

# Big Data Analytics

## Lecture 5: Cleaning and Transformation of Big Data

*Prof. Dr. Ulrich Matter  
(University of St. Gallen)*

*19/03/2020*

## 1 Cleaning and Transformation of large data sets

Preceding the filtering/selection/aggregation of raw data, data cleaning and transformation typically have to be run on large parts of the overall dataset. In practice, the bottleneck is often a lack of RAM. In the following, we explore two strategies that broadly build on the idea of *virtual memory* (using parts of the hard disk as RAM).

### 1.1 ‘Out-of-memory’ strategies

Virtual memory is in simple words an approach to combining the RAM and mass storage components in order to cope with a lack of RAM. Modern operating systems come with a virtual memory manager that would automatically handle the swapping between RAM and the hard-disk, when running processes use up too much RAM. However, a virtual memory manager is not specifically developed to perform this task in the context of data analysis. Several strategies have thus been developed to build on the basic idea of virtual memory in the context of data analysis tasks.

- *Chunked data files on disk:* The data analytics software ‘partitions’ the large dataset, maps, and stores the chunks of raw data on disk. What is actually ‘read’ into RAM when importing the data file with this approach is the mapping to the partitions of the actual dataset (the data structure) and some metadata describing the dataset. In R, this approach is implemented in the **ff** package and several packages building on **ff**. In this approach, the usage of disk space and the linking between RAM and files on disk is very explicit (and well visible to the user).
- *Memory mapped files and shared memory:* The data analytics software uses segments of virtual memory for the dataset and allows different programs/processes to access it in the same memory segment. Thus, virtual memory is explicitly allocated for one or several specific data analytics tasks. In R, this approach is prominently implemented in the **bigrmemory** package and several packages building on **bigrmemory**.

#### 1.1.1 Chunking data with the **ff**-package

Before looking at the more detailed and applied code examples in Walkowiak (2016), we investigate how the **ff** package (and the concept of chunked files) basically works. In order to do so, we first install and load the **ff** and **ffbase** packages, as well as the **pryr** package. We use the already known **flights.csv**-dataset as an example. When importing data via the **ff** package, we first have to set up a directory where **ff** can store the partitioned dataset (recall that this is explicitly/visibly done on disk). As in the code examples of the book, we call this new directory **ffdf** (after **ff**-data.frame).

```
# SET UP -----  
  
# install.packages(c("ff", "ffbase"))  
# load packages  
library(ff)  
library(ffbase)
```

```
library(pryr)

# create directory for ff chunks, and assign directory to ff
system("mkdir ffd")
options(fftempdir = "ffd")
```

Now we can read in the data with `read.table.ffdf`. In order to better understand the underlying concept, we record the change in memory in the R environment with `mem_change()`.

```
mem_change(
  flights <-
    read.table.ffdf(file="./data/flights.csv",
                    sep=",",
                    VERBOSE=TRUE,
                    header=TRUE,
                    next.rows=100000,
                    colClasses=NA)
)
```

```
## read.table.ffdf 1..100000 (100000)  csv-read=0.854sec ffd-write=0.105sec
## read.table.ffdf 100001..200000 (100000)  csv-read=0.836sec ffd-write=0.082sec
## read.table.ffdf 200001..300000 (100000)  csv-read=1.045sec ffd-write=0.074sec
## read.table.ffdf 300001..336776 (36776)  csv-read=0.426sec ffd-write=0.059sec
## csv-read=3.161sec ffd-write=0.32sec TOTAL=3.481sec

## 783 kB
```

Note that there are two substantial differences to what we have previously seen when using `fread()`. It takes much longer to import a csv into the ffd structure. However, the RAM allocated to it is much smaller. This is exactly what we would expect, keeping in mind what `read.table.ffdf()` does in comparison to what `fread()` does.

Now we can actually have a look at the data chunks created by `ff`, as well as how the structure of the dataset is represented in the `flights` object.

```
# show the files in the directory keeping the chunks
list.files("ffd")
```

```
##      [1] "clone44ba368f4473.ff" "clone44ba370bb5a8.ff" "clone44ba396a8bf9.ff" "clone44ba3d938c6.ff"
##      [5] "clone44ba3dbd0f26.ff" "clone44ba41b0b777.ff" "clone44ba472da774.ff" "clone44ba507b2d5f.ff"
##      [9] "clone44ba520e77bd.ff" "clone44ba5e091c1c.ff" "clone44ba61c344d3.ff" "clone44ba62b85467.ff"
##     [13] "clone535a103392bb.ff" "clone535a20051d8e.ff" "clone535a27a0edea.ff" "clone535a321a902c.ff"
##     [17] "clone535a4cb9c0b7.ff" "clone535a7551366a.ff" "clone5462102f18d5.ff" "clone54621bd03c9f.ff"
##     [21] "clone54622b1f269.ff" "clone5462576a7e5e.ff" "clone546261f5c43c.ff" "clone54629ca6258.ff"
##     [25] "clone554e13a203a9.ff" "clone554e4fc2869b.ff" "clone554e5234d876.ff" "clone554e6bf5b63e.ff"
##     [29] "clone554e6c2db28b.ff" "clone554eb6f5524.ff" "clone88ea1794e71f.ff" "clone88ea355eedfb.ff"
##     [33] "clone88ea52ff329c.ff" "clone88ea65871f82.ff" "clone88ea67201853.ff" "clone88ea6c1d2609.ff"
##     [37] "clone9df456e1144.ff" "clone9df4819f2a.ff" "clone9df49ca1938.ff" "clone9df713cecf0.ff"
##     [41] "clone9df733dd160.ff" "clone9df8b9e5a7.ff" "cloneb4311de2ec.ff" "cloneb4321de271f.ff"
##     [45] "cloneb4370e8e45f.ff" "cloneb4375409956.ff" "cloneb437ee969e8.ff" "cloneb437fda7597.ff"
##     [49] "cloneb6c2eb4777f.ff" "cloneb6c42677f51.ff" "cloneb6c685940c9.ff" "cloneb6c7cb948c7.ff"
##     [53] "cloneb6c7e5fa5ee.ff" "cloneb6c97ba47d.ff" "clonebbbe11050343.ff" "clonebbbe12bea52b.ff"
##     [57] "clonebbbe1b4e6f7e.ff" "clonebbbe2916f69e.ff" "clonebbbe60192c6f.ff" "clonebbbede2640c.ff"
##     [61] "ff44ba143d551a.ff" "ff44ba2814a27f.ff" "ff44ba2a1e5123.ff" "ff44ba2c60e66f.ff"
##     [65] "ff44ba3f8ff3f4.ff" "ff44ba46dfe140.ff" "ff44ba6f53764a.ff" "ff44ba77d6f600.ff"
##     [69] "ff535a2302d6a4.ff" "ff535a52d98720.ff" "ff535a691d6bdc.ff" "ff535a6d8c5f75.ff"
##     [73] "ff54621f662045.ff" "ff54623bad36f1.ff" "ff54625fe2991f.ff" "ff5462736c0808.ff"
```

##	[77]	"ff554e2c24fb29.ff"	"ff554e2fe55763.ff"	"ff554e39b259ec.ff"	"ff554e508fd7d1.ff"
##	[81]	"ff88ea168eae1.ff"	"ff88ea1a3b005d.ff"	"ff88ea30d1b051.ff"	"ff88eaa25c3e1.ff"
##	[85]	"ff9df3d2faaf8.ff"	"ff9df56e7baa4.ff"	"ff9df616fc631.ff"	"ff9df78e918f.ff"
##	[89]	"ffb4343f9b304.ff"	"ffb4349e952e2.ff"	"ffb43511cf486.ff"	"ffb4367245073.ff"
##	[93]	"ffb6c146c0f68.ff"	"ffb6c3986ed13.ff"	"ffb6c493697e6.ff"	"ffb6c59632dfa.ff"
##	[97]	"ffbbbe224db3aa.ff"	"ffbbbe22991425.ff"	"ffbbbe25c8e87b.ff"	"ffbbbe6ff99ae1.ff"
##	[101]	"ffdf44ba10fcdcf7.ff"	"ffdf44ba1183a7b4.ff"	"ffdf44ba119ee09e.ff"	"ffdf44ba13505151.ff"
##	[105]	"ffdf44ba14f62d1f.ff"	"ffdf44ba156d96e0.ff"	"ffdf44ba15fb6ceb.ff"	"ffdf44ba168f5fb6.ff"
##	[109]	"ffdf44ba19e433a4.ff"	"ffdf44ba1ab15e80.ff"	"ffdf44ba1be65408.ff"	"ffdf44ba1bfcc961.ff"
##	[113]	"ffdf44ba1d41381e.ff"	"ffdf44ba1e3fc23d.ff"	"ffdf44ba1ef5ba23.ff"	"ffdf44ba1f00463b.ff"
##	[117]	"ffdf44ba201ab4e4.ff"	"ffdf44ba20ba5e2.ff"	"ffdf44ba20c9e030.ff"	"ffdf44ba21e30e9c.ff"
##	[121]	"ffdf44ba222995d1.ff"	"ffdf44ba226b58d2.ff"	"ffdf44ba22be586c.ff"	"ffdf44ba259d8021.ff"
##	[125]	"ffdf44ba26248c06.ff"	"ffdf44ba262dfdbd.ff"	"ffdf44ba263620ea.ff"	"ffdf44ba26d7c648.ff"
##	[129]	"ffdf44ba272aab7c.ff"	"ffdf44ba273c0b51.ff"	"ffdf44ba2891670e.ff"	"ffdf44ba28d46a8b.ff"
##	[133]	"ffdf44ba2964f17a.ff"	"ffdf44ba2b592b76.ff"	"ffdf44ba2baaf63f.ff"	"ffdf44ba2ca55b0d.ff"
##	[137]	"ffdf44ba2cad1017.ff"	"ffdf44ba2d56880d.ff"	"ffdf44ba2dabc393.ff"	"ffdf44ba2e0c58f9.ff"
##	[141]	"ffdf44ba2e96e787.ff"	"ffdf44ba2ffc5080.ff"	"ffdf44ba306e23c3.ff"	"ffdf44ba33d74361.ff"
##	[145]	"ffdf44ba353a67f6.ff"	"ffdf44ba359454a5.ff"	"ffdf44ba38e0c51d.ff"	"ffdf44ba3959f533.ff"
##	[149]	"ffdf44ba3bda9a8e.ff"	"ffdf44ba3c2df73b.ff"	"ffdf44ba3d22588f.ff"	"ffdf44ba3d7d6f3.ff"
##	[153]	"ffdf44ba3e55ff43.ff"	"ffdf44ba3ff4cf71.ff"	"ffdf44ba40f104af.ff"	"ffdf44ba413a6f5.ff"
##	[157]	"ffdf44ba41d43525.ff"	"ffdf44ba429cc734.ff"	"ffdf44ba42da4916.ff"	"ffdf44ba42e7bda2.ff"
##	[161]	"ffdf44ba43b56d97.ff"	"ffdf44ba441f1bd8.ff"	"ffdf44ba44401f85.ff"	"ffdf44ba44ff9076.ff"
##	[165]	"ffdf44ba469e2857.ff"	"ffdf44ba489ae20a.ff"	"ffdf44ba489c5c9a.ff"	"ffdf44ba48ccc422.ff"
##	[169]	"ffdf44ba48d5bdb9.ff"	"ffdf44ba490ee4d7.ff"	"ffdf44ba49ed29ac.ff"	"ffdf44ba4ab7997a.ff"
##	[173]	"ffdf44ba4b02db63.ff"	"ffdf44ba4bc6a0c2.ff"	"ffdf44ba4d72466c.ff"	"ffdf44ba4dcc571d.ff"
##	[177]	"ffdf44ba4dd577c4.ff"	"ffdf44ba4e041a5.ff"	"ffdf44ba4f60ed7d.ff"	"ffdf44ba501dd98b.ff"
##	[181]	"ffdf44ba52671d92.ff"	"ffdf44ba52a8b87b.ff"	"ffdf44ba52e5cf24.ff"	"ffdf44ba52ed31e6.ff"
##	[185]	"ffdf44ba54634a34.ff"	"ffdf44ba54fa6143.ff"	"ffdf44ba554f03fe.ff"	"ffdf44ba55acae1b.ff"
##	[189]	"ffdf44ba56060d2.ff"	"ffdf44ba565d24d8.ff"	"ffdf44ba566ad79e.ff"	"ffdf44ba570b06f6.ff"
##	[193]	"ffdf44ba57b55dc4.ff"	"ffdf44ba582adddc.ff"	"ffdf44ba584dc108.ff"	"ffdf44ba58bc2182.ff"
##	[197]	"ffdf44ba5954b747.ff"	"ffdf44ba5a772367.ff"	"ffdf44ba5bbd3ee8.ff"	"ffdf44ba5d64476b.ff"
##	[201]	"ffdf44ba5f69fae8.ff"	"ffdf44ba615f2484.ff"	"ffdf44ba6204dcf0.ff"	"ffdf44ba62606dfa.ff"
##	[205]	"ffdf44ba628cf1b3.ff"	"ffdf44ba64488a39.ff"	"ffdf44ba6487d241.ff"	"ffdf44ba64cd52e.ff"
##	[209]	"ffdf44ba650cb87e.ff"	"ffdf44ba65364d8e.ff"	"ffdf44ba653cf693.ff"	"ffdf44ba656d2065.ff"
##	[213]	"ffdf44ba664bc3bc.ff"	"ffdf44ba6665565d.ff"	"ffdf44ba67b4ff51.ff"	"ffdf44ba6a2d6aeb.ff"
##	[217]	"ffdf44ba6b346635.ff"	"ffdf44ba6be9424e.ff"	"ffdf44ba6c34478c.ff"	"ffdf44ba6d494497.ff"
##	[221]	"ffdf44ba6d826f00.ff"	"ffdf44ba6db8096a.ff"	"ffdf44ba6dc2615a.ff"	"ffdf44ba6fc68681.ff"
##	[225]	"ffdf44ba7092830.ff"	"ffdf44ba70d204b9.ff"	"ffdf44ba71090da1.ff"	"ffdf44ba714b011e.ff"
##	[229]	"ffdf44ba71f265.ff"	"ffdf44ba72b33330.ff"	"ffdf44ba733692ad.ff"	"ffdf44ba74709d2.ff"
##	[233]	"ffdf44ba74859401.ff"	"ffdf44ba74a922fa.ff"	"ffdf44ba7500ab30.ff"	"ffdf44ba75ebc567.ff"
##	[237]	"ffdf44ba75edf094.ff"	"ffdf44ba781fa01b.ff"	"ffdf44ba7988afdd.ff"	"ffdf44ba79aa0cc7.ff"
##	[241]	"ffdf44ba7a109f74.ff"	"ffdf44ba7a632c09.ff"	"ffdf44ba7a747dbf.ff"	"ffdf44ba7b34ff45.ff"
##	[245]	"ffdf44ba7b60b984.ff"	"ffdf44ba7c52f8fe.ff"	"ffdf44ba7cd21480.ff"	"ffdf44ba7d1e0d33.ff"
##	[249]	"ffdf44ba7d58155e.ff"	"ffdf44ba7e0aa6be.ff"	"ffdf44ba7e9f2c45.ff"	"ffdf44ba7ee71852.ff"
##	[253]	"ffdf44ba7fe9a7ad.ff"	"ffdf44ba8f465a4.ff"	"ffdf44ba95dff01.ff"	"ffdf44ba96d842d.ff"
##	[257]	"ffdf44bab2d1b38.ff"	"ffdf44bab33c864.ff"	"ffdf44bac3786ec.ff"	"ffdf44bacd8ff55.ff"
##	[261]	"ffdf44bad7f62c7.ff"	"ffdf44badb8982c.ff"	"ffdf44bae45a12b.ff"	"ffdf44baeedf00c.ff"
##	[265]	"ffdf44baf6a9e78.ff"	"ffdf535a128ef651.ff"	"ffdf535a14527019.ff"	"ffdf535a1464cee7.ff"
##	[269]	"ffdf535a1511f799.ff"	"ffdf535a1988ceb7.ff"	"ffdf535a199f0ab9.ff"	"ffdf535a1bea1d50.ff"
##	[273]	"ffdf535a1c09e97b.ff"	"ffdf535a1c3a425f.ff"	"ffdf535a1cb380eb.ff"	"ffdf535a1efb0d29.ff"
##	[277]	"ffdf535a1f484b94.ff"	"ffdf535a2028f21c.ff"	"ffdf535a247e967f.ff"	"ffdf535a24e81e9a.ff"
##	[281]	"ffdf535a2979634d.ff"	"ffdf535a29e41b21.ff"	"ffdf535a2c4140b7.ff"	"ffdf535a31184fd2.ff"
##	[285]	"ffdf535a318cbd2f.ff"	"ffdf535a33f80b59.ff"	"ffdf535a369b6fb6.ff"	"ffdf535a3883dbe0.ff"
##	[289]	"ffdf535a3a0cadba.ff"	"ffdf535a3b65e60.ff"	"ffdf535a3c38dac4.ff"	"ffdf535a3faaf607.ff"

##	[293]	"ffdf535a3ff8bd78.ff"	"ffdf535a4054491.ff"	"ffdf535a4262853e.ff"	"ffdf535a430df409.ff"
##	[297]	"ffdf535a46ef399.ff"	"ffdf535a487b75ee.ff"	"ffdf535a48c88dfa.ff"	"ffdf535a4b450b2.ff"
##	[301]	"ffdf535a4badec37.ff"	"ffdf535a4c26e547.ff"	"ffdf535a4d61e970.ff"	"ffdf535a4d7d4e6d.ff"
##	[305]	"ffdf535a524e5fc6.ff"	"ffdf535a52d427e2.ff"	"ffdf535a5380ad38.ff"	"ffdf535a57606422.ff"
##	[309]	"ffdf535a5a09a379.ff"	"ffdf535a5ba99f1.ff"	"ffdf535a5e9c497f.ff"	"ffdf535a5f41090c.ff"
##	[313]	"ffdf535a5fa49636.ff"	"ffdf535a60746624.ff"	"ffdf535a60d8d4d3.ff"	"ffdf535a62f7676c.ff"
##	[317]	"ffdf535a6394c363.ff"	"ffdf535a63b90963.ff"	"ffdf535a63ef9e43.ff"	"ffdf535a6547c86d.ff"
##	[321]	"ffdf535a65f76d2b.ff"	"ffdf535a66df1cda.ff"	"ffdf535a67be4df4.ff"	"ffdf535a6892a4d1.ff"
##	[325]	"ffdf535a69e1aeff.ff"	"ffdf535a6e955663.ff"	"ffdf535a708945cc.ff"	"ffdf535a70c345d9.ff"
##	[329]	"ffdf535a72f5c389.ff"	"ffdf535a766ecb56.ff"	"ffdf535a76872a5a.ff"	"ffdf535a776007d2.ff"
##	[333]	"ffdf535a7be0f75f.ff"	"ffdf535a7d64c85a.ff"	"ffdf535a7dd2108a.ff"	"ffdf535a7f9677e5.ff"
##	[337]	"ffdf535a81359e2.ff"	"ffdf535a9b62932.ff"	"ffdf535ab4913e0.ff"	"ffdf535ad9adf04.ff"
##	[341]	"ffdf535ae6a79e4.ff"	"ffdf535aea02afd.ff"	"ffdf535aedb8e83.ff"	"ffdf535aff79a0d.ff"
##	[345]	"ffdf538510cffebe.ff"	"ffdf538516f59d49.ff"	"ffdf53851c137e46.ff"	"ffdf53851ca0fae9.ff"
##	[349]	"ffdf53851d25cfff.ff"	"ffdf53851d5a83c9.ff"	"ffdf53851d7a7d4c.ff"	"ffdf53852182102d.ff"
##	[353]	"ffdf538521bd292.ff"	"ffdf5385288ea278.ff"	"ffdf53852b104021.ff"	"ffdf53852c4c8915.ff"
##	[357]	"ffdf5385321cba51.ff"	"ffdf5385357c8aef.ff"	"ffdf538537e24c9d.ff"	"ffdf53853856c58.ff"
##	[361]	"ffdf538538c1578b.ff"	"ffdf53853eef0908.ff"	"ffdf53853f3a0c7f.ff"	"ffdf538543057a4f.ff"
##	[365]	"ffdf538543c73410.ff"	"ffdf538543fd8374.ff"	"ffdf5385475d7491.ff"	"ffdf538548900031.ff"
##	[369]	"ffdf5385494817d0.ff"	"ffdf53854b2b2933.ff"	"ffdf538550e6bc97.ff"	"ffdf5385539a9379.ff"
##	[373]	"ffdf538554b817b2.ff"	"ffdf538554bbd86b.ff"	"ffdf538557292f65.ff"	"ffdf538558f36a6.ff"
##	[377]	"ffdf53855c6786ff.ff"	"ffdf53855d21c343.ff"	"ffdf53855f07ffaa.ff"	"ffdf538561f40b8a.ff"
##	[381]	"ffdf538562222a4.ff"	"ffdf538565fcb62e.ff"	"ffdf53856a13f414.ff"	"ffdf538570395140.ff"
##	[385]	"ffdf5385709d7052.ff"	"ffdf5385787646da.ff"	"ffdf5385787d8442.ff"	"ffdf53857c0a8646.ff"
##	[389]	"ffdf53857c0c358a.ff"	"ffdf53857d4ccf23.ff"	"ffdf53857e9df9d2.ff"	"ffdf5385954f4b9.ff"
##	[393]	"ffdf5385cb8e5ae.ff"	"ffdf54621094e03e.ff"	"ffdf546216895e5.ff"	"ffdf5462197ed545.ff"
##	[397]	"ffdf5462198ae894.ff"	"ffdf54621d888677.ff"	"ffdf54621e2849f9.ff"	"ffdf54621f6ffc18.ff"
##	[401]	"ffdf54622132b1d.ff"	"ffdf54622165aab4.ff"	"ffdf5462221c286f.ff"	"ffdf546222337fa9.ff"
##	[405]	"ffdf5462243ff50a.ff"	"ffdf54622564d39.ff"	"ffdf546225e3fe40.ff"	"ffdf546225ed8fa8.ff"
##	[409]	"ffdf5462268fd69a.ff"	"ffdf546229a4660e.ff"	"ffdf54622a1fc8d2.ff"	"ffdf54622add7bd6.ff"
##	[413]	"ffdf54622b0cfbf3.ff"	"ffdf54622c2fa7cd.ff"	"ffdf54622d856dd0.ff"	"ffdf54622f1bbcf.ff"
##	[417]	"ffdf54622fc38d03.ff"	"ffdf5462306add23.ff"	"ffdf546230f7d9a1.ff"	"ffdf54623168fbc9.ff"
##	[421]	"ffdf5462329ce963.ff"	"ffdf5462334bd84c.ff"	"ffdf5462339c3921.ff"	"ffdf54623475508f.ff"
##	[425]	"ffdf546236647f13.ff"	"ffdf54623b6d666.ff"	"ffdf54623baf2121.ff"	"ffdf54623f270588.ff"
##	[429]	"ffdf5462418248b5.ff"	"ffdf5462431da2f2.ff"	"ffdf5462442eaa94.ff"	"ffdf5462459bb7c4.ff"
##	[433]	"ffdf546245d3ab1c.ff"	"ffdf54624683f317.ff"	"ffdf54624908c367.ff"	"ffdf54624aa381ed.ff"
##	[437]	"ffdf54624acfcba5.ff"	"ffdf54624d953dbc.ff"	"ffdf54624e632777.ff"	"ffdf54624efec2e.ff"
##	[441]	"ffdf54624fa5e4c4.ff"	"ffdf546250c5335c.ff"	"ffdf5462516e4c4d.ff"	"ffdf546251a84df9.ff"
##	[445]	"ffdf546253c00ae7.ff"	"ffdf54625405e7c2.ff"	"ffdf54625481e1e4.ff"	"ffdf5462594e4bb7.ff"
##	[449]	"ffdf54625adaafc2.ff"	"ffdf54625d4907f1.ff"	"ffdf54626192c86b.ff"	"ffdf54626238e658.ff"
##	[453]	"ffdf546263a5f388.ff"	"ffdf546265bffd5.ff"	"ffdf546266e1ebe2.ff"	"ffdf546267b386ce.ff"
##	[457]	"ffdf54626a4408a2.ff"	"ffdf54626fc8d22b.ff"	"ffdf546270a3bbe9.ff"	"ffdf54627375e3da.ff"
##	[461]	"ffdf5462741c0213.ff"	"ffdf5462786821d8.ff"	"ffdf546278e79011.ff"	"ffdf54627a9358a8.ff"
##	[465]	"ffdf54627f8eddd8.ff"	"ffdf546297c704f.ff"	"ffdf5462a1e6ae3.ff"	"ffdf5462ab74780.ff"
##	[469]	"ffdf5462bdfceed.ff"	"ffdf5462c77884b.ff"	"ffdf5462f6f2c08.ff"	"ffdf5462ffce758.ff"
##	[473]	"ffdf554e10a50d86.ff"	"ffdf554e129773c0.ff"	"ffdf554e12a32b8.ff"	"ffdf554e15304bc9.ff"
##	[477]	"ffdf554e15bf83fd.ff"	"ffdf554e17c49f68.ff"	"ffdf554e1941efe9.ff"	"ffdf554e1a75f5ae.ff"
##	[481]	"ffdf554e1a7b0649.ff"	"ffdf554e1ab42638.ff"	"ffdf554e1af9676f.ff"	"ffdf554e21bf2762.ff"
##	[485]	"ffdf554e22654f13.ff"	"ffdf554e226e9921.ff"	"ffdf554e23727c77.ff"	"ffdf554e23a0ee45.ff"
##	[489]	"ffdf554e23b1f9fc.ff"	"ffdf554e23eeac0.ff"	"ffdf554e25fb8426.ff"	"ffdf554e27ffce47.ff"
##	[493]	"ffdf554e28b99ad0.ff"	"ffdf554e292a7a09.ff"	"ffdf554e2939343a.ff"	"ffdf554e298f8daa.ff"
##	[497]	"ffdf554e2a896b79.ff"	"ffdf554e2b19c7d9.ff"	"ffdf554e2b852fc4.ff"	"ffdf554e2cd9ec6.ff"
##	[501]	"ffdf554e2d9b034.ff"	"ffdf554e2ed42383.ff"	"ffdf554e3047fc81.ff"	"ffdf554e31497acf.ff"
##	[505]	"ffdf554e3162e293.ff"	"ffdf554e3503032e.ff"	"ffdf554e36548014.ff"	"ffdf554e37d061e3.ff"

##	[509]	"ffdf554e37edb6f3.ff"	"ffdf554e38747774.ff"	"ffdf554e39760e32.ff"	"ffdf554e3a3eb459.ff"
##	[513]	"ffdf554e40a69473.ff"	"ffdf554e441ae0f3.ff"	"ffdf554e461291ea.ff"	"ffdf554e46618e86.ff"
##	[517]	"ffdf554e48a0daac.ff"	"ffdf554e4f1dc550.ff"	"ffdf554e4f5dfad7.ff"	"ffdf554e50726232.ff"
##	[521]	"ffdf554e51ca9ab5.ff"	"ffdf554e53093a39.ff"	"ffdf554e536db94e.ff"	"ffdf554e57862834.ff"
##	[525]	"ffdf554e58c5dc32.ff"	"ffdf554e58ce9e1.ff"	"ffdf554e5cb59ead.ff"	"ffdf554e5f44992f.ff"
##	[529]	"ffdf554e63659628.ff"	"ffdf554e63f343fb.ff"	"ffdf554e65a4bce9.ff"	"ffdf554e664e7da0.ff"
##	[533]	"ffdf554e664eb59f.ff"	"ffdf554e68afa4b1.ff"	"ffdf554e69678579.ff"	"ffdf554e6a93e0fa.ff"
##	[537]	"ffdf554e6ab25cee.ff"	"ffdf554e6b16d516.ff"	"ffdf554e6c135305.ff"	"ffdf554e6d29bd96.ff"
##	[541]	"ffdf554e6dbd7108.ff"	"ffdf554e6f3d546f.ff"	"ffdf554e71b8c97c.ff"	"ffdf554e759e1216.ff"
##	[545]	"ffdf554e77f44700.ff"	"ffdf554e785c2b2d.ff"	"ffdf554e7967cea0.ff"	"ffdf554e799321b2.ff"
##	[549]	"ffdf554e79cc3d25.ff"	"ffdf554e7f71dce6.ff"	"ffdf554e972da9a.ff"	"ffdf554eb075730.ff"
##	[553]	"ffdf554eb3fabd5.ff"	"ffdf554ec5dc168.ff"	"ffdf554ecb2a43d.ff"	"ffdf554ee2ea8a5.ff"
##	[557]	"ffdf554eed15f8.ff"	"ffdf554ef9da937.ff"	"ffdf88ea101482df.ff"	"ffdf88ea10b7ec72.ff"
##	[561]	"ffdf88ea10e3f37f.ff"	"ffdf88ea13cb2ff6.ff"	"ffdf88ea13f3888d.ff"	"ffdf88ea13fa8a7a.ff"
##	[565]	"ffdf88ea14c254ab.ff"	"ffdf88ea150bc30d.ff"	"ffdf88ea15b967d5.ff"	"ffdf88ea1695c606.ff"
##	[569]	"ffdf88ea16cf7cf4.ff"	"ffdf88ea1c37178b.ff"	"ffdf88ea1c6349d3.ff"	"ffdf88ea1cc74dd6.ff"
##	[573]	"ffdf88ea1dd94d34.ff"	"ffdf88ea1dfaa76c.ff"	"ffdf88ea20ff88f8.ff"	"ffdf88ea217d33fd.ff"
##	[577]	"ffdf88ea21d9d0f1.ff"	"ffdf88ea225039ef.ff"	"ffdf88ea227494a6.ff"	"ffdf88ea22cf04ce.ff"
##	[581]	"ffdf88ea22e03838.ff"	"ffdf88ea253802ac.ff"	"ffdf88ea255ba03b.ff"	"ffdf88ea259959d5.ff"
##	[585]	"ffdf88ea27732770.ff"	"ffdf88ea27fe862b.ff"	"ffdf88ea28465330.ff"	"ffdf88ea2acaed90.ff"
##	[589]	"ffdf88ea2ad8bad7.ff"	"ffdf88ea2e36cd33.ff"	"ffdf88ea2e8af66b.ff"	"ffdf88ea311043bc.ff"
##	[593]	"ffdf88ea32718ba2.ff"	"ffdf88ea329c06b8.ff"	"ffdf88ea3305b2d4.ff"	"ffdf88ea33593e23.ff"
##	[597]	"ffdf88ea33efd6a7.ff"	"ffdf88ea371a8590.ff"	"ffdf88ea3727ec0e.ff"	"ffdf88ea37329546.ff"
##	[601]	"ffdf88ea377d4b.ff"	"ffdf88ea37dfabd4.ff"	"ffdf88ea3a83ca34.ff"	"ffdf88ea3b02c709.ff"
##	[605]	"ffdf88ea3b1f9677.ff"	"ffdf88ea3c86528d.ff"	"ffdf88ea3c8d351f.ff"	"ffdf88ea3d8a07f4.ff"
##	[609]	"ffdf88ea3e58cc96.ff"	"ffdf88ea3f4fdc17.ff"	"ffdf88ea410b8b82.ff"	"ffdf88ea42fbbf7f.ff"
##	[613]	"ffdf88ea4544823c.ff"	"ffdf88ea46f58894.ff"	"ffdf88ea4bcd6fbe.ff"	"ffdf88ea5989eb36.ff"
##	[617]	"ffdf88ea59e642f9.ff"	"ffdf88ea5a9aa547.ff"	"ffdf88ea5cf4252a.ff"	"ffdf88ea5ed906c9.ff"
##	[621]	"ffdf88ea5f893250.ff"	"ffdf88ea5fdef81c.ff"	"ffdf88ea60c72b5c.ff"	"ffdf88ea6244b132.ff"
##	[625]	"ffdf88ea63248f93.ff"	"ffdf88ea64eab525.ff"	"ffdf88ea665c400e.ff"	"ffdf88ea687962b4.ff"
##	[629]	"ffdf88ea6d1ac3e5.ff"	"ffdf88ea6dac5052.ff"	"ffdf88ea6db15426.ff"	"ffdf88ea7053ca55.ff"
##	[633]	"ffdf88ea717b4158.ff"	"ffdf88ea754e2ebe.ff"	"ffdf88ea76d8c23b.ff"	"ffdf88ea78332758.ff"
##	[637]	"ffdf88ea79223e4b.ff"	"ffdf88ea793f4d01.ff"	"ffdf88ea7a4ca6c4.ff"	"ffdf88ea7ad28f3a.ff"
##	[641]	"ffdf88ea9ae7609.ff"	"ffdf88eabcd7c05.ff"	"ffdf88ead08cc10.ff"	"ffdf88eafd9e6f9.ff"
##	[645]	"ffdf9df1119ca8f.ff"	"ffdf9df11c18c63.ff"	"ffdf9df121efa97.ff"	"ffdf9df129bfae4.ff"
##	[649]	"ffdf9df131c7b75.ff"	"ffdf9df140a75f7.ff"	"ffdf9df140cc767.ff"	"ffdf9df14b23af2.ff"
##	[653]	"ffdf9df1b58b5c6.ff"	"ffdf9df1cd8d05b.ff"	"ffdf9df1e1b8083.ff"	"ffdf9df1f9227e6.ff"
##	[657]	"ffdf9df20f64729.ff"	"ffdf9df24fe720b.ff"	"ffdf9df272e257b.ff"	"ffdf9df28accf57.ff"
##	[661]	"ffdf9df2e1304ac.ff"	"ffdf9df2eced706.ff"	"ffdf9df2fd5d8cc.ff"	"ffdf9df32ed4820.ff"
##	[665]	"ffdf9df34417b0e.ff"	"ffdf9df34797d3b.ff"	"ffdf9df36f5eeef.ff"	"ffdf9df36f930c.ff"
##	[669]	"ffdf9df39bc9477.ff"	"ffdf9df39e7c0b0.ff"	"ffdf9df3a53d572.ff"	"ffdf9df3ac6b568.ff"
##	[673]	"ffdf9df3cf14326.ff"	"ffdf9df3d4fee1a.ff"	"ffdf9df3e749847.ff"	"ffdf9df3fc8dfbc.ff"
##	[677]	"ffdf9df3fd43e41.ff"	"ffdf9df413bf07b.ff"	"ffdf9df41673a6.ff"	"ffdf9df45091856.ff"
##	[681]	"ffdf9df456e376c.ff"	"ffdf9df46293cb2.ff"	"ffdf9df469ec255.ff"	"ffdf9df46f8b710.ff"
##	[685]	"ffdf9df4bafdcfd.ff"	"ffdf9df4c19a919.ff"	"ffdf9df4cf2ad3a.ff"	"ffdf9df4e0c12ff.ff"
##	[689]	"ffdf9df4fa4b135.ff"	"ffdf9df504ae24d.ff"	"ffdf9df5157d920.ff"	"ffdf9df52b0b4c.ff"
##	[693]	"ffdf9df52f74579.ff"	"ffdf9df554865d8.ff"	"ffdf9df55e24d48.ff"	"ffdf9df57e6dbd9.ff"
##	[697]	"ffdf9df581cd982.ff"	"ffdf9df58840cf2.ff"	"ffdf9df58b97d59.ff"	"ffdf9df5d05e830.ff"
##	[701]	"ffdf9df5e029962.ff"	"ffdf9df5e571528.ff"	"ffdf9df5e6fed98.ff"	"ffdf9df5f7d30bb.ff"
##	[705]	"ffdf9df5f85fb96.ff"	"ffdf9df60e87eab.ff"	"ffdf9df6297cfd5.ff"	"ffdf9df644f5669.ff"
##	[709]	"ffdf9df64943b6a.ff"	"ffdf9df657c698.ff"	"ffdf9df68f06e95.ff"	"ffdf9df69606d9d.ff"
##	[713]	"ffdf9df6caf2868.ff"	"ffdf9df6d58cdd.ff"	"ffdf9df7116d96c.ff"	"ffdf9df711a1b35.ff"
##	[717]	"ffdf9df72ba9fd6.ff"	"ffdf9df7433e24.ff"	"ffdf9df756bb0a4.ff"	"ffdf9df781a46fe.ff"
##	[721]	"ffdf9df79d48b90.ff"	"ffdf9df7ab34b60.ff"	"ffdf9df7b7c4410.ff"	"ffdf9df7c3fde04.ff"

##	[725]	"ffdf9df7c421ab2.ff"	"ffdf9df7ca42125.ff"	"ffdf9df8c519d.ff"	"ffdf9dfbb72e3e.ff"
##	[729]	"ffdf9dfd86e2ec.ff"	"ffdf9dfec2e06.ff"	"ffdfb431507eab8.ff"	"ffdfb43152725cd.ff"
##	[733]	"ffdfb4315307f76.ff"	"ffdfb431862fc6e.ff"	"ffdfb43196dfcf2.ff"	"ffdfb4319c4666f.ff"
##	[737]	"ffdfb431b238cdf.ff"	"ffdfb431bf15b89.ff"	"ffdfb431bff5be0.ff"	"ffdfb431feeb5e9.ff"
##	[741]	"ffdfb43201b7ca7.ff"	"ffdfb43227edd6a.ff"	"ffdfb43230026d8.ff"	"ffdfb43241463b5.ff"
##	[745]	"ffdfb43255b4ab.ff"	"ffdfb43285833f6.ff"	"ffdfb432b08db4.ff"	"ffdfb432d210ea0.ff"
##	[749]	"ffdfb432f3b1950.ff"	"ffdfb432f5e6b68.ff"	"ffdfb433035cc6a.ff"	"ffdfb4330e72f16.ff"
##	[753]	"ffdfb43311b7da7.ff"	"ffdfb43349153e7.ff"	"ffdfb43369e1294.ff"	"ffdfb4336bb702b.ff"
##	[757]	"ffdfb4336fa6bd7.ff"	"ffdfb433ac2e0d1.ff"	"ffdfb433af59264.ff"	"ffdfb433d22c094.ff"
##	[761]	"ffdfb433d67821.ff"	"ffdfb433e57ee79.ff"	"ffdfb433f53a31f.ff"	"ffdfb433f980e9.ff"
##	[765]	"ffdfb433fa27fbd.ff"	"ffdfb4343d5ce8f.ff"	"ffdfb43457071ee.ff"	"ffdfb4346fc6e09.ff"
##	[769]	"ffdfb434754af49.ff"	"ffdfb4348c847bf.ff"	"ffdfb434989bfd4.ff"	"ffdfb434a80b2b5.ff"
##	[773]	"ffdfb434a8d0f44.ff"	"ffdfb434b63c200.ff"	"ffdfb434bb2d827.ff"	"ffdfb434bc1331f.ff"
##	[777]	"ffdfb434c571676.ff"	"ffdfb435035d77a.ff"	"ffdfb43505e0179.ff"	"ffdfb4350faeb69.ff"
##	[781]	"ffdfb4354de71ed.ff"	"ffdfb4355474185.ff"	"ffdfb435809aa22.ff"	"ffdfb435947e3df.ff"
##	[785]	"ffdfb435ef640db.ff"	"ffdfb435f5e4e14.ff"	"ffdfb435fdd9123.ff"	"ffdfb4361c42847.ff"
##	[789]	"ffdfb436280cb55.ff"	"ffdfb4364f00790.ff"	"ffdfb43669c3547.ff"	"ffdfb436983c421.ff"
##	[793]	"ffdfb436c94c868.ff"	"ffdfb436e7f862c.ff"	"ffdfb43717e4308.ff"	"ffdfb43718a1d22.ff"
##	[797]	"ffdfb4372911d03.ff"	"ffdfb4372be0c73.ff"	"ffdfb4372e55f98.ff"	"ffdfb43745cd952.ff"
##	[801]	"ffdfb437568c07e.ff"	"ffdfb4375a56bf8.ff"	"ffdfb437641eb60.ff"	"ffdfb4376d7860f.ff"
##	[805]	"ffdfb4379035f13.ff"	"ffdfb43791c1a91.ff"	"ffdfb4379fa4e11.ff"	"ffdfb437bbfd22c.ff"
##	[809]	"ffdfb437ed5b3b9.ff"	"ffdfb438c00635.ff"	"ffdfb439756408.ff"	"ffdfb439ec3114.ff"
##	[813]	"ffdfb43a77bd.ff"	"ffdfb43aaaa644.ff"	"ffdfb43bfa68b4.ff"	"ffdfb43e283f70.ff"
##	[817]	"ffdfb6c10e4ec46.ff"	"ffdfb6c11266f86.ff"	"ffdfb6c117c569d.ff"	"ffdfb6c11a2d332.ff"
##	[821]	"ffdfb6c12729360.ff"	"ffdfb6c13a11068.ff"	"ffdfb6c13b6ad19.ff"	"ffdfb6c15d7b153.ff"
##	[825]	"ffdfb6c179f5734.ff"	"ffdfb6c1893ac7a.ff"	"ffdfb6c18b911b3.ff"	"ffdfb6c1de9d1ce.ff"
##	[829]	"ffdfb6c1e390b73.ff"	"ffdfb6c1efc0a62.ff"	"ffdfb6c1f2ea297.ff"	"ffdfb6c20298d16.ff"
##	[833]	"ffdfb6c21d69d1d.ff"	"ffdfb6c23b9bf3d.ff"	"ffdfb6c2553f054.ff"	"ffdfb6c26678bfb.ff"
##	[837]	"ffdfb6c267d2b63.ff"	"ffdfb6c26aa6a8e.ff"	"ffdfb6c29f3eeb7.ff"	"ffdfb6c2ad73116.ff"
##	[841]	"ffdfb6c2c4004d0.ff"	"ffdfb6c3092a8d0.ff"	"ffdfb6c31dcfe46.ff"	"ffdfb6c331eba85.ff"
##	[845]	"ffdfb6c333c25e9.ff"	"ffdfb6c34083f6b.ff"	"ffdfb6c350db7d2.ff"	"ffdfb6c3545feb8.ff"
##	[849]	"ffdfb6c37ae72a9.ff"	"ffdfb6c38a0ff43.ff"	"ffdfb6c3bacb59.ff"	"ffdfb6c3df0ea8c.ff"
##	[853]	"ffdfb6c3e1e4e6b.ff"	"ffdfb6c4029c4f8.ff"	"ffdfb6c4489a9f0.ff"	"ffdfb6c4537e542.ff"
##	[857]	"ffdfb6c4a114acd.ff"	"ffdfb6c4bc7f5a.ff"	"ffdfb6c4dbd45e5.ff"	"ffdfb6c4ec50623.ff"
##	[861]	"ffdfb6c4fd0733c.ff"	"ffdfb6c50e72aaa.ff"	"ffdfb6c5150dc15.ff"	"ffdfb6c51d6a4c0.ff"
##	[865]	"ffdfb6c52876b79.ff"	"ffdfb6c52d45b9b.ff"	"ffdfb6c543ac1ed.ff"	"ffdfb6c5610dd97.ff"
##	[869]	"ffdfb6c58171f6f.ff"	"ffdfb6c58ac738e.ff"	"ffdfb6c58d780a9.ff"	"ffdfb6c5b3d3a0.ff"
##	[873]	"ffdfb6c5bf30801.ff"	"ffdfb6c5c7c41f8.ff"	"ffdfb6c5d91d9ce.ff"	"ffdfb6c5e09fb1b.ff"
##	[877]	"ffdfb6c5e162309.ff"	"ffdfb6c610ab04c.ff"	"ffdfb6c6144d9d8.ff"	"ffdfb6c61cd3ab9.ff"
##	[881]	"ffdfb6c61cf3037.ff"	"ffdfb6c62da6339.ff"	"ffdfb6c63ab8452.ff"	"ffdfb6c6522878b.ff"
##	[885]	"ffdfb6c655fd5b.ff"	"ffdfb6c65f37ba5.ff"	"ffdfb6c68ca79d8.ff"	"ffdfb6c68e1dd53.ff"
##	[889]	"ffdfb6c69160c3b.ff"	"ffdfb6c69509a9e.ff"	"ffdfb6c69669d0b.ff"	"ffdfb6c6c890899.ff"
##	[893]	"ffdfb6c6d417b04.ff"	"ffdfb6c6d4410a5.ff"	"ffdfb6c6f3a25cb.ff"	"ffdfb6c6fa865e2.ff"
##	[897]	"ffdfb6c7181fc4e.ff"	"ffdfb6c7208a31.ff"	"ffdfb6c734cf023.ff"	"ffdfb6c73ca4798.ff"
##	[901]	"ffdfb6c764296b0.ff"	"ffdfb6c776c649c.ff"	"ffdfb6c77ba501d.ff"	"ffdfb6c78a22864.ff"
##	[905]	"ffdfb6c7a5c7fb7.ff"	"ffdfb6c7bffb724.ff"	"ffdfb6c7cad422f.ff"	"ffdfb6c7e39a476.ff"
##	[909]	"ffdfb6c7e3d9113.ff"	"ffdfb6c7edafd9.ff"	"ffdfb6c7f565420.ff"	"ffdfb6ccdb9f1a.ff"
##	[913]	"ffdfb6cdb33978.ff"	"ffdfb6cfa0f8ae.ff"	"ffdfbbbe10305529.ff"	"ffdfbbbe142da4e3.ff"
##	[917]	"ffdfbbbe14b45cb3.ff"	"ffdfbbbe14d4c782.ff"	"ffdfbbbe157778f3.ff"	"ffdfbbbe1894908a.ff"
##	[921]	"ffdfbbbe1966ebb.ff"	"ffdfbbbe1bbeb549.ff"	"ffdfbbbe1d76327d.ff"	"ffdfbbbe1e14b451.ff"
##	[925]	"ffdfbbbe1edd9eb5.ff"	"ffdfbbbe1fd81fc3.ff"	"ffdfbbbe1ffeba1b.ff"	"ffdfbbbe260d4b8a.ff"
##	[929]	"ffdfbbbe29c27de8.ff"	"ffdfbbbe2d50f81b.ff"	"ffdfbbbe2ec4d313.ff"	"ffdfbbbe32421dfc.ff"
##	[933]	"ffdfbbbe329a8ef6.ff"	"ffdfbbbe32da648a.ff"	"ffdfbbbe33c99ac6.ff"	"ffdfbbbe35608872.ff"
##	[937]	"ffdfbbbe362f70a6.ff"	"ffdfbbbe37c44068.ff"	"ffdfbbbe39ed99ed.ff"	"ffdfbbbe3aecbba7.ff"

```
## [941] "ffdfbbbe3b3fb7cd.ff" "ffdfbbbe3b70df46.ff" "ffdfbbbe402d0bd0.ff" "ffdfbbbe40a43d6e.ff"
## [945] "ffdfbbbe40a670b1.ff" "ffdfbbbe411fd26e.ff" "ffdfbbbe419d60a1.ff" "ffdfbbbe43330bae.ff"
## [949] "ffdfbbbe43f4c6f7.ff" "ffdfbbbe463f1645.ff" "ffdfbbbe4725e3df.ff" "ffdfbbbe4c43eef9.ff"
## [953] "ffdfbbbe4c4ca33c.ff" "ffdfbbbe4cc8e298.ff" "ffdfbbbe4d31f663.ff" "ffdfbbbe4d62a59a.ff"
## [957] "ffdfbbbe52030942.ff" "ffdfbbbe52fd7896.ff" "ffdfbbbe5413469e.ff" "ffdfbbbe543c121e.ff"
## [961] "ffdfbbbe549e0fbe.ff" "ffdfbbbe551bae58.ff" "ffdfbbbe5526ff0c.ff" "ffdfbbbe56a5a487.ff"
## [965] "ffdfbbbe56fd9da.ff" "ffdfbbbe578fc154.ff" "ffdfbbbe57cf300a.ff" "ffdfbbbe5eaf32c0.ff"
## [969] "ffdfbbbe5ebd21e9.ff" "ffdfbbbe614a4144.ff" "ffdfbbbe61bc3ac8.ff" "ffdfbbbe635b0d8f.ff"
## [973] "ffdfbbbe648cbe14.ff" "ffdfbbbe67de8ce7.ff" "ffdfbbbe6c276335.ff" "ffdfbbbe6ed75a3e.ff"
## [977] "ffdfbbbe6f30959f.ff" "ffdfbbbe6f528b26.ff" "ffdfbbbe70bab497.ff" "ffdfbbbe71a65f52.ff"
## [981] "ffdfbbbe724779ff.ff" "ffdfbbbe73329880.ff" "ffdfbbbe7355b132.ff" "ffdfbbbe7664927.ff"
## [985] "ffdfbbbe78d057b9.ff" "ffdfbbbe7b32a4fb.ff" "ffdfbbbe7bfd9c61.ff" "ffdfbbbe7c65d54b.ff"
## [989] "ffdfbbbe7eba3695.ff" "ffdfbbbe7ee484c5.ff" "ffdfbbbe7ef7ea6a.ff" "ffdfbbbe7ffe448c.ff"
## [993] "ffdfbbbe89369f7.ff" "ffdfbbbea1656f3.ff" "ffdfbbbeb27052d.ff" "ffdfbbbec13ef2a.ff"
## [997] "ffdfbbbec533f0f.ff" "ffdfbbbec6c5306.ff" "ffdfbbbecb75299.ff" "ffdfbbbee4684fa.ff"
```

```
# investigate the structure of the object created in the R environment
str(flights)
```

```
## List of 3
## $ virtual: 'data.frame': 19 obs. of 7 variables:
## .. $ VirtualVmode : chr "integer" "integer" "integer" "integer" ...
## .. $ AsIs : logi FALSE FALSE FALSE FALSE FALSE FALSE ...
## .. $ VirtualIsMatrix : logi FALSE FALSE FALSE FALSE FALSE FALSE ...
## .. $ PhysicalIsMatrix : logi FALSE FALSE FALSE FALSE FALSE FALSE ...
## .. $ PhysicalElementNo: int 1 2 3 4 5 6 7 8 9 10 ...
## .. $ PhysicalFirstCol : int 1 1 1 1 1 1 1 1 1 1 ...
## .. $ PhysicalLastCol : int 1 1 1 1 1 1 1 1 1 1 ...
## .. - attr(*, "Dim")= int 336776 19
## .. - attr(*, "Dimorder")= int 1 2
## $ physical: List of 19
## .. $ year : list()
## .. ..- attr(*, "physical")=Class 'ff_pointer' <externalptr>
## .. ..- attr(*, "vmode")= chr "integer"
## .. ..- attr(*, "maxlength")= int 336776
## .. ..- attr(*, "pattern")= chr "ffdf"
## .. ..- attr(*, "filename")= chr "/Users/umatter/Dropbox/Teaching/HSG/BigData/BigData/materials/"
## .. ..- attr(*, "pagesize")= int 65536
## .. ..- attr(*, "finalizer")= chr "close"
## .. ..- attr(*, "finonexit")= logi TRUE
## .. ..- attr(*, "readonly")= logi FALSE
## .. ..- attr(*, "caching")= chr "mmnoflush"
## .. ..- attr(*, "virtual")= list()
## .. ..- attr(*, "Length")= int 336776
## .. ..- attr(*, "Symmetric")= logi FALSE
## .. ..- attr(*, "class")= chr "virtual"
## .. ..- attr(*, "class")= chr [1:2] "ff_vector" "ff"
## .. $ month : list()
## .. ..- attr(*, "physical")=Class 'ff_pointer' <externalptr>
## .. ..- attr(*, "vmode")= chr "integer"
## .. ..- attr(*, "maxlength")= int 336776
## .. ..- attr(*, "pattern")= chr "ffdf"
## .. ..- attr(*, "filename")= chr "/Users/umatter/Dropbox/Teaching/HSG/BigData/BigData/materials/"
## .. ..- attr(*, "pagesize")= int 65536
## .. ..- attr(*, "finalizer")= chr "close"
```

```

## .. ..- attr(*, "finonexit")= logi TRUE
## .. ..- attr(*, "readonly")= logi FALSE
## .. ..- attr(*, "caching")= chr "mmnoflush"
## .. ..- attr(*, "virtual")= list()
## .. ..- attr(*, "Length")= int 336776
## .. ..- attr(*, "Symmetric")= logi FALSE
## .. ..- attr(*, "class")= chr "virtual"
## .. ..- attr(*, "class") = chr [1:2] "ff_vector" "ff"
## .. $ day : list()
## .. ..- attr(*, "physical")=Class 'ff_pointer' <externalptr>
## .. ..- attr(*, "vmode")= chr "integer"
## .. ..- attr(*, "maxlength")= int 336776
## .. ..- attr(*, "pattern")= chr "ffdf"
## .. ..- attr(*, "filename")= chr "/Users/umatter/Dropbox/Teaching/HSG/BigData/BigData/materials/"
## .. ..- attr(*, "pagesize")= int 65536
## .. ..- attr(*, "finalizer")= chr "close"
## .. ..- attr(*, "finonexit")= logi TRUE
## .. ..- attr(*, "readonly")= logi FALSE
## .. ..- attr(*, "caching")= chr "mmnoflush"
## .. ..- attr(*, "virtual")= list()
## .. ..- attr(*, "Length")= int 336776
## .. ..- attr(*, "Symmetric")= logi FALSE
## .. ..- attr(*, "class")= chr "virtual"
## .. ..- attr(*, "class") = chr [1:2] "ff_vector" "ff"
## .. $ dep_time : list()
## .. ..- attr(*, "physical")=Class 'ff_pointer' <externalptr>
## .. ..- attr(*, "vmode")= chr "integer"
## .. ..- attr(*, "maxlength")= int 336776
## .. ..- attr(*, "pattern")= chr "ffdf"
## .. ..- attr(*, "filename")= chr "/Users/umatter/Dropbox/Teaching/HSG/BigData/BigData/materials/"
## .. ..- attr(*, "pagesize")= int 65536
## .. ..- attr(*, "finalizer")= chr "close"
## .. ..- attr(*, "finonexit")= logi TRUE
## .. ..- attr(*, "readonly")= logi FALSE
## .. ..- attr(*, "caching")= chr "mmnoflush"
## .. ..- attr(*, "virtual")= list()
## .. ..- attr(*, "Length")= int 336776
## .. ..- attr(*, "Symmetric")= logi FALSE
## .. ..- attr(*, "class")= chr "virtual"
## .. ..- attr(*, "class") = chr [1:2] "ff_vector" "ff"
## .. $ sched_dep_time: list()
## .. ..- attr(*, "physical")=Class 'ff_pointer' <externalptr>
## .. ..- attr(*, "vmode")= chr "integer"
## .. ..- attr(*, "maxlength")= int 336776
## .. ..- attr(*, "pattern")= chr "ffdf"
## .. ..- attr(*, "filename")= chr "/Users/umatter/Dropbox/Teaching/HSG/BigData/BigData/materials/"
## .. ..- attr(*, "pagesize")= int 65536
## .. ..- attr(*, "finalizer")= chr "close"
## .. ..- attr(*, "finonexit")= logi TRUE
## .. ..- attr(*, "readonly")= logi FALSE
## .. ..- attr(*, "caching")= chr "mmnoflush"
## .. ..- attr(*, "virtual")= list()
## .. ..- attr(*, "Length")= int 336776
## .. ..- attr(*, "Symmetric")= logi FALSE

```



```

## .. ..- attr(*, "class")= chr "virtual"
## .. ..- attr(*, "class") = chr [1:2] "ff_vector" "ff"
## .. $ dep_delay      : list()
## .. ..- attr(*, "physical")=Class 'ff_pointer' <externalptr>
## .. ..- attr(*, "vmode")= chr "integer"
## .. ..- attr(*, "maxlength")= int 336776
## .. ..- attr(*, "pattern")= chr "ffdf"
## .. ..- attr(*, "filename")= chr "/Users/umatter/Dropbox/Teaching/HSG/BigData/BigData/materials/"
## .. ..- attr(*, "pagesize")= int 65536
## .. ..- attr(*, "finalizer")= chr "close"
## .. ..- attr(*, "finonexit")= logi TRUE
## .. ..- attr(*, "readonly")= logi FALSE
## .. ..- attr(*, "caching")= chr "mmnoflush"
## .. ..- attr(*, "virtual")= list()
## .. ..- attr(*, "Length")= int 336776
## .. ..- attr(*, "Symmetric")= logi FALSE
## .. ..- attr(*, "class")= chr "virtual"
## .. ..- attr(*, "class") = chr [1:2] "ff_vector" "ff"
## .. $ arr_time      : list()
## .. ..- attr(*, "physical")=Class 'ff_pointer' <externalptr>
## .. ..- attr(*, "vmode")= chr "integer"
## .. ..- attr(*, "maxlength")= int 336776
## .. ..- attr(*, "pattern")= chr "ffdf"
## .. ..- attr(*, "filename")= chr "/Users/umatter/Dropbox/Teaching/HSG/BigData/BigData/materials/"
## .. ..- attr(*, "pagesize")= int 65536
## .. ..- attr(*, "finalizer")= chr "close"
## .. ..- attr(*, "finonexit")= logi TRUE
## .. ..- attr(*, "readonly")= logi FALSE
## .. ..- attr(*, "caching")= chr "mmnoflush"
## .. ..- attr(*, "virtual")= list()
## .. ..- attr(*, "Length")= int 336776
## .. ..- attr(*, "Symmetric")= logi FALSE
## .. ..- attr(*, "class")= chr "virtual"
## .. ..- attr(*, "class") = chr [1:2] "ff_vector" "ff"
## .. $ sched_arr_time: list()
## .. ..- attr(*, "physical")=Class 'ff_pointer' <externalptr>
## .. ..- attr(*, "vmode")= chr "integer"
## .. ..- attr(*, "maxlength")= int 336776
## .. ..- attr(*, "pattern")= chr "ffdf"
## .. ..- attr(*, "filename")= chr "/Users/umatter/Dropbox/Teaching/HSG/BigData/BigData/materials/"
## .. ..- attr(*, "pagesize")= int 65536
## .. ..- attr(*, "finalizer")= chr "close"
## .. ..- attr(*, "finonexit")= logi TRUE
## .. ..- attr(*, "readonly")= logi FALSE
## .. ..- attr(*, "caching")= chr "mmnoflush"
## .. ..- attr(*, "virtual")= list()
## .. ..- attr(*, "Length")= int 336776
## .. ..- attr(*, "Symmetric")= logi FALSE
## .. ..- attr(*, "class")= chr "virtual"
## .. ..- attr(*, "class") = chr [1:2] "ff_vector" "ff"
## .. $ arr_delay      : list()
## .. ..- attr(*, "physical")=Class 'ff_pointer' <externalptr>
## .. ..- attr(*, "vmode")= chr "integer"
## .. ..- attr(*, "maxlength")= int 336776

```

```

## .. ..- attr(*, "pattern")= chr "ffdf"
## .. ..- attr(*, "filename")= chr "/Users/umatter/Dropbox/Teaching/HSG/BigData/BigData/materials/"
## .. ..- attr(*, "pagesize")= int 65536
## .. ..- attr(*, "finalizer")= chr "close"
## .. ..- attr(*, "finonexit")= logi TRUE
## .. ..- attr(*, "readonly")= logi FALSE
## .. ..- attr(*, "caching")= chr "mmnoflush"
## .. ..- attr(*, "virtual")= list()
## .. ..- attr(*, "Length")= int 336776
## .. ..- attr(*, "Symmetric")= logi FALSE
## .. ..- attr(*, "class")= chr "virtual"
## .. ..- attr(*, "class") = chr [1:2] "ff_vector" "ff"
## .. $ carrier      : list()
## .. ..- attr(*, "physical")=Class 'ff_pointer' <externalptr>
## .. ..- attr(*, "vmode")= chr "integer"
## .. ..- attr(*, "maxlength")= int 336776
## .. ..- attr(*, "pattern")= chr "ffdf"
## .. ..- attr(*, "filename")= chr "/Users/umatter/Dropbox/Teaching/HSG/BigData/BigData/materials/"
## .. ..- attr(*, "pagesize")= int 65536
## .. ..- attr(*, "finalizer")= chr "close"
## .. ..- attr(*, "finonexit")= logi TRUE
## .. ..- attr(*, "readonly")= logi FALSE
## .. ..- attr(*, "caching")= chr "mmnoflush"
## .. ..- attr(*, "virtual")= list()
## .. ..- attr(*, "Length")= int 336776
## .. ..- attr(*, "Symmetric")= logi FALSE
## .. ..- attr(*, "Levels")= chr [1:16] "9E" "AA" "AS" "B6" ...
## .. ..- attr(*, "ramclass")= chr "factor"
## .. ..- attr(*, "class")= chr "virtual"
## .. ..- attr(*, "class") = chr [1:2] "ff_vector" "ff"
## .. $ flight      : list()
## .. ..- attr(*, "physical")=Class 'ff_pointer' <externalptr>
## .. ..- attr(*, "vmode")= chr "integer"
## .. ..- attr(*, "maxlength")= int 336776
## .. ..- attr(*, "pattern")= chr "ffdf"
## .. ..- attr(*, "filename")= chr "/Users/umatter/Dropbox/Teaching/HSG/BigData/BigData/materials/"
## .. ..- attr(*, "pagesize")= int 65536
## .. ..- attr(*, "finalizer")= chr "close"
## .. ..- attr(*, "finonexit")= logi TRUE
## .. ..- attr(*, "readonly")= logi FALSE
## .. ..- attr(*, "caching")= chr "mmnoflush"
## .. ..- attr(*, "virtual")= list()
## .. ..- attr(*, "Length")= int 336776
## .. ..- attr(*, "Symmetric")= logi FALSE
## .. ..- attr(*, "class")= chr "virtual"
## .. ..- attr(*, "class") = chr [1:2] "ff_vector" "ff"
## .. $ tailnum     : list()
## .. ..- attr(*, "physical")=Class 'ff_pointer' <externalptr>
## .. ..- attr(*, "vmode")= chr "integer"
## .. ..- attr(*, "maxlength")= int 336776
## .. ..- attr(*, "pattern")= chr "ffdf"
## .. ..- attr(*, "filename")= chr "/Users/umatter/Dropbox/Teaching/HSG/BigData/BigData/materials/"
## .. ..- attr(*, "pagesize")= int 65536
## .. ..- attr(*, "finalizer")= chr "close"

```

```

## .. ..- attr(*, "finonexit")= logi TRUE
## .. ..- attr(*, "readonly")= logi FALSE
## .. ..- attr(*, "caching")= chr "mmnoflush"
## .. ..- attr(*, "virtual")= list()
## .. ..- attr(*, "Length")= int 336776
## .. ..- attr(*, "Symmetric")= logi FALSE
## .. ..- attr(*, "Levels")= chr [1:4044] "" "NOEGMQ" "N10156" "N102UW" ...
## .. ..- attr(*, "ramclass")= chr "factor"
## .. ..- attr(*, "class")= chr "virtual"
## .. ..- attr(*, "class") = chr [1:2] "ff_vector" "ff"
## .. $ origin      : list()
## .. ..- attr(*, "physical")=Class 'ff_pointer' <externalptr>
## .. ..- attr(*, "vmode")= chr "integer"
## .. ..- attr(*, "maxlength")= int 336776
## .. ..- attr(*, "pattern")= chr "ffdf"
## .. ..- attr(*, "filename")= chr "/Users/umatter/Dropbox/Teaching/HSG/BigData/BigData/materials/"
## .. ..- attr(*, "pagesize")= int 65536
## .. ..- attr(*, "finalizer")= chr "close"
## .. ..- attr(*, "finonexit")= logi TRUE
## .. ..- attr(*, "readonly")= logi FALSE
## .. ..- attr(*, "caching")= chr "mmnoflush"
## .. ..- attr(*, "virtual")= list()
## .. ..- attr(*, "Length")= int 336776
## .. ..- attr(*, "Symmetric")= logi FALSE
## .. ..- attr(*, "Levels")= chr [1:3] "EWR" "JFK" "LGA"
## .. ..- attr(*, "ramclass")= chr "factor"
## .. ..- attr(*, "class")= chr "virtual"
## .. ..- attr(*, "class") = chr [1:2] "ff_vector" "ff"
## .. $ dest        : list()
## .. ..- attr(*, "physical")=Class 'ff_pointer' <externalptr>
## .. ..- attr(*, "vmode")= chr "integer"
## .. ..- attr(*, "maxlength")= int 336776
## .. ..- attr(*, "pattern")= chr "ffdf"
## .. ..- attr(*, "filename")= chr "/Users/umatter/Dropbox/Teaching/HSG/BigData/BigData/materials/"
## .. ..- attr(*, "pagesize")= int 65536
## .. ..- attr(*, "finalizer")= chr "close"
## .. ..- attr(*, "finonexit")= logi TRUE
## .. ..- attr(*, "readonly")= logi FALSE
## .. ..- attr(*, "caching")= chr "mmnoflush"
## .. ..- attr(*, "virtual")= list()
## .. ..- attr(*, "Length")= int 336776
## .. ..- attr(*, "Symmetric")= logi FALSE
## .. ..- attr(*, "Levels")= chr [1:105] "ABQ" "ACK" "ALB" "ATL" ...
## .. ..- attr(*, "ramclass")= chr "factor"
## .. ..- attr(*, "class")= chr "virtual"
## .. ..- attr(*, "class") = chr [1:2] "ff_vector" "ff"
## .. $ air_time    : list()
## .. ..- attr(*, "physical")=Class 'ff_pointer' <externalptr>
## .. ..- attr(*, "vmode")= chr "integer"
## .. ..- attr(*, "maxlength")= int 336776
## .. ..- attr(*, "pattern")= chr "ffdf"
## .. ..- attr(*, "filename")= chr "/Users/umatter/Dropbox/Teaching/HSG/BigData/BigData/materials/"
## .. ..- attr(*, "pagesize")= int 65536
## .. ..- attr(*, "finalizer")= chr "close"

```

```

## .. ..- attr(*, "finonexit")= logi TRUE
## .. ..- attr(*, "readonly")= logi FALSE
## .. ..- attr(*, "caching")= chr "mmnoflush"
## .. ..- attr(*, "virtual")= list()
## .. ..- attr(*, "Length")= int 336776
## .. ..- attr(*, "Symmetric")= logi FALSE
## .. ..- attr(*, "class")= chr "virtual"
## .. ..- attr(*, "class") = chr [1:2] "ff_vector" "ff"
## .. $ distance      : list()
## .. ..- attr(*, "physical")=Class 'ff_pointer' <externalptr>
## .. ..- attr(*, "vmode")= chr "integer"
## .. ..- attr(*, "maxlength")= int 336776
## .. ..- attr(*, "pattern")= chr "ffdf"
## .. ..- attr(*, "filename")= chr "/Users/umatter/Dropbox/Teaching/HSG/BigData/BigData/materials/"
## .. ..- attr(*, "pagesize")= int 65536
## .. ..- attr(*, "finalizer")= chr "close"
## .. ..- attr(*, "finonexit")= logi TRUE
## .. ..- attr(*, "readonly")= logi FALSE
## .. ..- attr(*, "caching")= chr "mmnoflush"
## .. ..- attr(*, "virtual")= list()
## .. ..- attr(*, "Length")= int 336776
## .. ..- attr(*, "Symmetric")= logi FALSE
## .. ..- attr(*, "class")= chr "virtual"
## .. ..- attr(*, "class") = chr [1:2] "ff_vector" "ff"
## .. $ hour          : list()
## .. ..- attr(*, "physical")=Class 'ff_pointer' <externalptr>
## .. ..- attr(*, "vmode")= chr "integer"
## .. ..- attr(*, "maxlength")= int 336776
## .. ..- attr(*, "pattern")= chr "ffdf"
## .. ..- attr(*, "filename")= chr "/Users/umatter/Dropbox/Teaching/HSG/BigData/BigData/materials/"
## .. ..- attr(*, "pagesize")= int 65536
## .. ..- attr(*, "finalizer")= chr "close"
## .. ..- attr(*, "finonexit")= logi TRUE
## .. ..- attr(*, "readonly")= logi FALSE
## .. ..- attr(*, "caching")= chr "mmnoflush"
## .. ..- attr(*, "virtual")= list()
## .. ..- attr(*, "Length")= int 336776
## .. ..- attr(*, "Symmetric")= logi FALSE
## .. ..- attr(*, "class")= chr "virtual"
## .. ..- attr(*, "class") = chr [1:2] "ff_vector" "ff"
## .. $ minute        : list()
## .. ..- attr(*, "physical")=Class 'ff_pointer' <externalptr>
## .. ..- attr(*, "vmode")= chr "integer"
## .. ..- attr(*, "maxlength")= int 336776
## .. ..- attr(*, "pattern")= chr "ffdf"
## .. ..- attr(*, "filename")= chr "/Users/umatter/Dropbox/Teaching/HSG/BigData/BigData/materials/"
## .. ..- attr(*, "pagesize")= int 65536
## .. ..- attr(*, "finalizer")= chr "close"
## .. ..- attr(*, "finonexit")= logi TRUE
## .. ..- attr(*, "readonly")= logi FALSE
## .. ..- attr(*, "caching")= chr "mmnoflush"
## .. ..- attr(*, "virtual")= list()
## .. ..- attr(*, "Length")= int 336776
## .. ..- attr(*, "Symmetric")= logi FALSE

```

```
## .. ..- attr(*, "class")= chr "virtual"
## .. .. - attr(*, "class") = chr [1:2] "ff_vector" "ff"
## .. $ time_hour      : list()
## .. ..- attr(*, "physical")=Class 'ff_pointer' <externalptr>
## .. ..- attr(*, "vmode")= chr "integer"
## .. ..- attr(*, "maxlength")= int 336776
## .. ..- attr(*, "pattern")= chr "ffdf"
## .. ..- attr(*, "filename")= chr "/Users/umatter/Dropbox/Teaching/HSG/BigData/BigData/materials/"
## .. ..- attr(*, "pagesize")= int 65536
## .. ..- attr(*, "finalizer")= chr "close"
## .. ..- attr(*, "finonexit")= logi TRUE
## .. ..- attr(*, "readonly")= logi FALSE
## .. ..- attr(*, "caching")= chr "mmnoflush"
## .. ..- attr(*, "virtual")= list()
## .. ..- attr(*, "Length")= int 336776
## .. ..- attr(*, "Symmetric")= logi FALSE
## .. ..- attr(*, "Levels")= chr [1:6936] "2013-01-01T10:00:00Z" "2013-01-01T11:00:00Z" "2013-01-01T12:00:00Z"
## .. ..- attr(*, "ramclass")= chr "factor"
## .. ..- attr(*, "class")= chr "virtual"
## .. .. - attr(*, "class") = chr [1:2] "ff_vector" "ff"
## $ row.names:  NULL
## - attributes: List of 2
## .. $ names: chr [1:3] "virtual" "physical" "row.names"
## .. $ class: chr "ffdf"
```

### 1.1.2 Memory mapping with bigmemory

The `bigmemory`-package handles data in matrices, and therefore only accepts variables in the same data type. Before importing data via the `bigmemory`-package, we thus have to ensure that all variables in the raw data can be imported in a common type. This example follows the example of the package authors given here.<sup>1</sup>

```
# SET UP -----

# load packages
library(bigmemory)
library(biganalytics)

# import the data
flights <- read.big.matrix("../data/flights.csv",
  type="integer",
  header=TRUE,
  backingfile="flights.bin",
  descriptorfile="flights.desc")
```

Note that, similar to the `ff`-example, `read.big.matrix()` initiates a local file-backing `flights.bin` on disk which is linked to the `flights`-object in RAM. From looking at the imported file, we see that various variable values have been discarded. This is due to the fact that we have forced all variables to be of type `"integer"` when importing the dataset.

```
summary(flights)
```

```
##               min               max               mean               NAs
## year          2013.000000    2013.000000    2013.000000         0.000000
```

<sup>1</sup>We only use a fraction of the data used in the package vignette example, the full raw data used there can be downloaded [here](#).

## month	1.000000	12.000000	6.548510	0.000000
## day	1.000000	31.000000	15.710787	0.000000
## dep_time	1.000000	2400.000000	1349.109947	8255.000000
## sched_dep_time	106.000000	2359.000000	1344.254840	0.000000
## dep_delay	-43.000000	1301.000000	12.639070	8255.000000
## arr_time	1.000000	2400.000000	1502.054999	8713.000000
## sched_arr_time	1.000000	2359.000000	1536.380220	0.000000
## arr_delay	-86.000000	1272.000000	6.895377	9430.000000
## carrier	9.000000	9.000000	9.000000	318316.000000
## flight	1.000000	8500.000000	1971.923620	0.000000
## tailnum				336776.000000
## origin				336776.000000
## dest				336776.000000
## air_time	20.000000	695.000000	150.686460	9430.000000
## distance	17.000000	4983.000000	1039.912604	0.000000
## hour	1.000000	23.000000	13.180247	0.000000
## minute	0.000000	59.000000	26.230100	0.000000
## time_hour	2013.000000	2014.000000	2013.000261	0.000000

## 2 Cleaning and Transformation

### 2.1 Typical tasks (independent of data set size)

- Normalize/standardize.
- Code additional variables (indicators, strings to categorical, etc.).
- Remove, add covariates.
- Merge data sets.
- Set data types.

### 2.2 Typical workflow

1. Import raw data.
2. Clean/transform.
3. Store for analysis.
  - Write to file.
  - Write to database.

### 2.3 Bottlenecks

- RAM:
  - Raw data does not fit into memory.
  - Transformations enlarge RAM allocation (copying).
- Mass Storage: Reading/Writing
- CPU: Parsing (data types)

## 3 Data Preparation with ff

### 3.1 Set up

The following examples are based on Walkowiak (2016), Chapter 3.

```
## SET UP -----

#Set working directory to the data and airline_id files.
# setwd("materials/code_book/B05396_Ch03_Code")
system("mkdir ffd")
options(fftempdir = "ffd")

# load packages
library(ff)
library(ffbase)
library(pryr)

# fix vars
FLIGHTS_DATA <- "../code_book/B05396_Ch03_Code/flights_sep_oct15.txt"
AIRLINES_DATA <- "../code_book/B05396_Ch03_Code/airline_id.csv"
```

### 3.2 Data import

```
# DATA IMPORT -----

# 1. Upload flights_sep_oct15.txt and airline_id.csv files from flat files.

system.time(flights.ff <- read.table.ffdf(file=FLIGHTS_DATA,
                                           sep=",",
                                           VERBOSE=TRUE,
                                           header=TRUE,
                                           next.rows=100000,
                                           colClasses=NA))

## read.table.ffdf 1..100000 (100000)  csv-read=0.961sec ffd-write=0.186sec
## read.table.ffdf 100001..200000 (100000)  csv-read=1.42sec ffd-write=0.13sec
## read.table.ffdf 200001..300000 (100000)  csv-read=1.107sec ffd-write=0.143sec
## read.table.ffdf 300001..400000 (100000)  csv-read=1.178sec ffd-write=0.114sec
## read.table.ffdf 400001..500000 (100000)  csv-read=1.255sec ffd-write=0.096sec
## read.table.ffdf 500001..600000 (100000)  csv-read=1.238sec ffd-write=0.128sec
## read.table.ffdf 600001..700000 (100000)  csv-read=1.29sec ffd-write=0.114sec
## read.table.ffdf 700001..800000 (100000)  csv-read=1.419sec ffd-write=0.103sec
## read.table.ffdf 800001..900000 (100000)  csv-read=1.154sec ffd-write=0.108sec
## read.table.ffdf 900001..951111 (51111)  csv-read=0.701sec ffd-write=0.1sec
##  csv-read=11.723sec ffd-write=1.222sec  TOTAL=12.945sec

##   user  system elapsed
## 10.980   0.779  12.953

airlines.ff <- read.csv.ffdf(file= AIRLINES_DATA,
                             VERBOSE=TRUE,
                             header=TRUE,
```

```

next.rows=100000,
colClasses=NA)

## read.table.ffdf 1..1607 (1607) csv-read=0.016sec ffdf-write=0.009sec
## csv-read=0.016sec ffdf-write=0.009sec TOTAL=0.025sec

# check memory used
mem_used()

## 1.33 GB

```

### 3.3 Comparison with read.table

```

##Using read.table()
system.time(flights.table <- read.table(FLIGHTS_DATA,
                                         sep="," ,
                                         header=TRUE))

##      user  system elapsed
##    9.036    0.496    9.861

gc()

##           used   (Mb) gc trigger   (Mb) limit (Mb) max used   (Mb)
## Ncells   1309358   70.0  4518625   241.4      NA   3183030   170.0
## Vcells 157045432 1198.2 256657206 1958.2    16384 256657206 1958.2

system.time(airlines.table <- read.csv(AIRLINES_DATA,
                                         header = TRUE))

##      user  system elapsed
##    0.006    0.000    0.007

# check memory used
mem_used()

## 1.33 GB

```

### 3.4 Inspect imported files

```

# 2. Inspect the ffdf objects.
## For flights.ff object:
class(flights.ff)

## [1] "ffdf"

dim(flights.ff)

## [1] 951111    28
## For airlines.ff object:
class(airlines.ff)

## [1] "ffdf"

dim(airlines.ff)

## [1] 1607     2

```



### 3.5 Data cleaning and transformation

Goal: merge airline data to flights data

```
# step 1:
## Rename "Code" variable from airlines.ff to "AIRLINE_ID" and "Description" into "AIRLINE_NM".
names(airlines.ff) <- c("AIRLINE_ID", "AIRLINE_NM")
names(airlines.ff)

## [1] "AIRLINE_ID" "AIRLINE_NM"

str(airlines.ff[1:20,])

## 'data.frame':    20 obs. of  2 variables:
## $ AIRLINE_ID: int  19031 19032 19033 19034 19035 19036 19037 19038 19039 19040 ...
## $ AIRLINE_NM: Factor w/ 1607 levels "40-Mile Air: Q5",...: 945 1025 503 721 64 725 1194 99 1395 276
```

### 3.6 Data cleaning and transformation

Goal: merge airline data to flights data

```
# merge of ffd and ffd objects
mem_change(flights.data.ff <- merge.ffdf(flights.ff, airlines.ff, by="AIRLINE_ID"))

## -26.2 kB

##The new object is only 551.2 Kb in size
class(flights.data.ff)

## [1] "ffdf"

dim(flights.data.ff)

## [1] 951111      29

dimnames.ffdf(flights.data.ff)

## [[1]]
## NULL
##
## [[2]]
## [1] "YEAR"           "MONTH"           "DAY_OF_MONTH"    "DAY_OF_WEEK"
## [5] "FL_DATE"        "UNIQUE_CARRIER" "AIRLINE_ID"      "TAIL_NUM"
## [9] "FL_NUM"         "ORIGIN_AIRPORT_ID" "ORIGIN"          "ORIGIN_CITY_NAME"
## [13] "ORIGIN_STATE_NM" "ORIGIN_WAC"       "DEST_AIRPORT_ID" "DEST"
## [17] "DEST_CITY_NAME" "DEST_STATE_NM"    "DEST_WAC"        "DEP_TIME"
## [21] "DEP_DELAY"      "ARR_TIME"         "ARR_DELAY"       "CANCELLED"
## [25] "CANCELLATION_CODE" "DIVERTED"         "AIR_TIME"        "DISTANCE"
## [29] "AIRLINE_NM"
```

### 3.7 Inspect difference to in-memory operation

```
##For flights.table:
names(airlines.table) <- c("AIRLINE_ID", "AIRLINE_NM")
names(airlines.table)

## [1] "AIRLINE_ID" "AIRLINE_NM"
```

```
str(airlines.table[1:20,])

## 'data.frame': 20 obs. of 2 variables:
## $ AIRLINE_ID: int 19031 19032 19033 19034 19035 19036 19037 19038 19039 19040 ...
## $ AIRLINE_NM: Factor w/ 1607 levels "40-Mile Air: Q5",...: 945 1025 503 721 64 725 1194 99 1395 276

# check memory usage of merge in RAM
mem_change(flights.data.table <- merge(flights.table,
                                       airlines.table,
                                       by="AIRLINE_ID"))

## -59 kB

#The new object is already 105.7 Mb in size
#A rapid spike in RAM use when processing
```

### 3.8 Type conversion: ff factor

```
# Inspect the current variable
table.ff(flights.data.ff$DAY_OF_WEEK)

##
##      1      2      3      4      5      6      7
## 131267 143057 145057 148544 148159 112239 122788

head(flights.data.ff$DAY_OF_WEEK)

## [1] 1 1 1 1 1 1

# Convert numeric ff DAY_OF_WEEK vector to a ff factor:
flights.data.ff$WEEKDAY <- cut.ff(flights.data.ff$DAY_OF_WEEK,
                                breaks = 7,
                                labels = c("Monday", "Tuesday",
                                           "Wednesday", "Thursday",
                                           "Friday", "Saturday",
                                           "Sunday"))

# inspect the result
head(flights.data.ff$WEEKDAY)

## [1] Monday Monday Monday Monday Monday Monday
## Levels: Monday Tuesday Wednesday Thursday Friday Saturday Sunday

table.ff(flights.data.ff$WEEKDAY)

##
##      Monday      Tuesday Wednesday      Thursday      Friday      Saturday      Sunday
##      131267      143057      145057      148544      148159      112239      122788
```

### 3.9 Subsetting

```
mem_used()

## 1.33 GB
```

```
# Subset the ffdf object flights.data.ff:
subs1.ff <- subset.ffdf(flights.data.ff, CANCELLED == 1,
                        select = c(FL_DATE, AIRLINE_ID,
                                   ORIGIN_CITY_NAME,
                                   ORIGIN_STATE_NM,
                                   DEST_CITY_NAME,
                                   DEST_STATE_NM,
                                   CANCELLATION_CODE))
```

```
dim(subs1.ff)
```

```
## [1] 4529    7
```

```
mem_used()
```

```
## 1.33 GB
```

### 3.10 Save to ffdf-files

(For further processing with ff)

```
# Save a newly created ffdf object to a data file:
```

```
save.ffdf(subs1.ff, overwrite = TRUE) #7 files (one for each column) created in the ffdb directory
```

### 3.11 Load ffdf-files

```
# Loading previously saved ffdf files:
```

```
rm(subs1.ff)
```

```
gc()
```

```
##           used   (Mb) gc trigger   (Mb) limit (Mb) max used   (Mb)
## Ncells  1309134  70.0   4518625   241.4      NA   3230054  172.6
## Vcells 157031788 1198.1 256657206 1958.2    16384 256657206 1958.2
```

```
load.ffdf("ffdb")
```

```
str(subs1.ff)
```

```
## List of 3
## $ virtual: 'data.frame':   7 obs. of  7 variables:
## .. $ VirtualVmode      : chr  "integer" "integer" "integer" "integer" ...
## .. $ AsIs              : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
## .. $ VirtualIsMatrix   : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
## .. $ PhysicalIsMatrix  : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
## .. $ PhysicalElementNo: int   1 2 3 4 5 6 7
## .. $ PhysicalFirstCol  : int   1 1 1 1 1 1 1
## .. $ PhysicalLastCol   : int   1 1 1 1 1 1 1
## .. - attr(*, "Dim")= int   4529 7
## .. - attr(*, "Dimorder")= int   1 2
## $ physical: List of 7
## .. $ FL_DATE           : list()
## .. ..- attr(*, "physical")=Class 'ff_pointer' <externalptr>
## .. ..- attr(*, "vmode")= chr  "integer"
## .. ..- attr(*, "maxlength")= int 4529
```

```

## .. ..- attr(*, "pattern")= chr "ffdf"
## .. ..- attr(*, "filename")= chr "/Users/umatter/Dropbox/Teaching/HSG/BigData/BigData/materials/"
## .. ..- attr(*, "pagesize")= int 65536
## .. ..- attr(*, "finalizer")= chr "close"
## .. ..- attr(*, "finonexit")= logi TRUE
## .. ..- attr(*, "readonly")= logi FALSE
## .. ..- attr(*, "caching")= chr "mmnoflush"
## .. ..- attr(*, "virtual")= list()
## .. ..- attr(*, "Length")= int 4529
## .. ..- attr(*, "Symmetric")= logi FALSE
## .. ..- attr(*, "Levels")= chr [1:61] "2015-09-01" "2015-09-02" "2015-09-03" "2015-09-04" ...
## .. ..- attr(*, "ramclass")= chr "factor"
## .. ..- attr(*, "class") = chr [1:2] "ff_vector" "ff"
## .. $ AIRLINE_ID : list()
## .. ..- attr(*, "physical")=Class 'ff_pointer' <externalptr>
## .. ..- attr(*, "vmode")= chr "integer"
## .. ..- attr(*, "maxlength")= int 4529
## .. ..- attr(*, "pattern")= chr "ffdf"
## .. ..- attr(*, "filename")= chr "/Users/umatter/Dropbox/Teaching/HSG/BigData/BigData/materials/"
## .. ..- attr(*, "pagesize")= int 65536
## .. ..- attr(*, "finalizer")= chr "close"
## .. ..- attr(*, "finonexit")= logi TRUE
## .. ..- attr(*, "readonly")= logi FALSE
## .. ..- attr(*, "caching")= chr "mmnoflush"
## .. ..- attr(*, "virtual")= list()
## .. ..- attr(*, "Length")= int 4529
## .. ..- attr(*, "Symmetric")= logi FALSE
## .. ..- attr(*, "class") = chr [1:2] "ff_vector" "ff"
## .. $ ORIGIN_CITY_NAME : list()
## .. ..- attr(*, "physical")=Class 'ff_pointer' <externalptr>
## .. ..- attr(*, "vmode")= chr "integer"
## .. ..- attr(*, "maxlength")= int 4529
## .. ..- attr(*, "pattern")= chr "ffdf"
## .. ..- attr(*, "filename")= chr "/Users/umatter/Dropbox/Teaching/HSG/BigData/BigData/materials/"
## .. ..- attr(*, "pagesize")= int 65536
## .. ..- attr(*, "finalizer")= chr "close"
## .. ..- attr(*, "finonexit")= logi TRUE
## .. ..- attr(*, "readonly")= logi FALSE
## .. ..- attr(*, "caching")= chr "mmnoflush"
## .. ..- attr(*, "virtual")= list()
## .. ..- attr(*, "Length")= int 4529
## .. ..- attr(*, "Symmetric")= logi FALSE
## .. ..- attr(*, "Levels")= chr [1:305] "Abilene, TX" "Akron, OH" "Albany, GA" "Albany, NY" ...
## .. ..- attr(*, "ramclass")= chr "factor"
## .. ..- attr(*, "class") = chr [1:2] "ff_vector" "ff"
## .. $ ORIGIN_STATE_NM : list()
## .. ..- attr(*, "physical")=Class 'ff_pointer' <externalptr>
## .. ..- attr(*, "vmode")= chr "integer"
## .. ..- attr(*, "maxlength")= int 4529
## .. ..- attr(*, "pattern")= chr "ffdf"
## .. ..- attr(*, "filename")= chr "/Users/umatter/Dropbox/Teaching/HSG/BigData/BigData/materials/"
## .. ..- attr(*, "pagesize")= int 65536
## .. ..- attr(*, "finalizer")= chr "close"
## .. ..- attr(*, "finonexit")= logi TRUE

```

```

## .. ..- attr(*, "readonly")= logi FALSE
## .. ..- attr(*, "caching")= chr "mmnoflush"
## .. ..- attr(*, "virtual")= list()
## .. ..- attr(*, "Length")= int 4529
## .. ..- attr(*, "Symmetric")= logi FALSE
## .. ..- attr(*, "Levels")= chr [1:52] "Alabama" "Alaska" "Arizona" "Arkansas" ...
## .. ..- attr(*, "ramclass")= chr "factor"
## .. ..- attr(*, "class") = chr [1:2] "ff_vector" "ff"
## .. $ DEST_CITY_NAME : list()
## .. ..- attr(*, "physical")=Class 'ff_pointer' <externalptr>
## .. ..- attr(*, "vmode")= chr "integer"
## .. ..- attr(*, "maxlength")= int 4529
## .. ..- attr(*, "pattern")= chr "ffdf"
## .. ..- attr(*, "filename")= chr "/Users/umatter/Dropbox/Teaching/HSG/BigData/BigData/materials/"
## .. ..- attr(*, "pagesize")= int 65536
## .. ..- attr(*, "finalizer")= chr "close"
## .. ..- attr(*, "finonexit")= logi TRUE
## .. ..- attr(*, "readonly")= logi FALSE
## .. ..- attr(*, "caching")= chr "mmnoflush"
## .. ..- attr(*, "virtual")= list()
## .. ..- attr(*, "Length")= int 4529
## .. ..- attr(*, "Symmetric")= logi FALSE
## .. ..- attr(*, "Levels")= chr [1:306] "Abilene, TX" "Akron, OH" "Albany, GA" "Albany, NY" ...
## .. ..- attr(*, "ramclass")= chr "factor"
## .. ..- attr(*, "class") = chr [1:2] "ff_vector" "ff"
## .. $ DEST_STATE_NM : list()
## .. ..- attr(*, "physical")=Class 'ff_pointer' <externalptr>
## .. ..- attr(*, "vmode")= chr "integer"
## .. ..- attr(*, "maxlength")= int 4529
## .. ..- attr(*, "pattern")= chr "ffdf"
## .. ..- attr(*, "filename")= chr "/Users/umatter/Dropbox/Teaching/HSG/BigData/BigData/materials/"
## .. ..- attr(*, "pagesize")= int 65536
## .. ..- attr(*, "finalizer")= chr "close"
## .. ..- attr(*, "finonexit")= logi TRUE
## .. ..- attr(*, "readonly")= logi FALSE
## .. ..- attr(*, "caching")= chr "mmnoflush"
## .. ..- attr(*, "virtual")= list()
## .. ..- attr(*, "Length")= int 4529
## .. ..- attr(*, "Symmetric")= logi FALSE
## .. ..- attr(*, "Levels")= chr [1:52] "Alabama" "Alaska" "Arizona" "Arkansas" ...
## .. ..- attr(*, "ramclass")= chr "factor"
## .. ..- attr(*, "class") = chr [1:2] "ff_vector" "ff"
## .. $ CANCELLATION_CODE: list()
## .. ..- attr(*, "physical")=Class 'ff_pointer' <externalptr>
## .. ..- attr(*, "vmode")= chr "integer"
## .. ..- attr(*, "maxlength")= int 4529
## .. ..- attr(*, "pattern")= chr "ffdf"
## .. ..- attr(*, "filename")= chr "/Users/umatter/Dropbox/Teaching/HSG/BigData/BigData/materials/"
## .. ..- attr(*, "pagesize")= int 65536
## .. ..- attr(*, "finalizer")= chr "close"
## .. ..- attr(*, "finonexit")= logi TRUE
## .. ..- attr(*, "readonly")= logi FALSE
## .. ..- attr(*, "caching")= chr "mmnoflush"
## .. ..- attr(*, "virtual")= list()

```

```
## .. .. - attr(*, "Length")= int 4529
## .. .. - attr(*, "Symmetric")= logi FALSE
## .. .. - attr(*, "Levels")= chr [1:4] "" "A" "B" "C"
## .. .. - attr(*, "ramclass")= chr "factor"
## .. .. - attr(*, "class") = chr [1:2] "ff_vector" "ff"
## $ row.names: NULL
## - attributes: List of 2
## .. $ names: chr [1:2] "virtual" "physical"
## .. $ class: chr "ffdf"

dim(subs1.ff)

## [1] 4529      7

dimnames(subs1.ff)

## [[1]]
## NULL
##
## [[2]]
## [1] "FL_DATE"          "AIRLINE_ID"      "ORIGIN_CITY_NAME" "ORIGIN_STATE_NM"
## [5] "DEST_CITY_NAME"   "DEST_STATE_NM"   "CANCELLATION_CODE"
```

### 3.12 Export to CSV

```
# Export subs1.ff into CSV and TXT files:
write.csv.ffdf(subs1.ff, "subset1.csv")
```

Walkowiak, Simkon. 2016. *Big Data Analytics with R*. Birmingham, UK: PACKT Publishing.