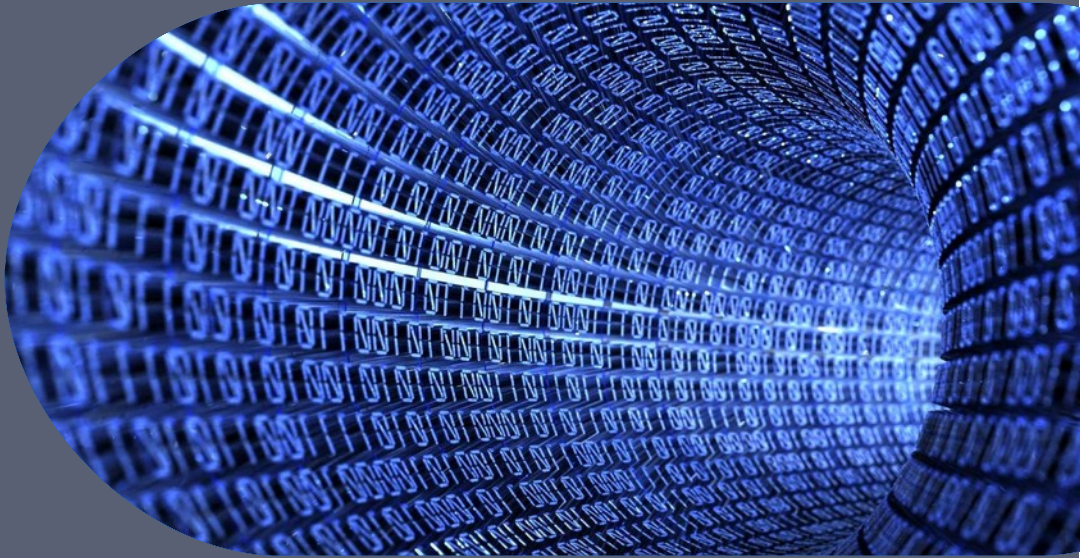
Databases & Data Design

The Meerkat Data Pipeline

Jonathan Berry



Overview

Talk: Overview

Exercise: Data collection

Talk: The data collection server

Exercise: Data transformation

Talk: The website server

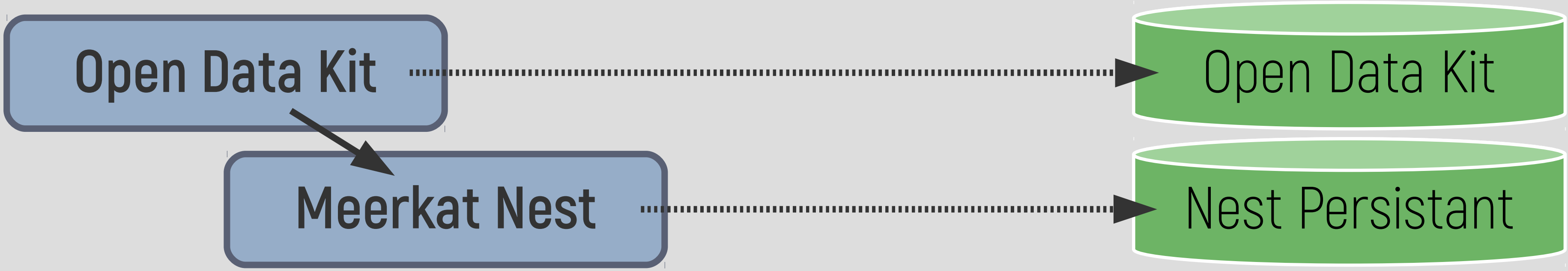
Restrictions on the data design

- Security
- Real Time
- Accessibility
- Open Data Kit
- Open Source
- Micro-services

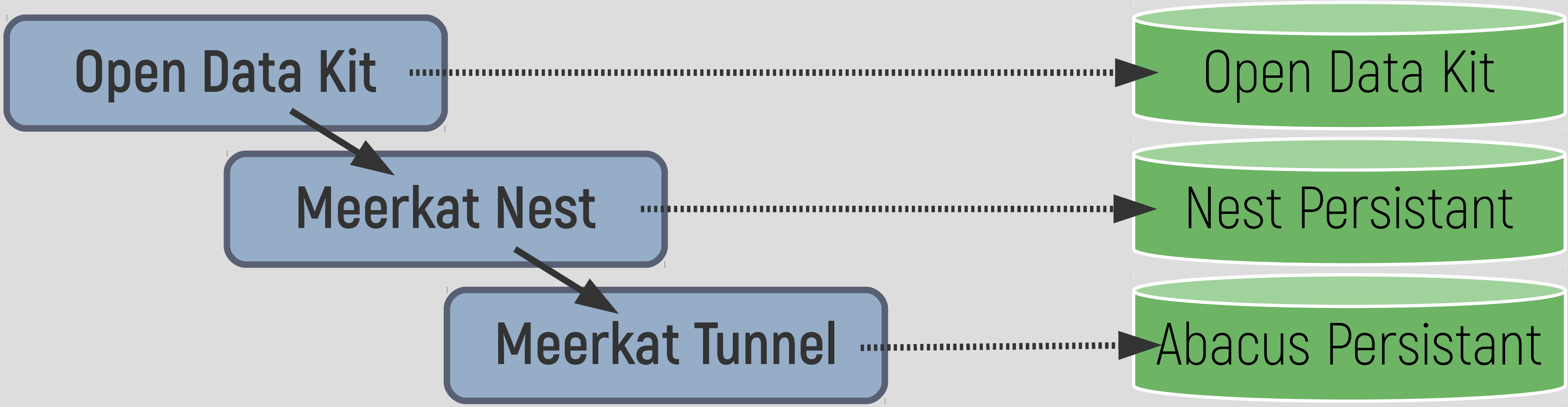
In a nutshell...

Open Data Kit
Open Data Kit

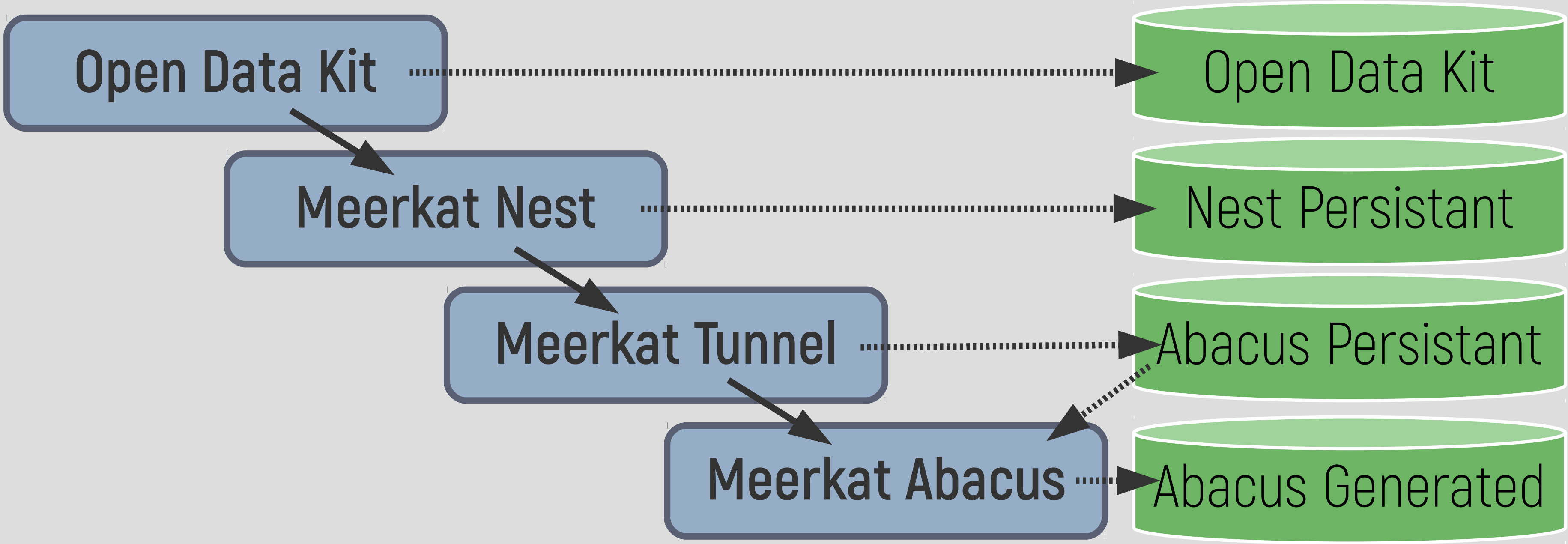
In a nutshell...



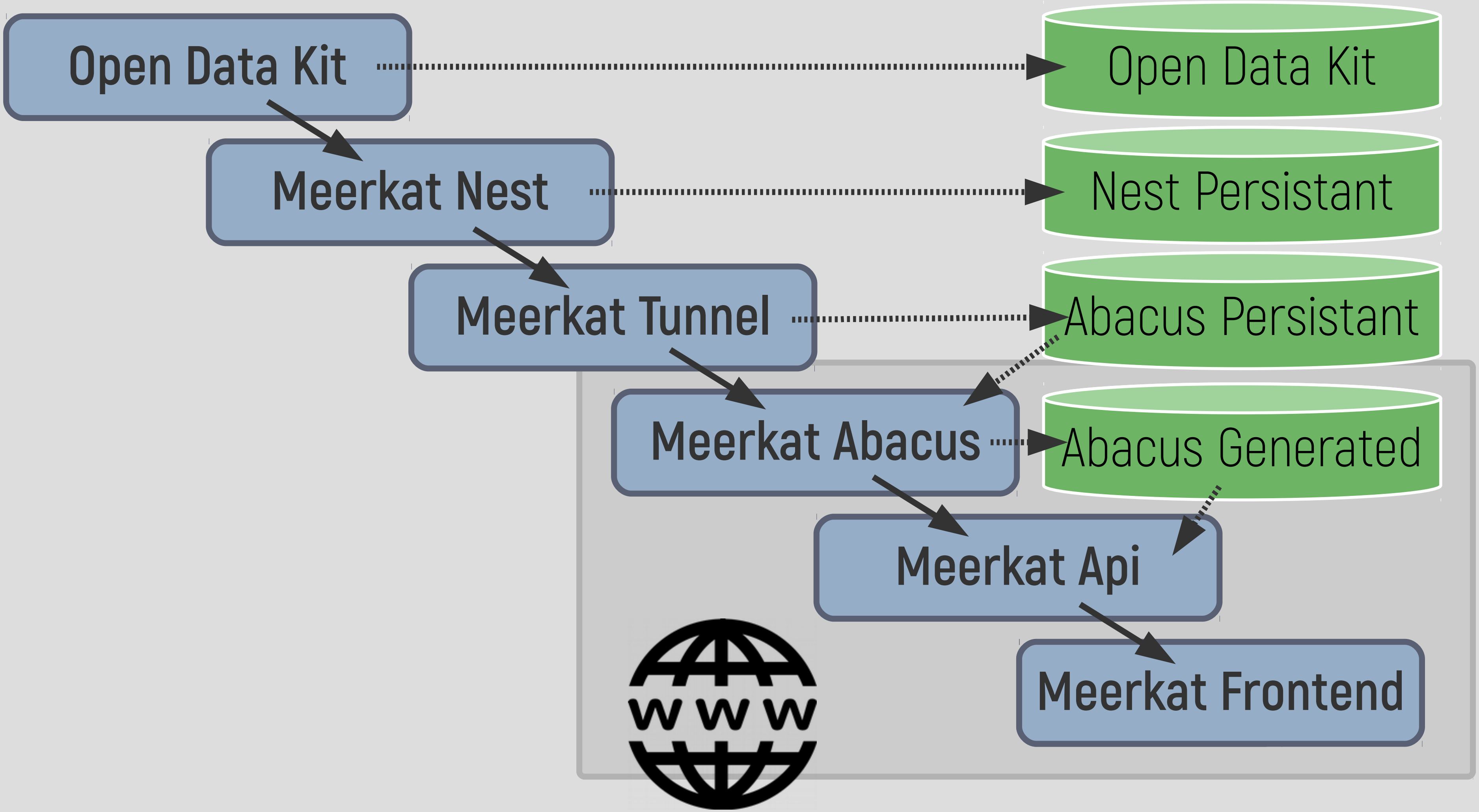
In a nutshell...



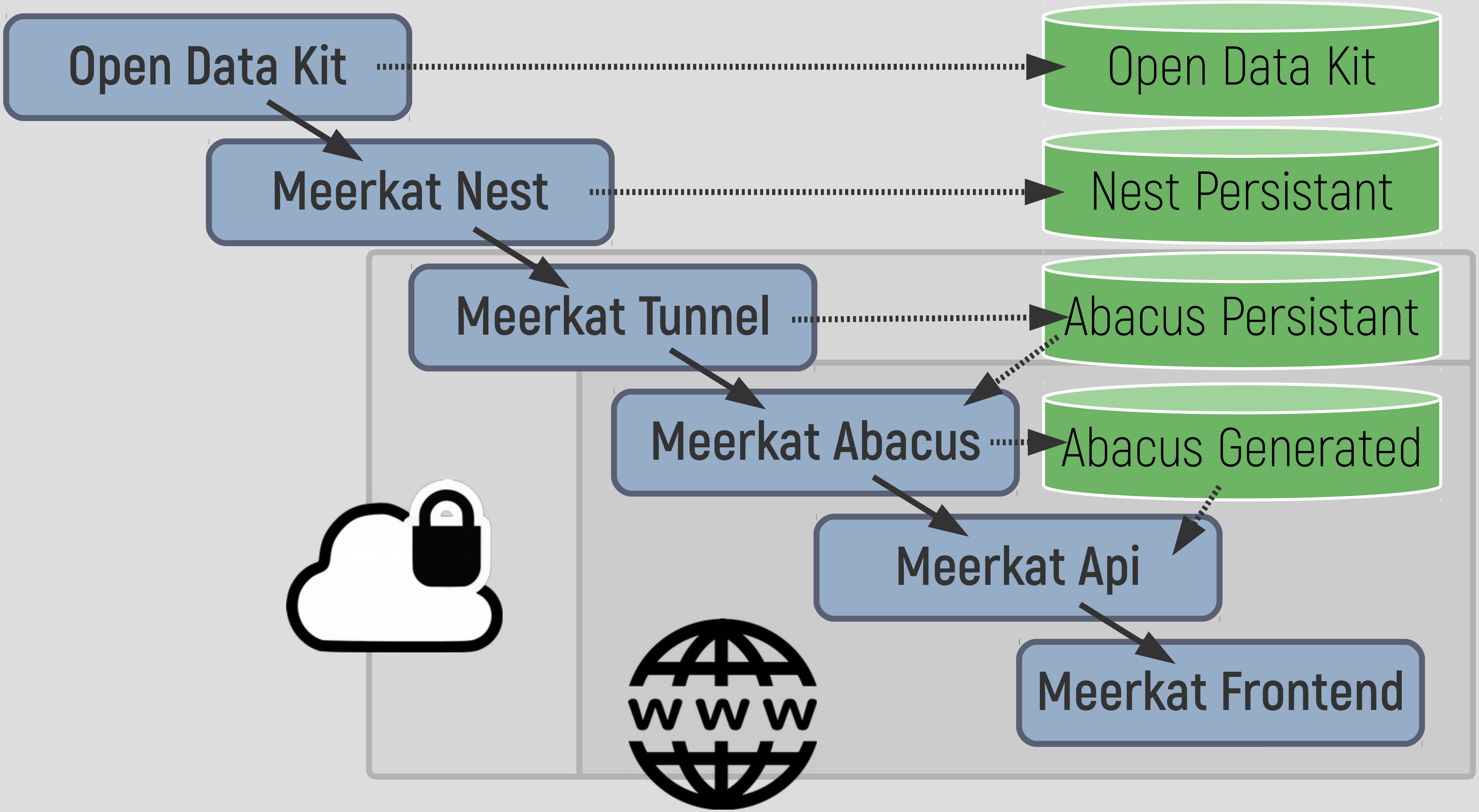
In a nutshell...



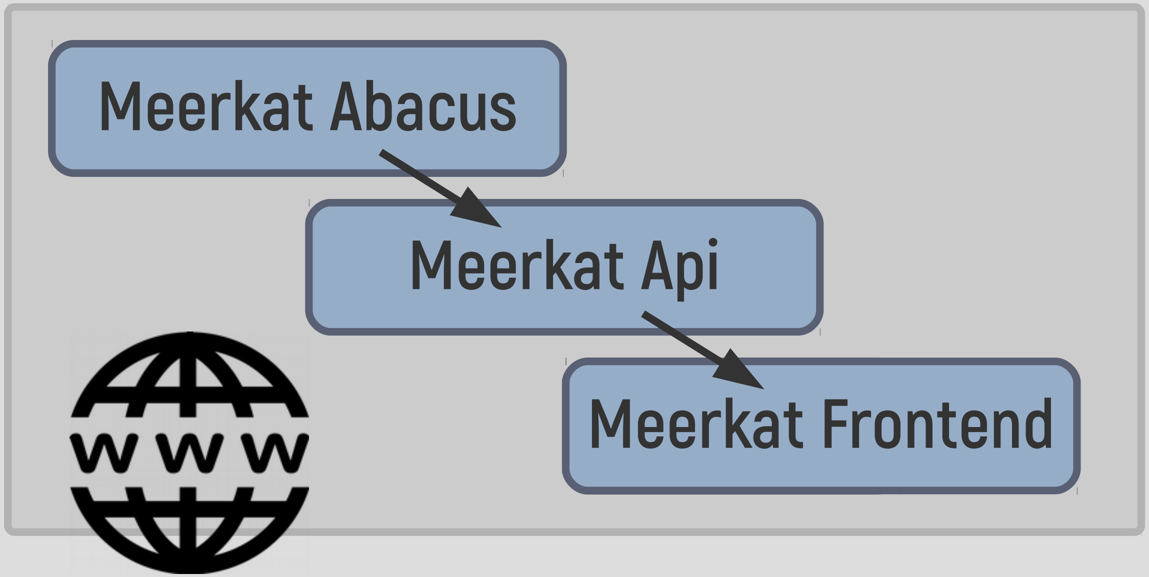
In a nutshell...



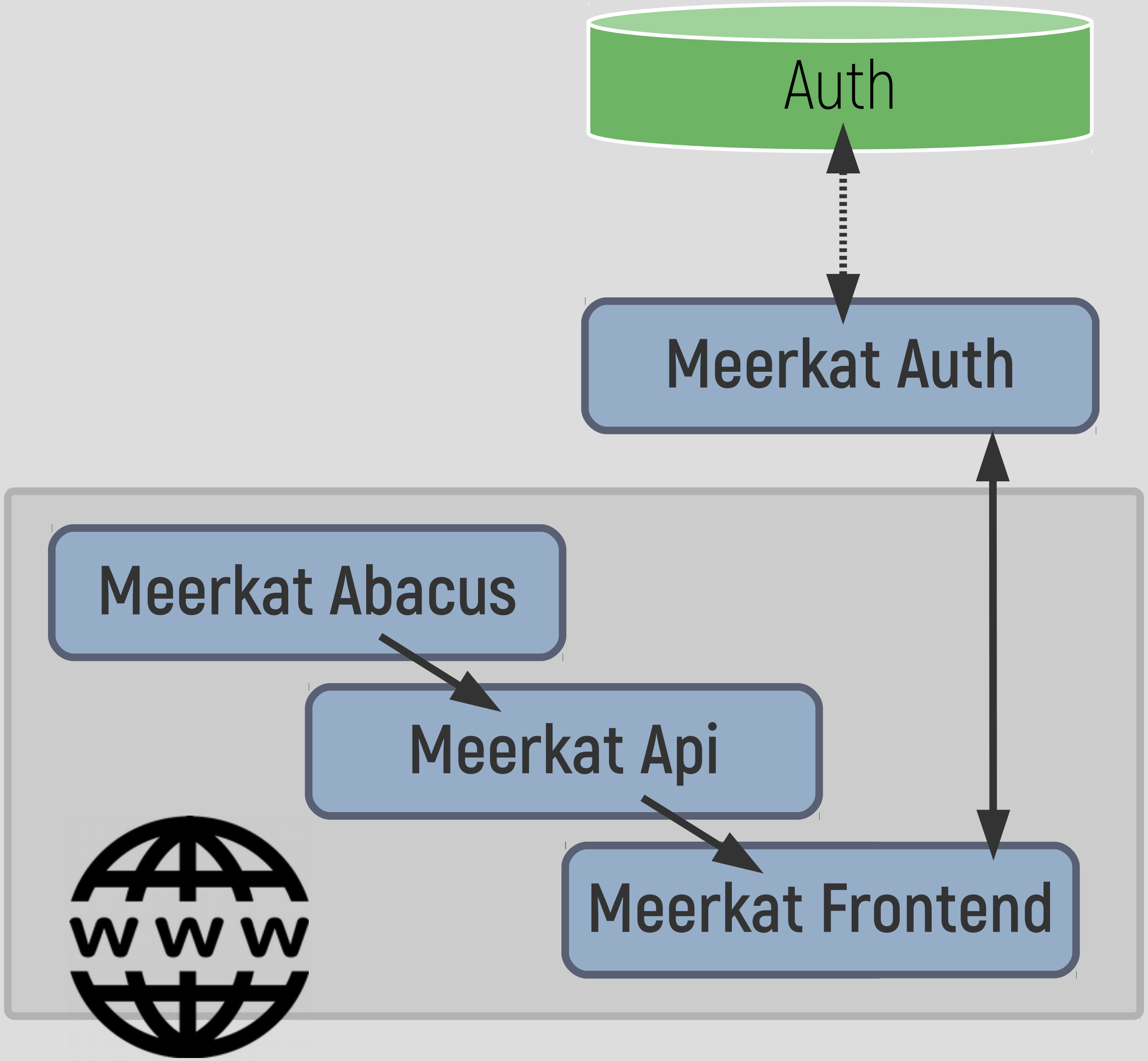
In a nutshell...



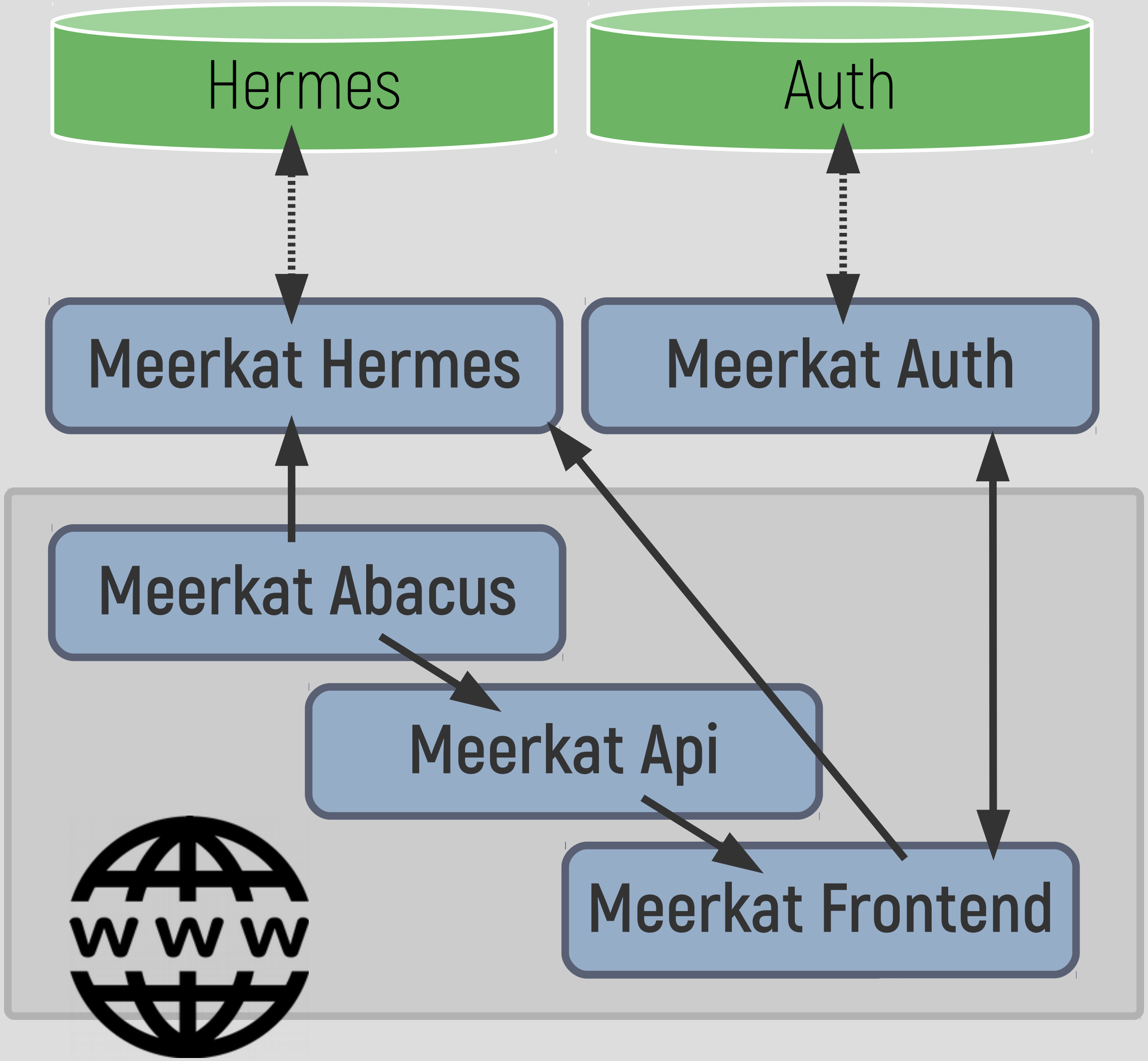
Other data



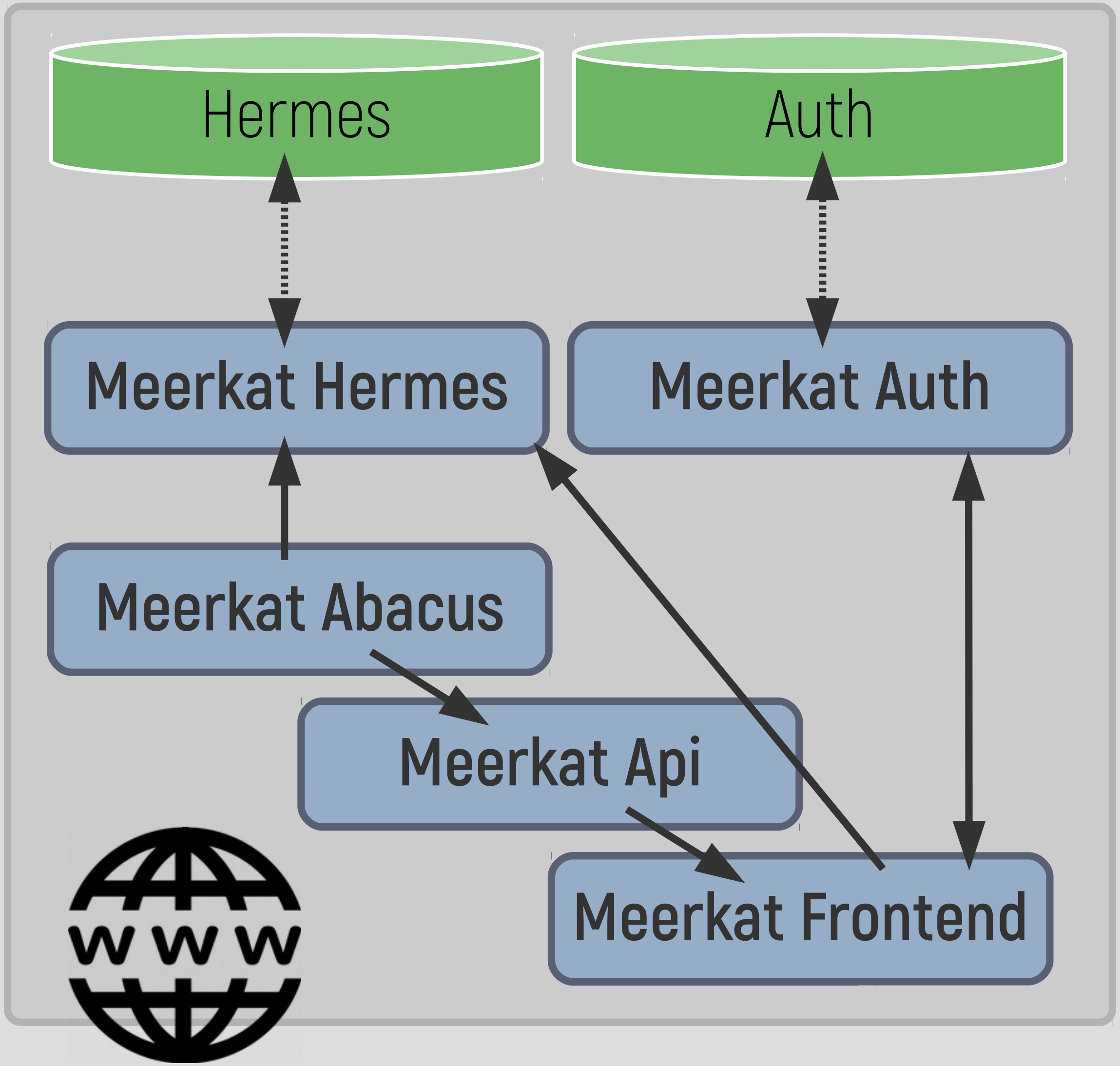
Other data


Auth
Meerkat Auth
Meerkat Abacus
Meerkat Api
Meerkat Frontend
www

Other data


Hermes
Auth
Meerkat Hermes
Meerkat Auth
Meerkat Abacus
Meerkat Api
Meerkat Frontend
www

Other data


Hermes
Auth
Meerkat Hermes
Meerkat Auth
Meerkat Abacus
Meerkat Api
Meerkat Frontend
www

Relational vs. Non-relational Databases

No fixed schema!

Scalable, flexible, with strong consistency.

PostgreSQL allows a half-way house with the JSON data type.

Exercise

Data Collection

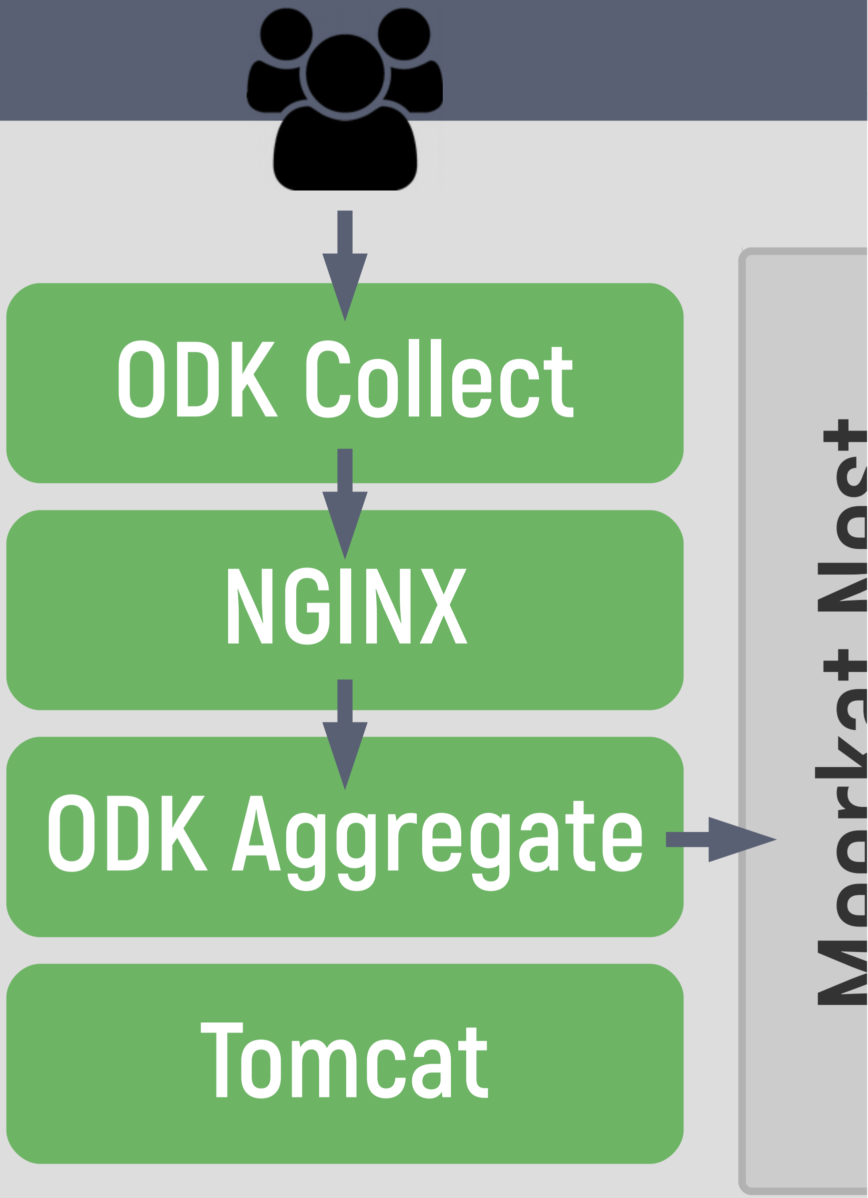
Jordan
Public Health Surveillance

Open Data Kit (ODK)

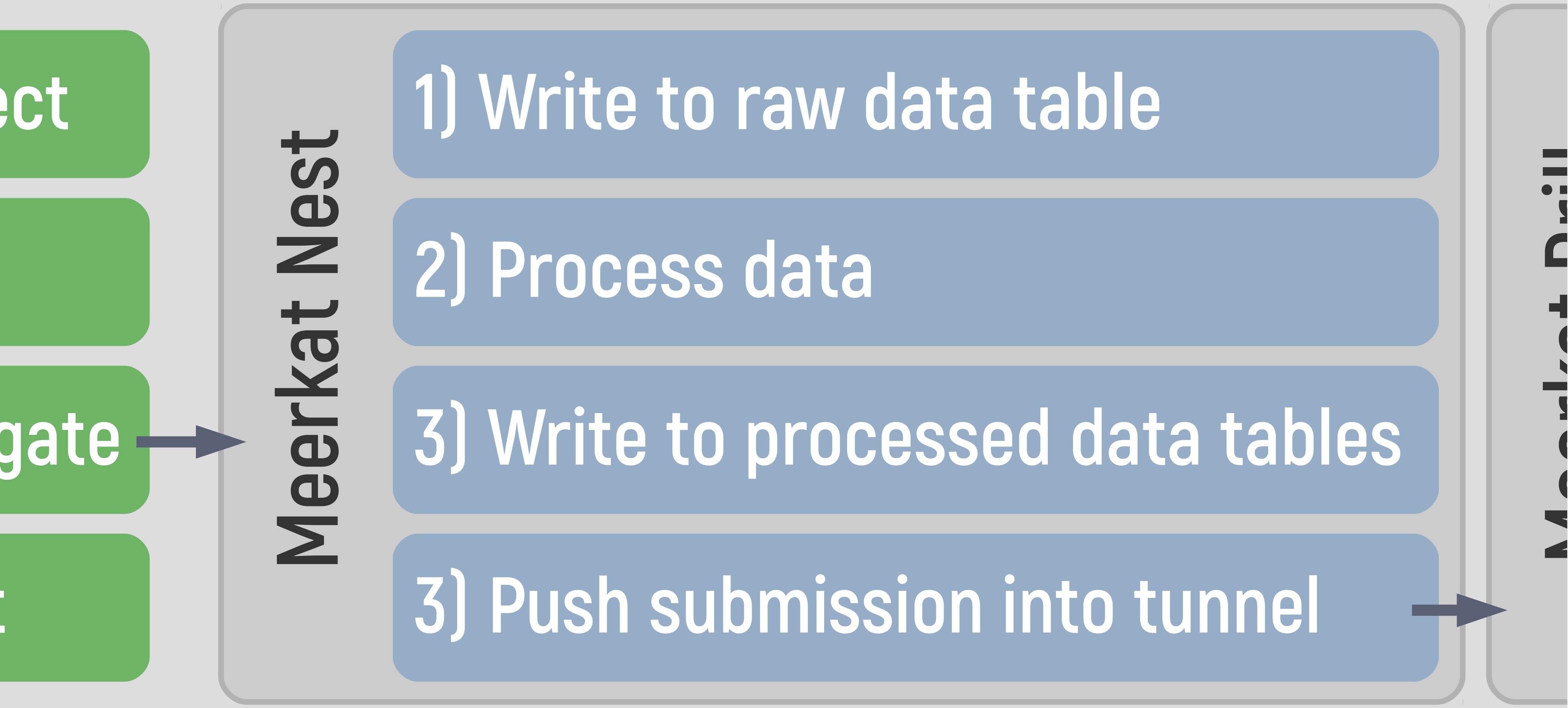
OPEN DATA KIT

NGINX

Apache Tomcat



Meerkat Nest



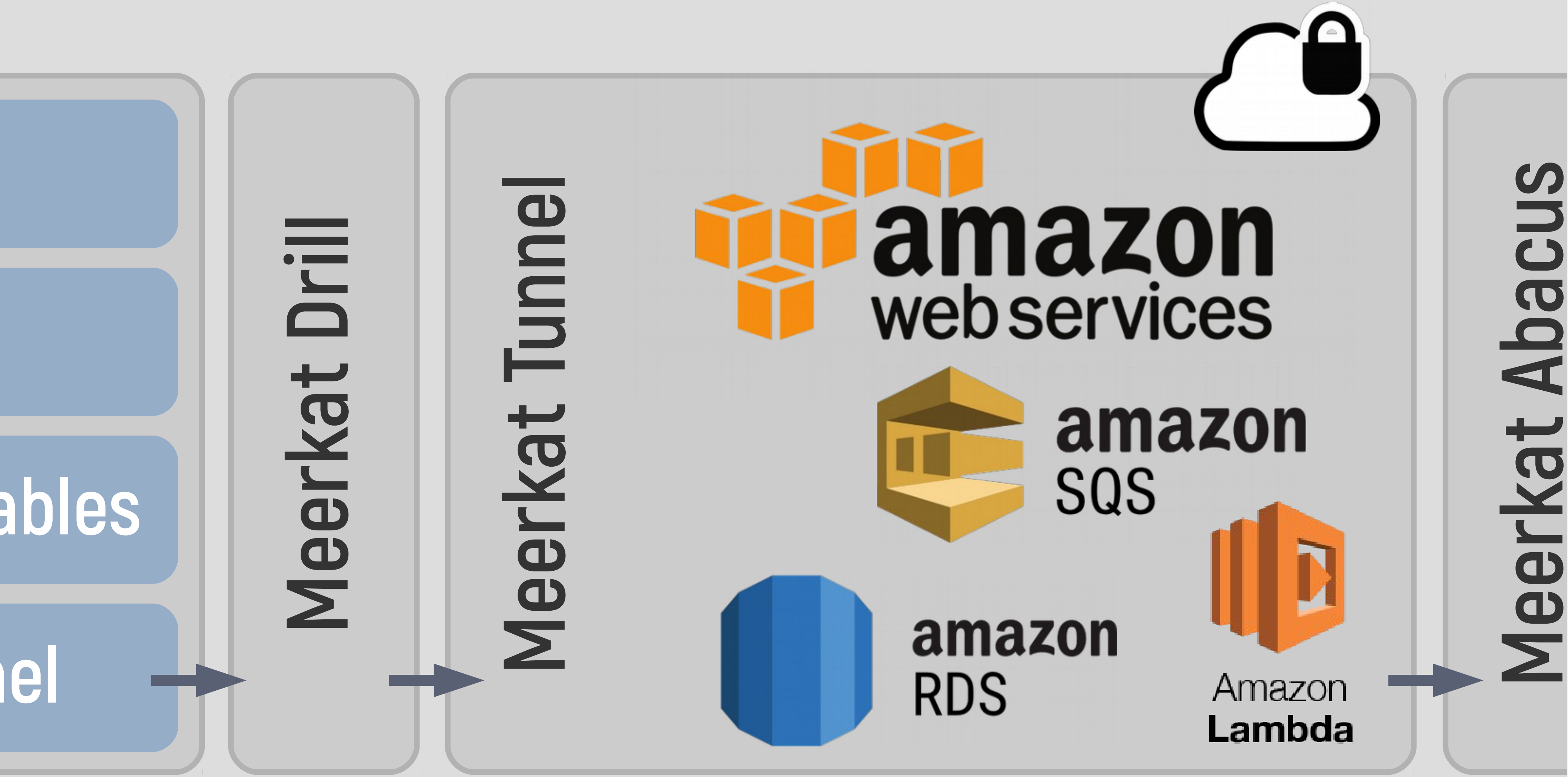
1) Write to raw data table

2) Process data

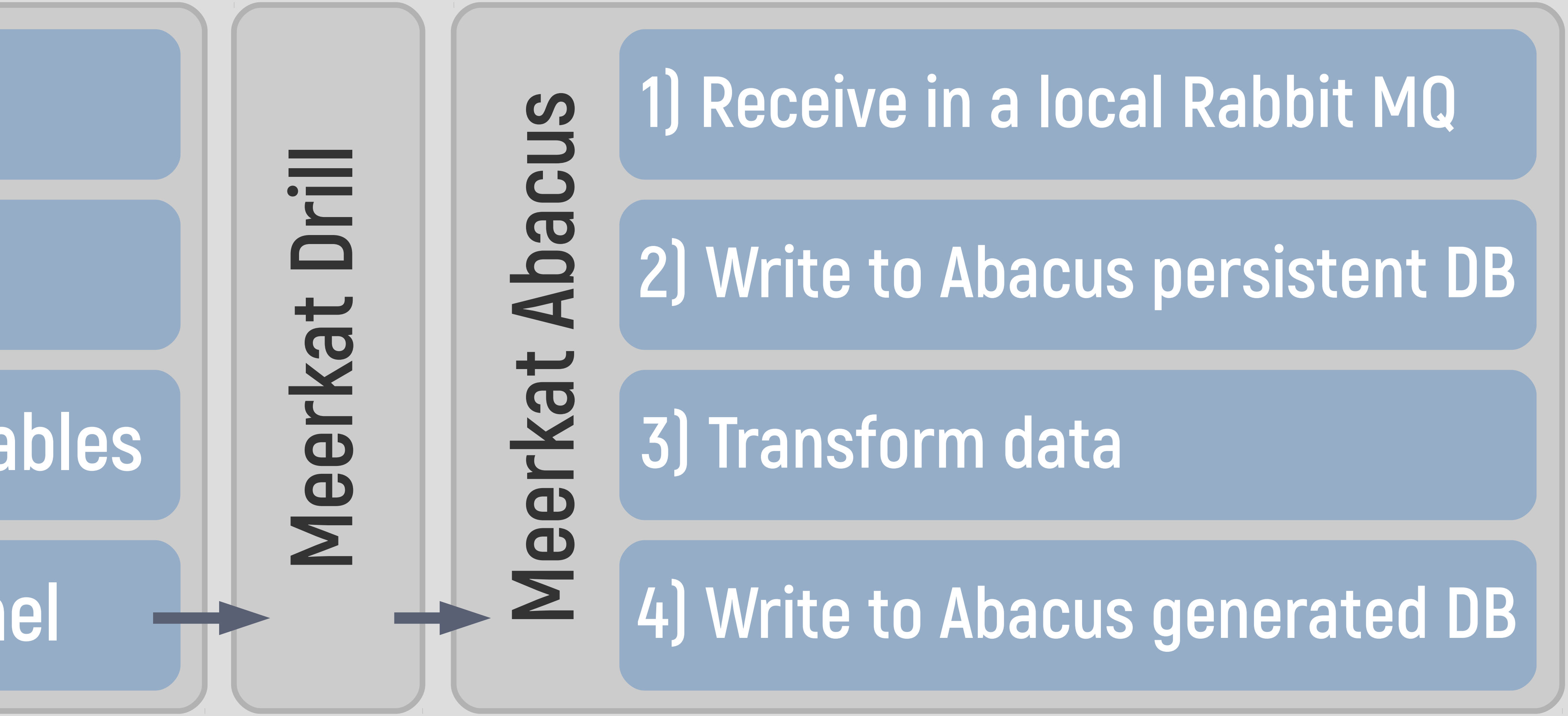
3) Write to processed data tables

3) Push submission into tunnel

Meerkat Tunnel



But Without Cloud?



ables

nel

Meerkat Drill

Meerkat Abacus

1) Receive in a local Rabbit MQ

2) Write to Abacus persistent DB

3) Transform data

4) Write to Abacus generated DB

There are improvements to be made!

Do we need the Abacus persistent DB?

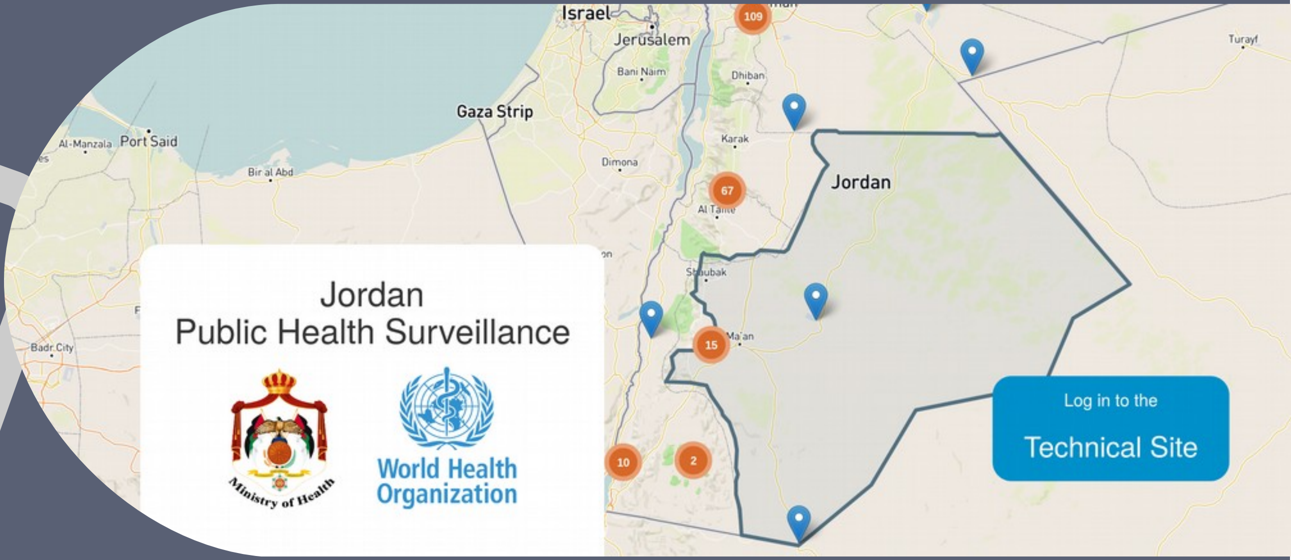
Does this need to be on two separate servers?

The Jordan fix was put together in a rush:

- Code quality and refactoring
- Testing
- Hardware backups and fail-safe methods

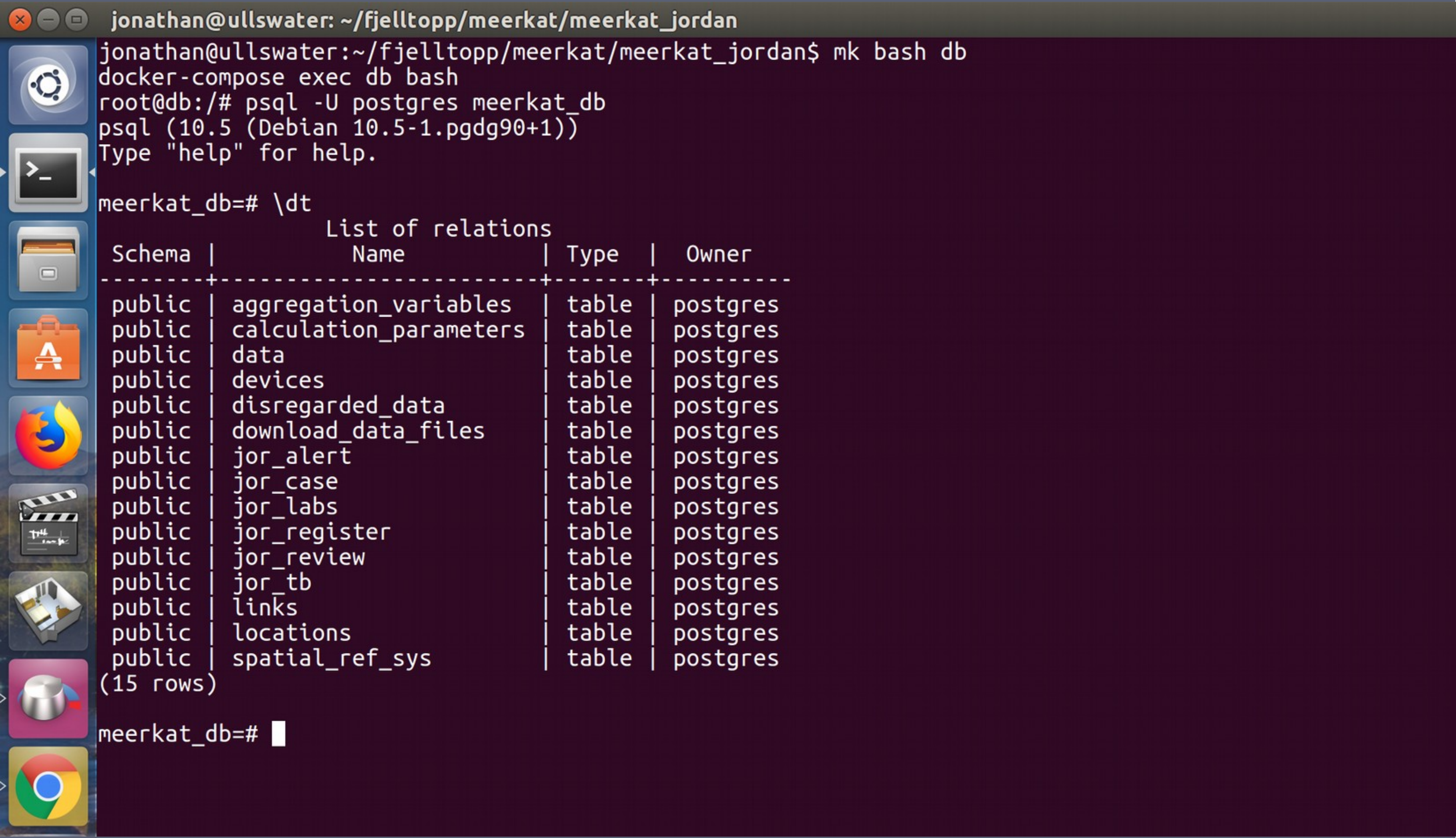
Exercise

The Website Server



Jordan
Public Health Surveillance

Meerkat Abacus Data Tables

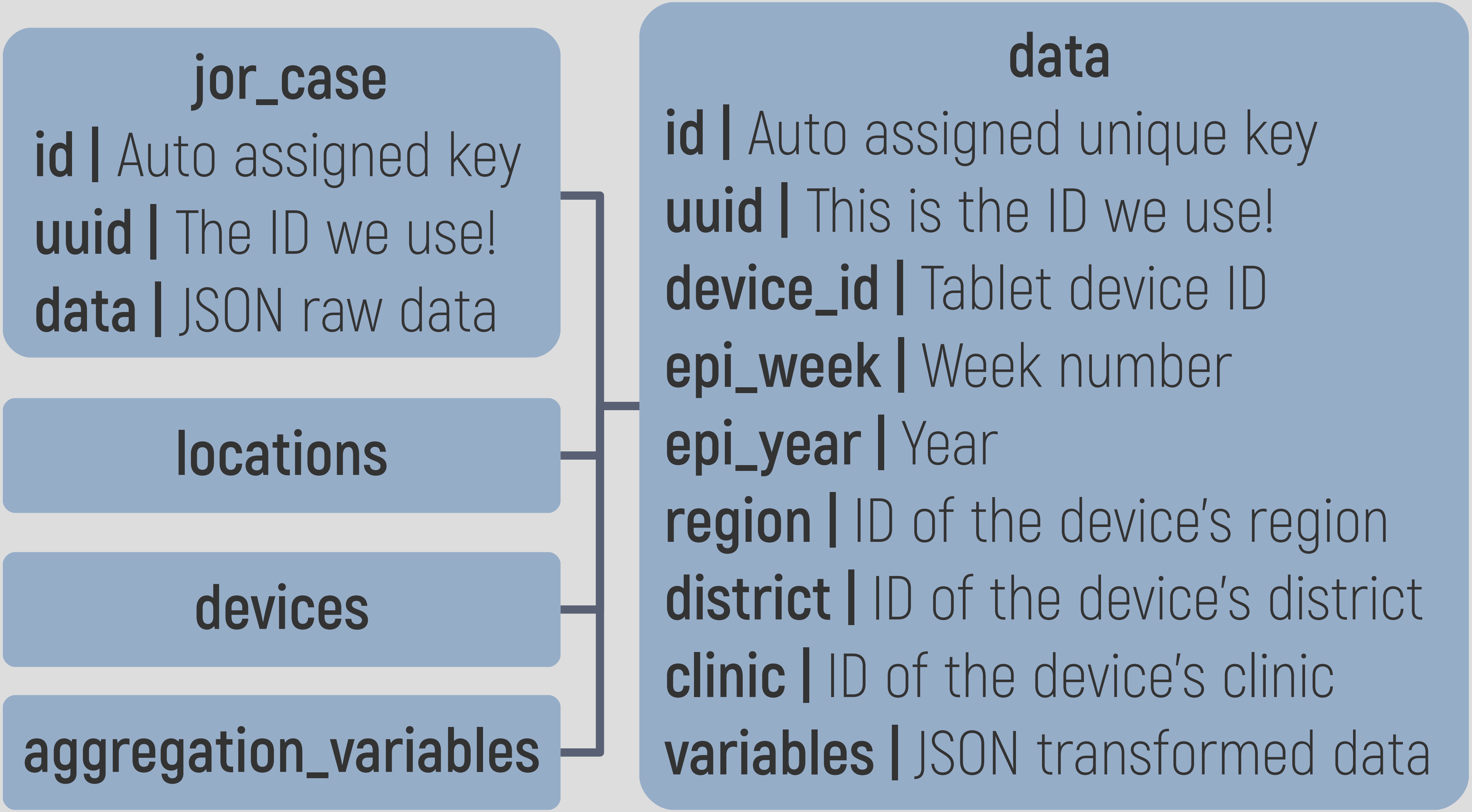


```
jonathan@ullswater:~/fjelltopp/meerkat/meerkat_jordan$ mk bash db
docker-compose exec db bash
root@db:/# psql -U postgres meerkat_db
psql (10.5 (Debian 10.5-1.pgdg90+1))
Type "help" for help.

meerkat_db=# \dt
                  List of relations
 Schema |          Name          | Type  |  Owner
--------+------------------------+-------+----------
 public | aggregation_variables  | table | postgres
 public | calculation_parameters | table | postgres
 public | data                   | table | postgres
 public | devices                | table | postgres
 public | disregarded_data       | table | postgres
 public | download_data_files    | table | postgres
 public | jor_alert              | table | postgres
 public | jor_case               | table | postgres
 public | jor_labs               | table | postgres
 public | jor_register           | table | postgres
 public | jor_review             | table | postgres
 public | jor_tb                 | table | postgres
 public | links                  | table | postgres
 public | locations              | table | postgres
 public | spatial_ref_sys        | table | postgres
(15 rows)

meerkat_db=# █
```

Meerkat Abacus: Data Transformation



jor_case

id | Auto assigned key

uuid | The ID we use!

data | JSON raw data

locations

devices

aggregation_variables

data

id | Auto assigned unique key

uuid | This is the ID we use!

device_id | Tablet device ID

epi_week | Week number

epi_year | Year

region | ID of the device's region

district | ID of the device's district

clinic | ID of the device's clinic

variables | JSON transformed data

Meerkat Abacus: Data Transformation

Where?

[Abacus Country Configs]/variable_codes

What?

A selection of CSV files that are concatenated together

Example?

Code: { “id”: “gen_1”,
“method”: “match”,
“db_column”: “pt1./gender”
“condition”: “male” }

Raw: { “pt1./gender”: “male” }
Variables: { “gen_1”: 1 }

Meerkat Runner: Asynchronous Tasks

Celery

Asynchronous tasks:

- Processing new submissions
- Generating data sets for download
- Sending reports by email

Remember to check

Meerkat Runner

logs as well!

Meerkat API: Data Aggregation

Provides a **HTTP RESTful API** interface to **explore** the data in meerkat_db.

Enter the world of Web Programming in Python…

Exercise

Data Transformation

	A	B	F	J	
1	id	name	db_column	method	co
2	tot_1	Total	SubmissionDate	not_null	
3	submission_date	Submission Date	SubmissionDate	value	
4	end_edit_date	End Edit date	end	value	
5	gen_1	Male	pt1./gender	match	male
6	gen_2	Female	pt1./gender	match	fema
7	nat_1	Jordan	nationality	match	jorda
8	nat_2	Syria	nationality	match	syria
		Egypt	nationality	match	egyp
		Iraq	nationality	match	iraq.i

Summary

Securely managing and processing data in real time is a complex task.

ODK aggregate is used to collect data only.

Meerkat Nest DB should be considered the single source of truth.

All data after that can be regenerated, though this takes a long time!

Nest & Abacus process and transform data.

Challenge: Can we refactor the data pipeline to simplify it for local infrastructure?