

OpenML

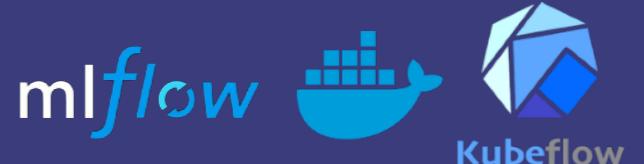
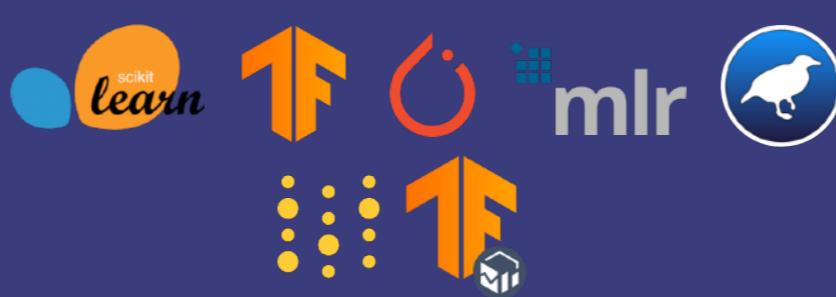
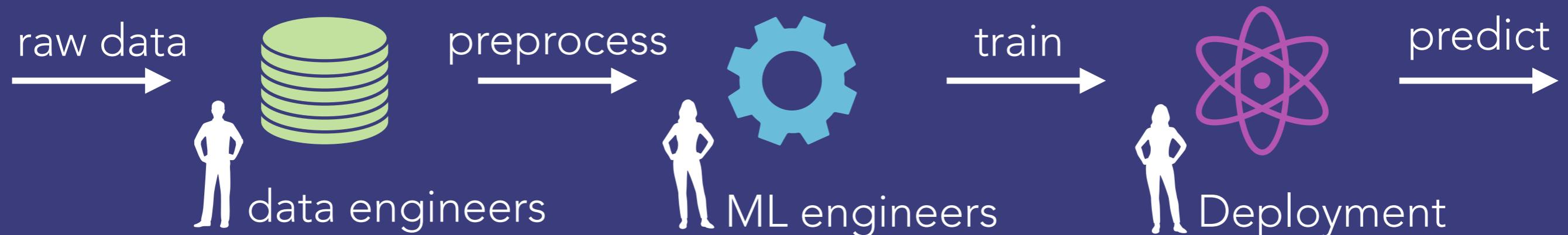
Sharing and reproducing
machine learning experiments

Joaquin Vanschoren, TU/e
and the OpenML team



Machine Learning: art or science?

Process with many actors and tools (model *lifecycle*)

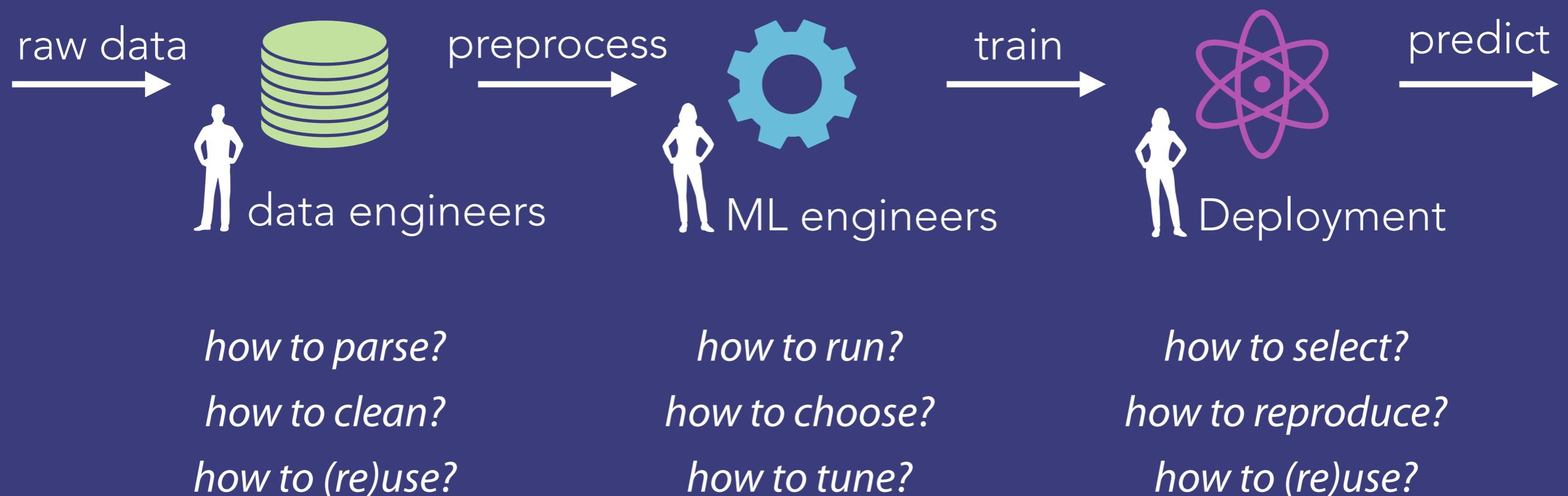




Machine Learning: art or science?

Process with many actors and tools (model *lifecycle*)

Requires *significant* knowledge and experience





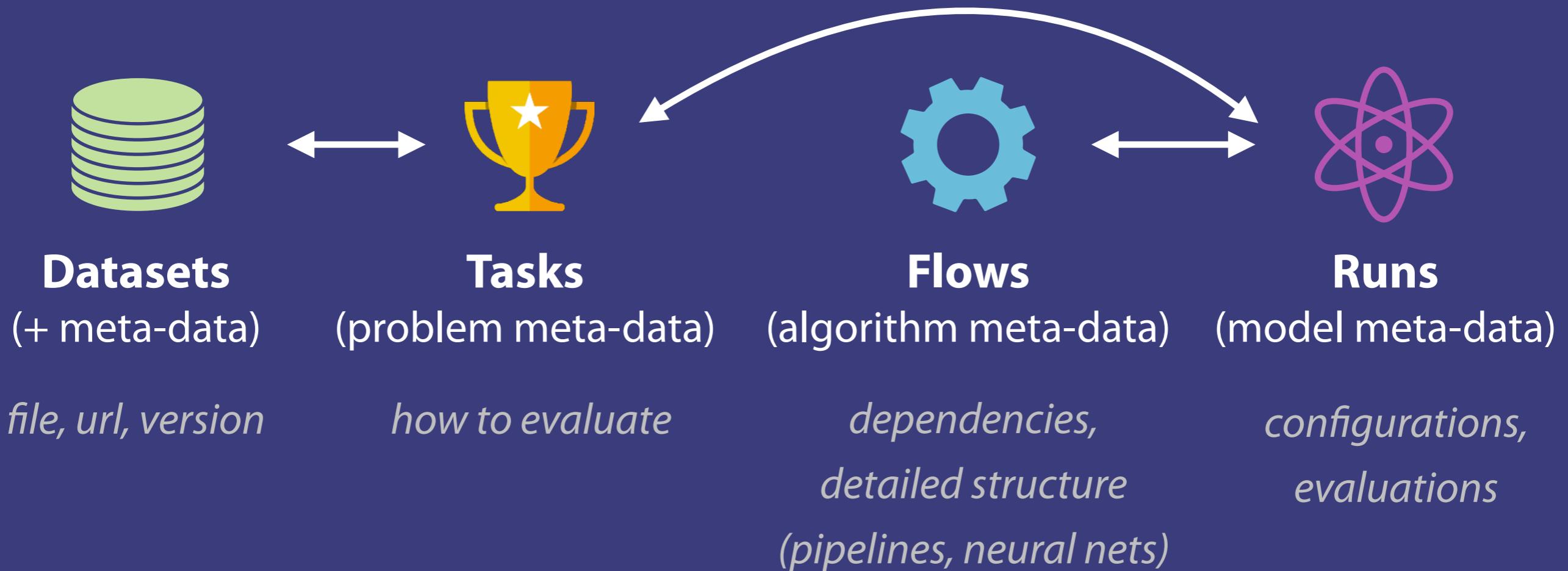
*Can we organize all our results and
learn from this prior experience?*

*What if...
we could organize the world's
machine learning information

and make it universally
accessible and useful?*



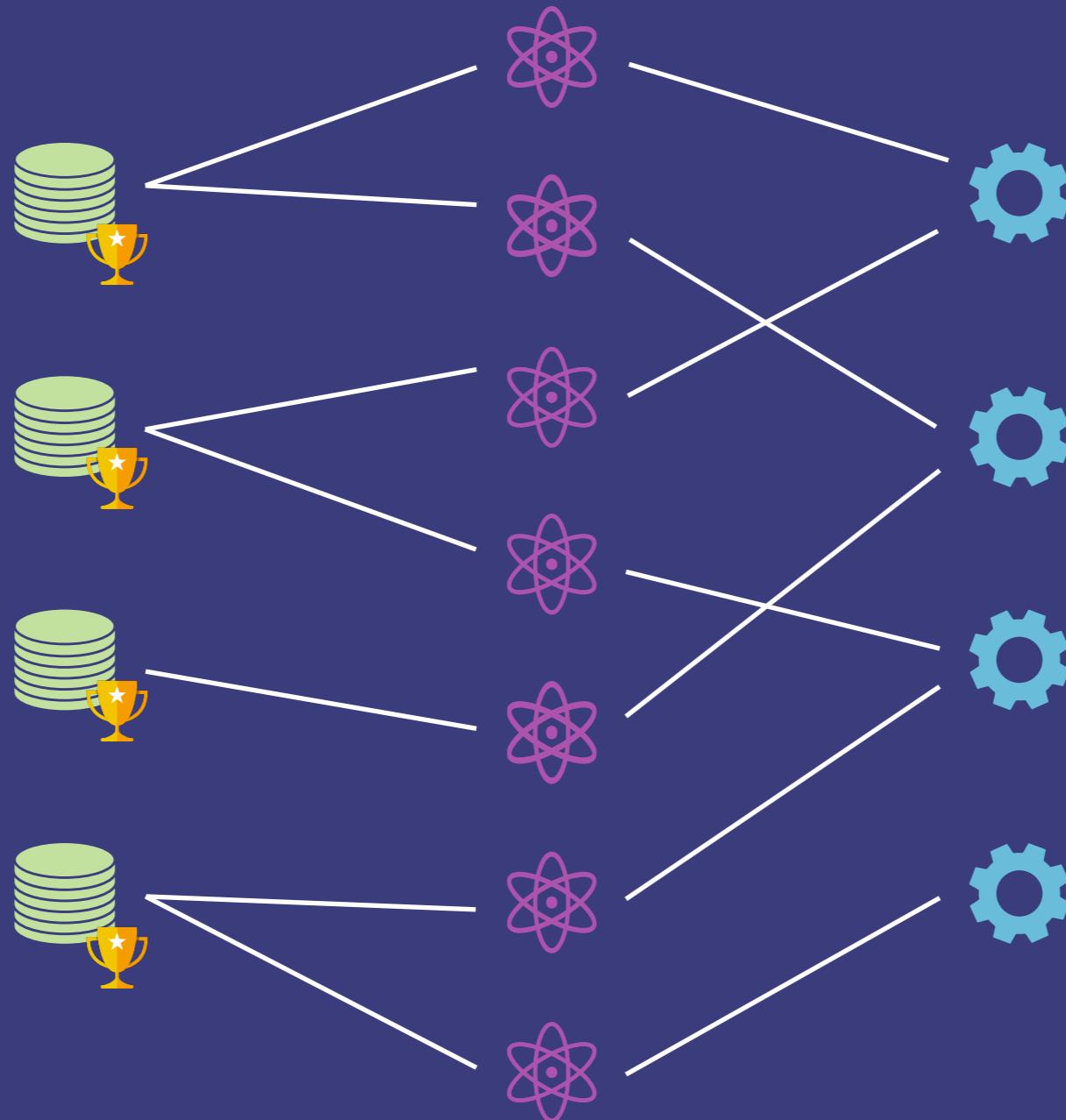
Machine Learning components



For every **dataset**, find all models built (and which are best)

For every **model**, get the *exact* dataset and algorithm used

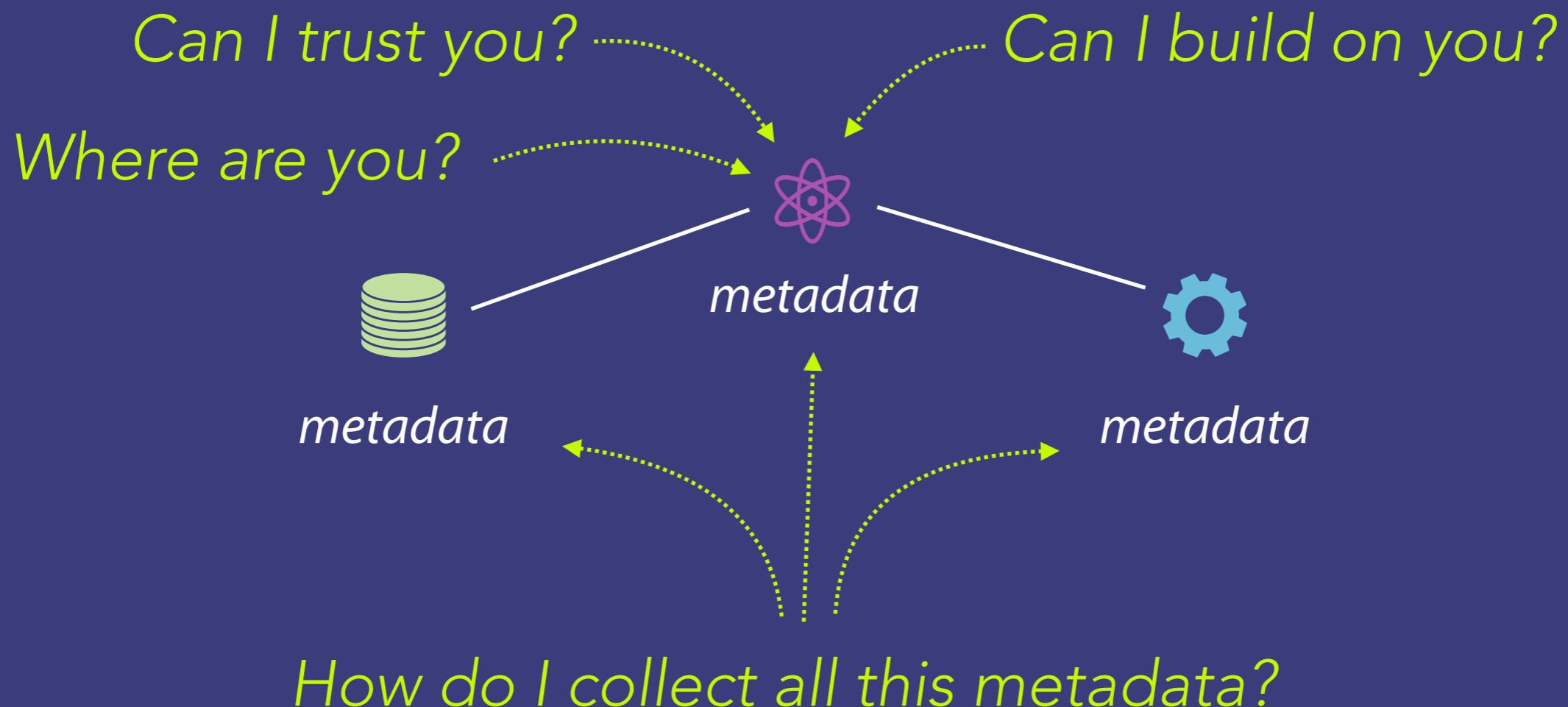
For every **algorithm**, find how useful it is for every dataset



Reproducibility

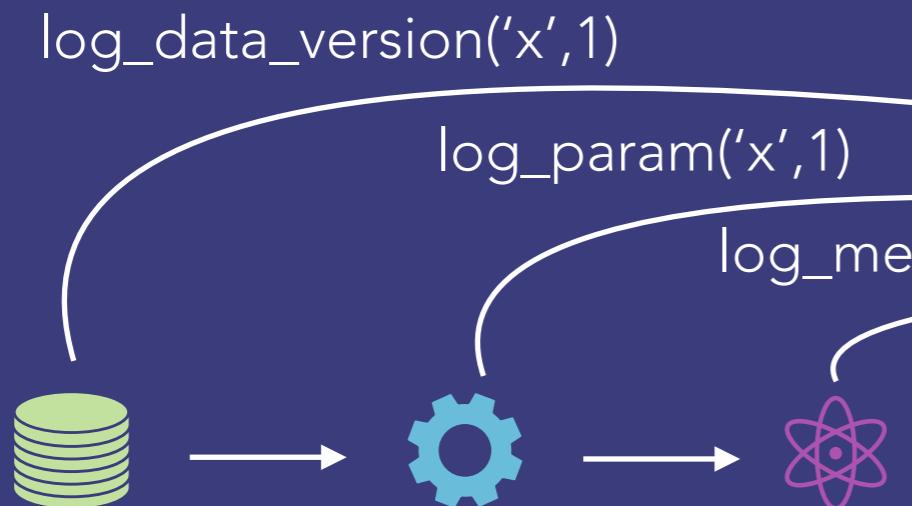
For every model, get the *exact* dataset and algorithm used

How do I *reproduce* this result?



Reproducibility

System of execution



*flexible, lightweight
manual annotation, heterogeneous metadata
data, code,... outside of system*

System of record

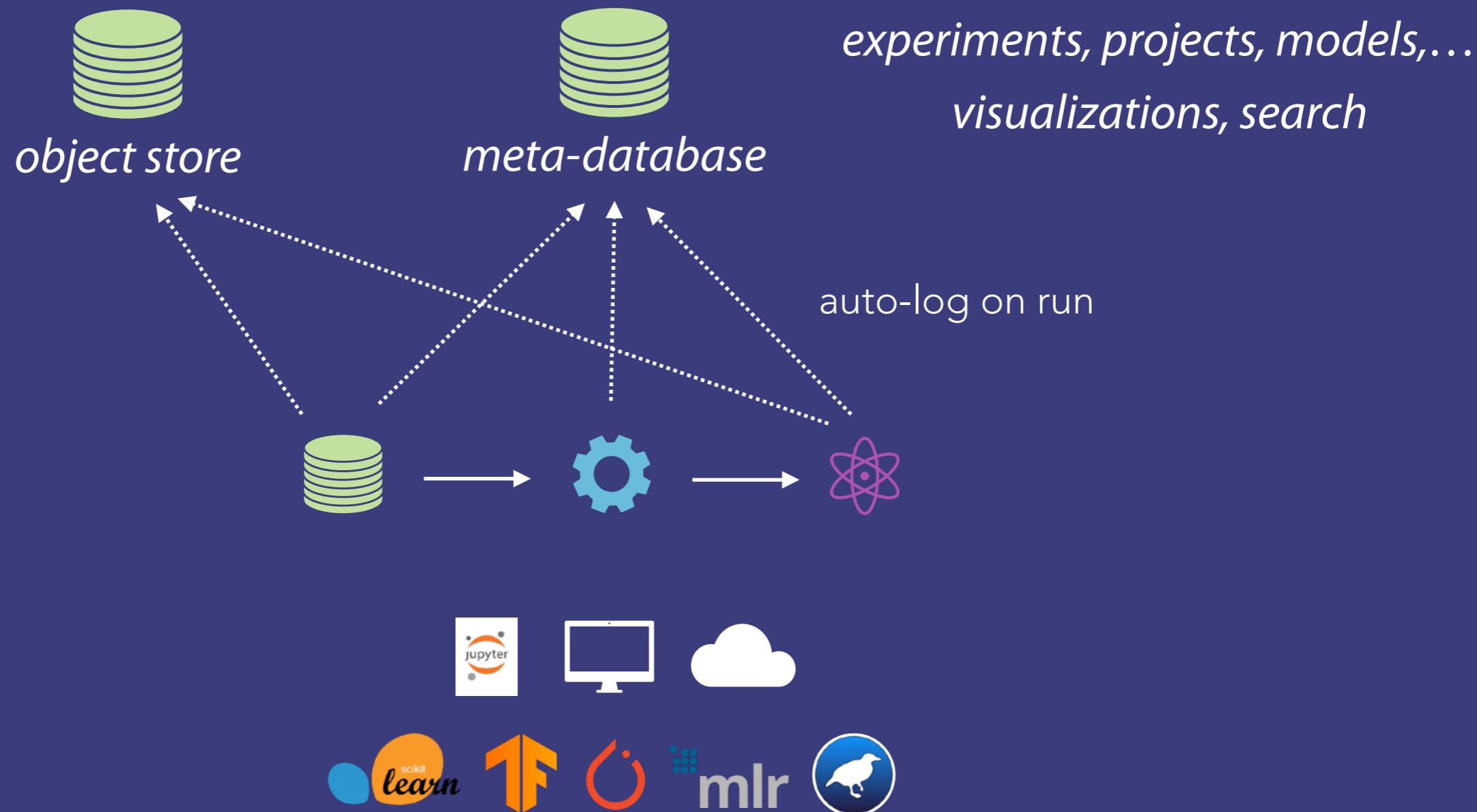
 *meta-database*
experiments, projects, models,...
visualizations, search



Reproducibility

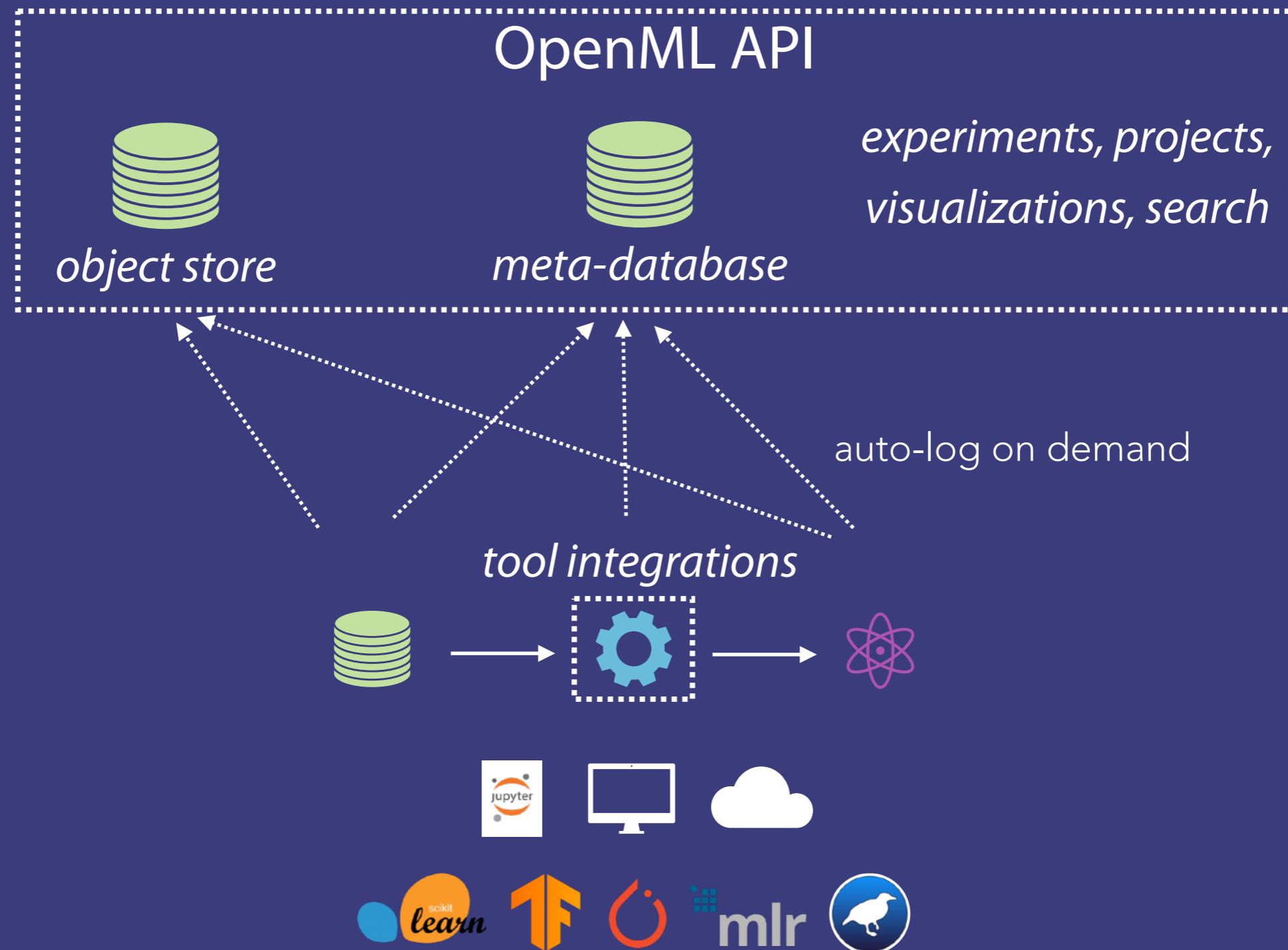
*auto-logging, homogeneous rich metadata
requires tool integration, less flexible
tool-specific / host-specific*

System of execution = system of record



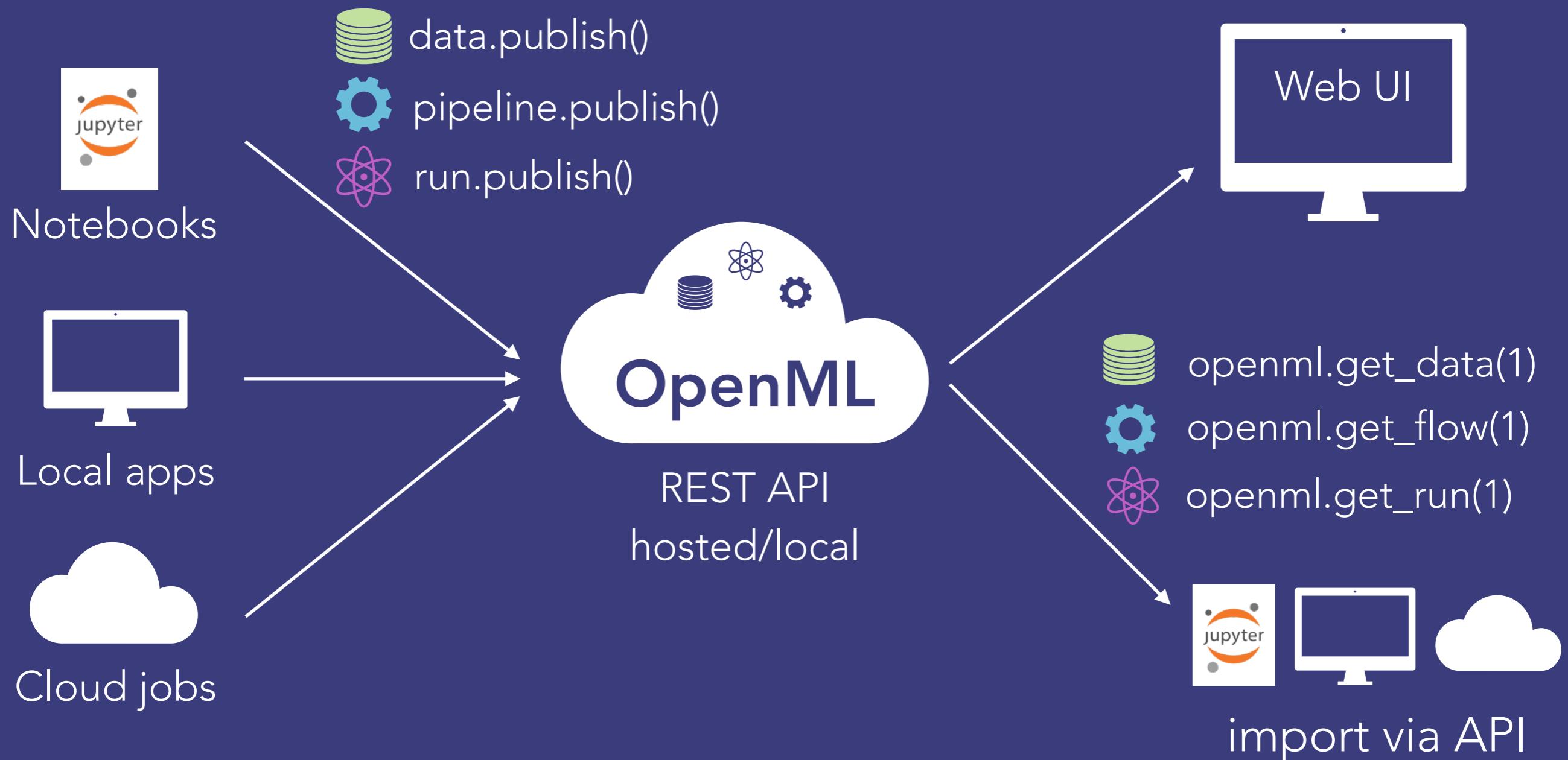
Reproducibility

*auto-logging, homogeneous rich metadata
easy sharing and reuse
less flexible, client-side compute*

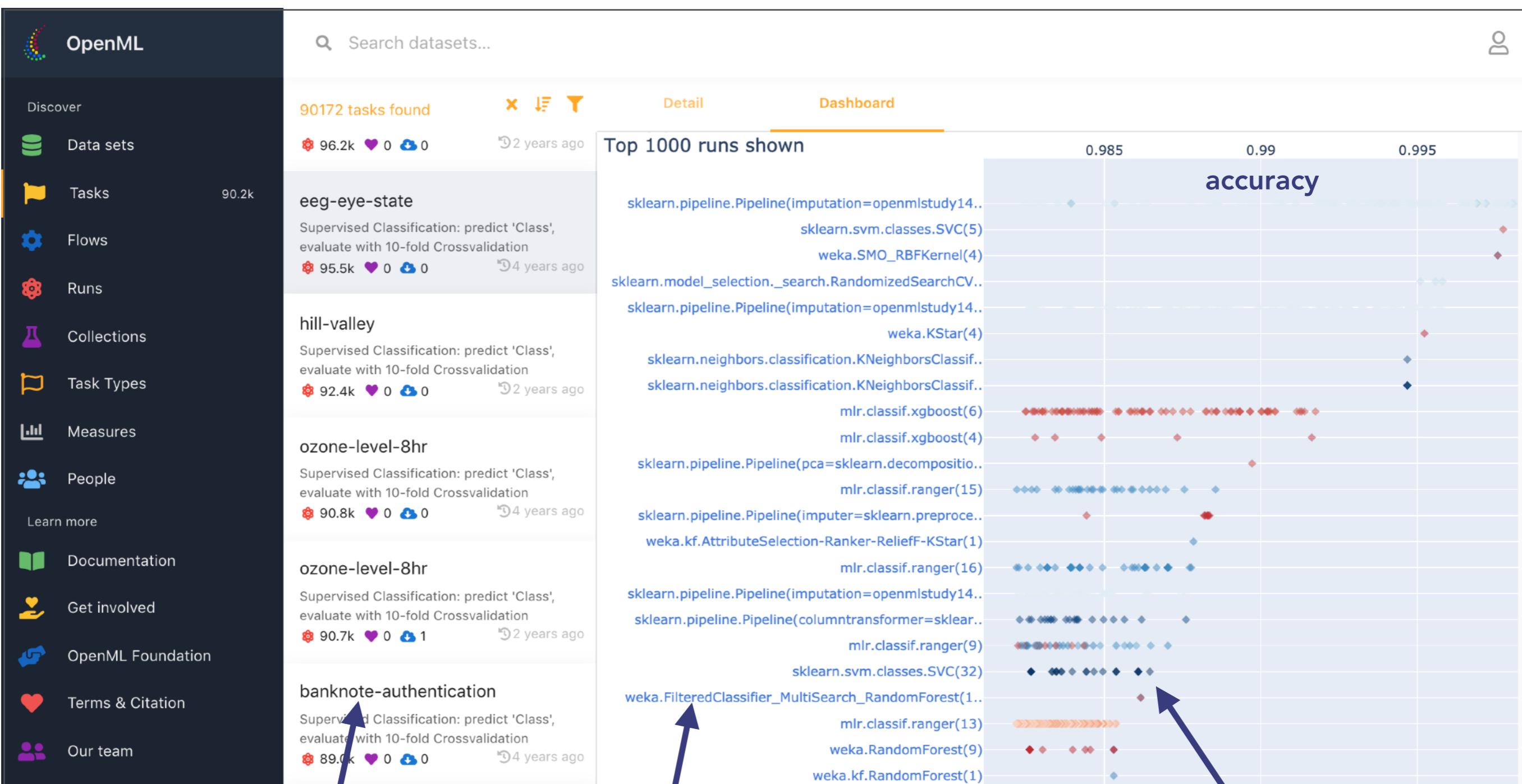


Easy sharing, discovery, reuse

All (meta)data is collected and organized *automatically*



Web UI - Browse everyone's shared data (new.openml.org)



datasets



flows (pipelines)



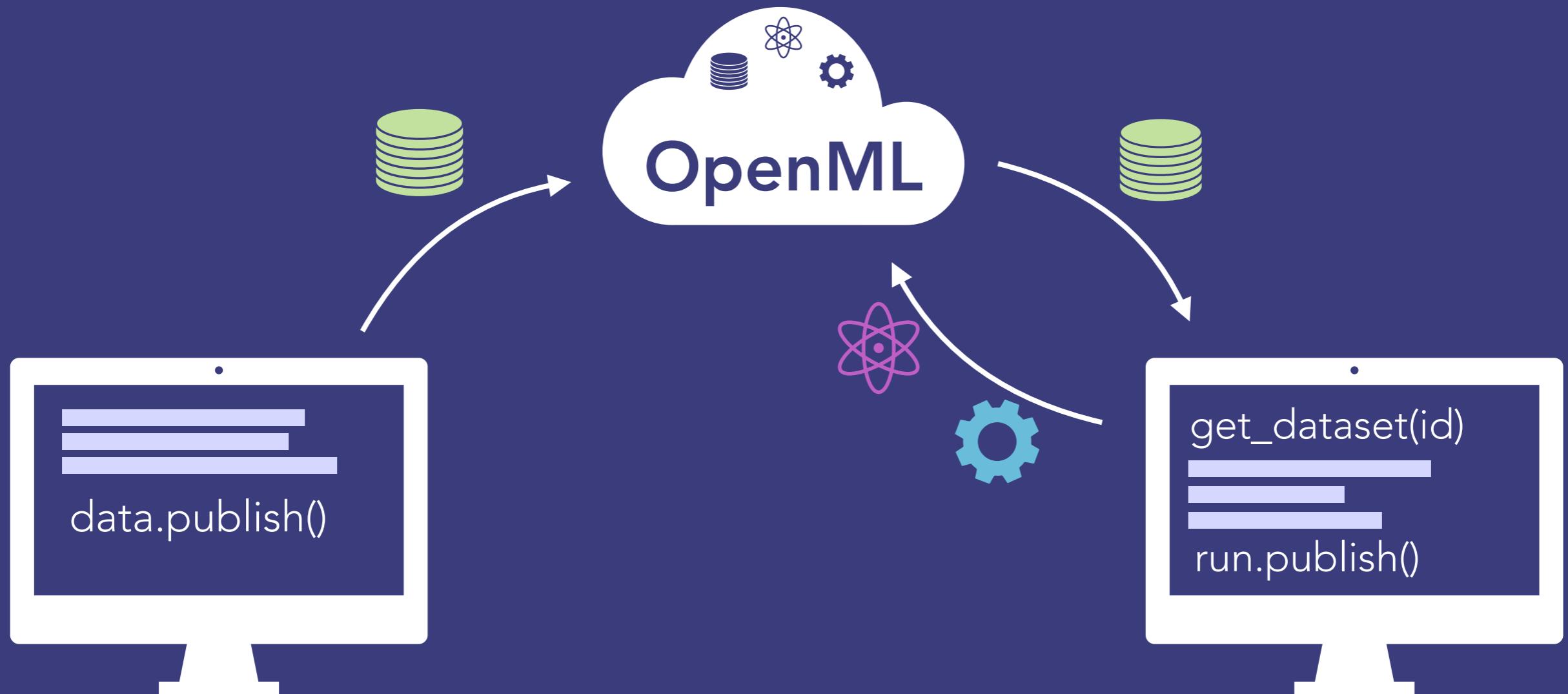
runs
(models + performance)

Frictionless machine learning

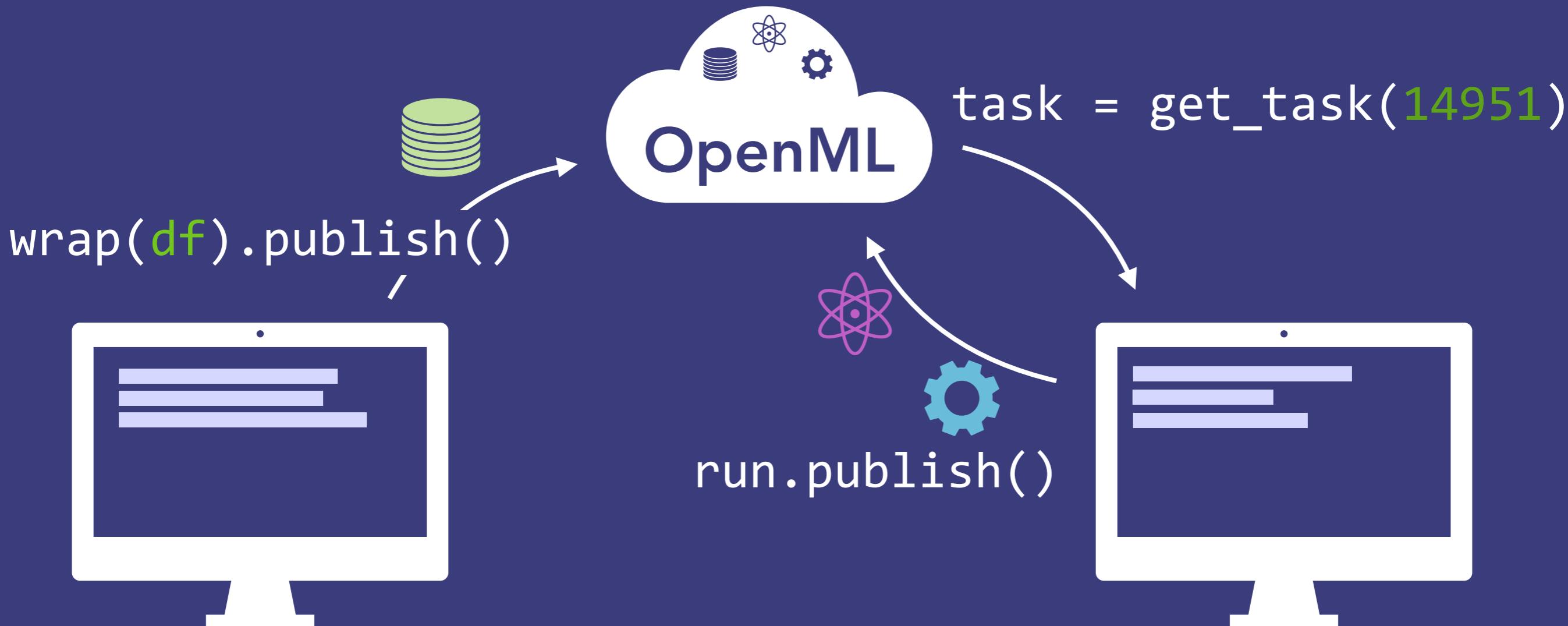
Share easily from where you create

Import easily into your working environment (in uniform formats)

Run wherever you want



Frictionless machine learning



```
clf = MyClassifier()  
run = run_model(clf,task)
```

APIs:

Integrations:

Examples



```
from sklearn import ensemble  
from openml import tasks, runs  
  
clf = ensemble.RandomForestClassifier()  
task = tasks.get_task(3954)  
run = runs.run_model_on_task(clf, task)  
run.publish()
```

More examples on <https://docs.openml.org/Python-examples/>

Examples



```
import torch.nn  
from openml import tasks, runs  
  
model = torch.nn.Sequential(  
    processing_net, features_net, results_net)  
task = tasks.get_task(3954)  
run = runs.run_model_on_task(clf, task)  
run.publish()
```

Full example on <https://openml.github.io/blog/>

Examples



```
library(mlr)
library(OpenML)

lrn = makeLearner("classif.randomForest")
task = getOMLTask(3954)
run = runTaskMlr(task, lrn)
uploadOMLRun(run)
```

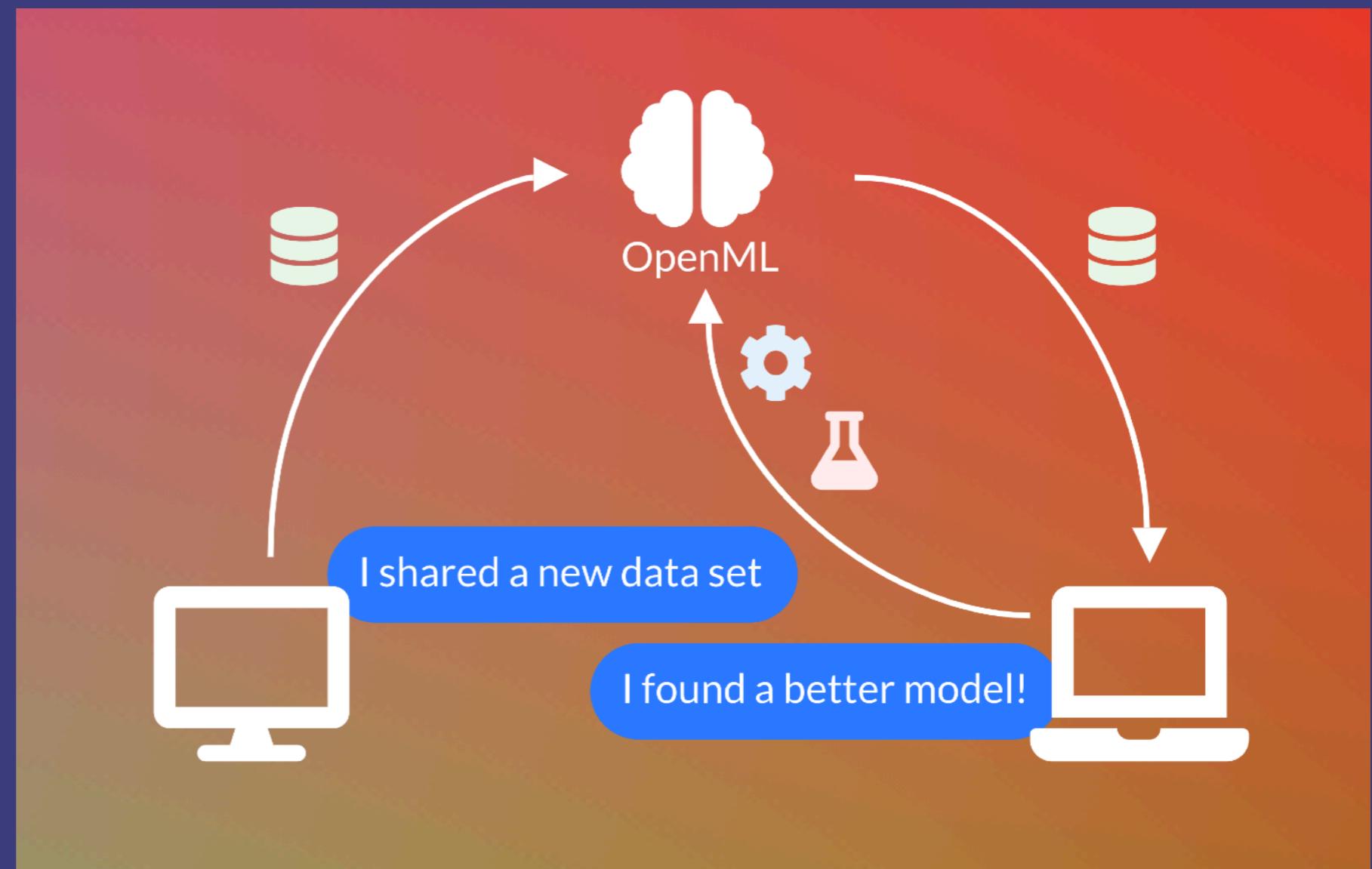
More examples on <https://docs.openml.org/R-API/>

Scalable collaborations

Anyone can share useful data

Anyone can import data, design algorithms, share models

Anyone can find and reuse the best algorithms/models



OpenML Community

150000+ yearly users

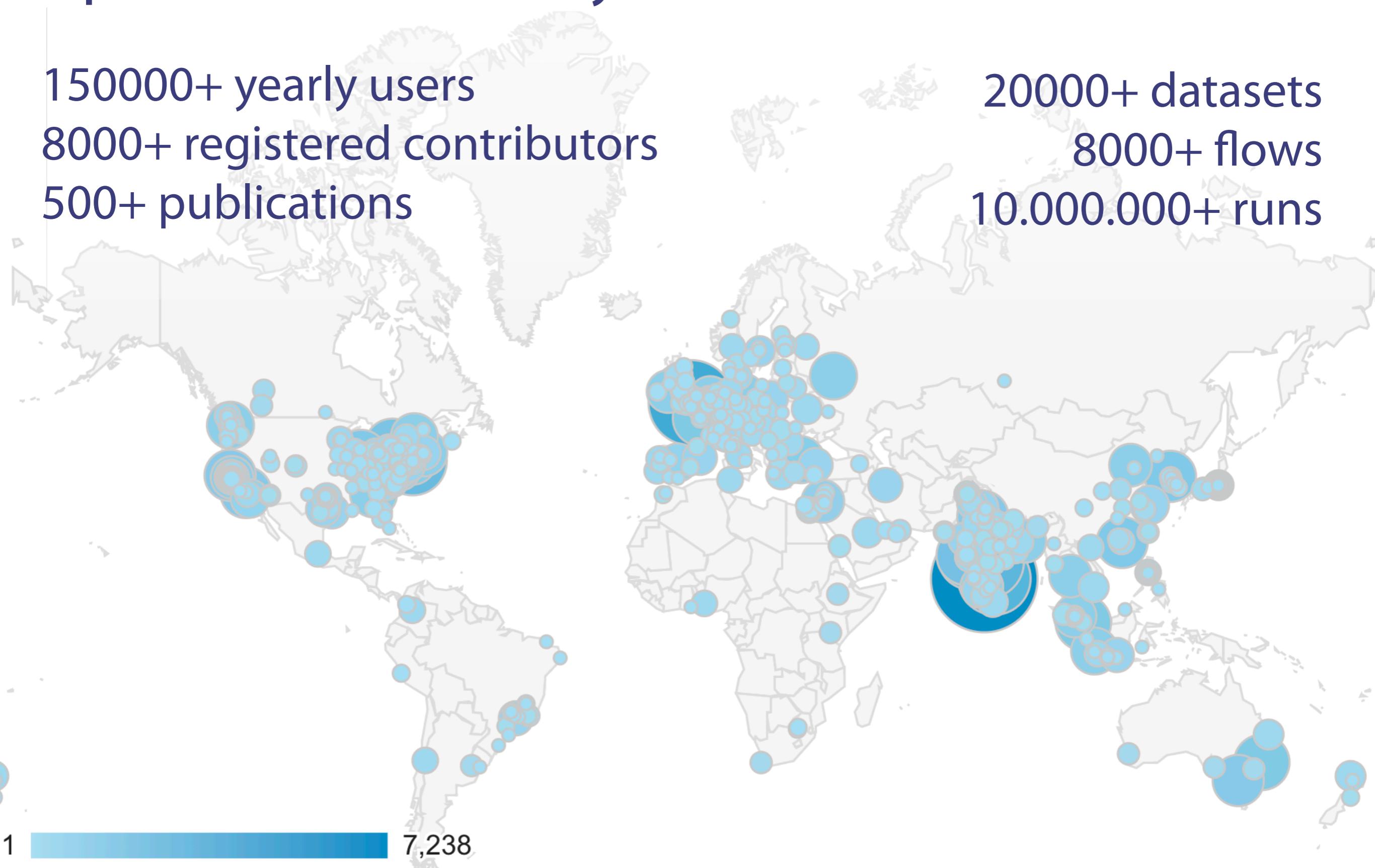
8000+ registered contributors

500+ publications

20000+ datasets

8000+ flows

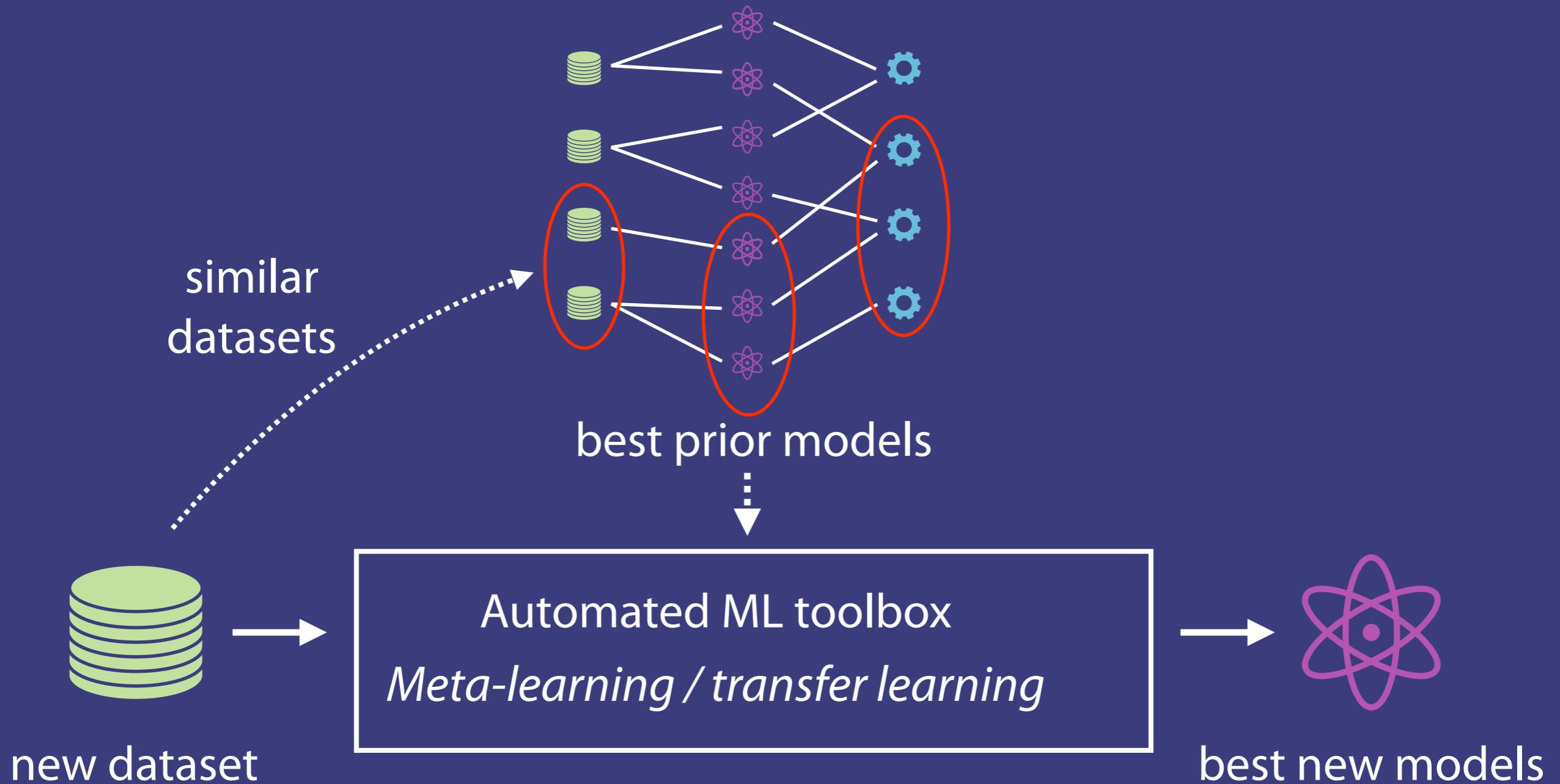
10.000.000+ runs



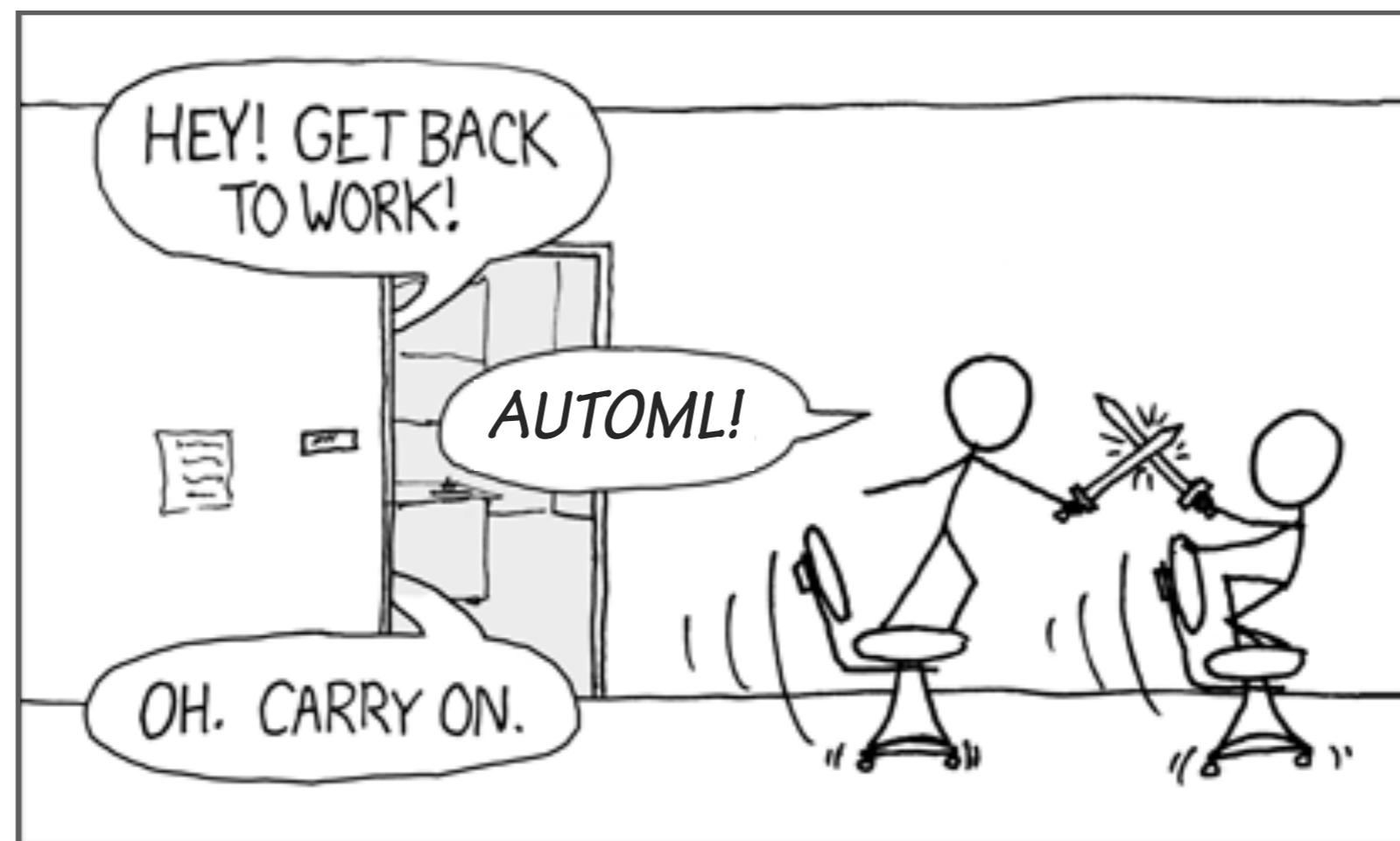
Automating machine learning

Reuse all shared metadata to learn how to learn

Lower barriers by automating hard or time-consuming aspects

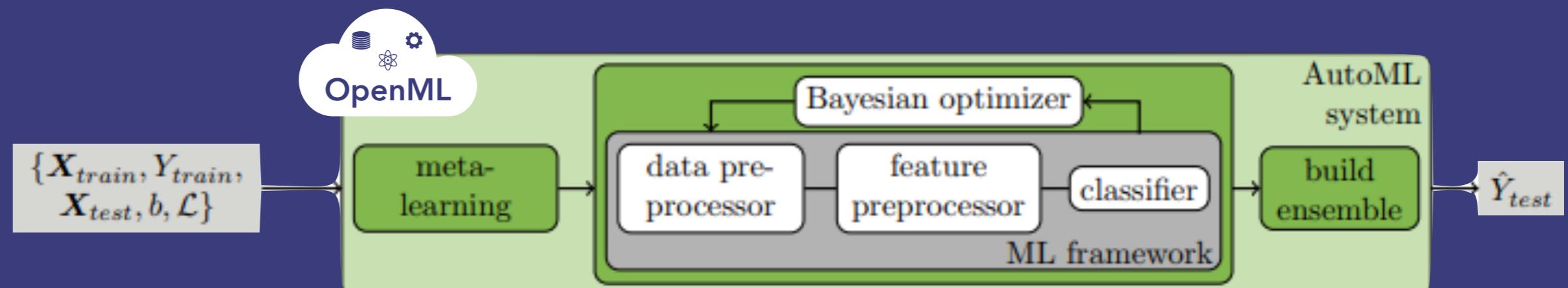


THE DATA SCIENTIST'S #1 EXCUSE FOR LEGITIMATELY SLACKING OFF: “THE AUTOML TOOL IS OPTIMIZING MY MODELS!”



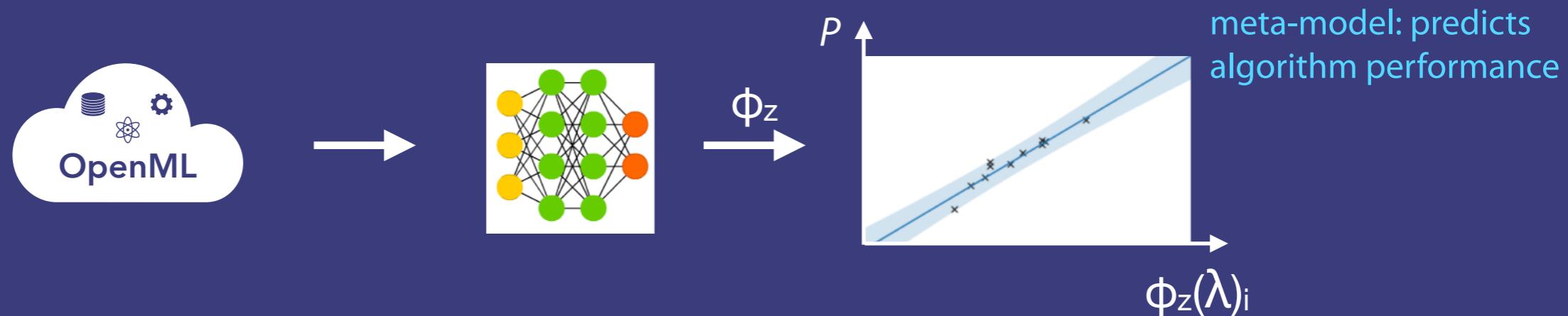
Automating machine learning

auto-sklearn: uses OpenML to *warm-start* the search for the best pipelines



Feurer et al. 2016

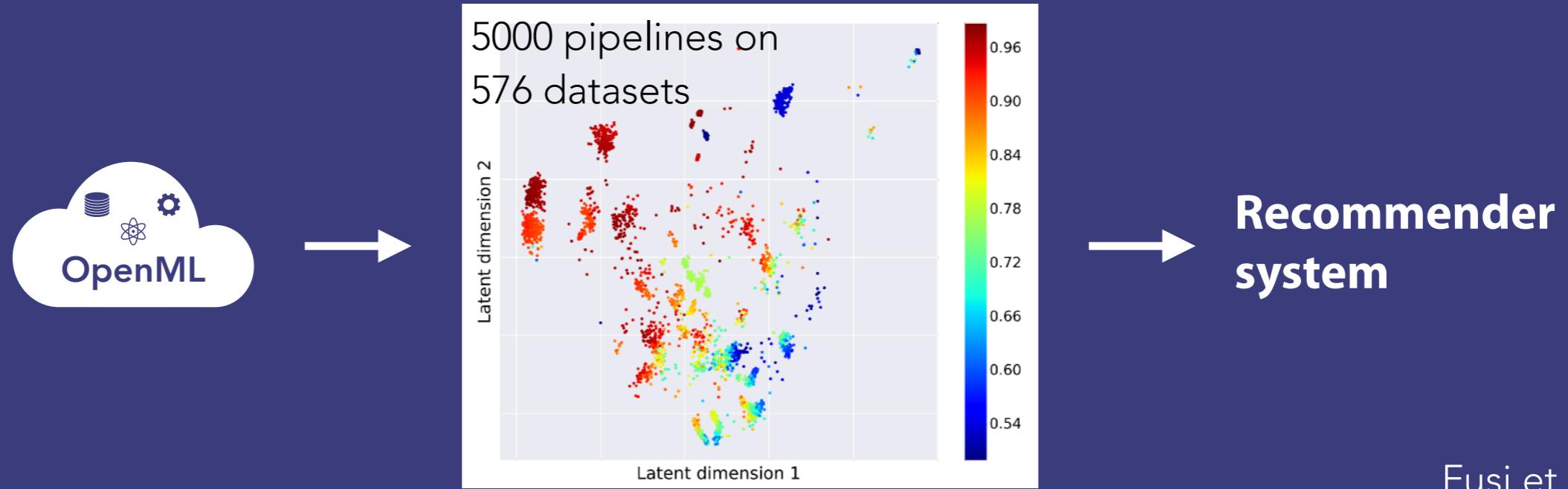
ABLR (Amazon): uses OpenML to learn how to search hyperparameters



Perrone et al. 2017

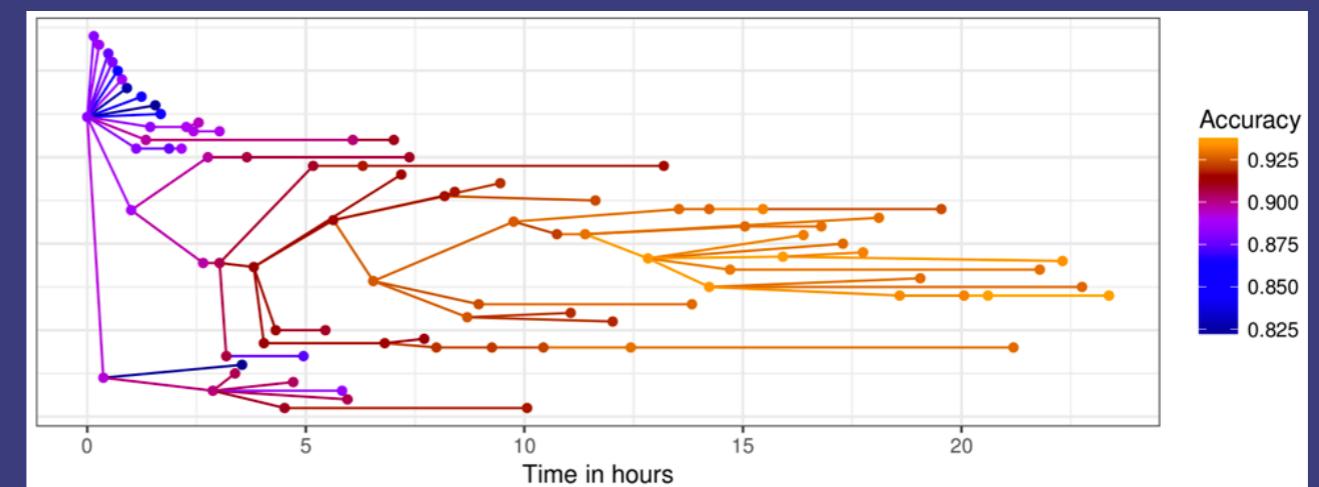
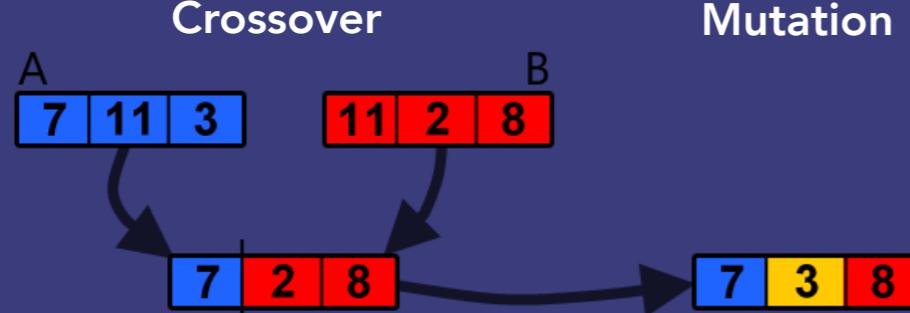
Automating machine learning

ProbMF (Microsoft): uses OpenML to recommend the best algorithms



Fusi et al. 2018

GAMA (TU/e): quickly evolves optimal pipelines for a given input dataset



Gijsbers et al. 2018



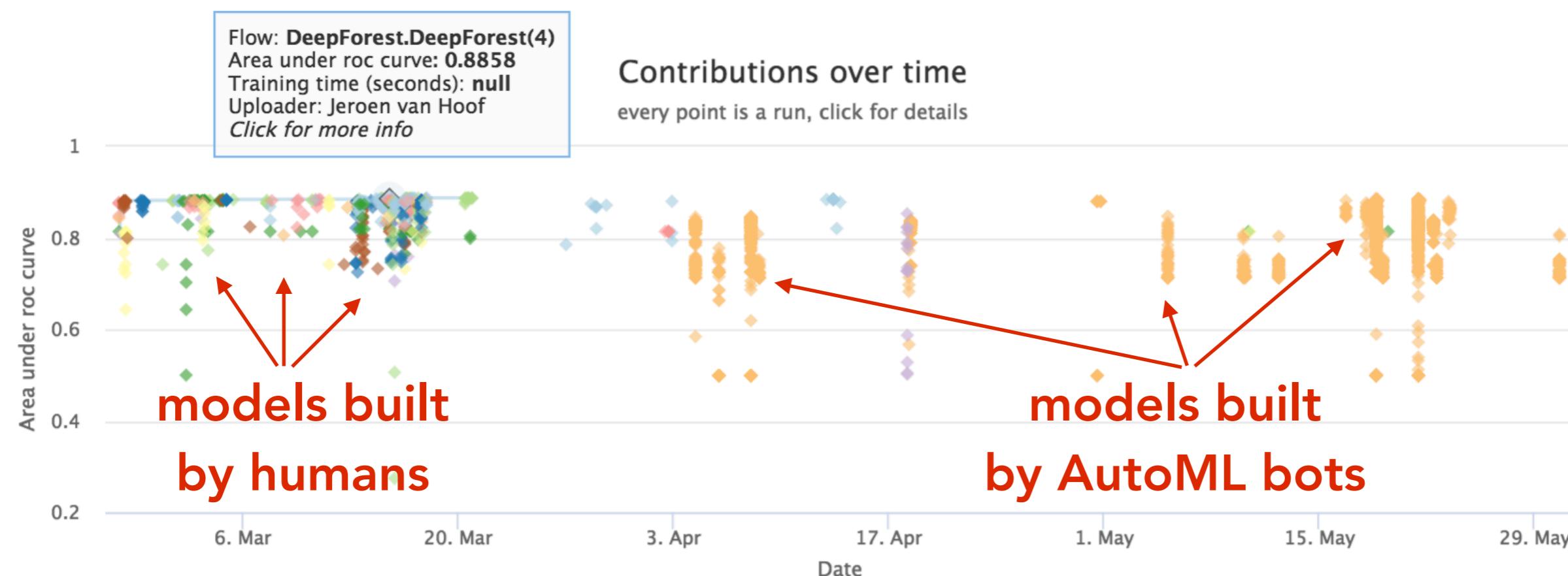
Human-AI interaction

Algorithms learn from models shared by humans

Humans learn from models built by bots

Timeline

Metric: AREA UNDER ROC CURVE



frontier	Joaquin Vanschoren	Hilde Weerts	edorigatti	Joel Goossens	Niels Hellinga	Mingpeiyu Zhang
Evertjan Peer	stevens jethefer	Hongliang Qiu	Yizi Zhu	János Szedelényi	Chin-Fang Lin	Wenting Xiong
M de Roode	Tianyu Zhou	Lirong Zhang	Ruud Andriessen	Stefan Majoer	Angelo Majoer	Changbin Lu
Irfan Nur Afif	Nan Yang	Niels de Jong	Thomas Hagebols	Stanley Clark	Joost Visser	Jeroen van Hoof
Xiaolei Wang	Timothy Aerts	Lieuwe Stooker	Corbin Joosen	Jos Mangnus	Luis Armando Perez Rey	Yongyu Fan
Jet van den Broek	Thijs Ledeboer	Brent van Strien	Arun Tom Skariah	Sako Arts	Xuqiang Fang	OpenML_Bot R
Suraj Iyer	Filip Obers	Laurens Reulink	Kevin van Eenige	Tong Wu	Jan van Rijn	y q
Raphaël Couronné	Mikaël Le Bars					

Join us! (and change the world)

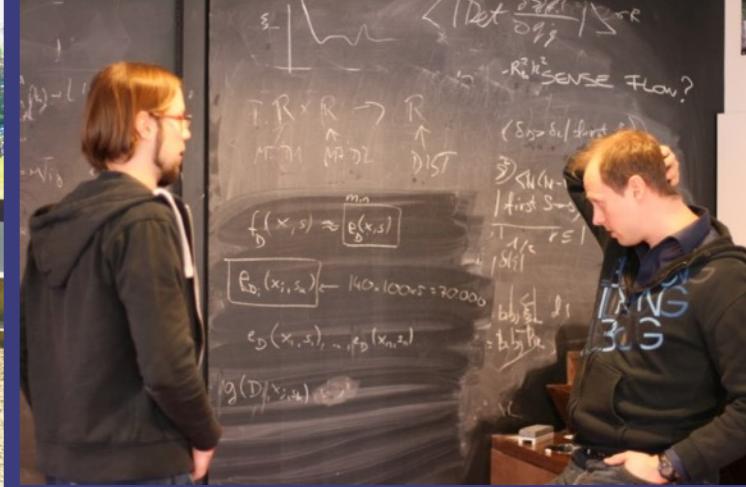
Active open source community

Hackathons 2-3x a year

We need bright people

- ML, Devs, UX

OpenML Foundation



Thank you!

谢谢

 @open_ml

 OpenML

 www.openml.org



Thanks to the entire OpenML star team



Jan van Rijn



Matthias Feurer



Heidi Seibold



Bernd Bischl



Andreas Müller



Prabhant Singh



Guiseppe Casalicchio



Michel Lang



Sahithya Ravi



Marcel Wever



Neil Lawrence



Erin Ledell



Bilge Celik



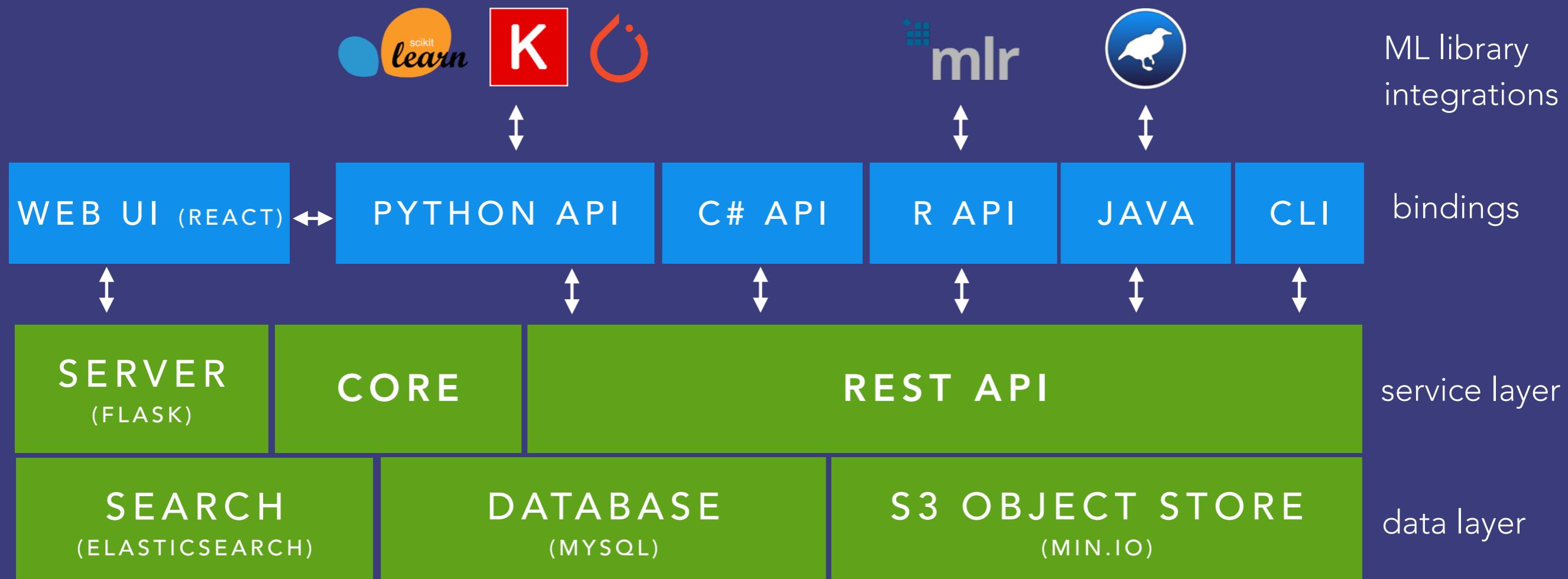
Janek Thomas



Markus Weimer

and many more!

Architecture



client-side (local)
server-side (remote)