

Dockerizing KNIME

Recipes for a KNIME Cocktail

Corpse Reviver

[knowyourcocktails.com](#)



KEY:

- Brandy
- Apple Brandy
- Sweet Vermouth
- Green Apple

DESCRIPTION:
A few variations of the Corpse Reviver exist, according to several drink historians, the original was created by Frank Meier at the Ritz Bar in Paris, France. The Corpse Reviver is taken "before 11 a.m. or whenever steam and energy are needed," according to Henry Craddock in *The Savoy Cocktail Book*. This drink is particularly popular at Halloween parties in the U.S.

BEST OCCASION:

TIME REQUIRED: Prep: 2 min Mix: 1 min

FINISHED DRINK: 

INSTRUCTIONS:

- Combine brandy, apple brandy, and sweet vermouth in a cocktail shaker.
- Shake with ice.
- Strain into a chilled cocktail glass.
- Garnish with an apple slice.

1 **2** **3** **4**



PRECISESADS gathers a wide range of data from thousands of patients suffering from auto-immune disorders that share common pathophysiological mechanisms in order to molecularly reclassify the diseases and eventually develop targeted therapies.

systemic lupus erythematosus
systemic sclerosis
Sjögren's syndrome
rheumatoid arthritis
primary antiphospholipid syndrome
mixed connective tissue disease



PRECISESADS

Molecular Reclassification to Find
Clinically Useful Biomarkers for
Systemic Autoimmune Diseases



CymeR

Advanced **cytometric** analysis using **KNIME**
leveraged with **Docker** and **R**

CymeR

Advanced cytometric
analysis using KNIME
leveraged with Docker
and R

 Download ZIP

 Download TAR

 View On GitHub

This project is
maintained by
[bmuchmore](#)

The prelude

CymeR Docker image

CymeR is open-source software that ties several separate components into a single program. The heart of the program is a Docker image that can be found at <https://hub.docker.com/r/bmuchmore/cymer/>

CymeR workspace

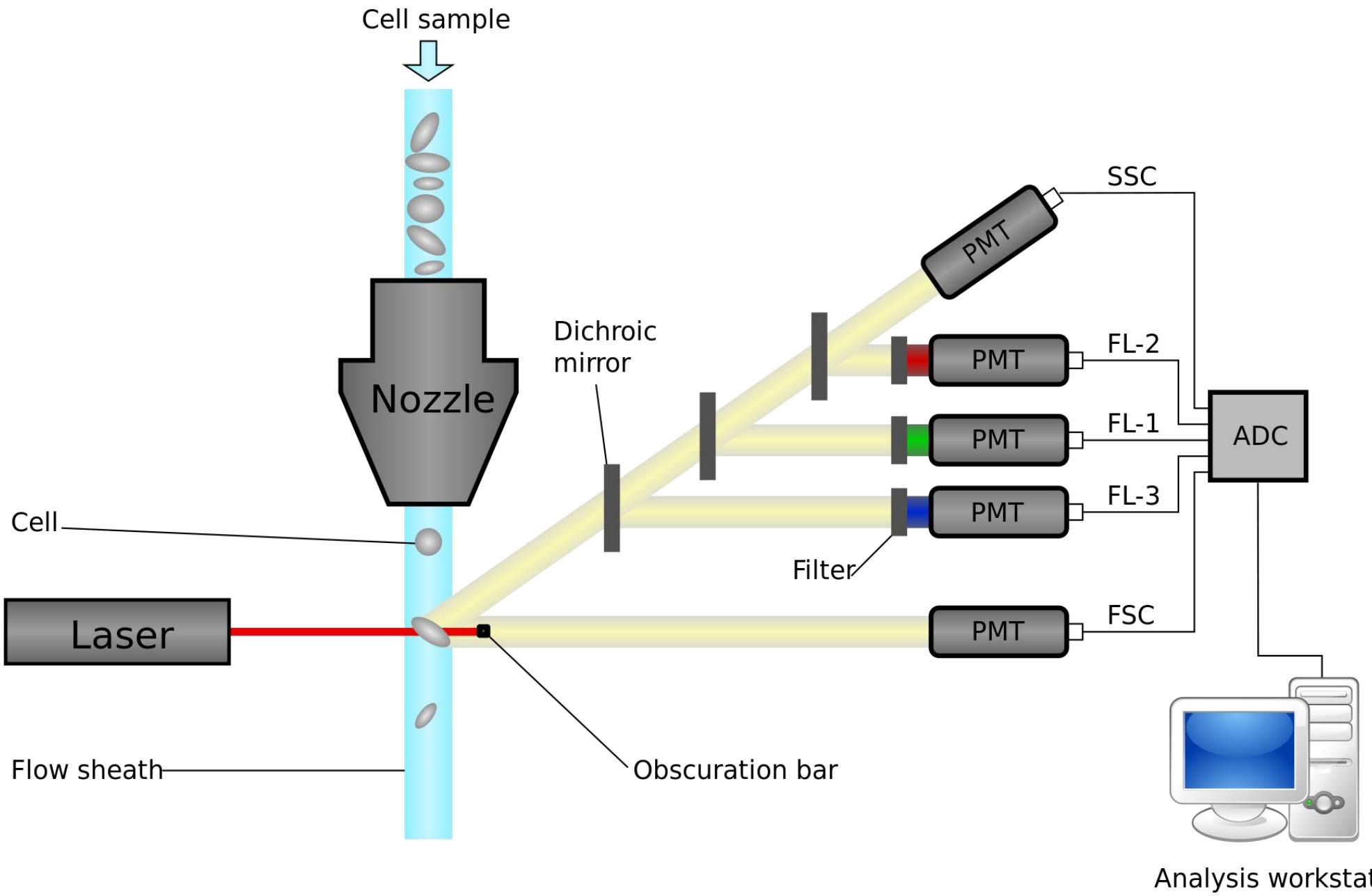
To make use of this Docker image, however, we provide an eclipse workspace along with a number of scripts for automated installation, start-up, updating and removal on GitHub at <https://github.com/bmuchmore/CymeR>

CymeR on different platforms

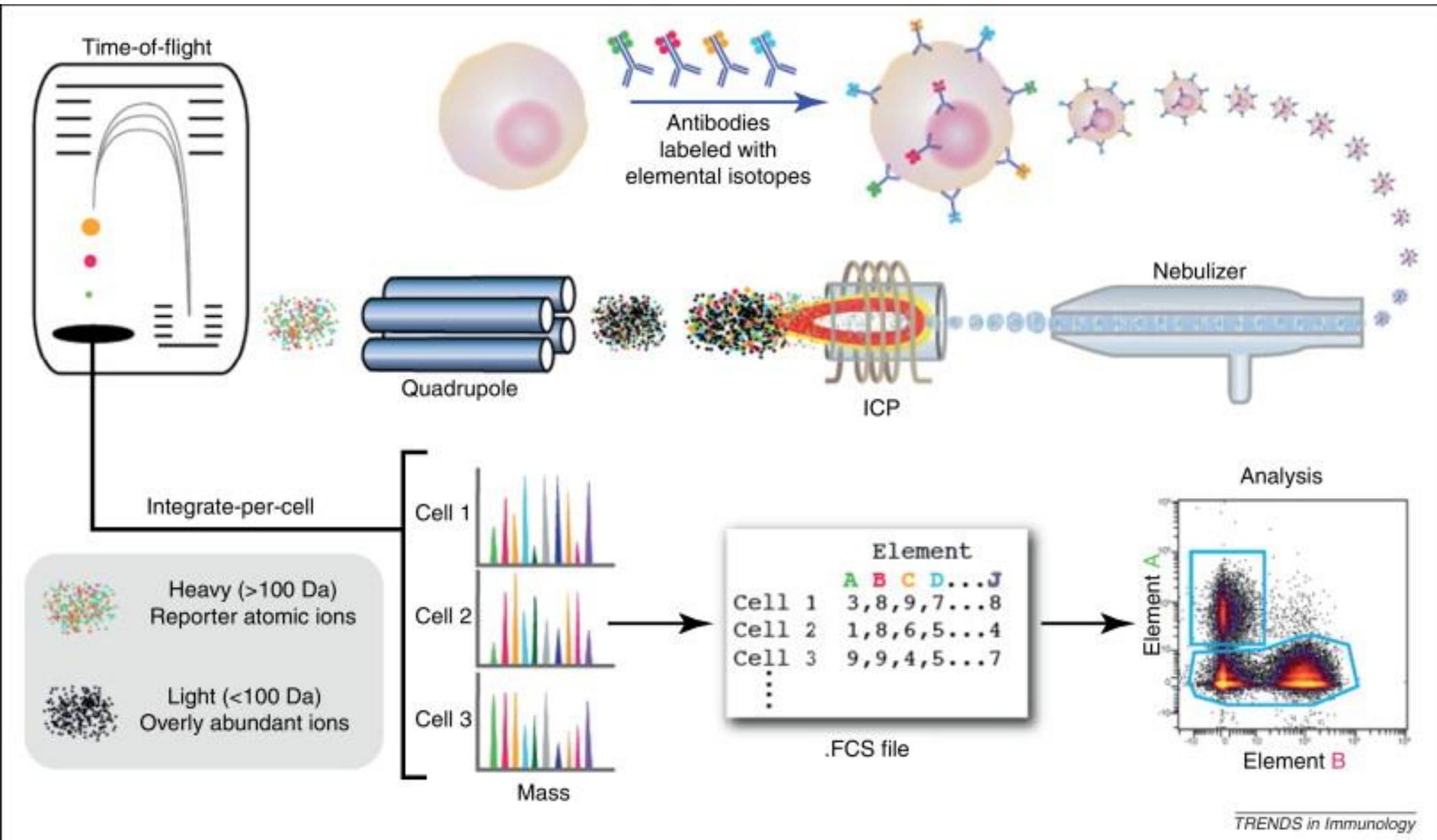
The Goal of CymeR

- | To implement graphical user interfaces (GUIs) for state of the art cytometry algorithms (e.g. clustering, dimension reduction, feature selection etc) in an open source environment.
- | To use these algorithms in ways that let you see your data in ways previously impossible.

Brief Introduction to Flow Cytometry



Brief Introduction to Mass Cytometry



Using KNIME as a simple GUI front-end for complex functions

What the biologist sees:

The image shows a KNIME dialog window titled "Dialog - 5:393 - 2D BHT-SNE continued (multicore)". The window has two tabs: "QuickForms" (selected) and "Job Manager Selection".

Include the columns you would like to run BHT-SNE on:

Exclude (red border): Column(s): Search Select all search hits
List: FSC-A, FSC-W, FSC-H, SSC-A, SSC-W, SSC-H, APC-AF750-A, Time

Select: add >> add all >> << remove << remove...

Include (green border): Column(s): Search Select all search hits
List: FITC-A, PE-A, PC5-5-A, PC7-A, APC-A, PB-A, KO-A

Would you like to check for duplicates?
 TRUE FALSE

Choose the theta value. Theta is a speed/accuracy trade-off (increase for less accuracy):

Choose the perplexity setting. Typical values are between 5-50 with higher values typically being used for denser datasets:

Would you like to perform PCA on your dataset to reduce the number of dimensions before you run bht-sne?
 TRUE FALSE

If you are performing PCA, choose the number of dimensions you would like PCA to downsample to. Fifty or fewer is common practice:

Choose the multiplier of the minimum median distance within which other observations are counted towards the density:

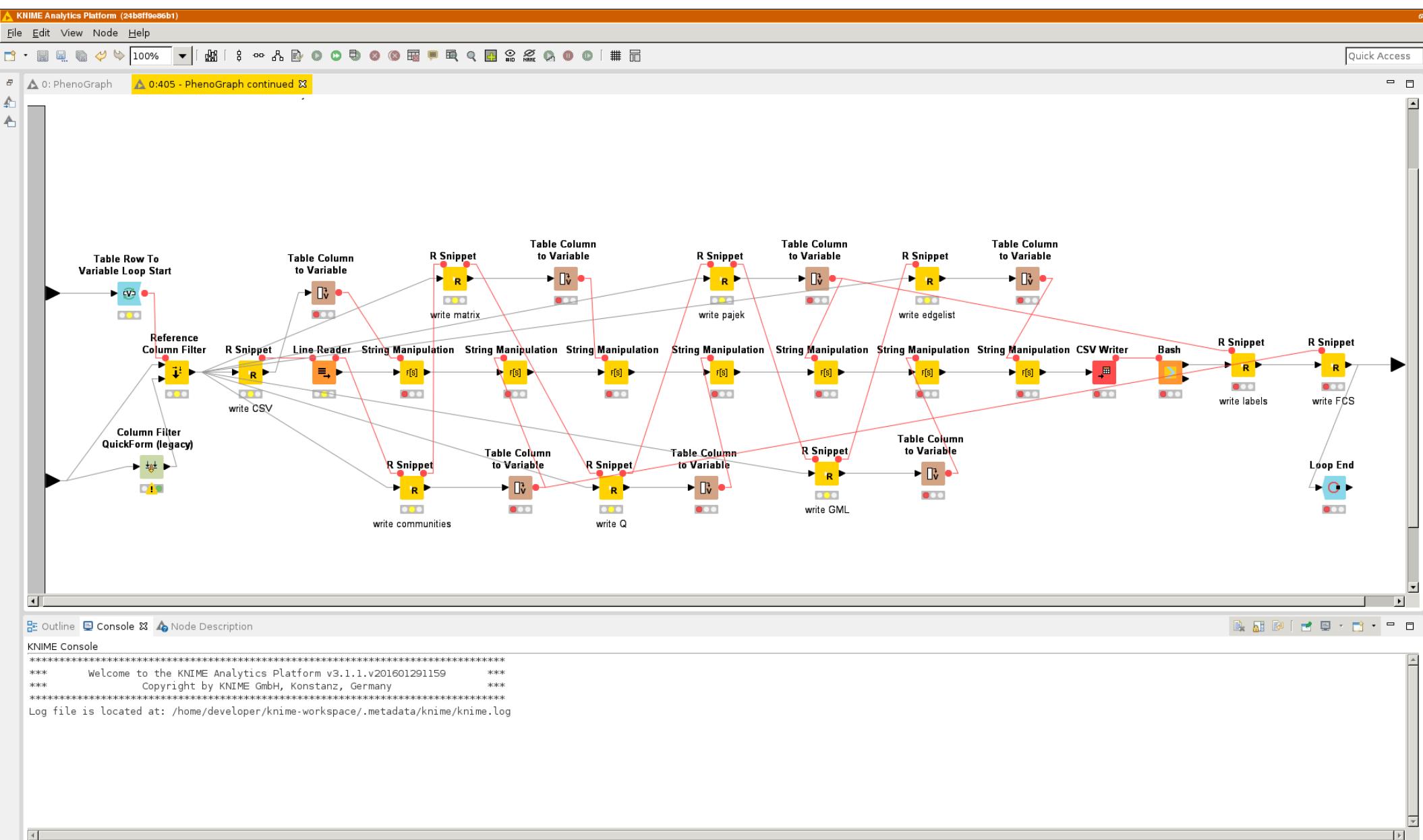
Choose the number of observations used to estimate the minimum median distance:

Choose the multiplier of the minimum median distance within which observations are approximated to have the same density:

Buttons at the bottom: OK, Apply, Cancel, ?

Using KNIME as a simple GUI front-end for complex functions

Underlying KNIME workflow:



Using KNIME as a simple GUI front-end for complex functions

Underlying R code:

Dialog - 5393361 - R Snippet

R Snippet Templates Advanced Flow Variables Memory Policy Create Template...

Column List

Flow Variable List

- knime.workspace
- PCA-dimensions
- perplexity
- theta
- check-duplicates (Index)
- check-duplicates
- PCA (Index)
- PCA
- kernel_mult
- med_samples
- apprx_mult
- Location
- dirname
- basename
- RowID
- new variable

R Script

```
1 library(flowCore)
2 library(BioBase)
3 library(Rtsne)
4 library(spade)
5 library(foreach)
6 library(doParallel)
7 library(parallel)
8 numCores <- detectCores()
9 cl <- makeCluster(numCores)
10 registerDoParallel(cl)
11 processInput <- function(i) {
12   cols <- knime.in
13   fs <- read.flowSet(files = NULL, path = knime.flow.in[["dirname"]], pattern = "fcs")
14   df <- as.data.frame(exprs(fs[[i]]))
15   df2 <- as.matrix(df[, which(names(df) %in% c(as.character(t(cols))))])
16   unique_mydata <- unique(df2)
17   rtsne_out <- Rtsne(unique_mydata, dims = 2, initial_dims = knime.flow.in[["PCA-dimensions"]], perplexity = knime.flow.in[["perplexity"]], theta = knime.flow.in[["theta"]], check_duplicate = TRUE)
18   new.cols <- rtsne_out$Y
19   channel_number <- ncol(fs[[i]]) + 1;
20   channel_id <- paste("$P", channel_number, sep = "");
21   channel_name <- "BHT.SNE.2D.Dim1";
22   channel_range <- max(new.cols[, 1]) + 1;
23   plist <- matrix(c(channel_name, channel_number, 0, channel_range - 1));
24   rownames(plist) <- c("name", "desc", "range", "minRange", "maxRange");
25   colnames(plist) <- c(channel_id);
26   channel_number2 <- ncol(fs[[i]]) + 2;
27   channel_id2 <- paste("$P", channel_number2, sep = "");
28   channel_name2 <- "BHT.SNE.2D.Dim2";
29   channel_range2 <- max(new.cols[, 2]) + 1;
30   plist2 <- matrix(c(channel_name2, channel_number2, 0, channel_range2 - 1));
31   rownames(plist2) <- c("name", "desc", "range", "minRange", "maxRange");
32   colnames(plist2) <- c(channel_id2);
33   Sys.setenv("OMP_NUM_THREADS" = detectCores())
34   density <- SPADE.density(as.data.frame(new.cols), kernel_mult = knime.flow.in[["kernel_mult"]], apprx_mult = knime.flow.in[["apprx_mult"]], med_samples = knime.flow.in[["med_samples"]])
35   dens <- as.matrix(density)
36   channel_number3 <- ncol(fs[[i]]) + 3;
37   channel_id3 <- paste("$P", channel_number3, sep = "");
38   channel_name3 <- "BHT.SNE.2D.Density";
39   channel_range3 <- max(dens[, 1]) + 1;
40   plist3 <- matrix(c(channel_name3, channel_number3, channel_range3, 0, channel_range3 - 1));
41   rownames(plist3) <- c("name", "desc", "range", "minRange", "maxRange");
```

Workspace

Name	Type

Eval Script Eval Selection Reset Workspace Show Plot

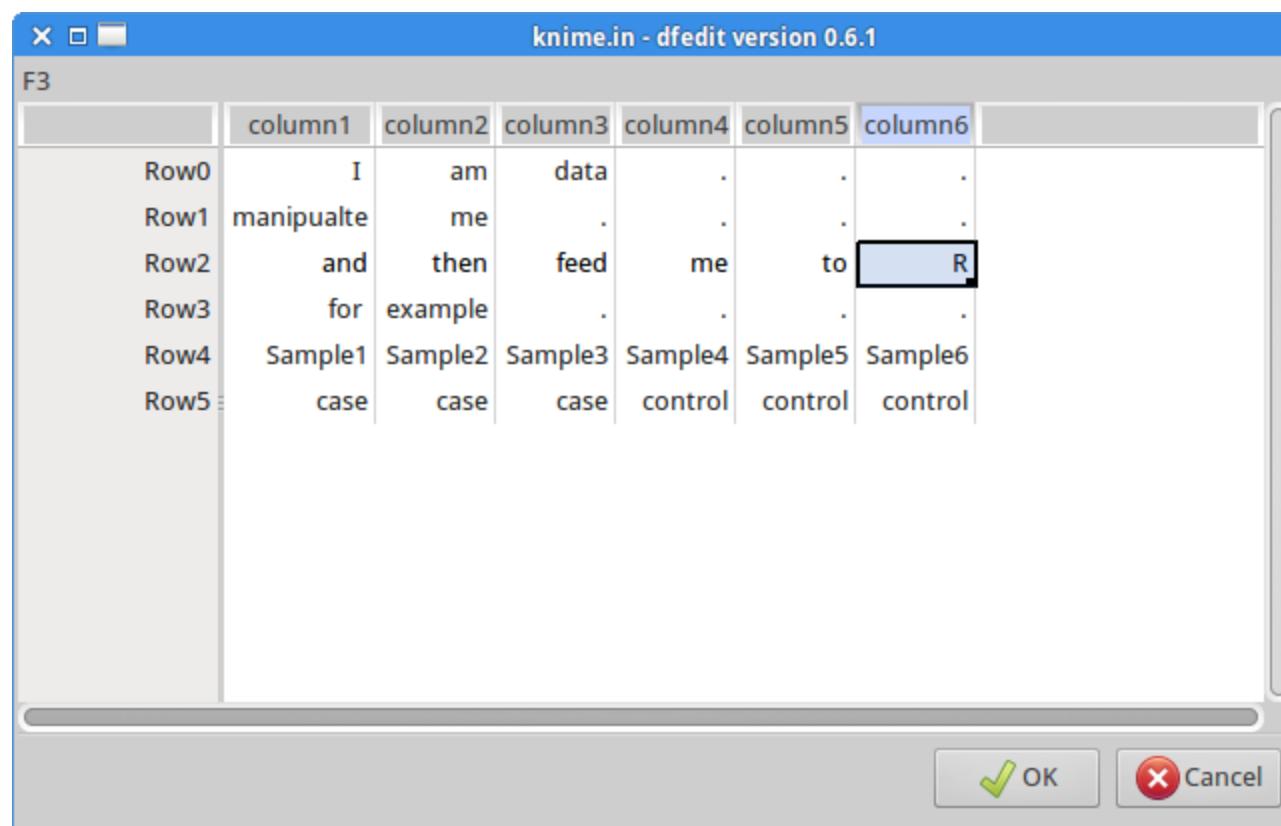
Console

R is busy waiting... 0% Cancel

OK Apply Cancel ?

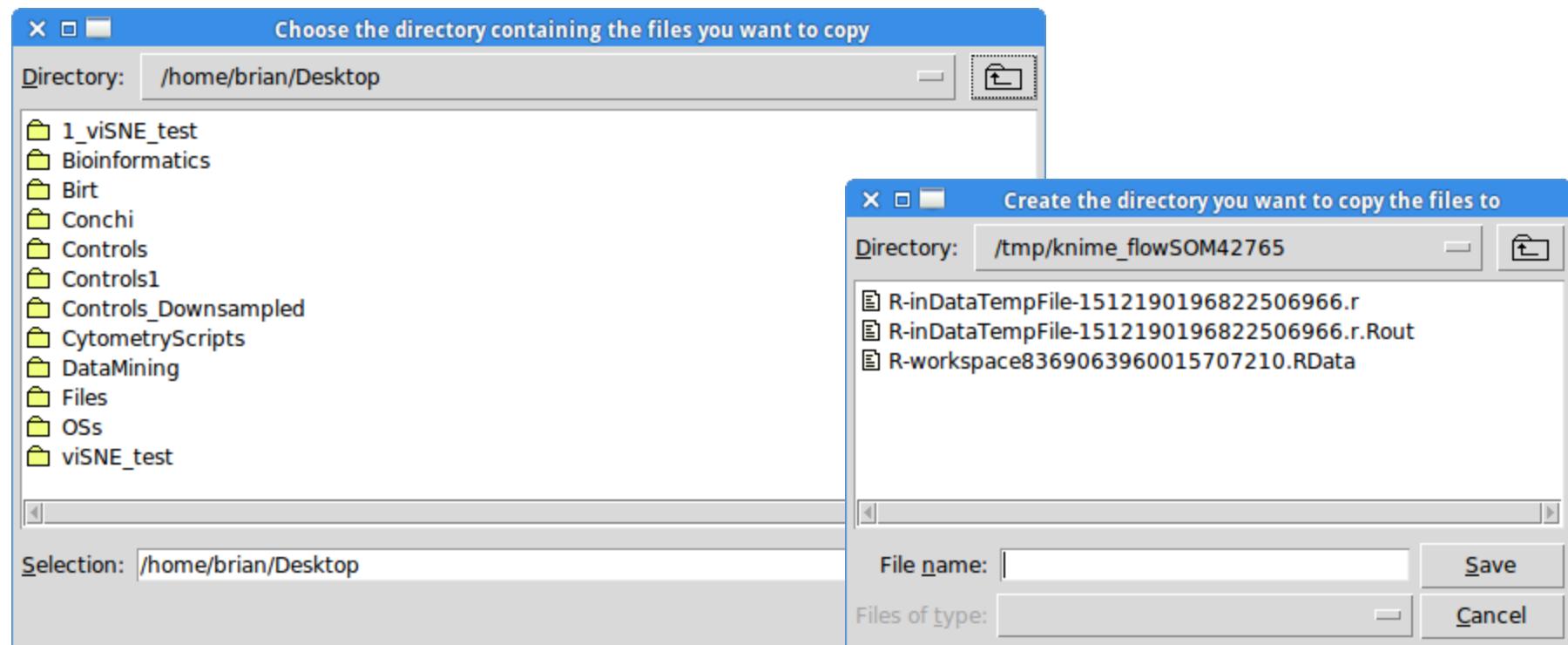
R can make KNIME (more) interactive

Sometimes, R needs interactive inputs that can not be pre-configured in KNIME. For these scenarios, I code in an interactive table editor that goes back into R as a data frame. All the biologist has to do is fill out the table when it pops up:



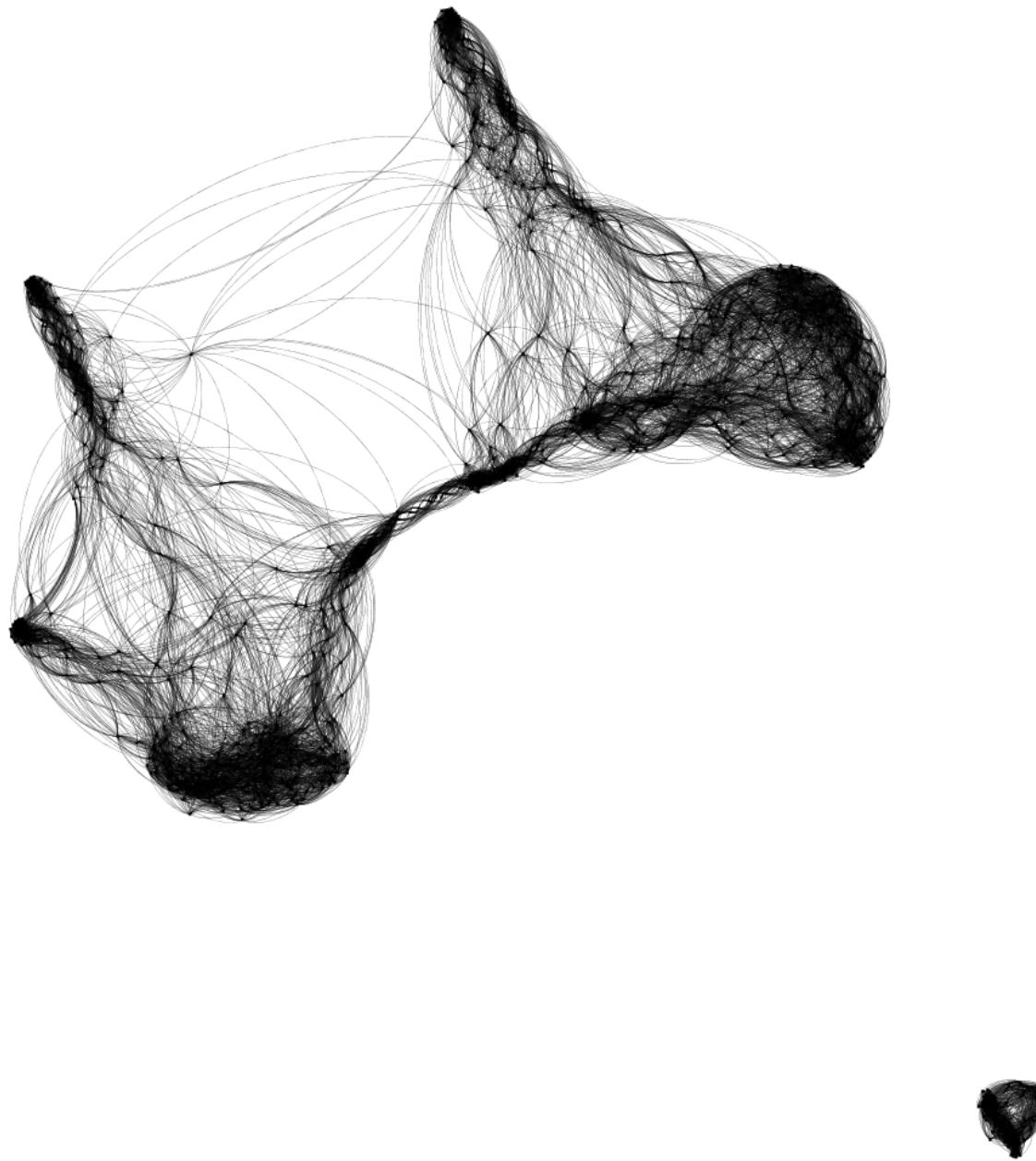
Essential workflow

- 1) Read in a FCS file
- 2) Apply function to FCS file
- 3) Write out new FCS file (and other files as needed)



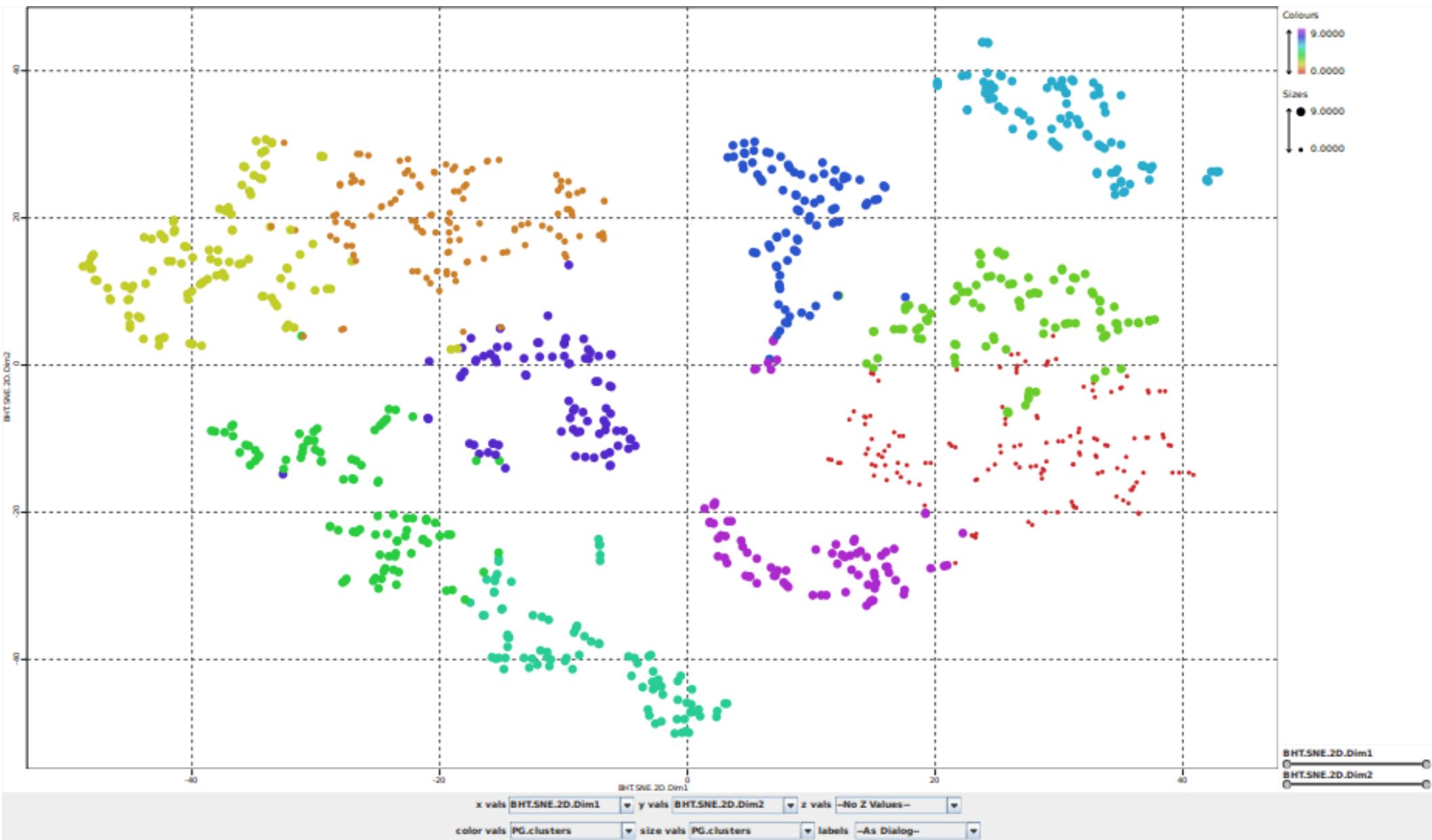
KNIME makes (scientific) data fun

PhenoGraph: A cellular social network that returns clustering designation based on Louvain communities



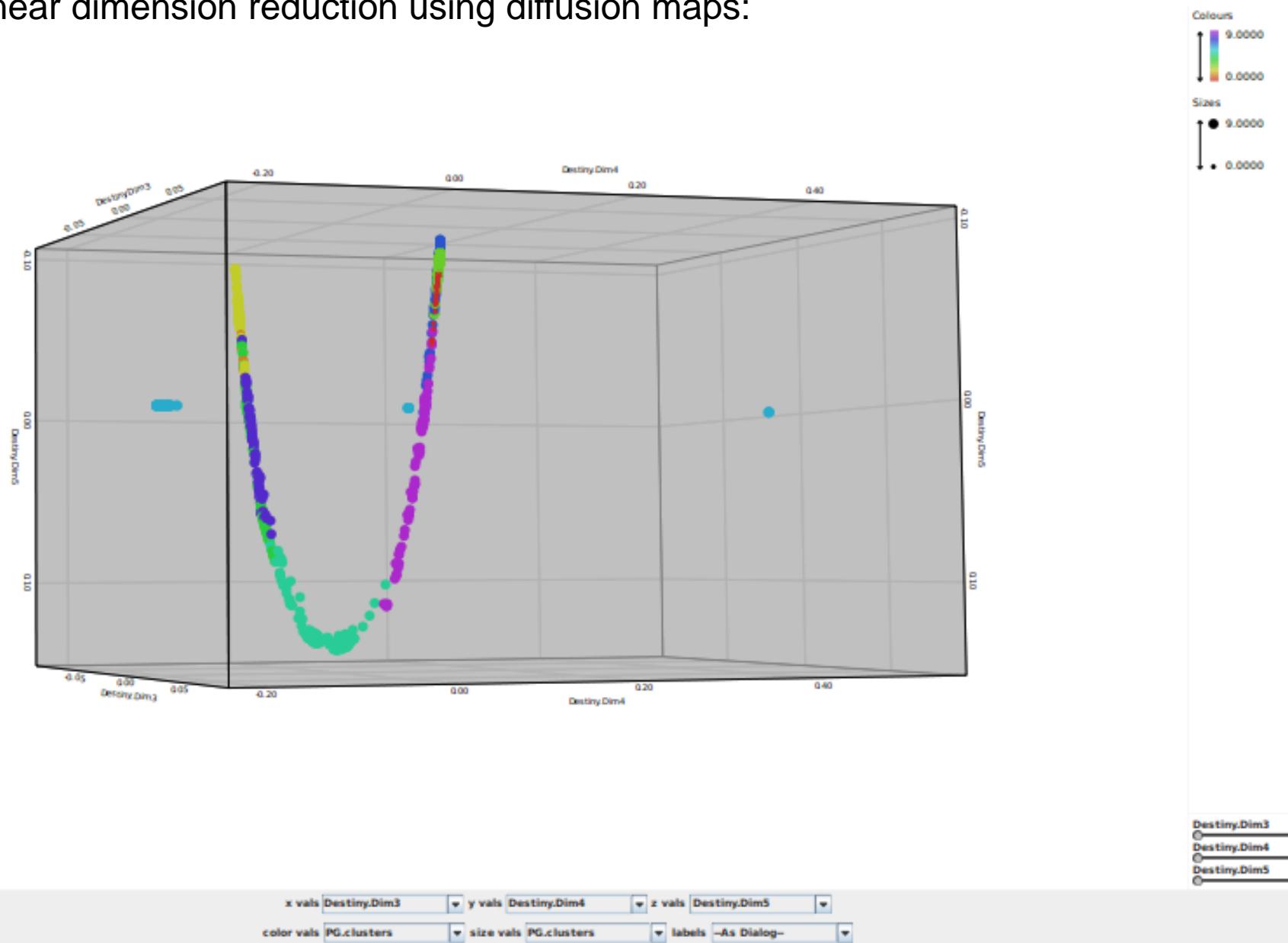
KNIME makes (scientific) data fun

BHt-SNE: Non-linear dimension reduction using Barnes-Hut t-Distributed Stochastic Neighbor Embedding



KNIME makes (scientific) data fun

Destiny: Non-linear dimension reduction using diffusion maps:



The abbreviated list of programs needed to make everything work:

KNIME

KNIME extensions

JAVA 8

R

Python 2.7

Python 3.4

X11 Server

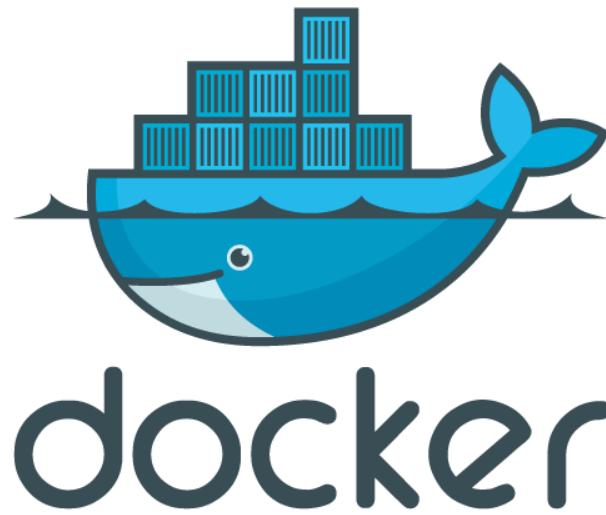
Firefox

60+ R or R/Bioconductor packages

Many other specific packages and...

All the dependencies required to make the above run

A brief introduction to...



The following Docker slides have been adapted from:
http://www.slideshare.net/dotCloud/docker-intro-november?from_action=save

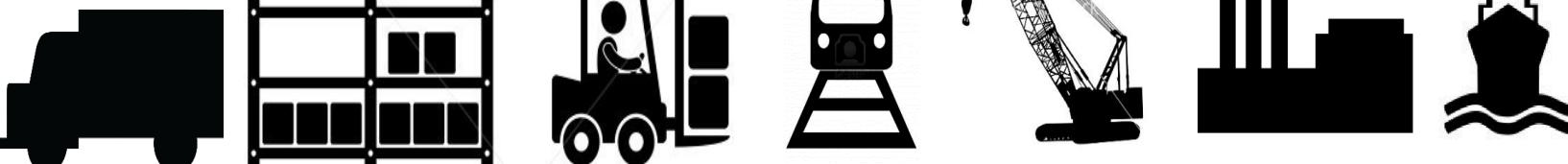
Docker metaphor: Intermodal Shipping Container



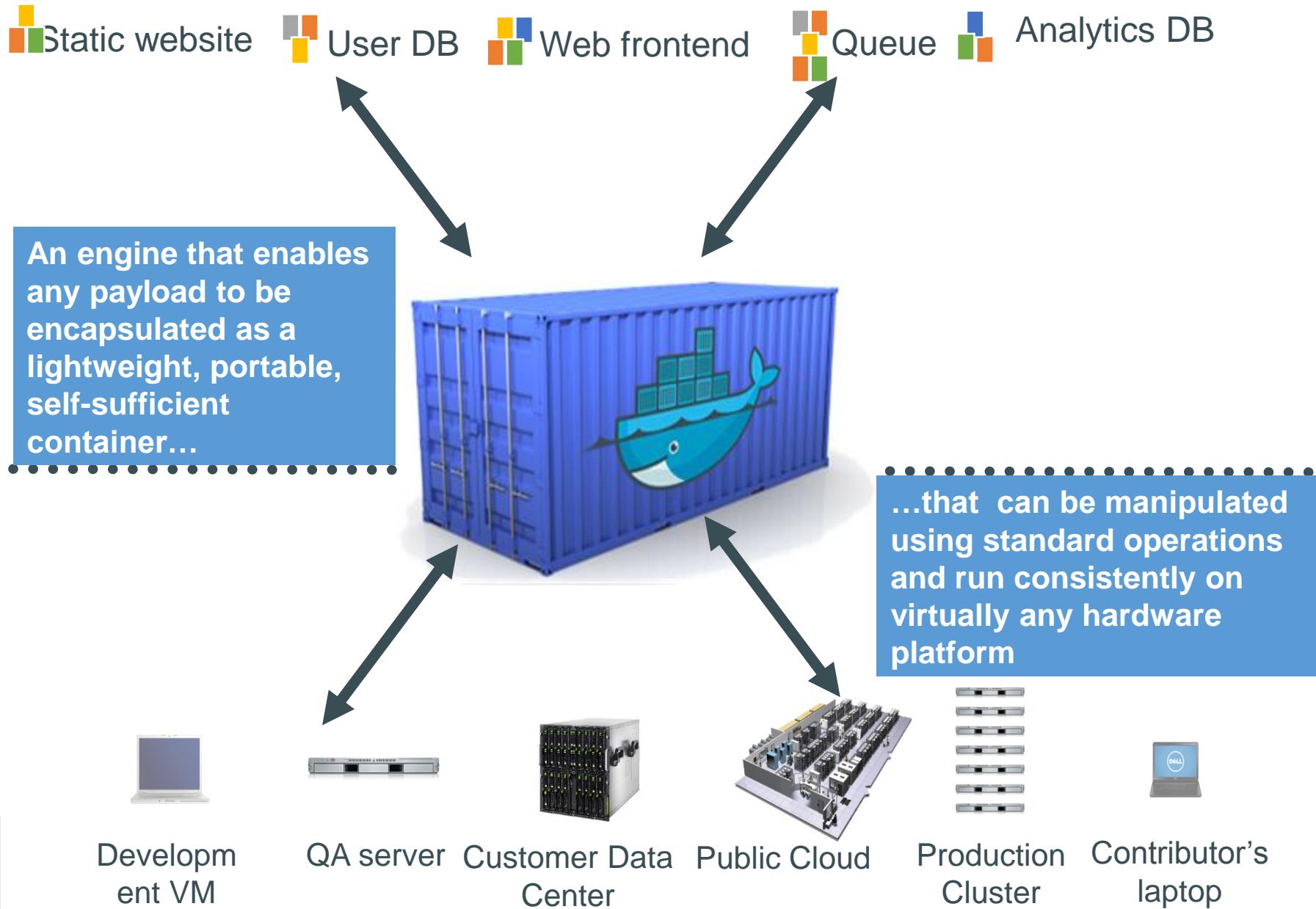
A standard container that is loaded with virtually any goods, and stays sealed until it reaches final delivery.



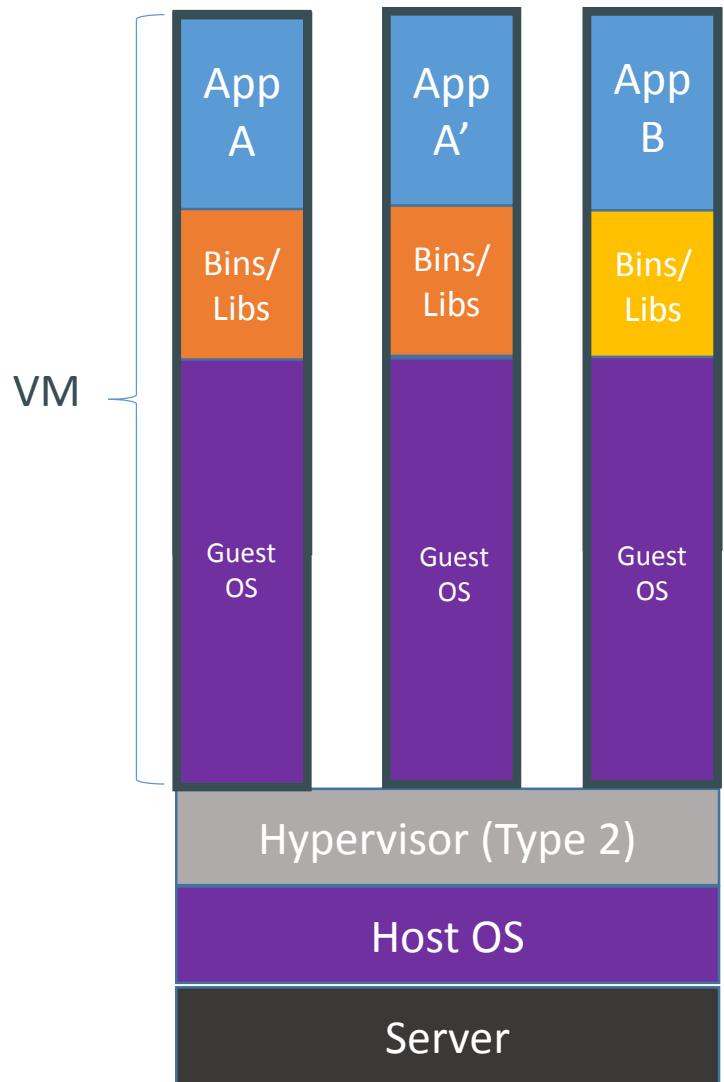
...in between, can be loaded and unloaded, stacked, transported efficiently over long distances, and transferred from one mode of transport to another



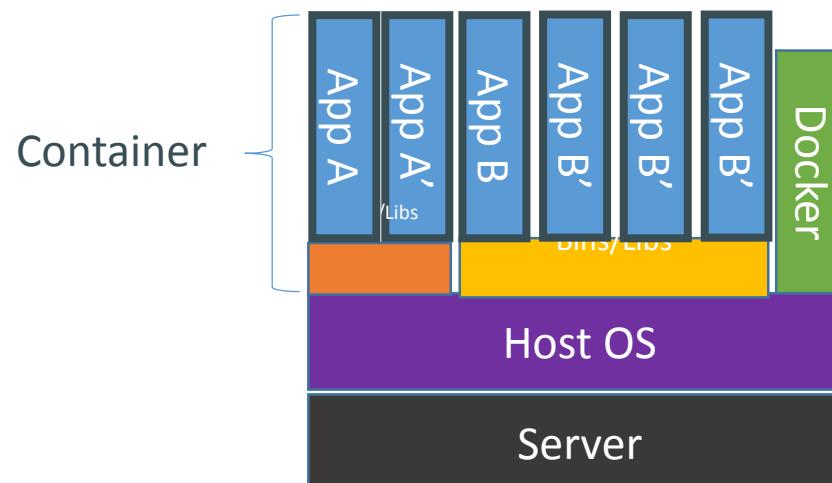
Docker is a shipping container system for code



Containers vs. VMs



Containers are isolated, but share OS and, where appropriate, bins/libraries



Example Dockerfile

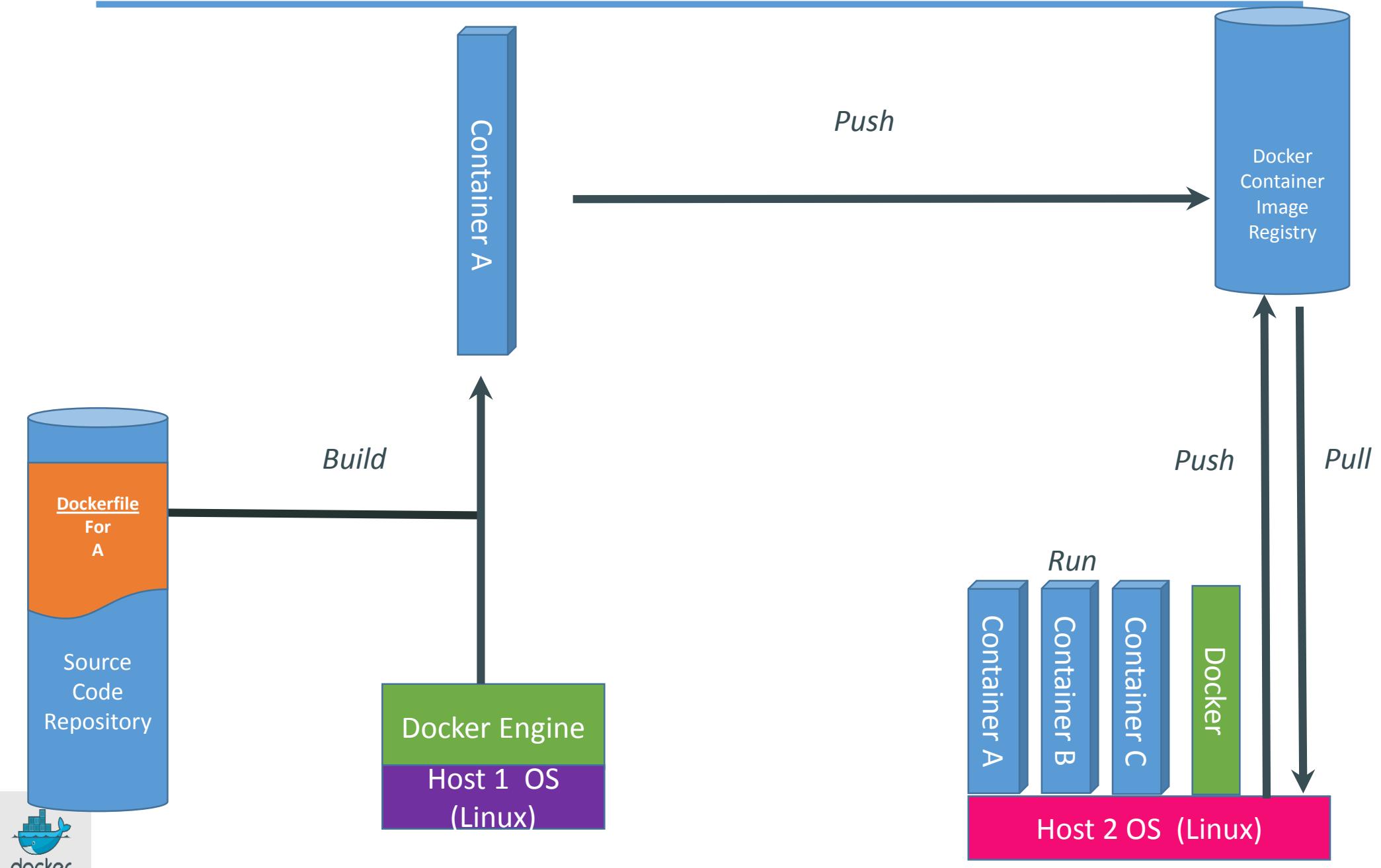
```
FROM ubuntu:14.04
MAINTAINER Brian Muchmore "brian.muchmore@genyo.es"

##Install Java8
RUN add-apt-repository ppa:webupd8team/java && \
    apt-get update -y && \
    echo oracle-java8-installer shared/accepted-oracle-license-v1-1 select true | /usr/bin/debconf-set-
selections && \
    xvfb-run -a apt-get install oracle-java8-installer libxext-dev libxrender-dev libxtst-dev -y && \
    xvfb-run -a apt-get install liblzma-dev -y && \
    xvfb-run -a apt-get install libglu1-mesa-dev -y

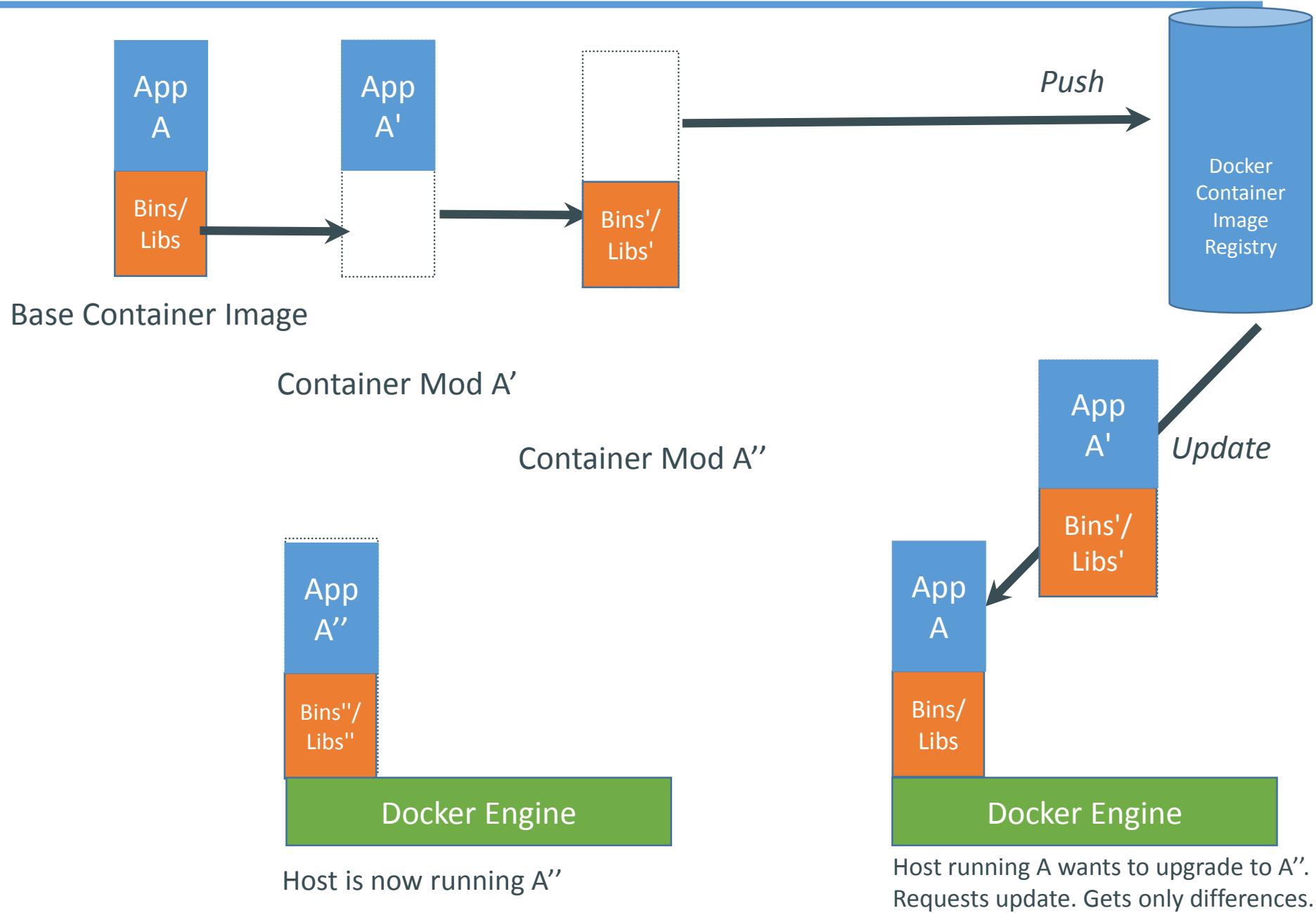
##Download KNIME
RUN echo 'Installing knime' && \
    wget http://download.knime.org/analytics-platform/linux/knime_3.1.0.linux.gtk.x86_64.tar.gz -O
/tmp/knime.tar.gz -q
    tar -xf /tmp/knime.tar.gz -C /home/developer && \
    rm /tmp/knime.tar.gz && \
    apt-get install libwebkitgtk-1.0-0 -y

USER developer
ENV HOME /home/developer
WORKDIR /home/developer
CMD /home/developer/knime_3.1.1/knime -data /home/developer/knime-workspace
```

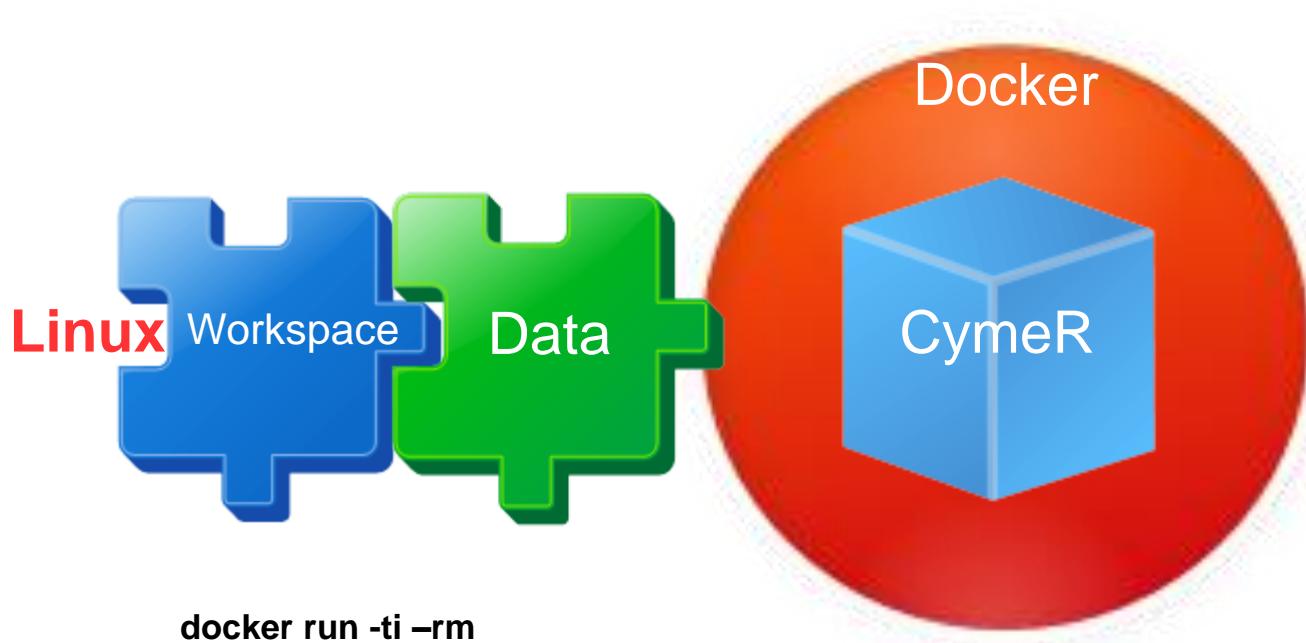
What are the basics of the Docker system?



Changes and Updates



CymeR in Linux



```
docker run -ti --rm
```

```
-e DISPLAY=$DISPLAY
```

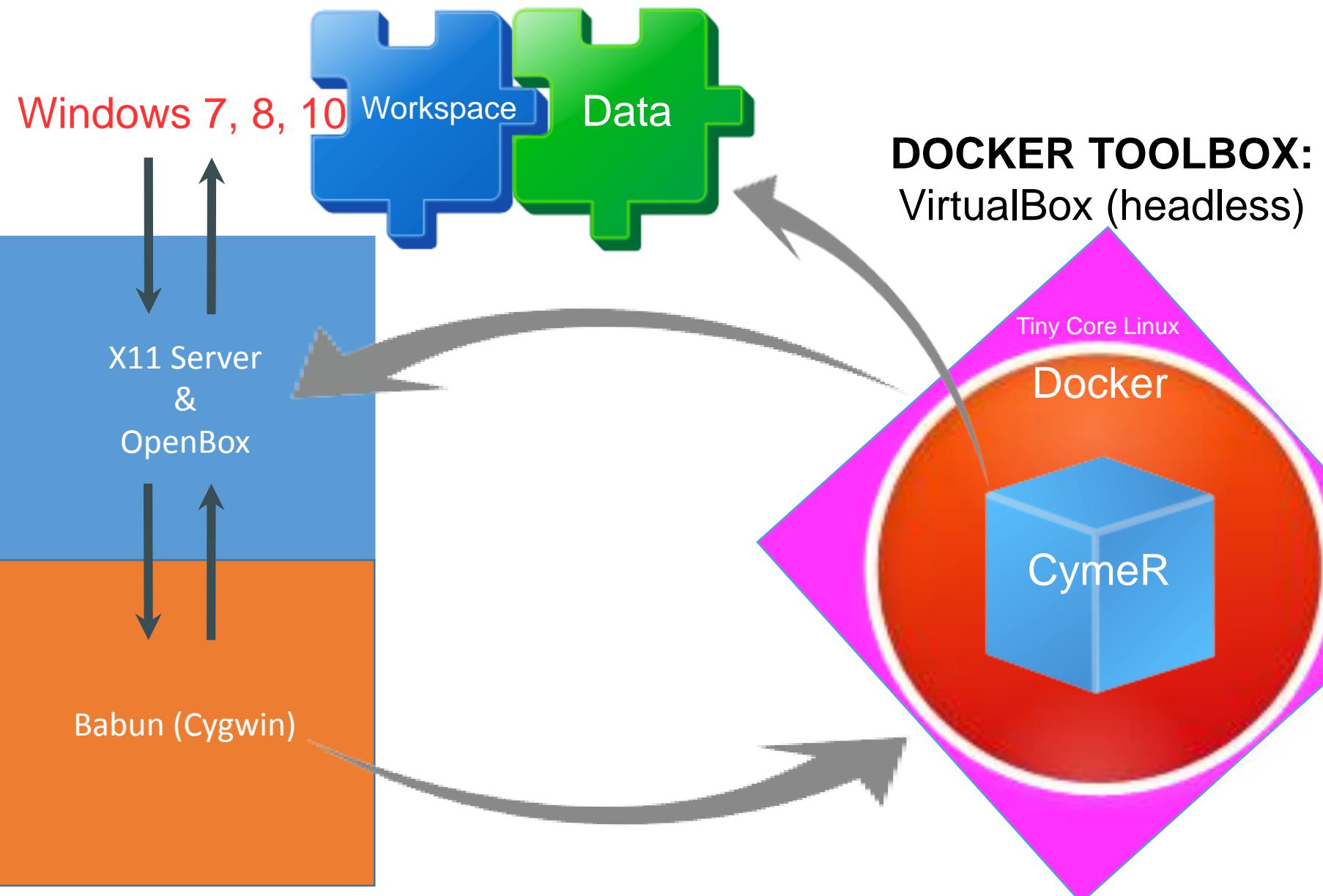
```
CymeR-Data:/home/developer/Data:rw
```

```
CymeR-Workspace:/home/developer/knime-workspace:rw
```

```
/tmp/.X11-unix:/tmp/.X11-unix:rw
```

```
bmuchmore/cymer
```

CymeR in Windows Using Less than 3 GB of RAM



CymeR in Mac

COMING SOON

(It should be much simpler to implement than CymeR in Windows)

CymeR

Advanced **cytometric** analysis using **KNIME**
leveraged with **Docker** and **R**

CymeR

Advanced cytometric
analysis using KNIME
leveraged with Docker
and R

 Download ZIP

 Download TAR

 View On GitHub

This project is
maintained by
[bmuchmore](#)

The prelude

CymeR Docker image

CymeR is open-source software that ties several separate components into a single program. The heart of the program is a Docker image that can be found at <https://hub.docker.com/r/bmuchmore/cymer/>

CymeR workspace

To make use of this Docker image, however, we provide an eclipse workspace along with a number of scripts for automated installation, start-up, updating and removal on GitHub at <https://github.com/bmuchmore/CymeR>

CymeR on different platforms

Thanks to...

Dr. Marta Alarcon-Riquelme

IMI and PRECISESADS

KNIME

DOCKER

All of you.



For much more info, go to:

<http://bmuchmore.github.io/Dock-o-KNIME/>

Or

<http://bmuchmore.github.io/CymeR/>

Questions?

Ask now or contact me at bmuchmore@gmail.com
(I will do my best to respond in a timely manner)