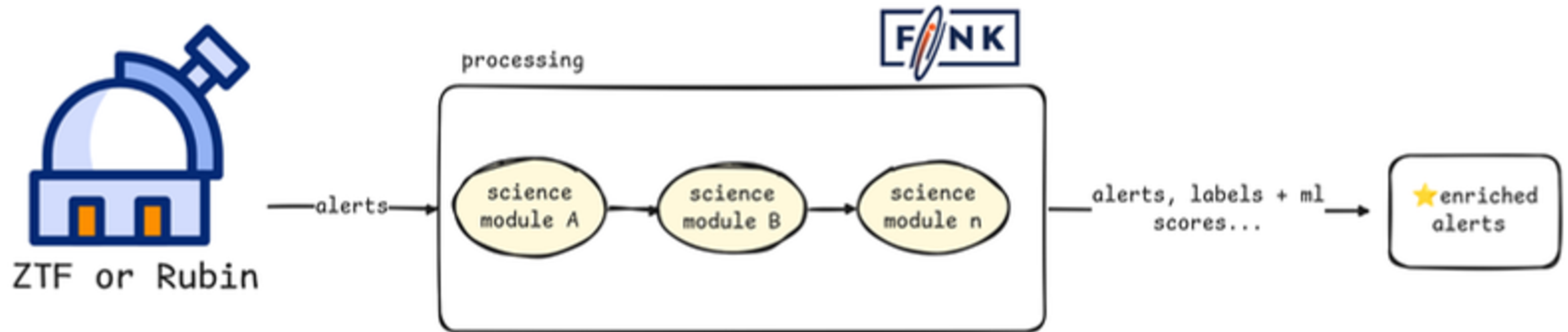




Running Your Own Machine Learning Models on Real Data

Farid MAMAN

How are alerts processed?

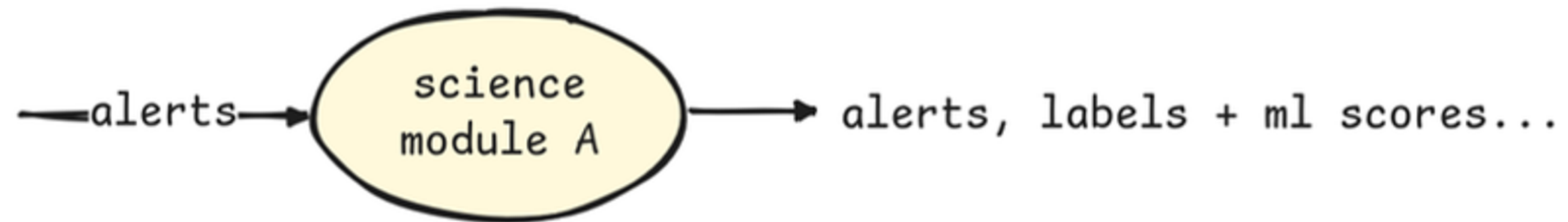


★ Enriched alerts:

- Saved in the database
- Filters are applied



Science modules

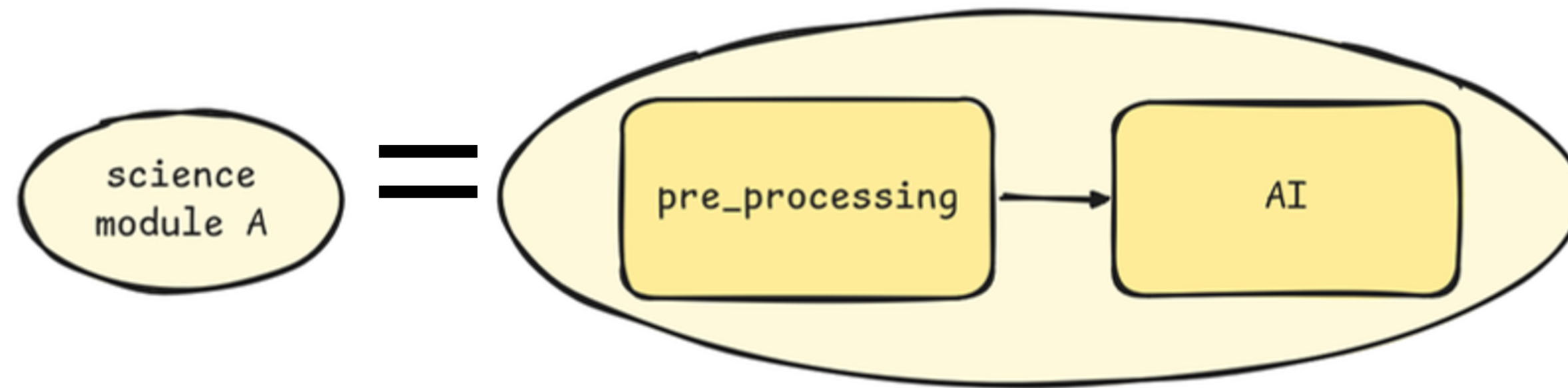


- Each module takes an alert as input
- Returns a score attached to the alert



Science modules

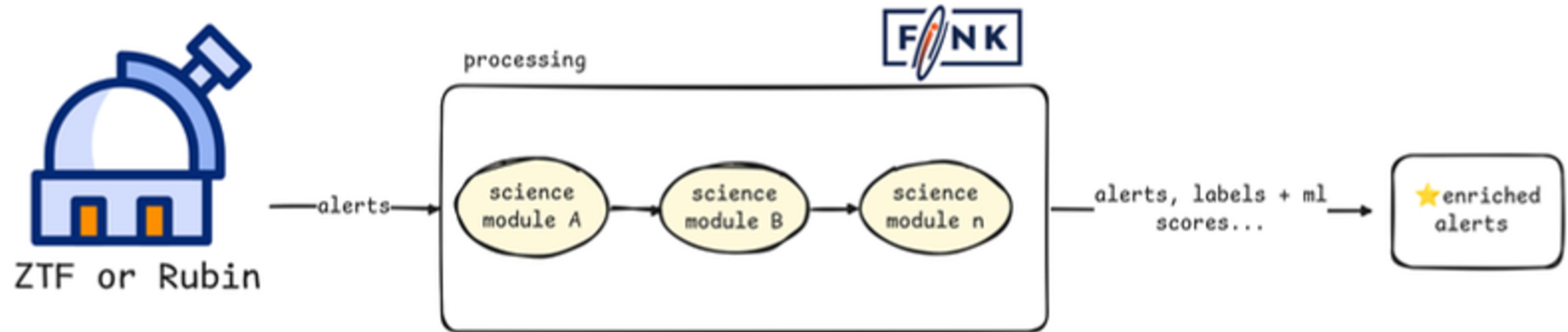
Each science module is an AI model dedicated to a specific classification task.



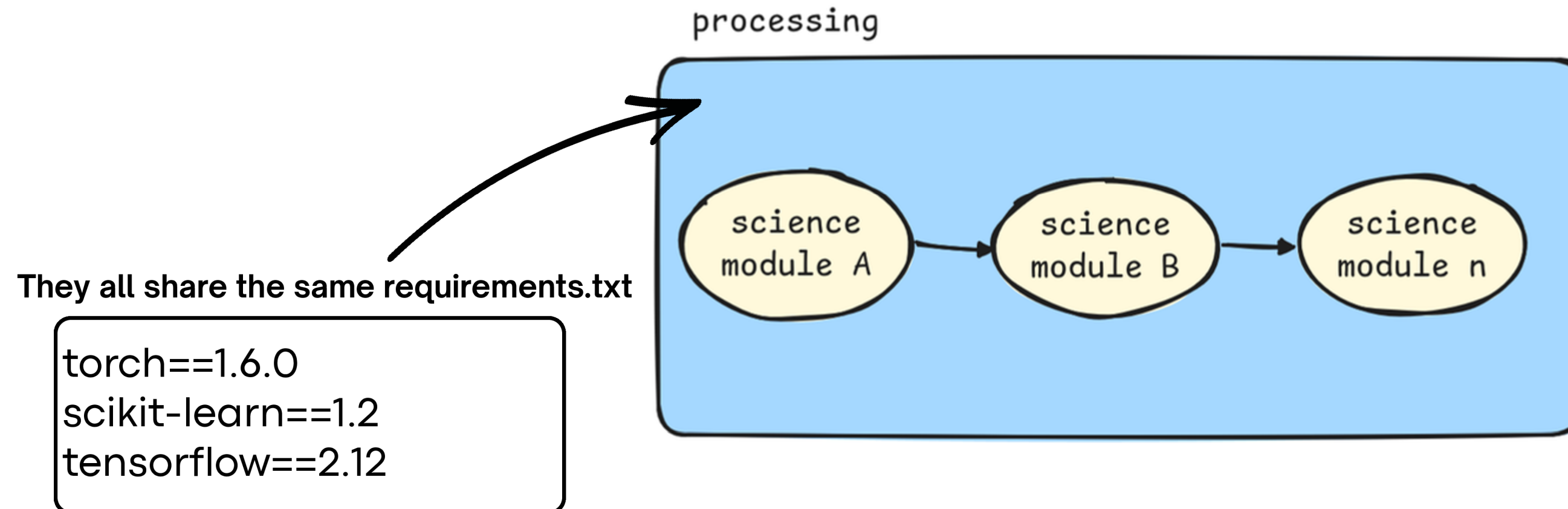
- **pre_processing**: cleans and formats the alert data before feeding it to the model
- **AI model** outputs a score reflecting the probability



What is the problem with this infrastructure?



Shared execution environment

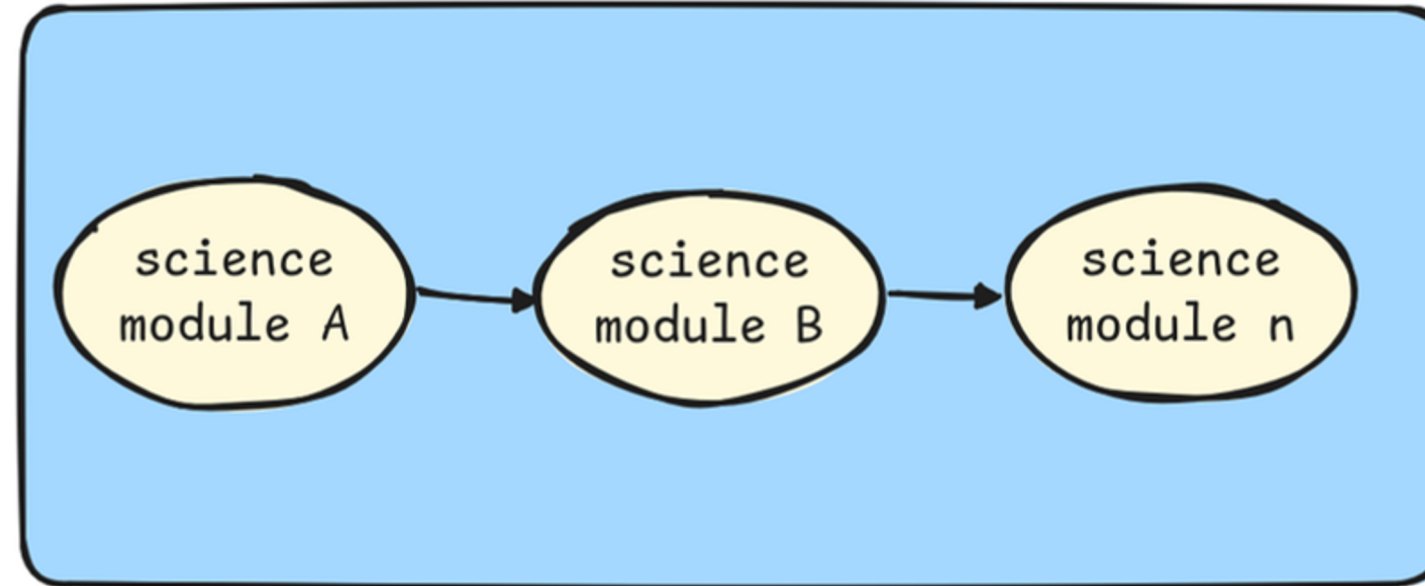


- Every science module uses the same packages and versions
- Need a different version? → Conflicts
- New dependency? => you need to open a PR

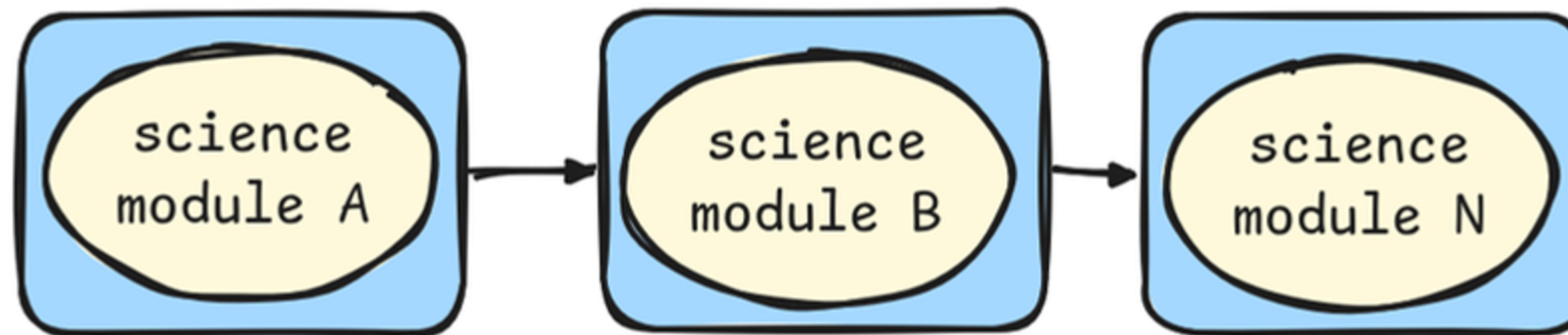


The solutions is to wrapper evreything

From this



To this



How ?



MLflow

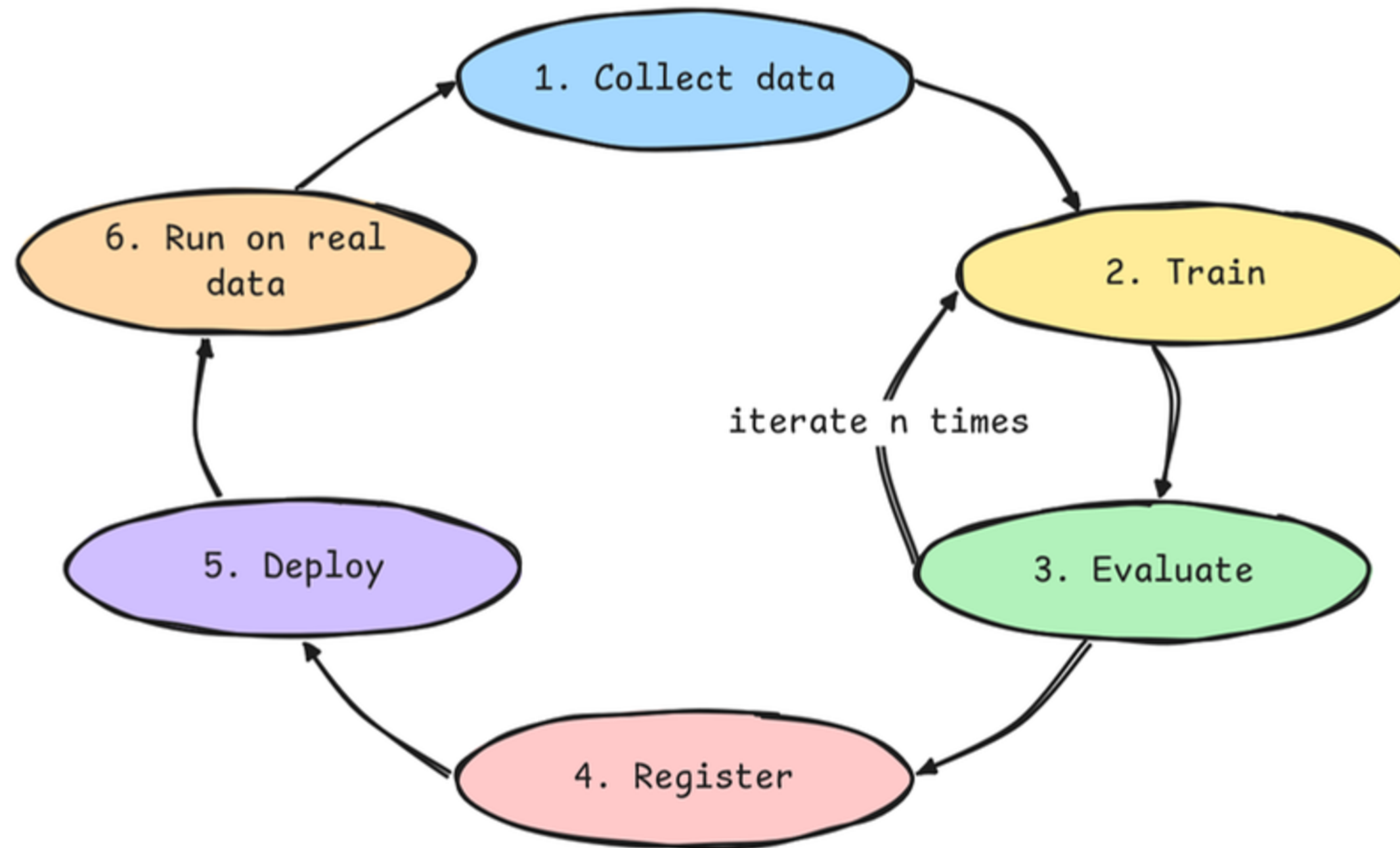
[MLflow](#) is an open-source platform to manage the full lifecycle of a machine learning model.



Think of it as: "**Git, but for trained models.**"



MLflow | ML lifecycle



- Tracks your parameters, metrics and results

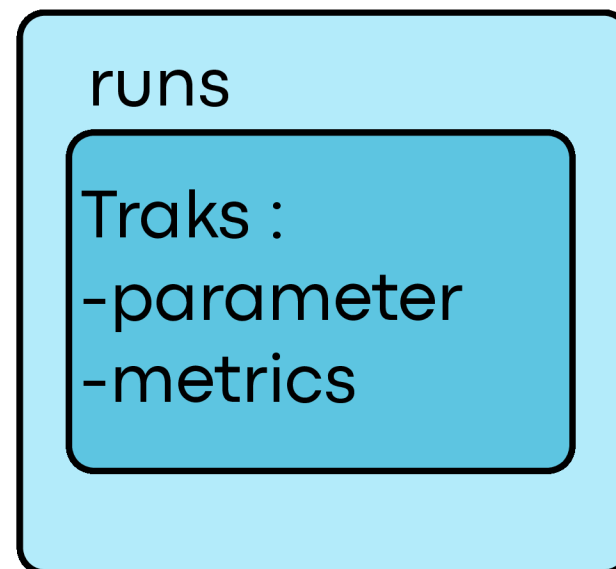


MLFlow | Key concepts

Experiment

- **Groups all your training runs**
- **A run = one training session**

Experiment



Artifacts

Artifacts = files stored
in mlflow



Registry

Registry = is a catalogue
where models are stored



Demonstration ?

- [MLflow Fink](#)
- [MLflow tutorial](#)

The image displays two screenshots of the MLflow web interface. The top screenshot shows the 'Experiments' page for 'MLflow 3.0 Tracking Example'. It features a table of runs with columns for Model name, Status, Created time, Source run, Dataset, accuracy, and activation. The bottom screenshot shows the 'Registered Models' page, listing various models like 'iris_model_dev' and 'mnist_model_prod' with their latest versions and aliases.

Model name	Status	Created	Source run	Dataset	accuracy	activation
torch-iris-100	Ready	26 seconds ago	popular-snake-452	train (#1f1c13b5)	0.9833333333333333	ReLU
torch-iris-90	Ready	29 seconds ago	popular-snake-452	train (#1f1c13b5)	0.9833333333333333	ReLU
torch-iris-80	Ready	31 seconds ago	popular-snake-452	train (#1f1c13b5)	0.9833333333333333	ReLU
torch-iris-70	Ready	33 seconds ago	popular-snake-452	train (#1f1c13b5)	0.9833333333333333	ReLU
torch-iris-60	Ready	35 seconds ago	popular-snake-452	train (#1f1c13b5)	0.9833333333333333	ReLU
torch-iris-50	Ready	37 seconds ago	popular-snake-452	train (#1f1c13b5)	0.9833333333333333	ReLU
torch-iris-40	Ready	39 seconds ago	popular-snake-452	train (#1f1c13b5)	0.975	ReLU
torch-iris-30	Ready	41 seconds ago	popular-snake-452	train (#1f1c13b5)	0.9416666666666667	ReLU
torch-iris-20	Ready	44 seconds ago	popular-snake-452	train (#1f1c13b5)	0.675	ReLU
torch-iris-10	Ready	46 seconds ago	popular-snake-452	train (#1f1c13b5)	0.6583333333333333	ReLU
torch-iris-0	Ready	48 seconds ago	popular-snake-452	train (#1f1c13b5)	0.325	ReLU

Name	Latest version	Aliased versions	Created by	Last modified	Tags
iris_model_dev	Version 17			2023-09-25 12:50:...	—
iris_model_prod	Version 11	@ champion : Version 11 +3		2023-10-26 17:10:...	—
iris_model_staging	Version 11			2023-09-25 12:46:...	—
iris_model_testing	Version 1			2023-09-27 13:17:...	—
mnist_model_dev	Version 12			2023-09-25 12:39:...	—
mnist_model_prod	Version 8	@ challenger : Version 8 +1		2024-01-19 10:35:...	—
mnist_model_staging	Version 8			2023-09-25 12:51:...	—



