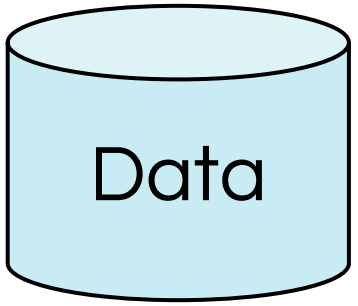
The yt Hub Architecture

Matthew Turk, iSchool, University of Illinois
Kacper Kowalik, NCSA, UIUC

Concepts

1. Logging in
2. Sharing data
3. Analyzing data

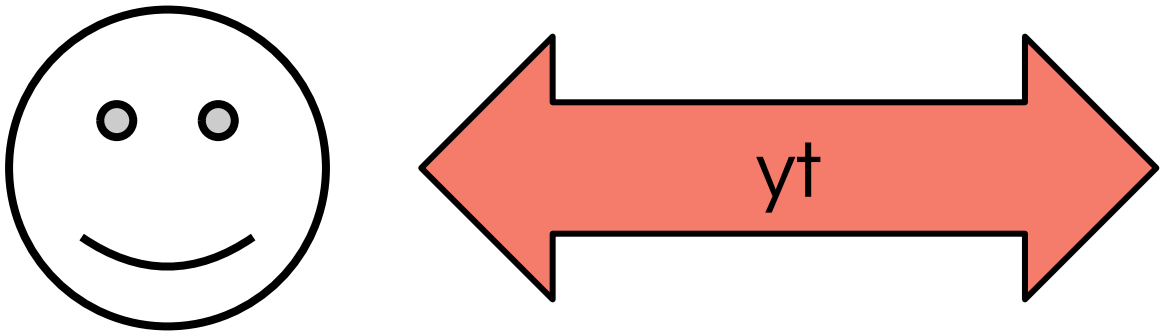
Driving force



Driving force

Size: 0.1PB
What: mostly astro data
(both sim and obs)
Where: Block storage
@ Nebula ;-(

Driving force



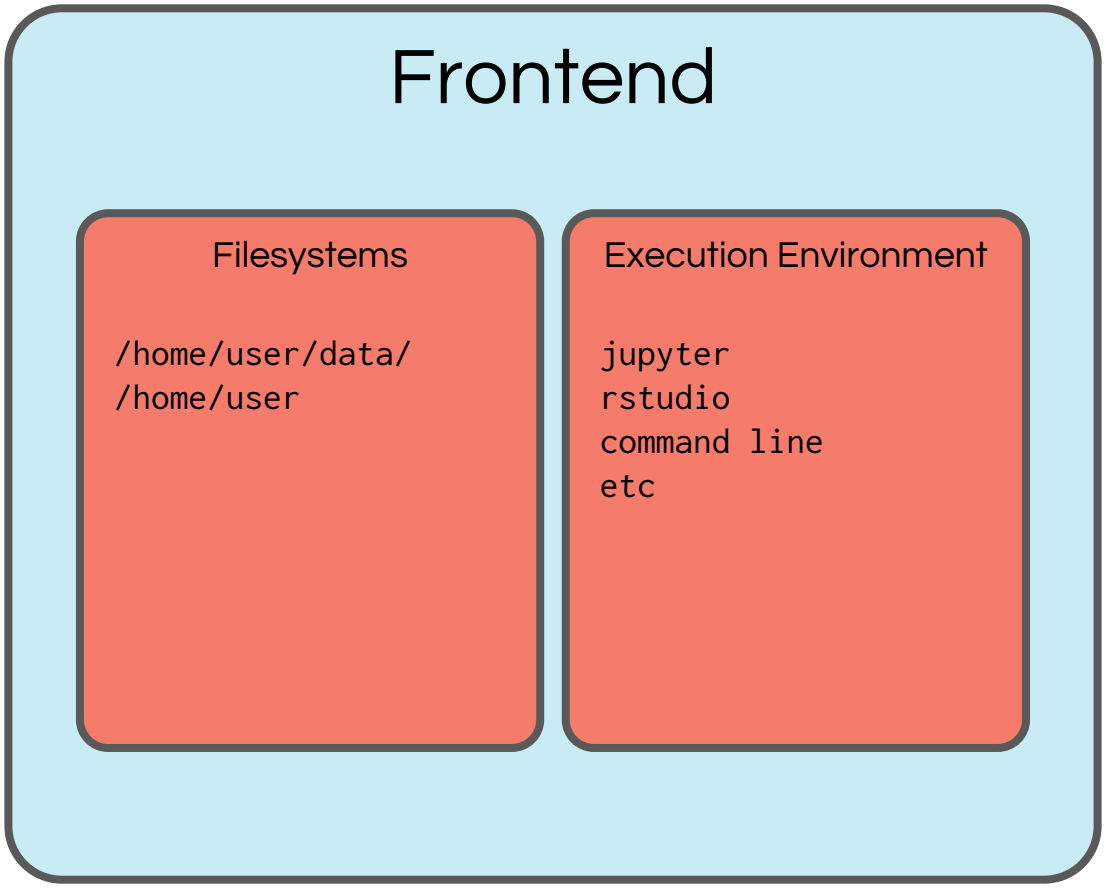
File Edit View Insert Cell Kernel Help

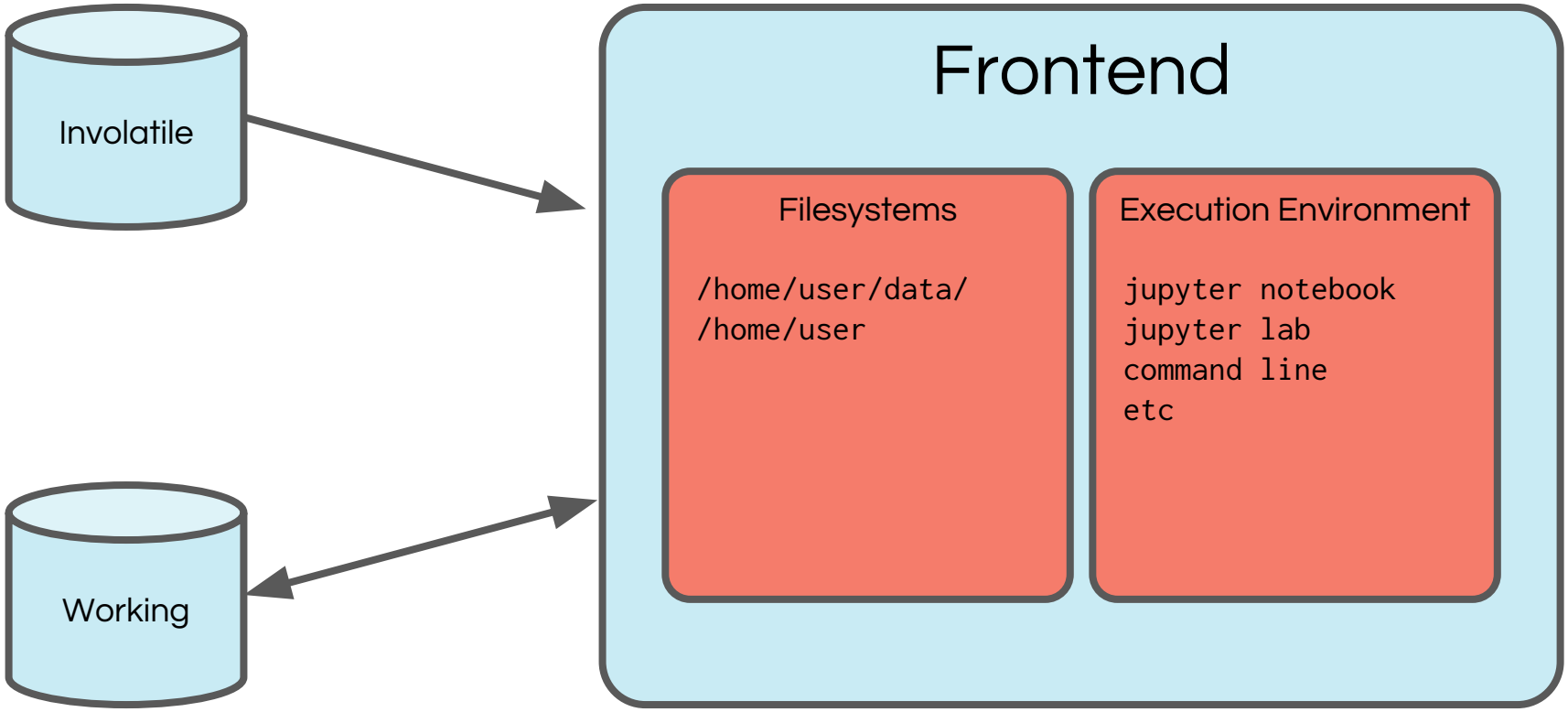
yt (dev) - py2

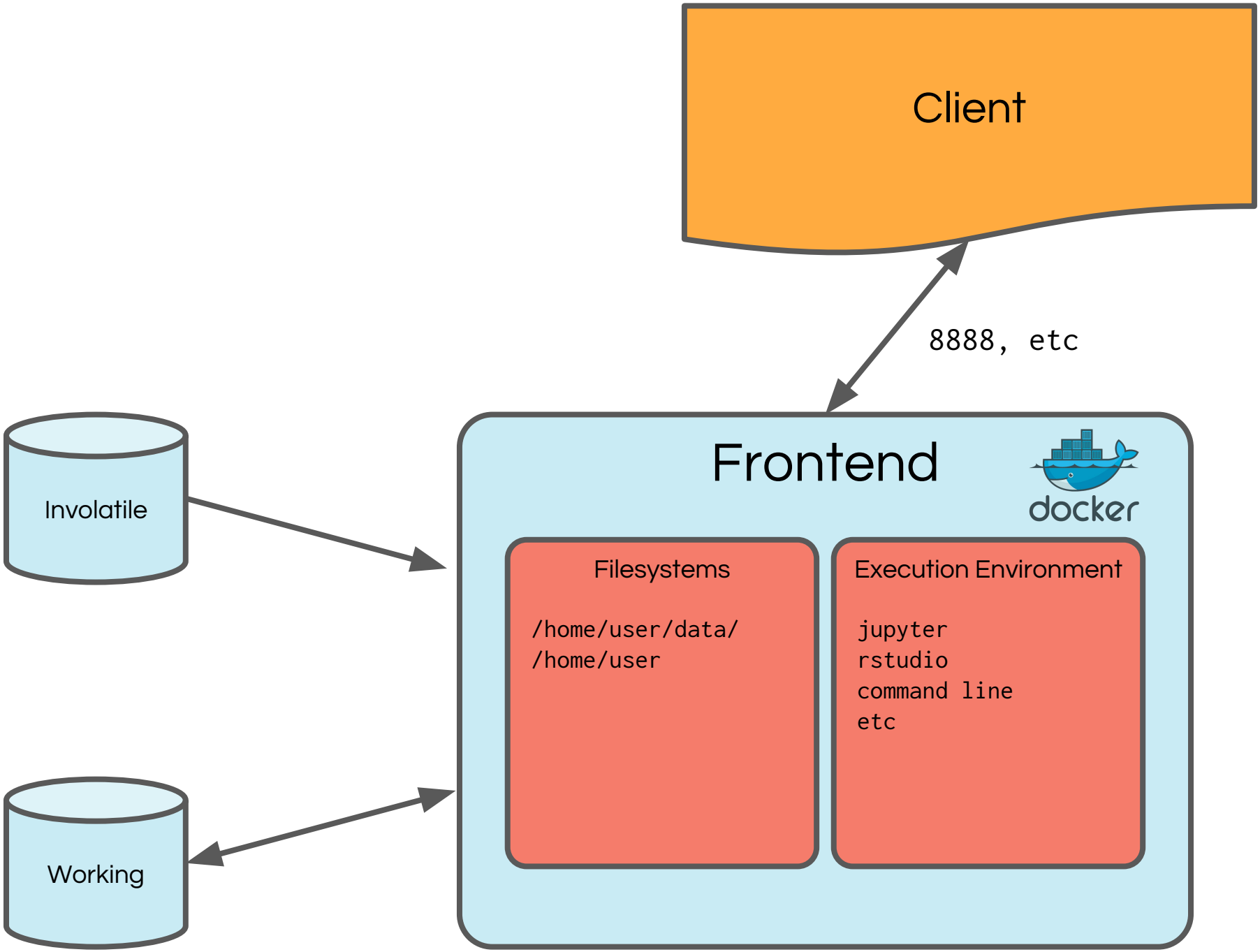
Code CellToolbar

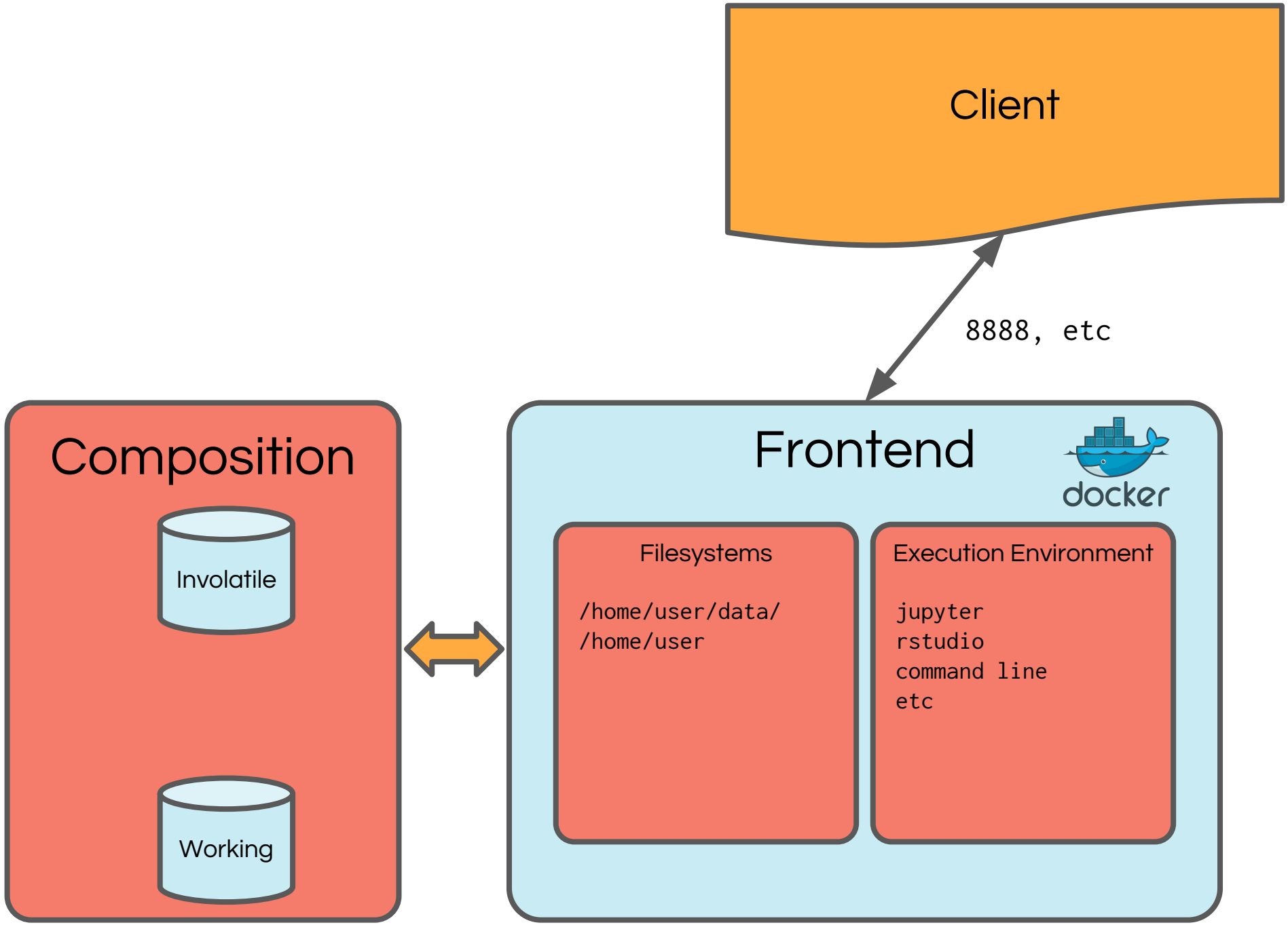
In [1]:
```
import yt
ds = yt.load("data/IsolatedGalaxy/galaxy0030/galaxy0030")
ds.r[:,0.5,:].plot("density")
```

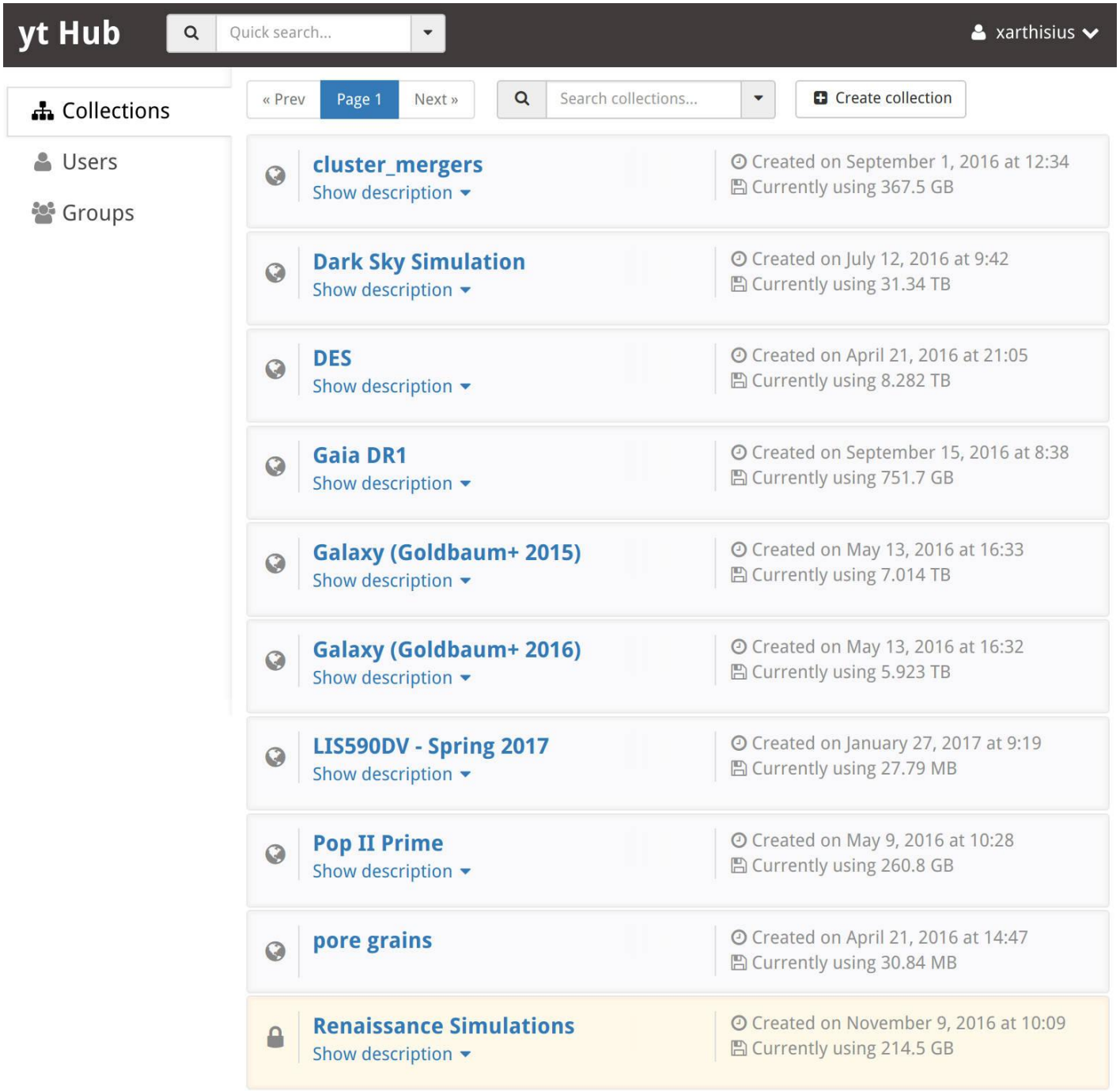
```
yt : [INFO     ] 2016-09-27 14:12:38,067 Parameters: current_time              = 0.00600002000283
yt : [INFO     ] 2016-09-27 14:12:38,069 Parameters: domain_dimensions         = [32 32 32]
yt : [INFO     ] 2016-09-27 14:12:38,071 Parameters: domain_left_edge          = [ 0.  0.  0.]
yt : [INFO     ] 2016-09-27 14:12:38,072 Parameters: domain_right_edge         = [ 1.  1.  1.]
yt : [INFO     ] 2016-09-27 14:12:38,073 Parameters: cosmological_simulation   = 0.0
/opt/conda/envs/py2-dev/lib/python2.7/site-packages/matplotlib/font_manager.py:273: UserWarning: Matplotlib is build
ing the font cache using fc-list. This may take a moment.
  warnings.warn('Matplotlib is building the font cache using fc-list. This may take a moment.')
yt : [INFO     ] 2016-09-27 14:12:39,189 xlim = 0.000000 1.000000
yt : [INFO     ] 2016-09-27 14:12:39,192 ylim = 0.000000 1.000000
Parsing Hierarchy : 100%|██████████| 173/173 [00:00<00:00, 4740.75it/s]
yt : [INFO     ] 2016-09-27 14:12:39,301 Gathering a field list (this may take a moment.)
yt : [INFO     ] 2016-09-27 14:12:43,657 Making a fixed resolution buffer of (density) 800 by 800
yt : [INFO     ] 2016-09-27 14:12:50,631 Making a fixed resolution buffer of (('gas', 'density')) 800 by 800
```


Frontend
Filesystems
/home/user/data/
/home/user
Execution Environment
jupyter
rstudio
command line
etc


Involatile
Working
Frontend
Filesystems
/home/user/data/
/home/user
Execution Environment
jupyter notebook
jupyter lab
command line
etc


Client
8888, etc
Involatile
Working
Frontend
docker
Filesystems
/home/user/data/
/home/user
Execution Environment
jupyter
rstudio
command line
etc


Client
8888, etc
Composition
Involatile
Working
Frontend
docker
Filesystems
/home/user/data/
/home/user
Execution Environment
jupyter
rstudio
command line
etc


yt Hub
Quick search...
xarthisius
Collections
Users
Groups
« Prev
Page 1
Next »
Search collections...
Create collection
cluster_mergers
Show description
Created on September 1, 2016 at 12:34
Currently using 367.5 GB
Dark Sky Simulation
Show description
Created on July 12, 2016 at 9:42
Currently using 31.34 TB
DES
Show description
Created on April 21, 2016 at 21:05
Currently using 8.282 TB
Gaia DR1
Show description
Created on September 15, 2016 at 8:38
Currently using 751.7 GB
Galaxy (Goldbaum+ 2015)
Show description
Created on May 13, 2016 at 16:33
Currently using 7.014 TB
Galaxy (Goldbaum+ 2016)
Show description
Created on May 13, 2016 at 16:32
Currently using 5.923 TB
LIS590DV - Spring 2017
Show description
Created on January 27, 2017 at 9:19
Currently using 27.79 MB
Pop II Prime
Show description
Created on May 9, 2016 at 10:28
Currently using 260.8 GB
pore grains
Created on April 21, 2016 at 14:47
Currently using 30.84 MB
Renaissance Simulations
Show description
Created on November 9, 2016 at 10:09
Currently using 214.5 GB

Collections and Filesystems

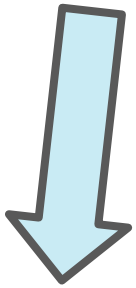
	Dataset 1
	Dataset 2
	Dataset 3

Collections and Filesystems

X	Dataset 1
	Dataset 2
X	Dataset 3

Collections and Filesystems

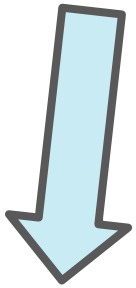
X	Dataset 1
	Dataset 2
X	Dataset 3



```
{ [ “dataset1”, [...], ...,
  [ “dataset3”, [...], ...  }
```

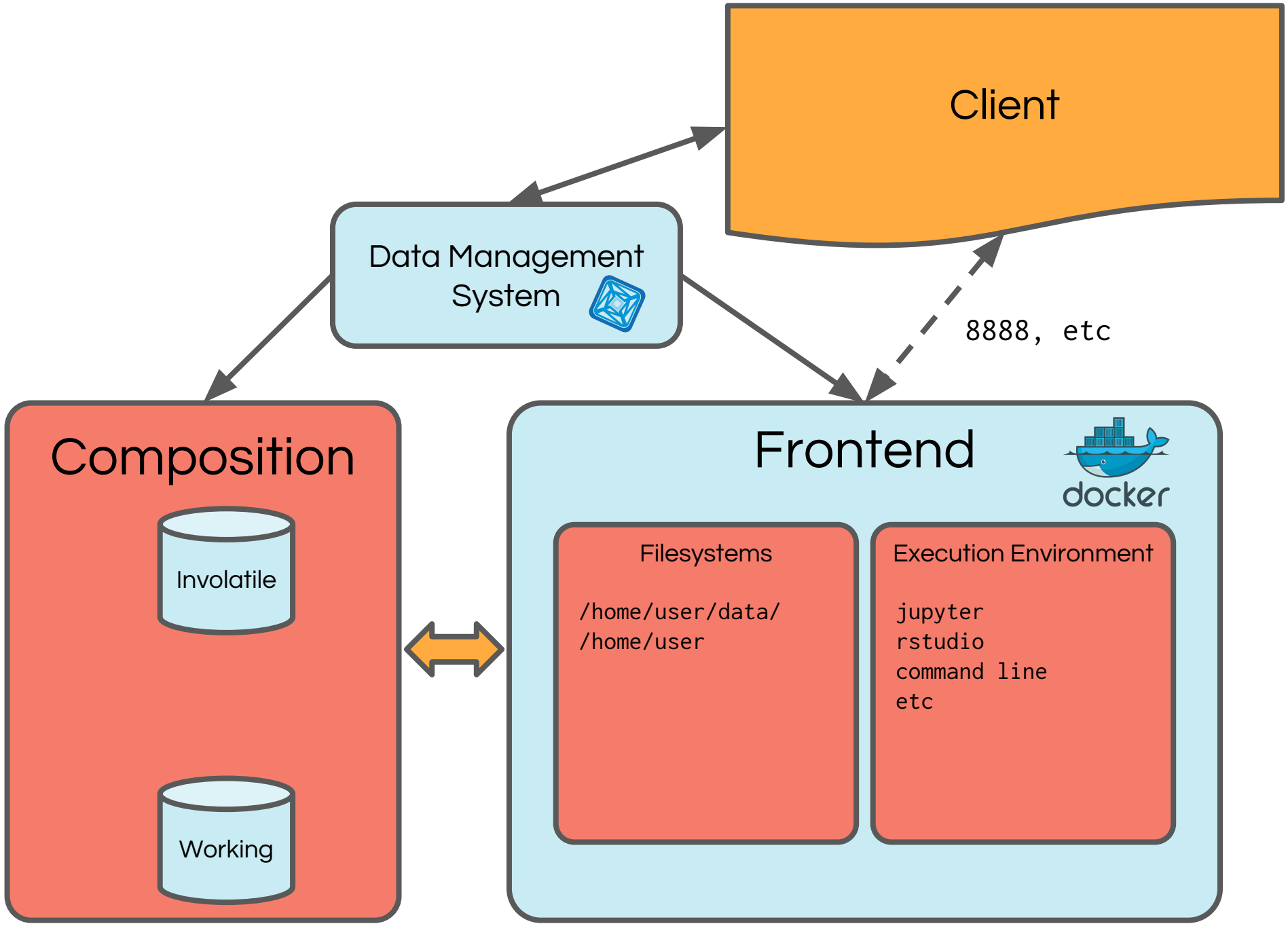
Collections and Filesystems

X	Dataset 1
	Dataset 2
X	Dataset 3

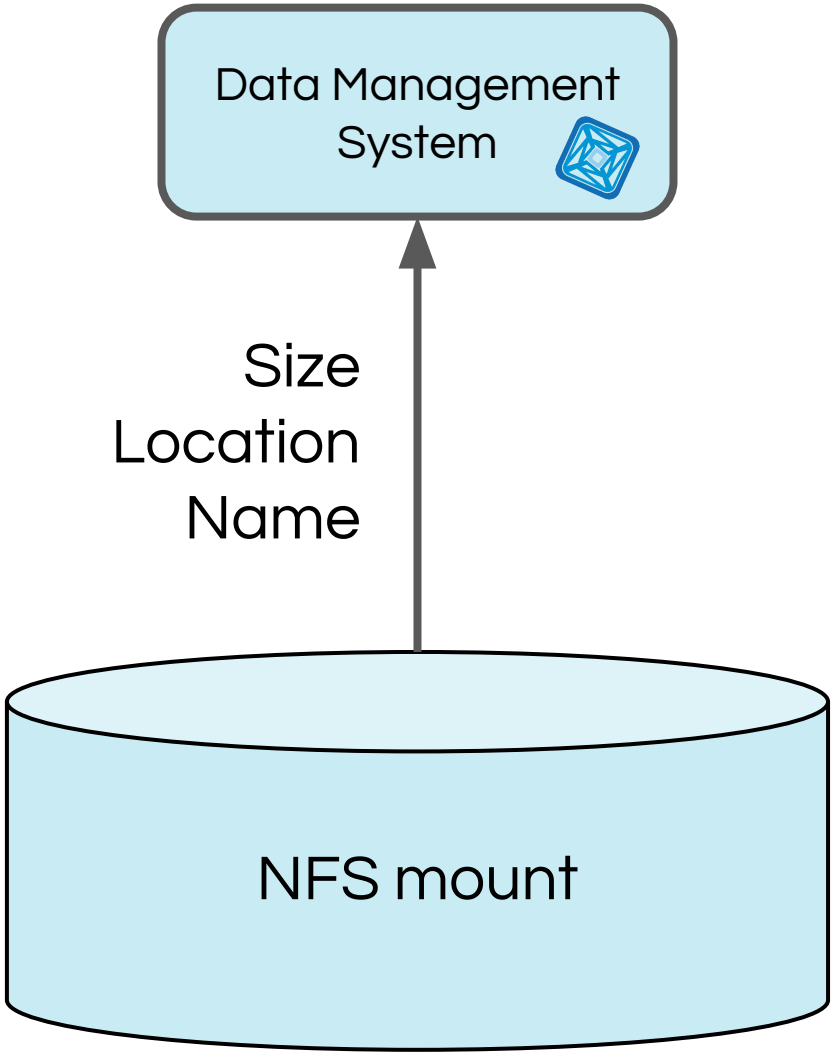


```
{ [ “dataset1”, [...], ...,
  [ “dataset3”, [...], ...  }
```

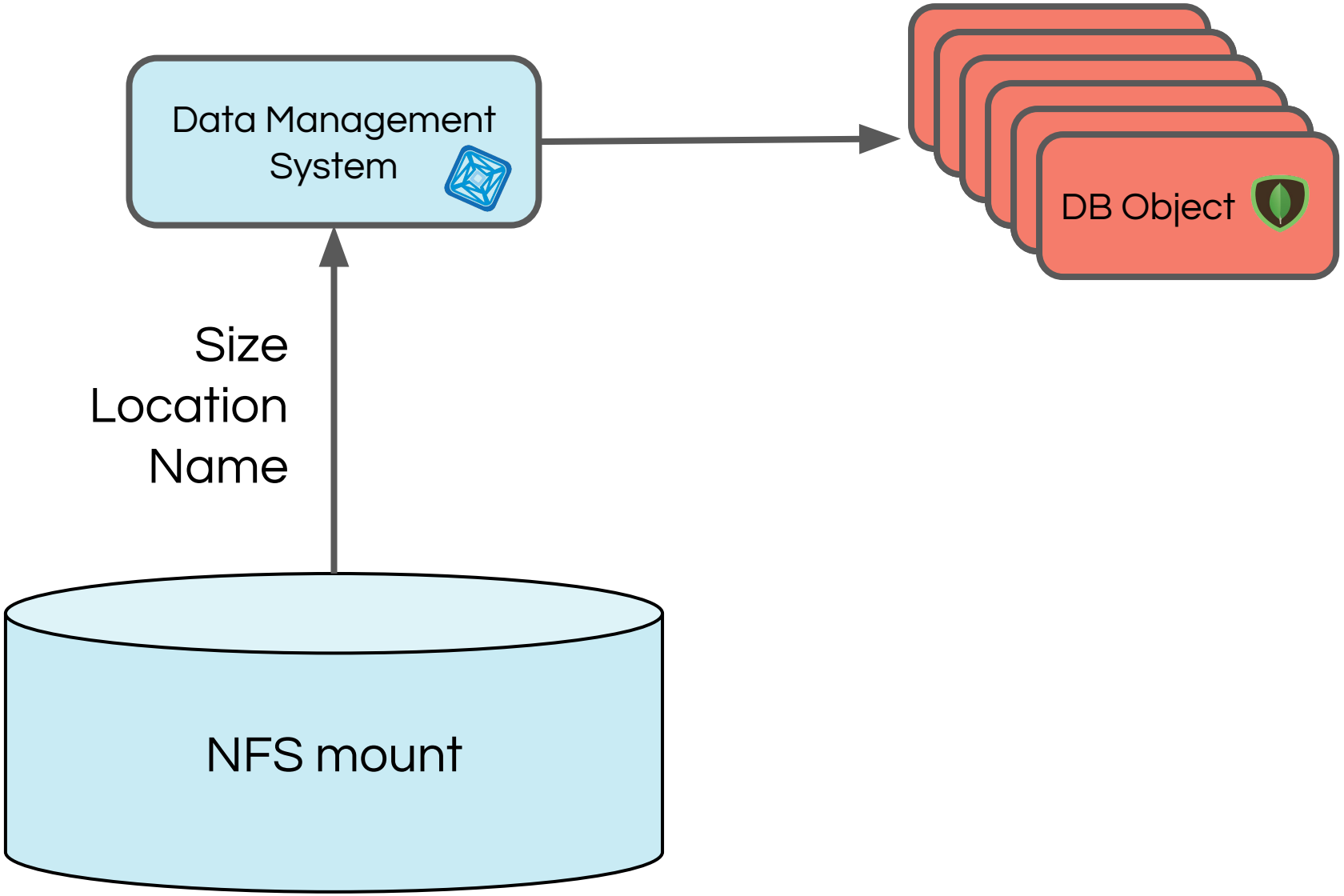
/data/
/data/dataset1/...
/data/dataset3/...


Client
Data Management System
8888, etc
Composition
Involatile
Working
Frontend
docker
Filesystems
/home/user/data/
/home/user
Execution Environment
jupyter
rstudio
command line
etc

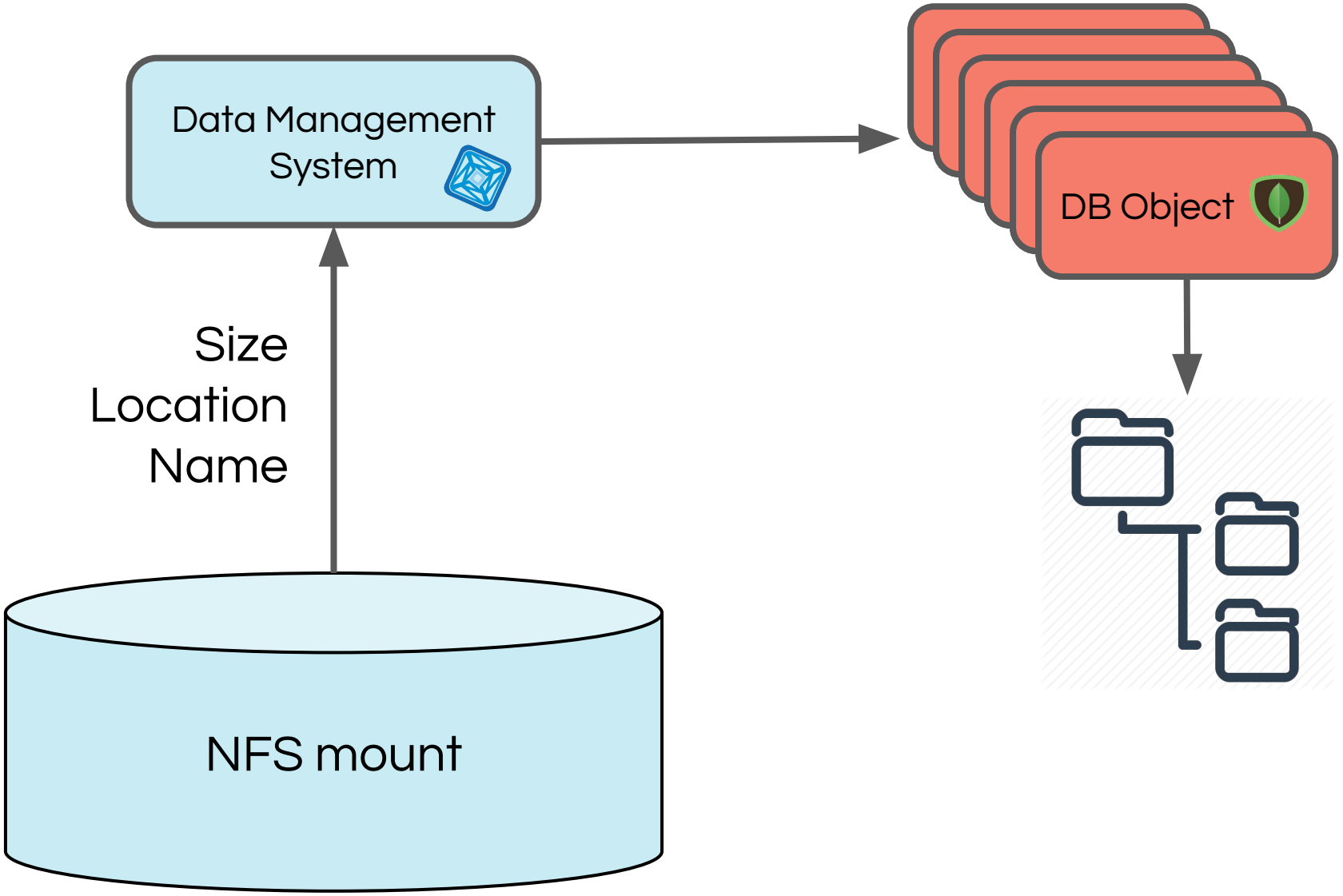
Under the hood



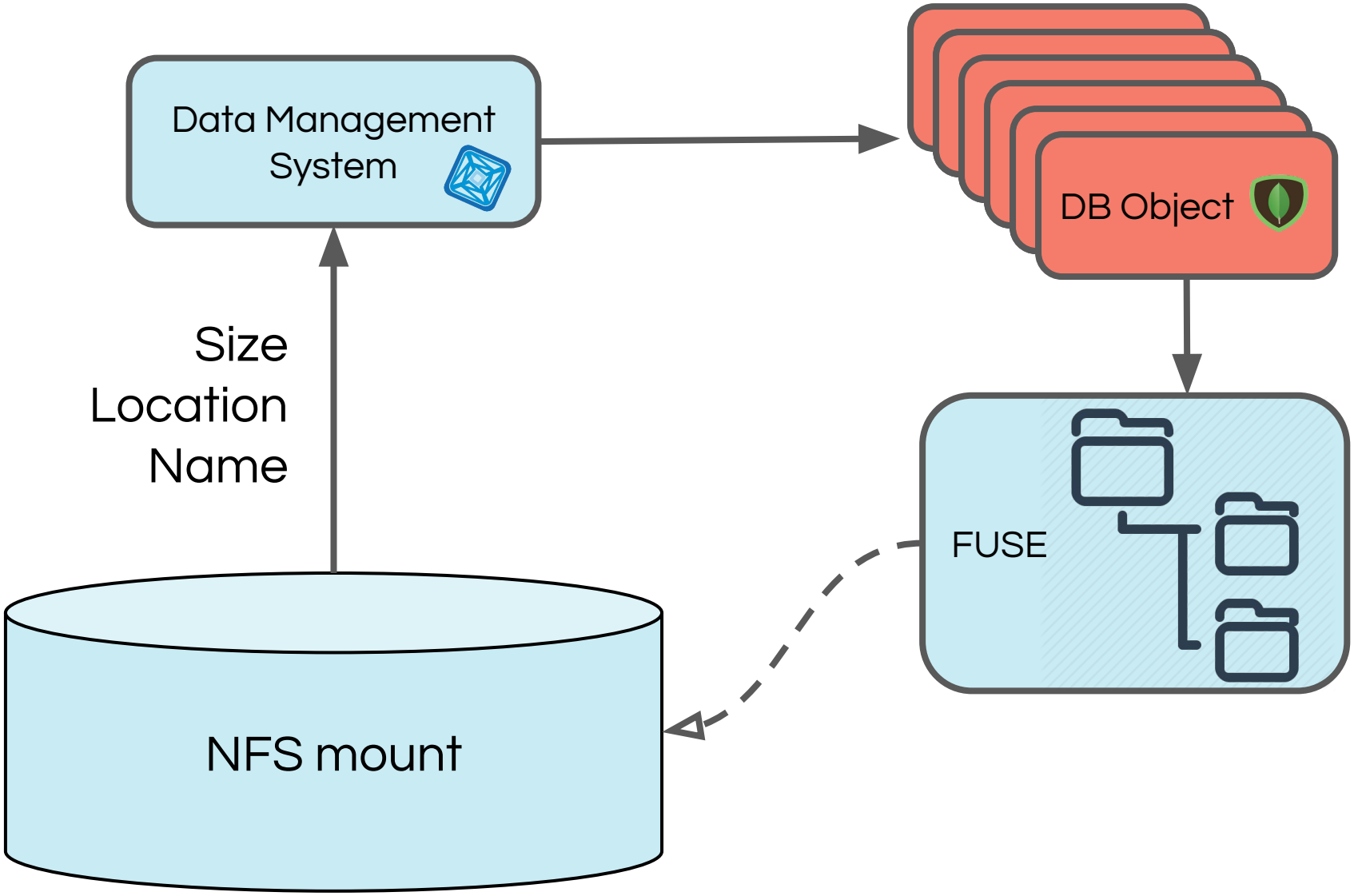
Under the hood



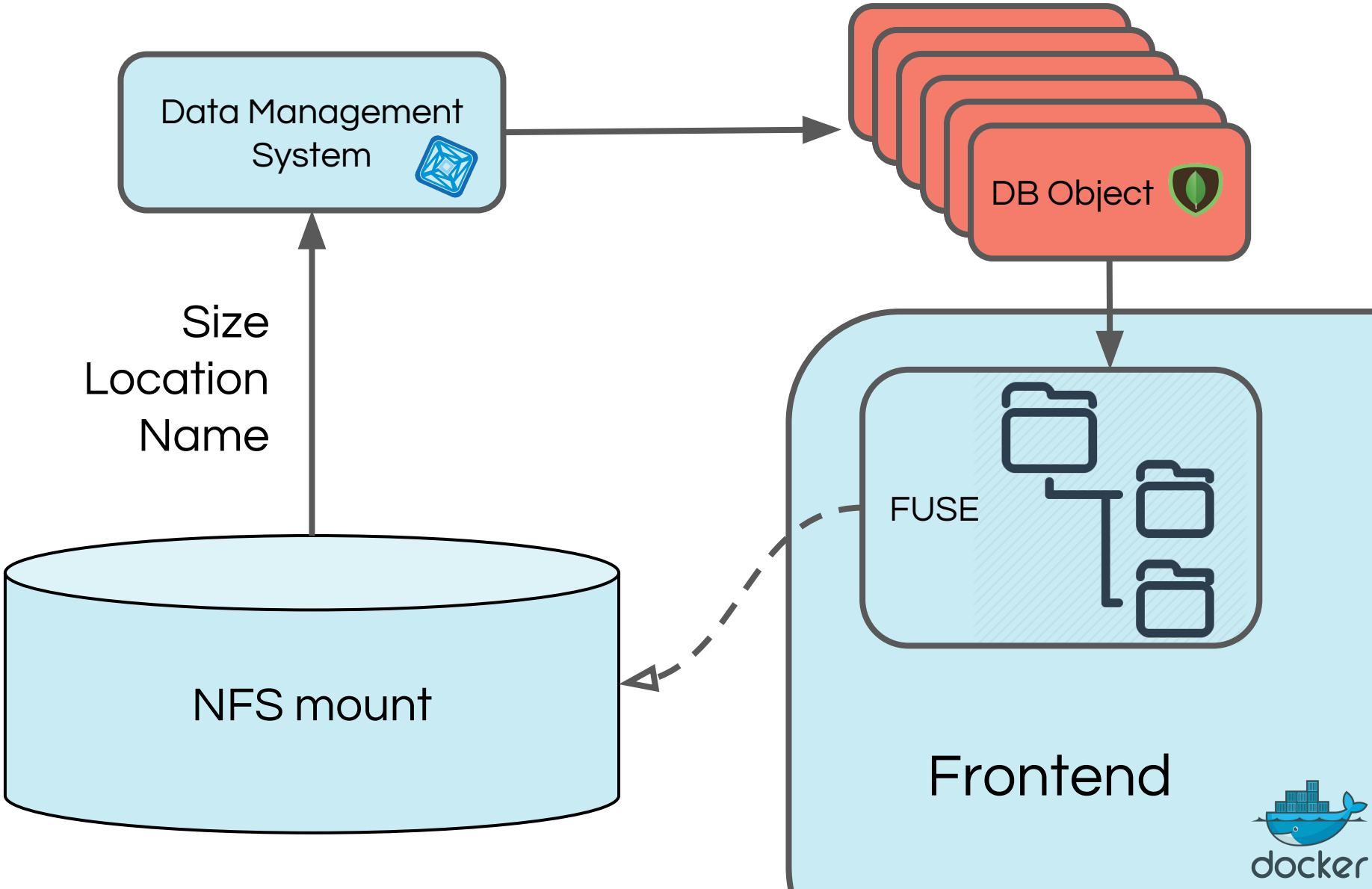
Under the hood



Under the hood



Under the hood



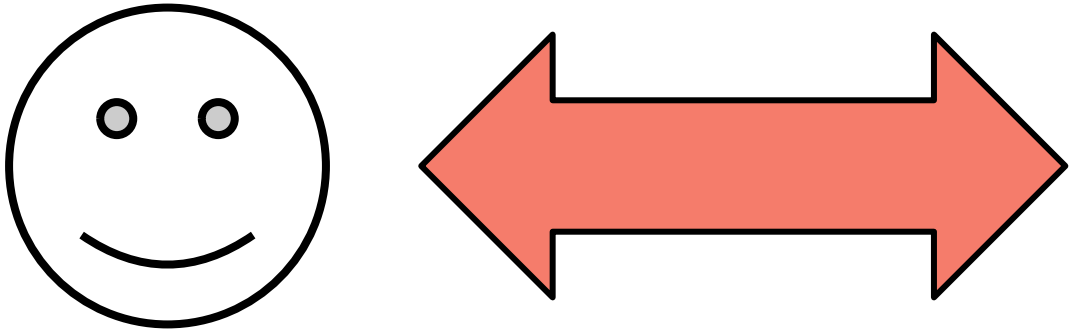
Whole Tale Architecture

Bertram Ludäscher, Adam Brinckman, Kyle Chard, Niall Gaffney, Mihael Hategan, Matthew B. Jones, Kacper Kowalik, Sivakumar Kulasekaran, Bryce Mecum, Jarosław Nabrzyski, Victoria Stodden, Ian Taylor, Matthew Turk, and Kandace Turner

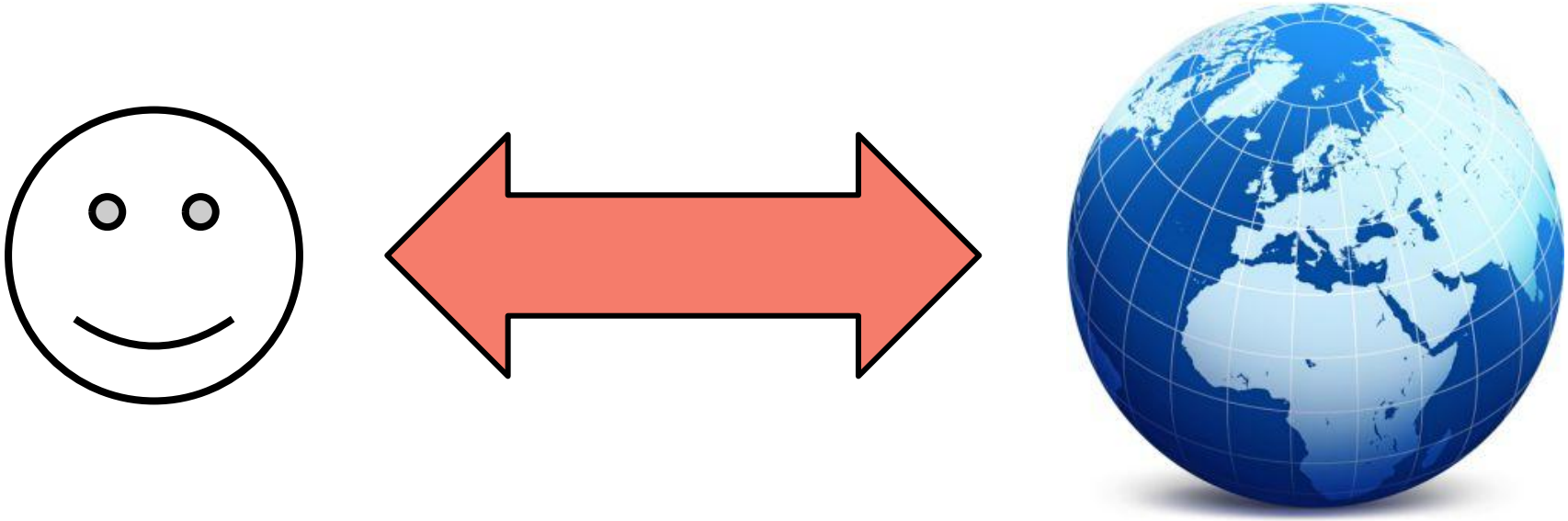
Concepts

1. Logging in
2. Ingesting data
3. Sharing data
4. Analyzing data
5. Storing artifacts
6. Provenance
7. Publishing

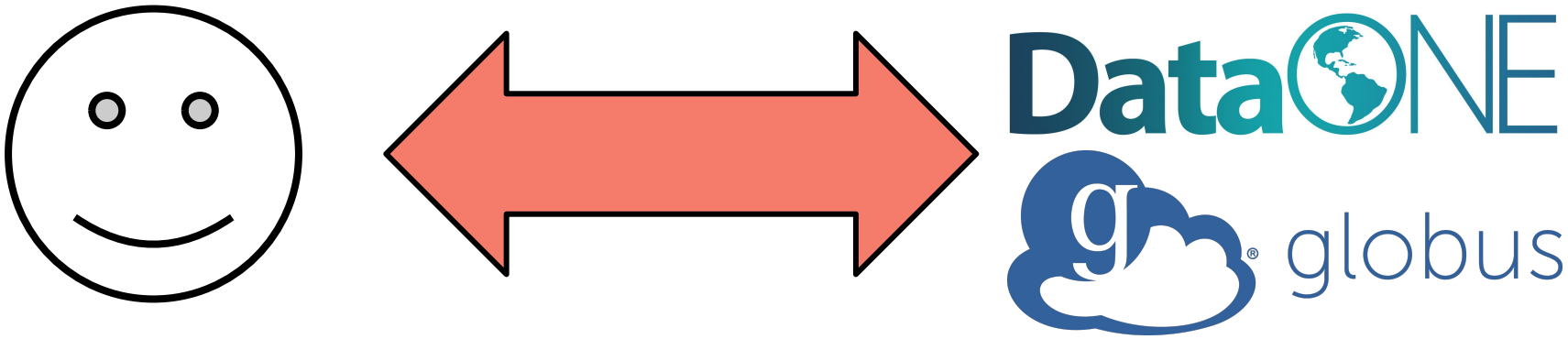
Driving force

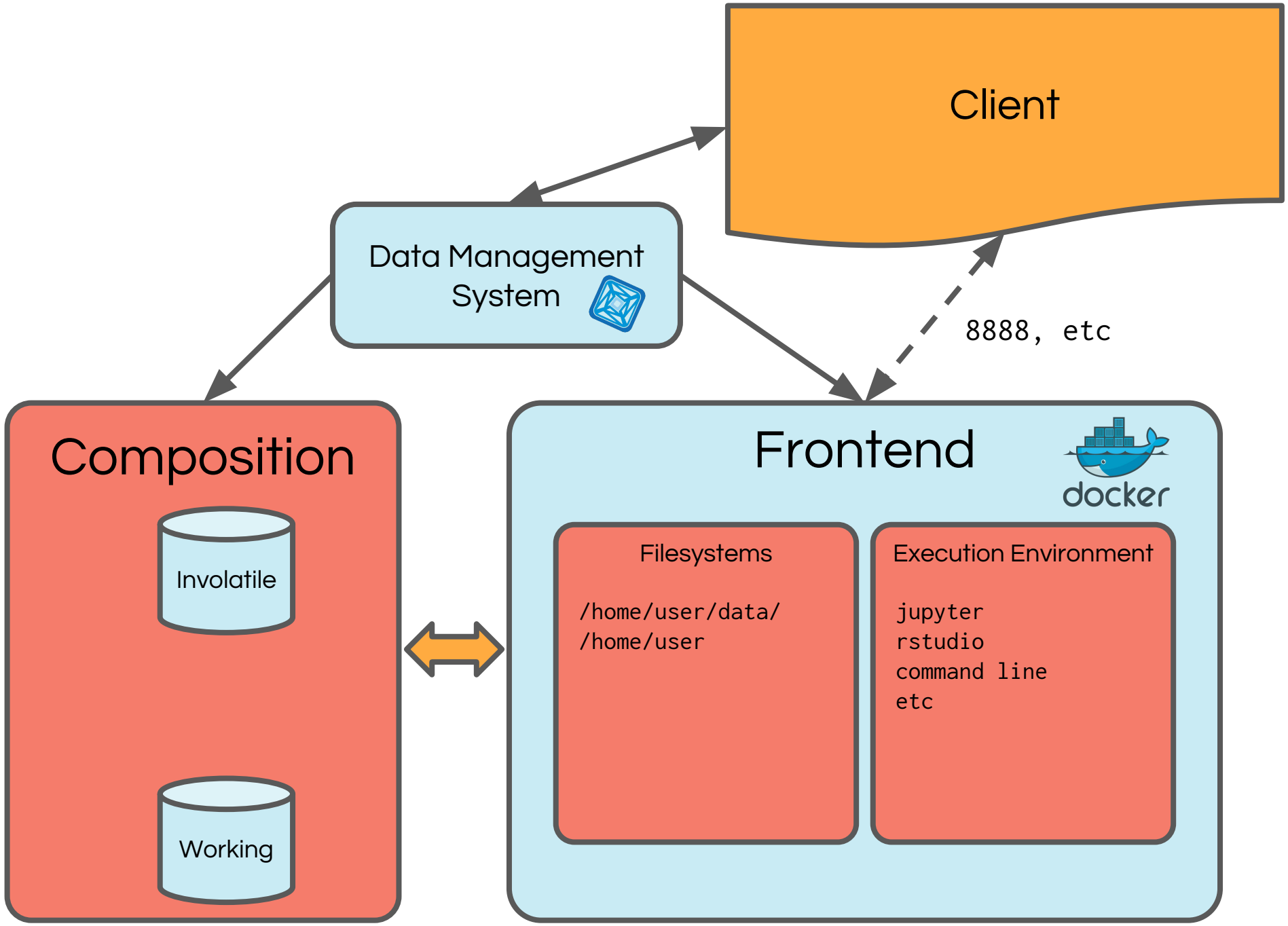


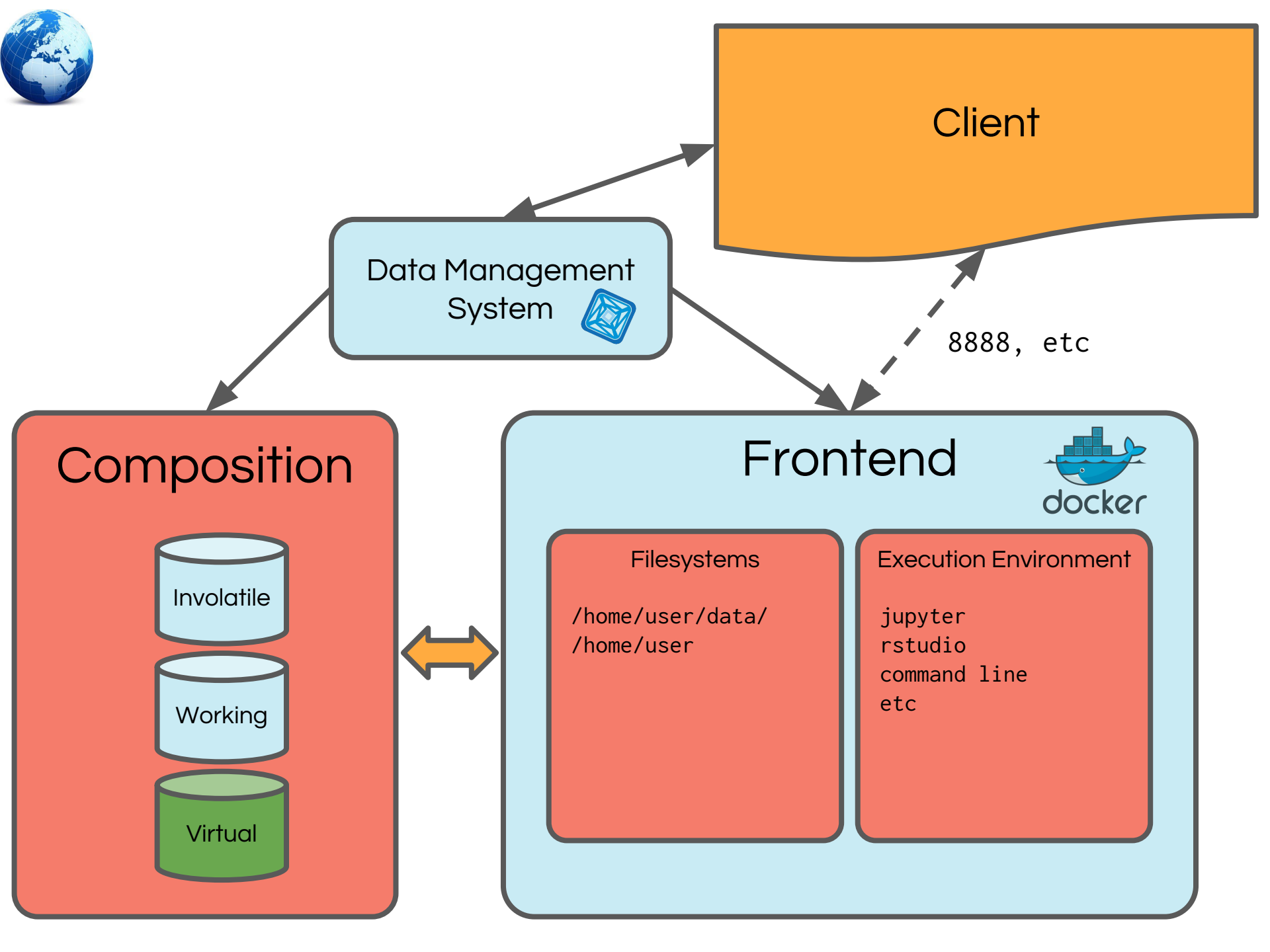
Driving force



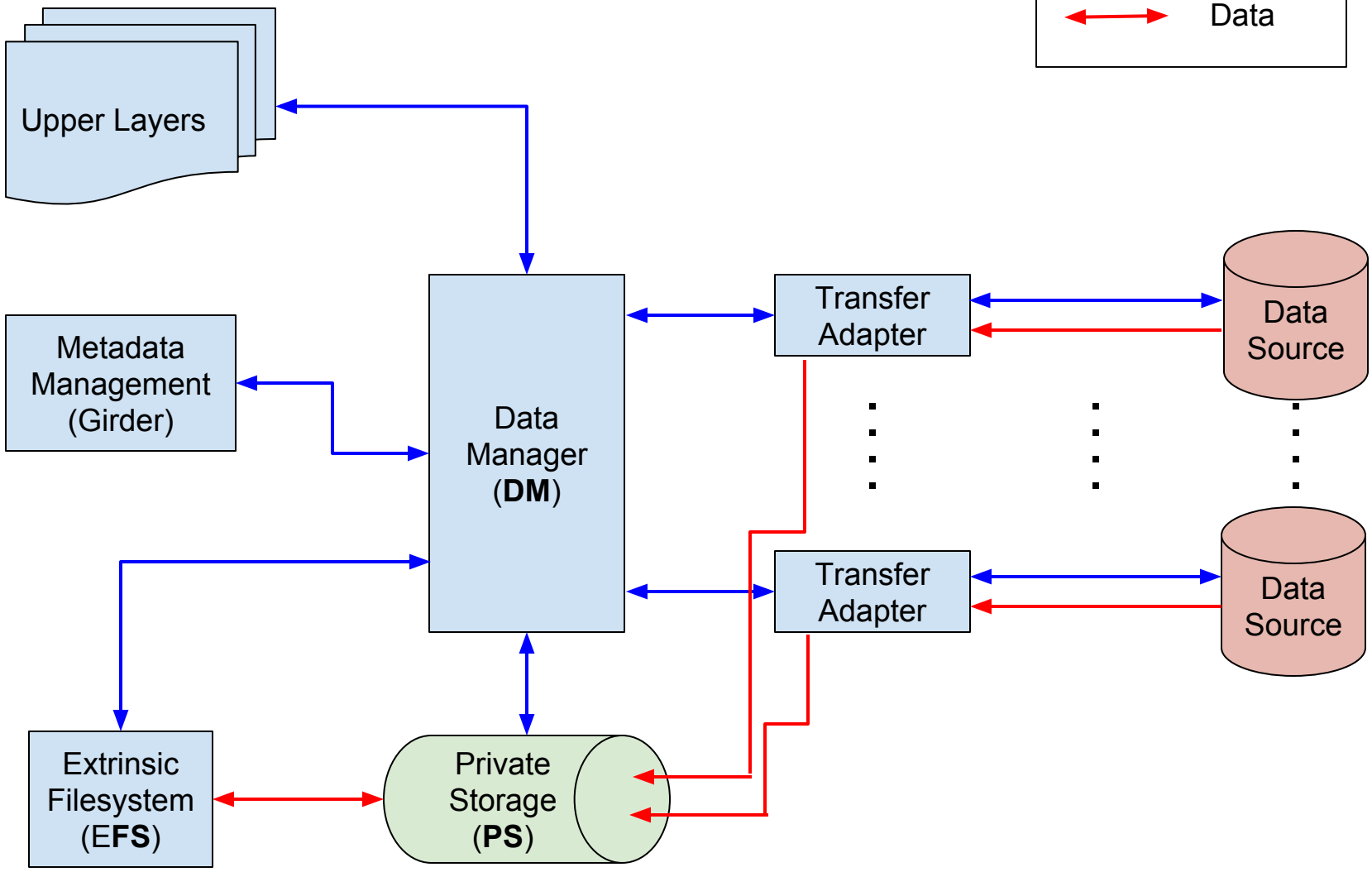
Driving force

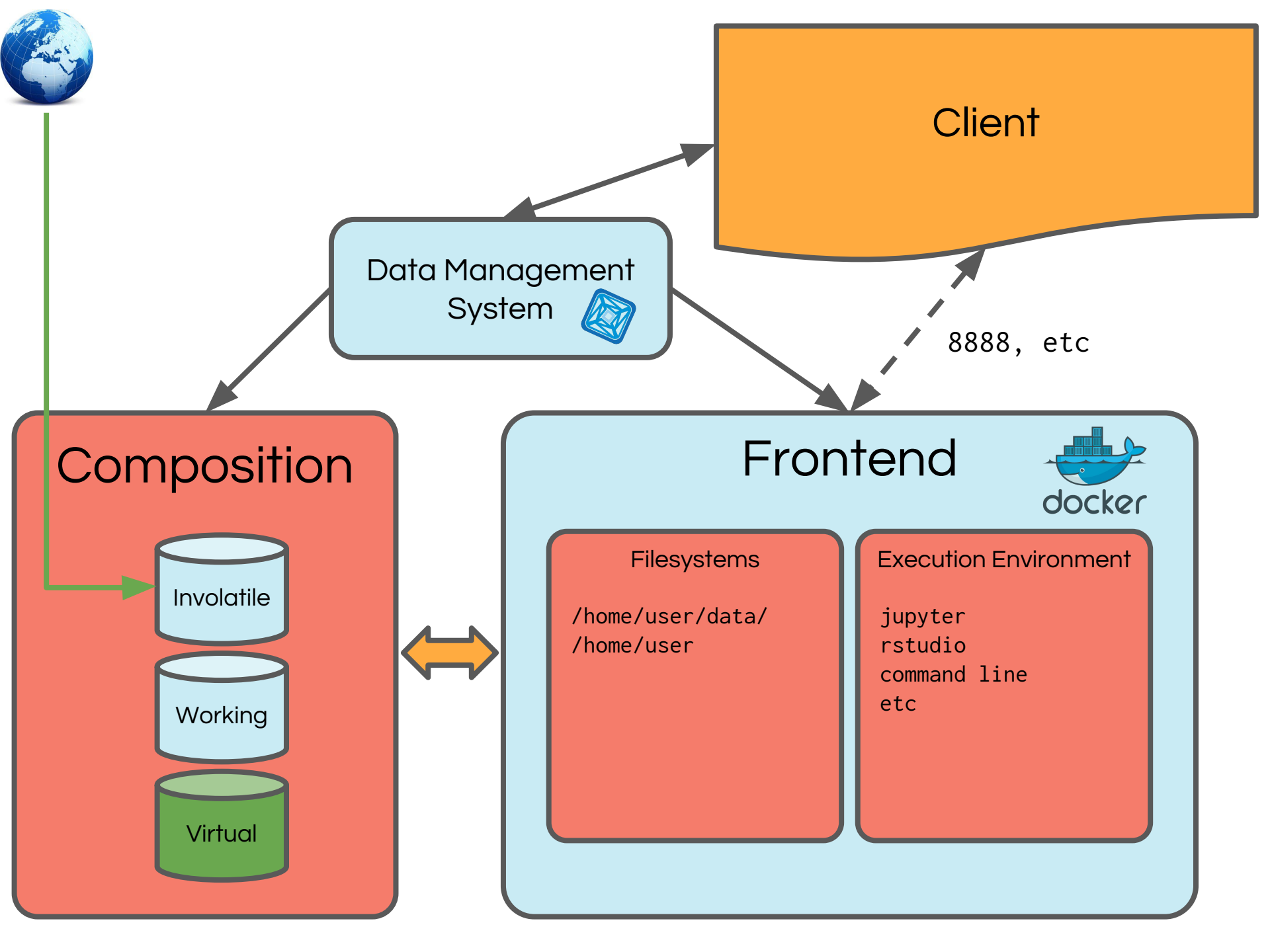


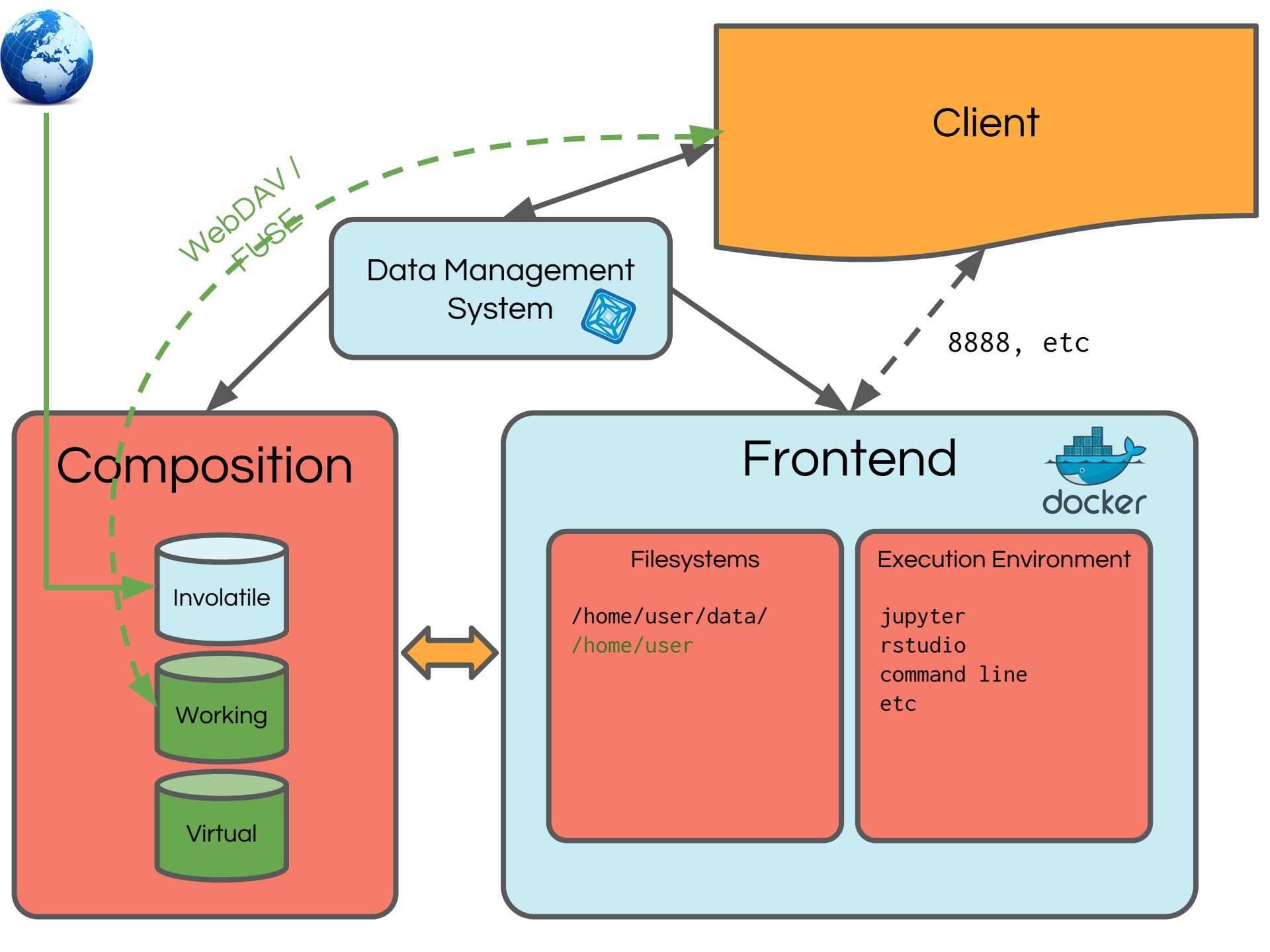

Client
Data Management System
8888, etc
Composition
Involatile
Working
Frontend
docker
Filesystems
/home/user/data/
/home/user
Execution Environment
jupyter
rstudio
command line
etc

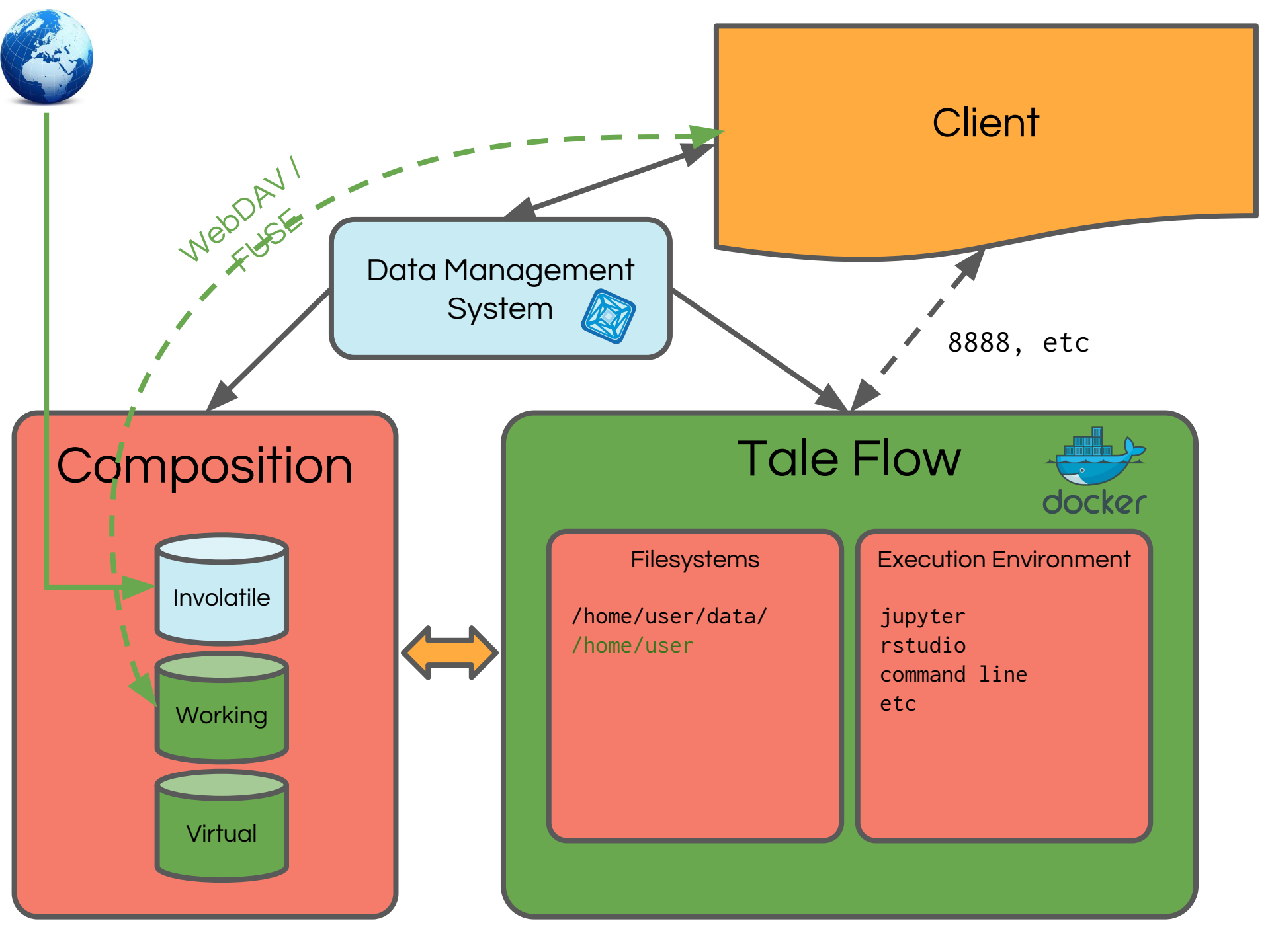

Client
Data Management System
8888, etc
Composition
Involatile
Working
Virtual
Frontend
docker
Filesystems
/home/user/data/
/home/user
Execution Environment
jupyter
rstudio
command line
etc

WT Data Management Plugin




Client
Data Management System
8888, etc
Composition
Involatile
Working
Virtual
Frontend
docker
Filesystems
/home/user/data/
/home/user
Execution Environment
jupyter
rstudio
command line
etc

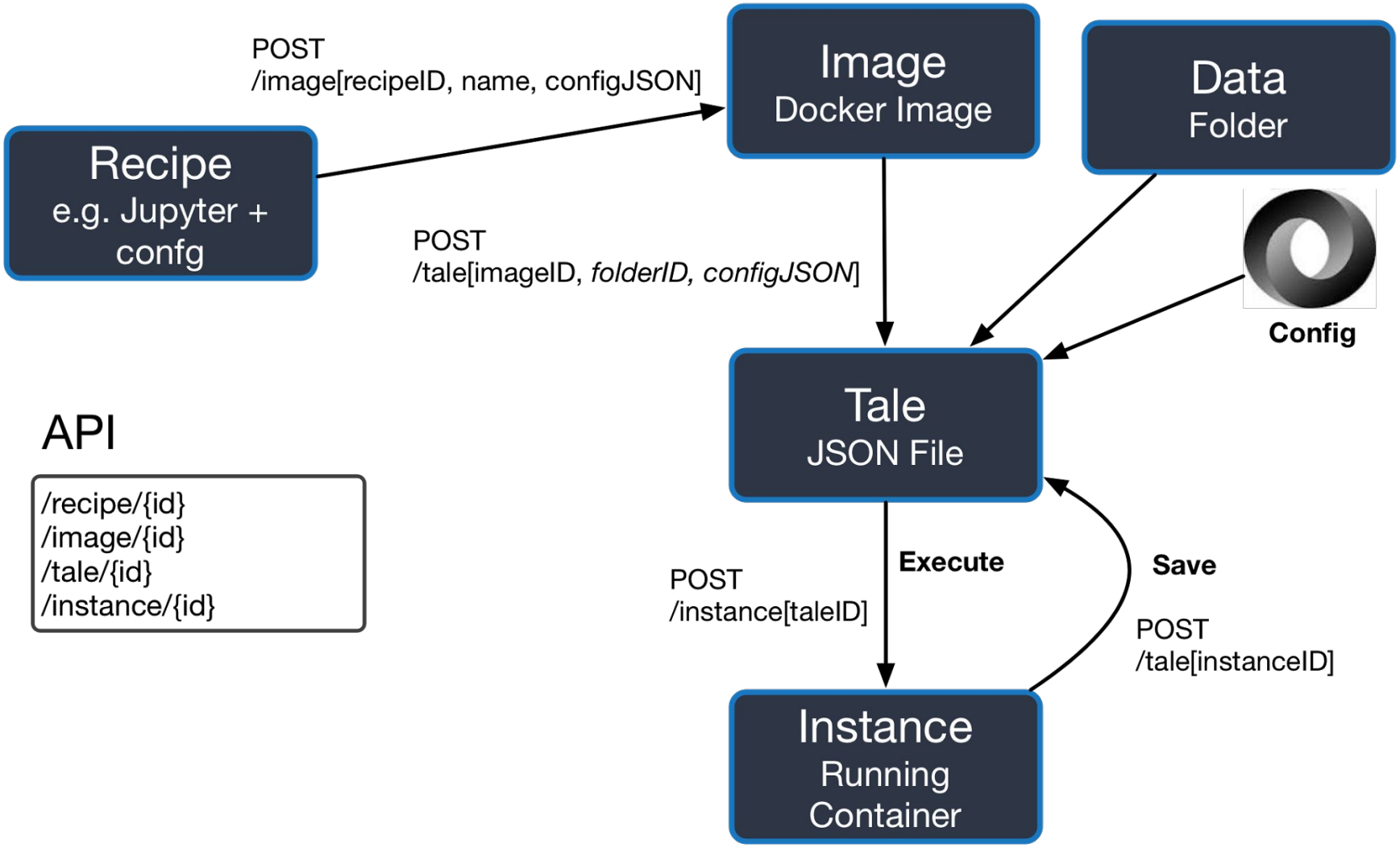

Client
WebDAV /
FUSE
Data Management
System
8888, etc
Composition
Involatile
Working
Virtual
Frontend
docker
Filesystems
/home/user/data/
/home/user
Execution Environment
jupyter
rstudio
command line
etc

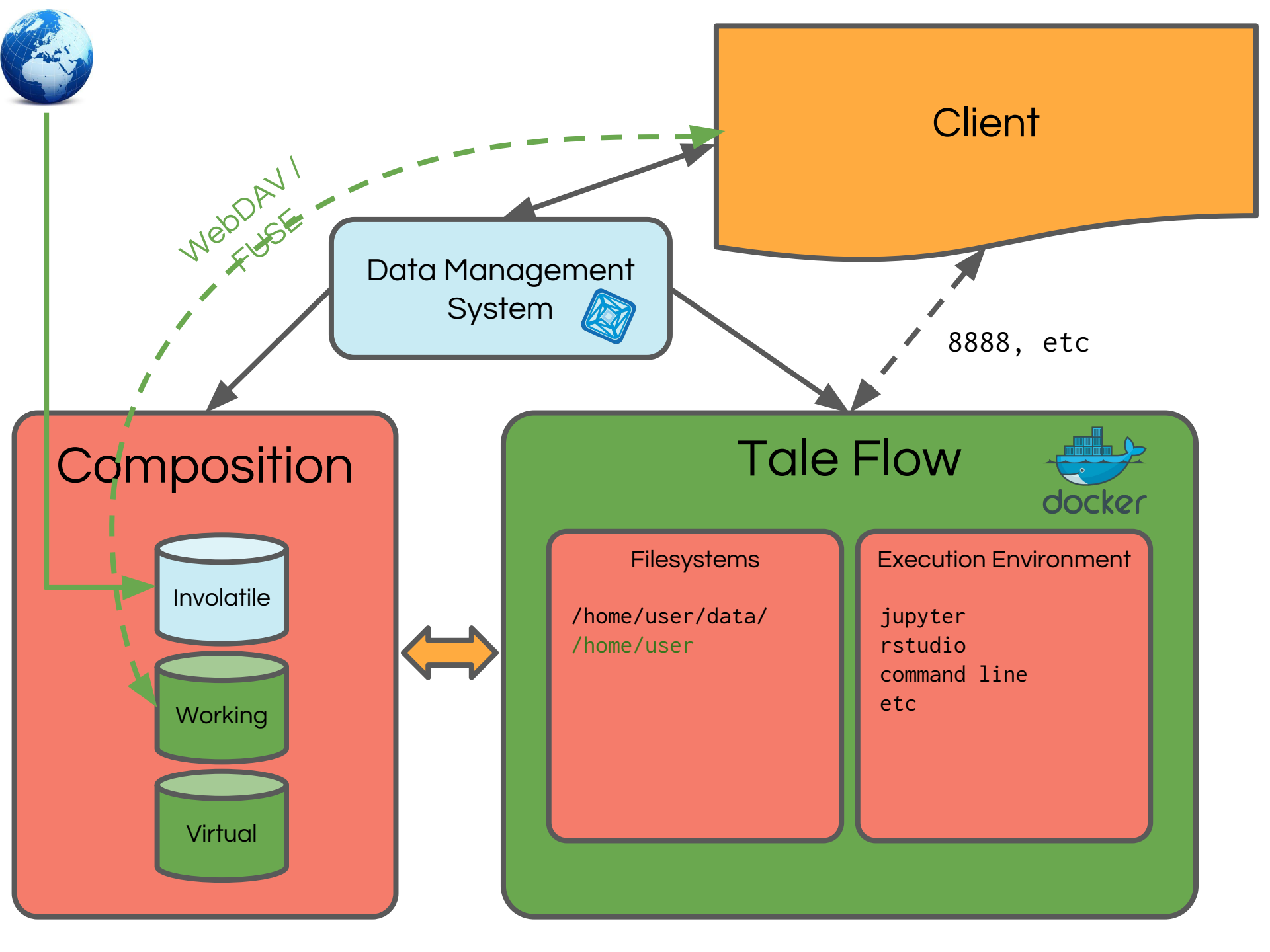

Client
WebDAV /
FUSE
Data Management
System
8888, etc
Composition
Involatile
Working
Virtual
Tale Flow
docker
Filesystems
/home/user/data/
/home/user
Execution Environment
jupyter
rstudio
command line
etc

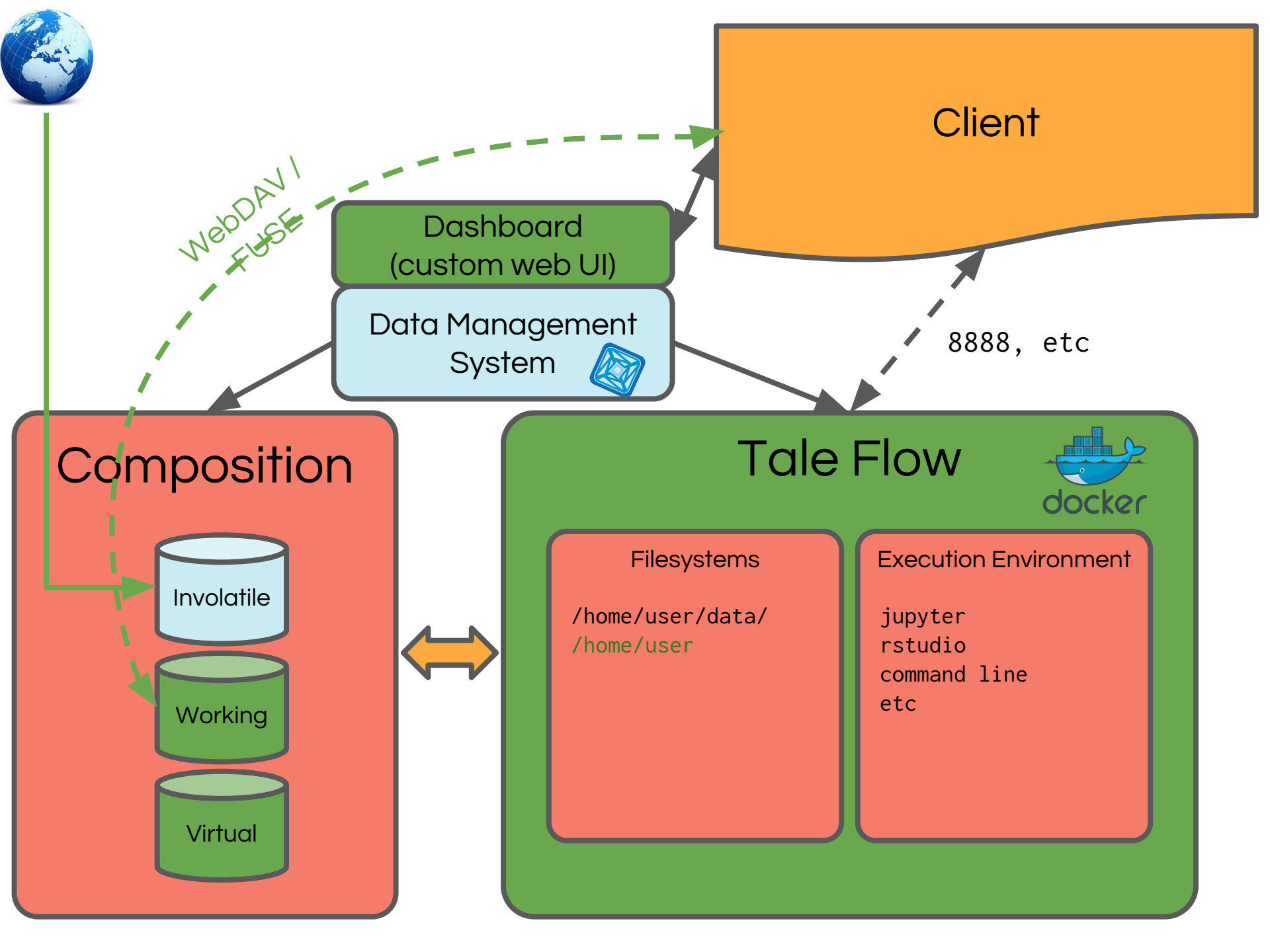
Whole **Tale** Flow

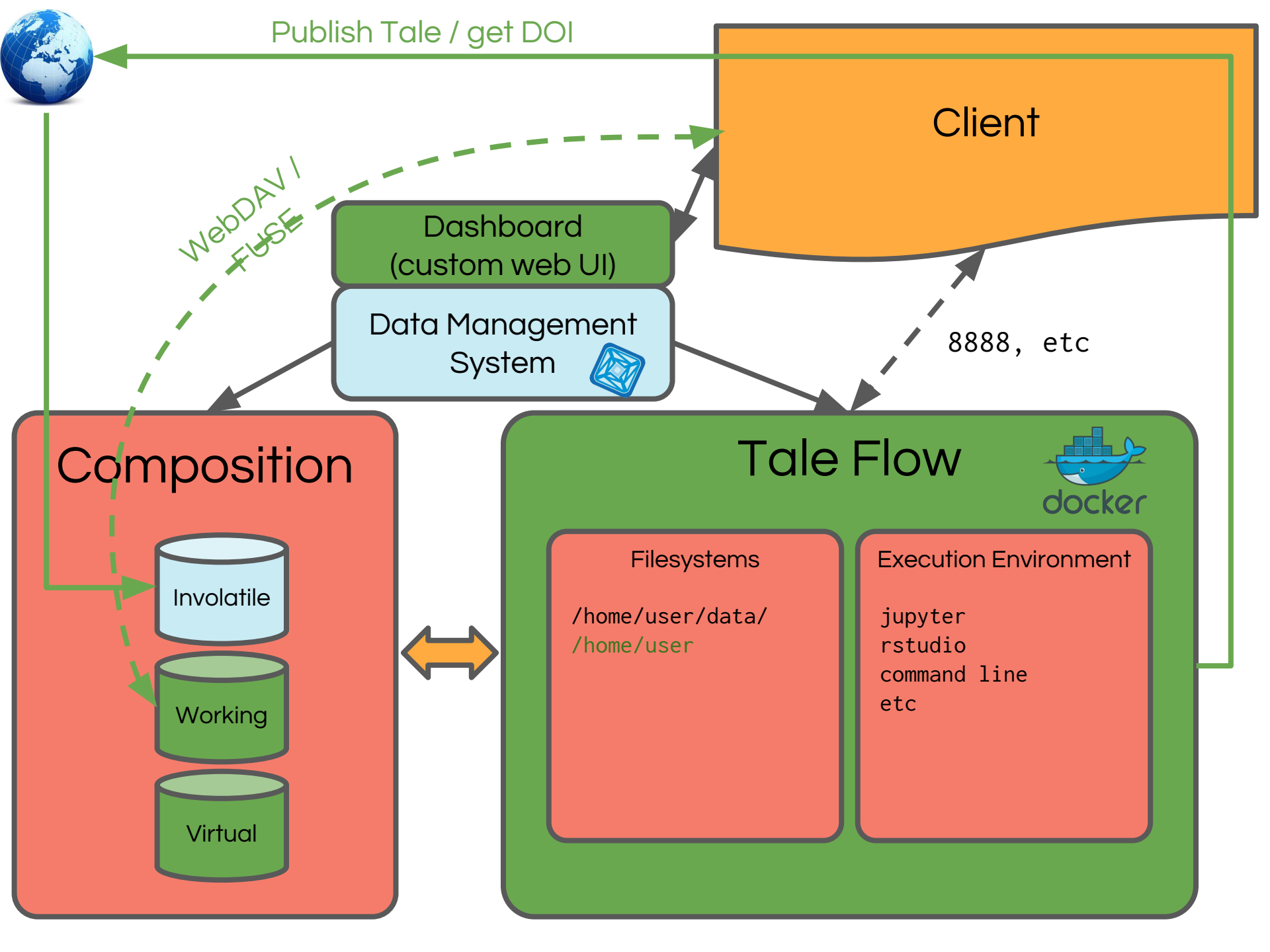
1. **Recipe** - describes how to build a working environment.
    - Current representation: a git(hub) repository buildable via `docker build` (url, commit)
    - Example: https://github.com/whole-tale/jupyter-yt
    - Immutable (keeping track of provenance)
2. **Image** - runnable research environment
    - Current representation: a docker image
    - Immutable (keeping track of provenance)
3. **Tale** - Image along with a default config and data reference
    - Current representation: a set of an Image, a girder folder, a JSON config for `docker start/run`
    - Immutable once published
4. **Instance** - a running Tale
    - Current representation: a docker instance
    - Interactive (or not), can be save as another tale (storing artifacts for publishing).

Whole **Tale** Flow




Client
WebDAV /
FUSE
Data Management
System
8888, etc
Composition
Involatile
Working
Virtual
Tale Flow
docker
Filesystems
/home/user/data/
/home/user
Execution Environment
jupyter
rstudio
command line
etc


Client
WebDAV /
FUSE
Dashboard
(custom web UI)
Data Management
System
8888, etc
Composition
Involatile
Working
Virtual
Tale Flow
docker
Filesystems
/home/user/data/
/home/user
Execution Environment
jupyter
rstudio
command line
etc


Publish Tale / get DOI
Client
WebDAV / FUSE
Dashboard
(custom web UI)
Data Management
System
8888, etc
Composition
Involatile
Working
Virtual
Tale Flow
docker
Filesystems
/home/user/data/
/home/user
Execution Environment
jupyter
rstudio
command line
etc