(k)
  value = sum(v)
  return (keyval(key,value))
}

data = to.dfs(text);

res = mapreduce(
  input = data,
  map = mapfunction,
  reduce = reducefunction,
  combine = TRUE
)

wordcount = from.dfs(res)
```



MAP



REDUCE

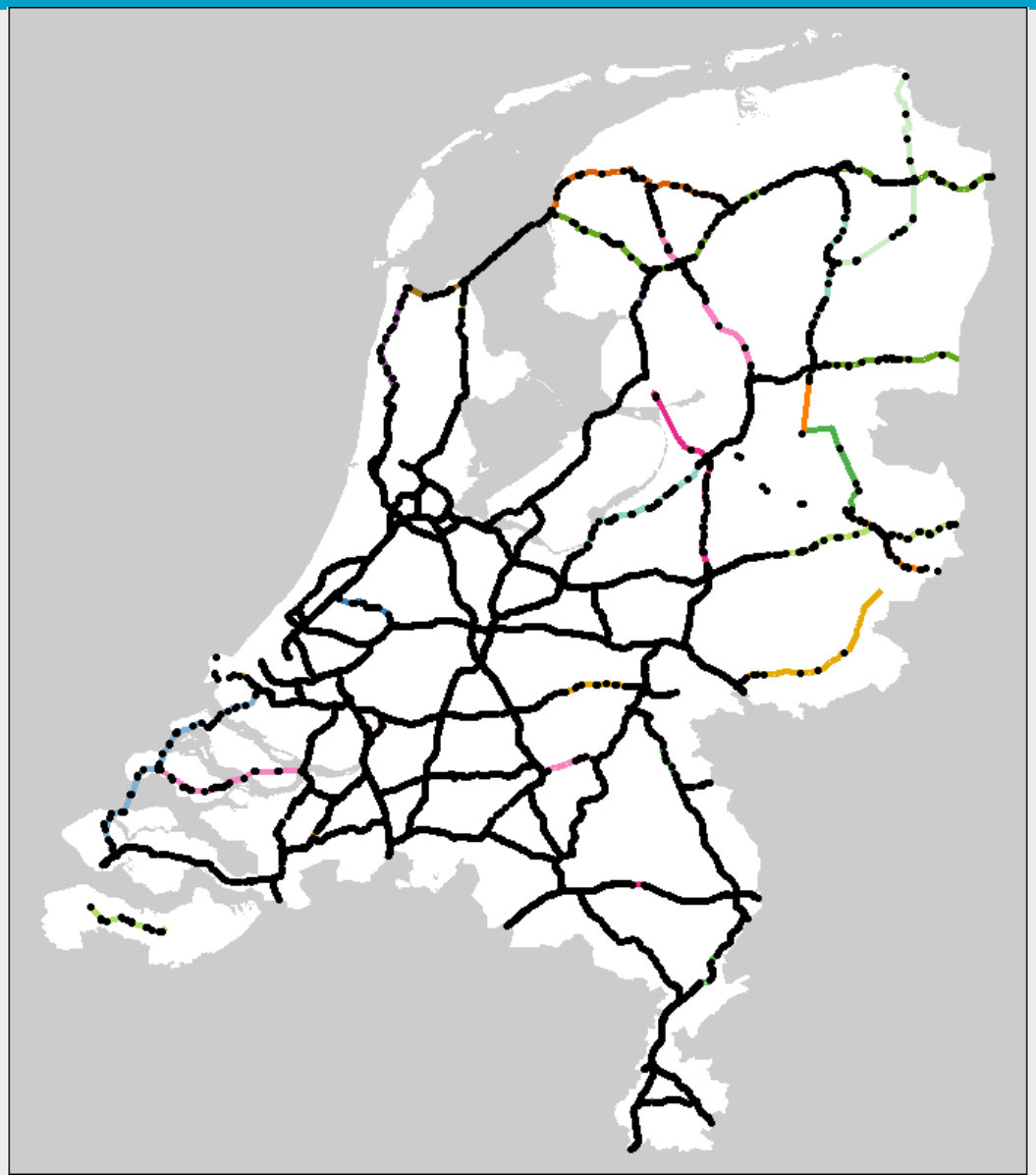
CALL

Road sensors

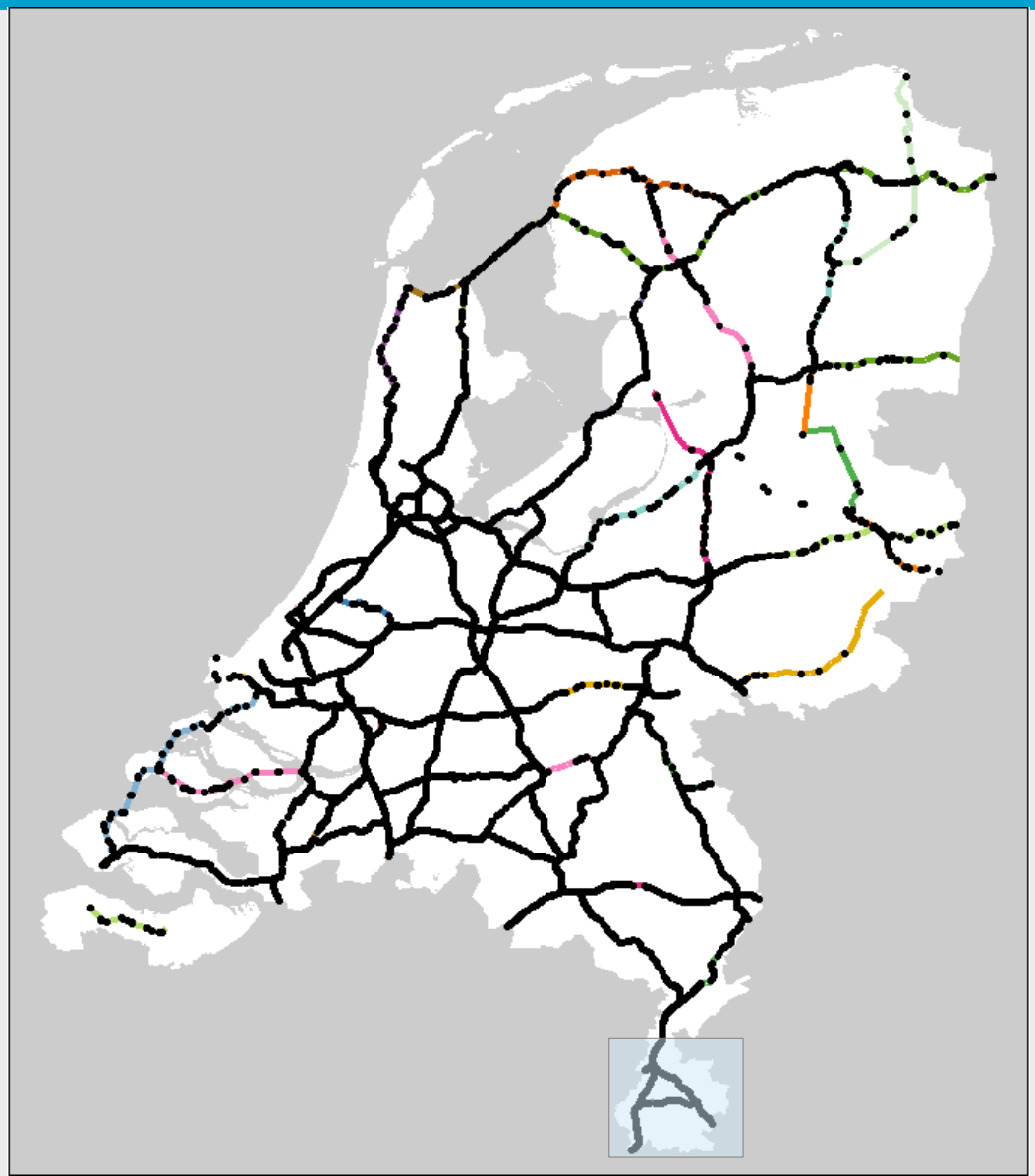
Road sensor data

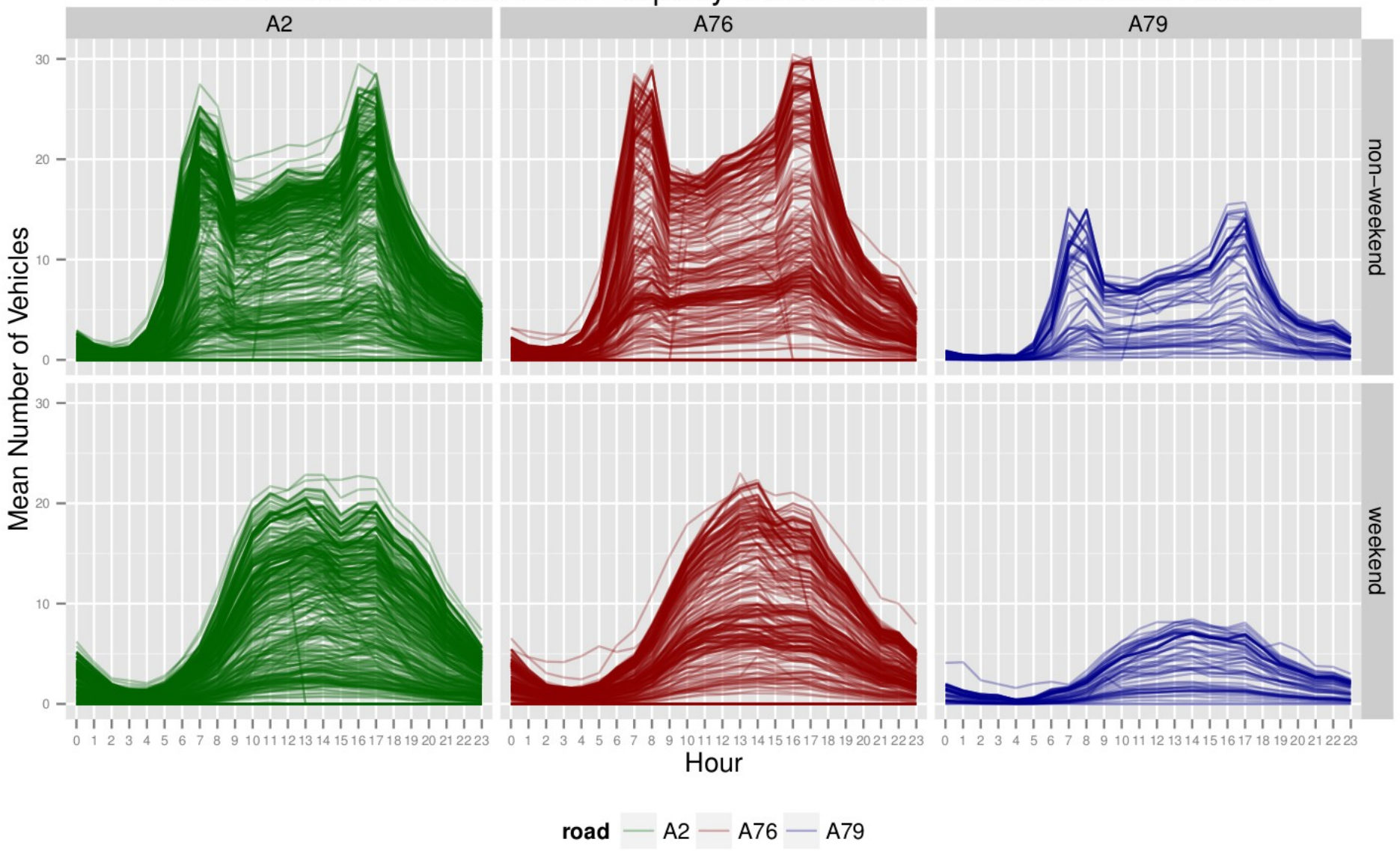
- Passing vehicle counts for each minute (24/7) at about 60.000 sensors in the Netherlands
- Types of sensors:
  - Induction loop
  - Camera
  - Bluetooth
- Length categories (e.g. small, medium, long vehicles)
- Large volume

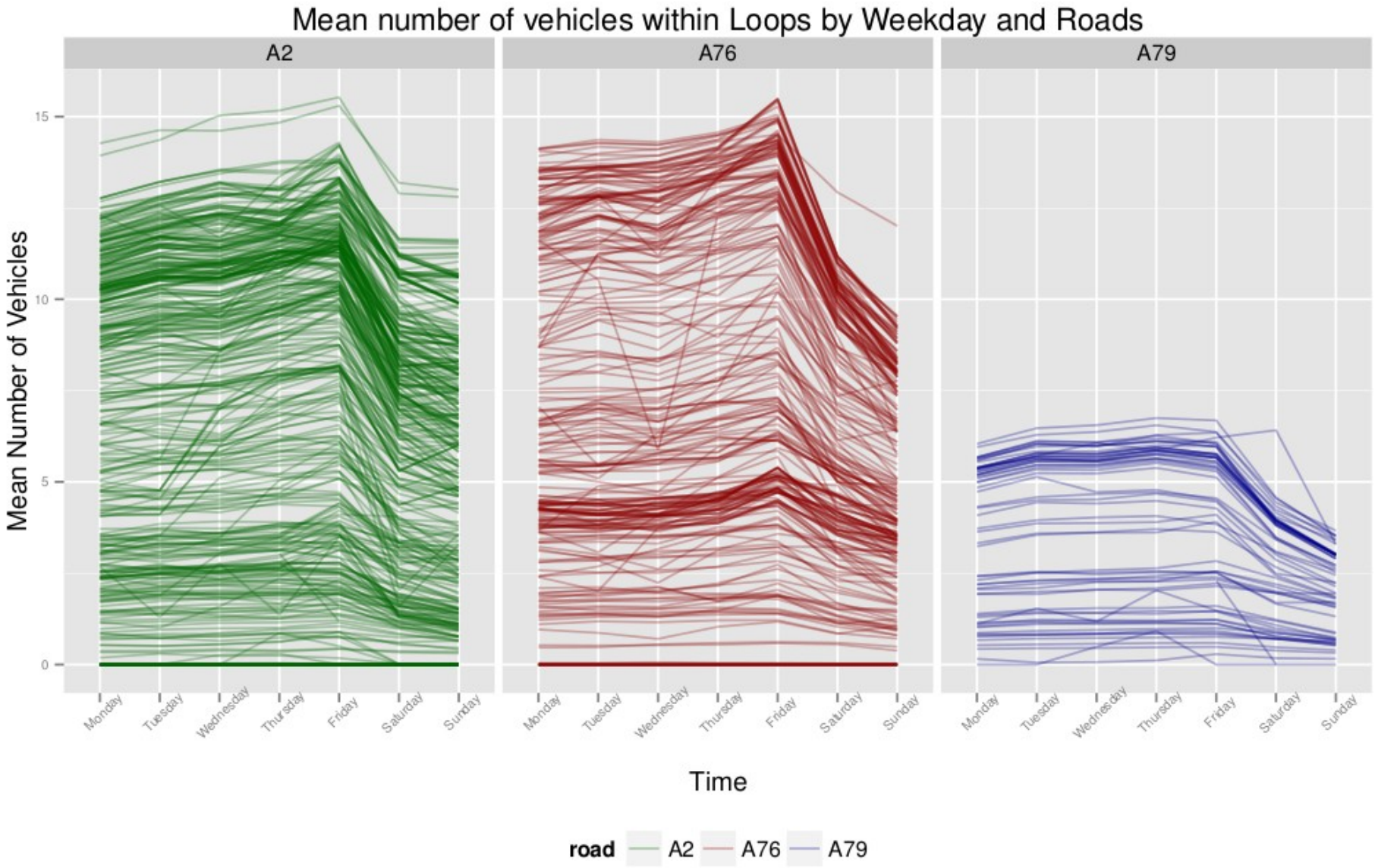
Dutch road sensors



Dutch road sensors




Mean number of vehicles within Loops by Weekends/Non−Weekends and Roads
A2
A76
A79
non−weekend
weekend
Mean Number of Vehicles
0
10
20
30
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
Hour
road
A2
A76
A79


Mean number of vehicles within Loops by Weekday and Roads
A2
A76
A79
Mean Number of Vehicles
15
10
5
0
Monday
Tuesday
Wednesday
Thursday
Friday
Saturday
Sunday
Time
road
A2
A76
A79

What about all data

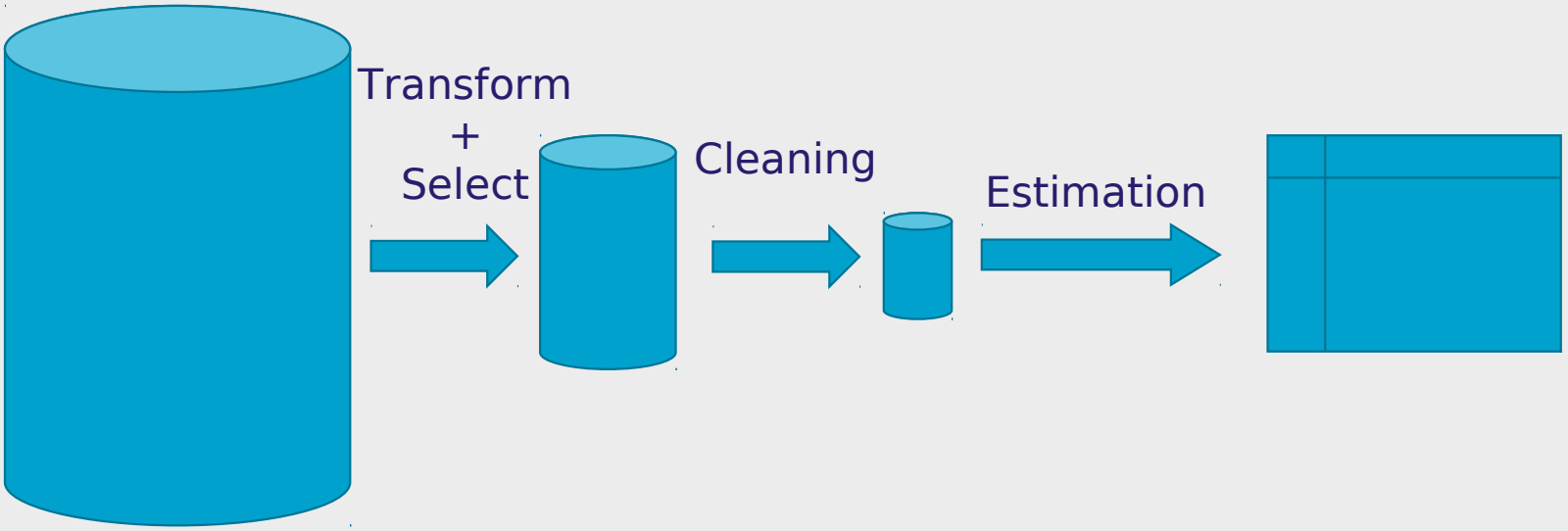
Size in Bytes

- 1 TB (compressed) per year
- 2.7 GB (compressed) per day

Size in Records

- 84 Billion records a year
- 230 Million records a day

Data Size



Raw Data
4 TB

2010 -
2014

Transformed
Data
60 GB

Clean Data
1 GB

Traffic Index
6 KB

Transform + Select

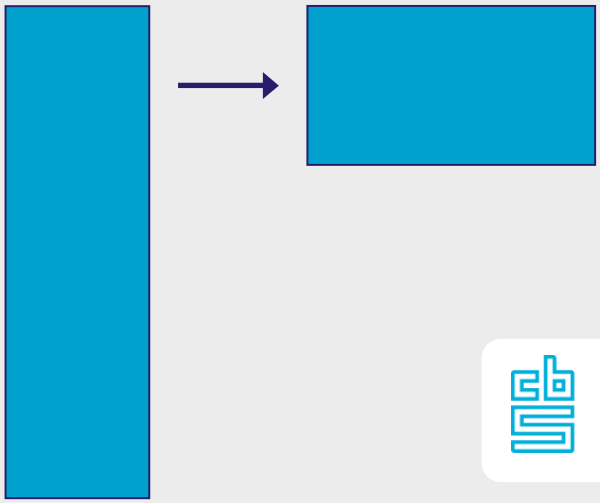
Reduce the Volume of the Data

Select

- Only necesary variables
- On the main routes (without ramps)

Transform

- Put one day in one record



Transform + Select

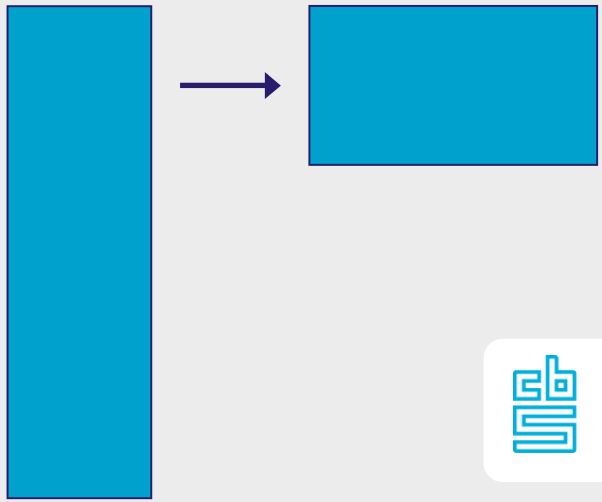
Experiment at Sandbox

Transport data to Ireland

- Mailing a NAS server serves more bandwidth than internet, hence faster (and cheaper)

Processing on RHadoop

- Slow
  - We used gnuzipped files
  - Chunks of data not optimal



Data Editing

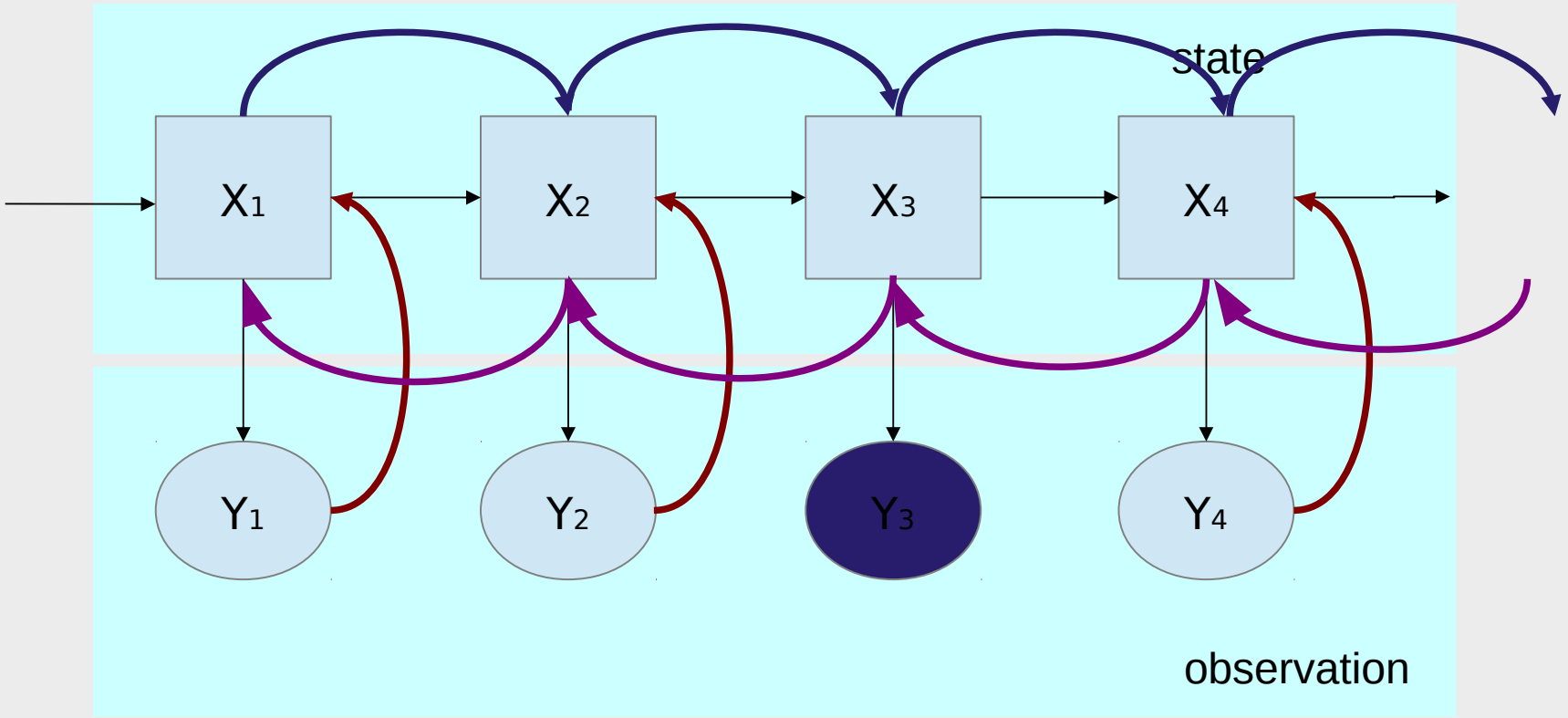
Traditionally:

- 80:20 rule automated/manual editing

Big Data:

- Manual editing on micro level impossible
- Alternative: Digital Filtering

Cleaning the Data
Recursive Bayesian Estimation



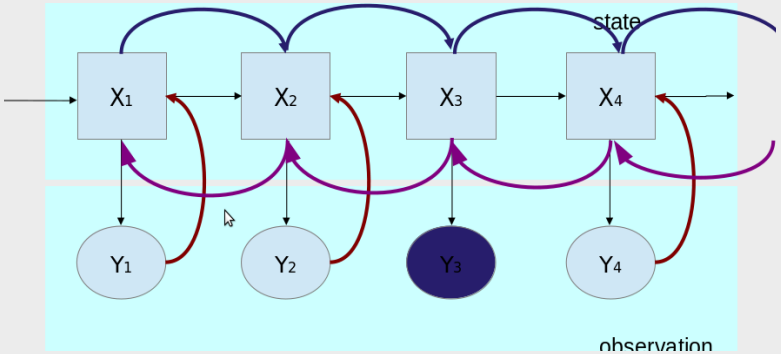
Smoothing

Filtering

Hadoop not the first choice

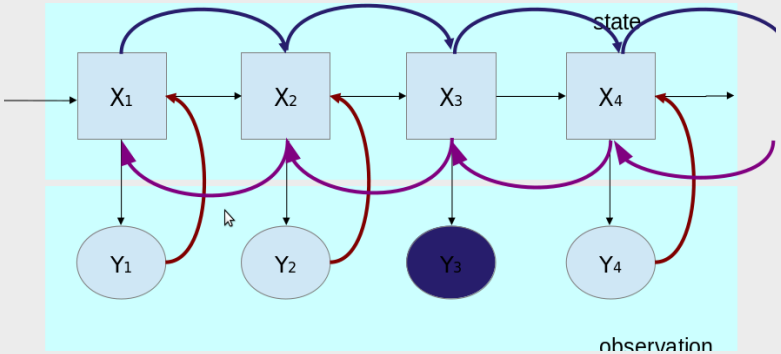
- Data already ordered
- Running the process on locally available datasets is better

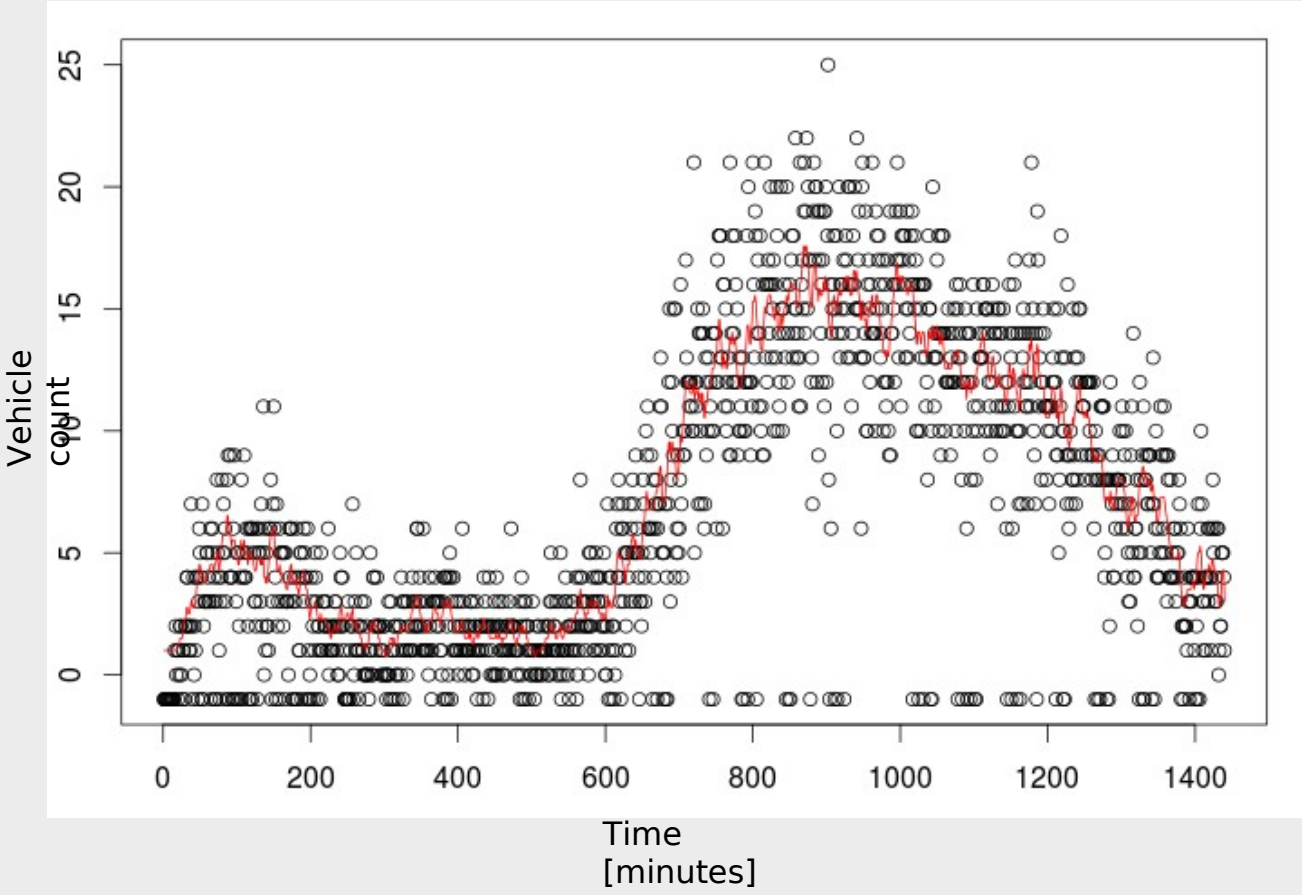
Spark!



Filtering

- *scp* chunks of data to different nodes
- Process chunks in parallel using *ssh*




Vehicle count
0
5
10
15
20
25
0
200
400
600
800
1000
1200
1400
Time
[minutes]

Conclusion

- RHadoop = R + Hadoop
  - Pro: Nice combination
  - Con: R only fast with dedicated packages
- Big data processing brings Big data challenges