Mastering a data pipeline with Python:

6 years of learned lessons from mistakes

to success

Robson Júnior

GitHub

Me

DEVELOPER
+ 16 YEARS

TELEGRAM:
BSAO0

TWITTER:
BSAO

GITHUB

Agenda

It’s not about code

Anatomy of a data product

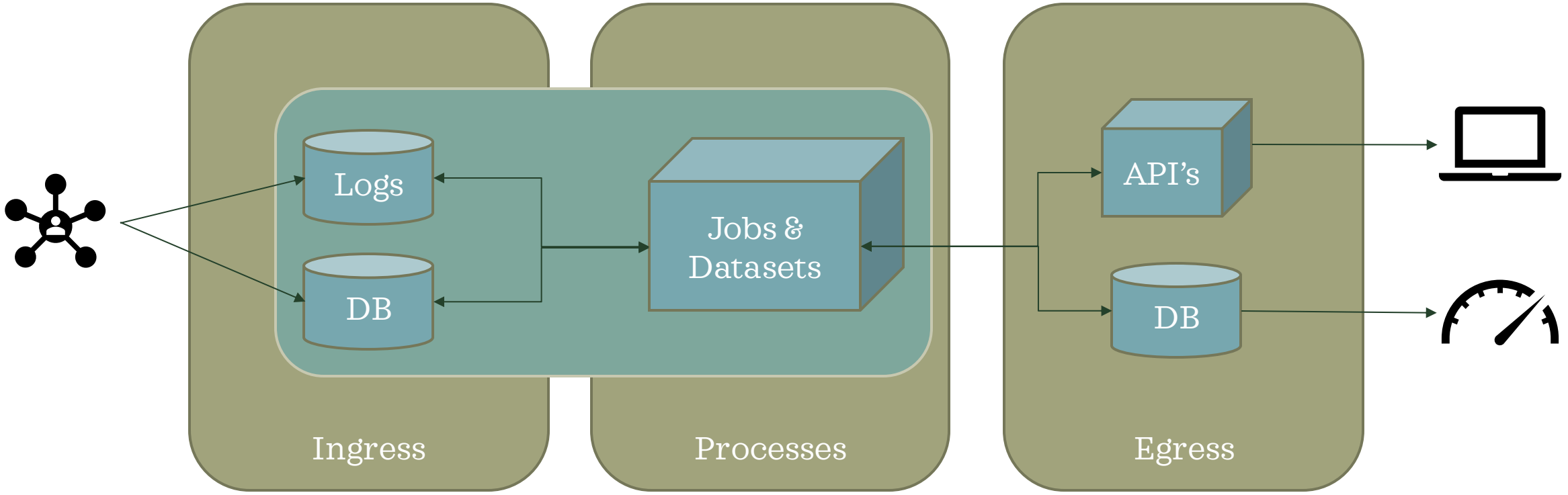
Lambda vs Kappa Architecture

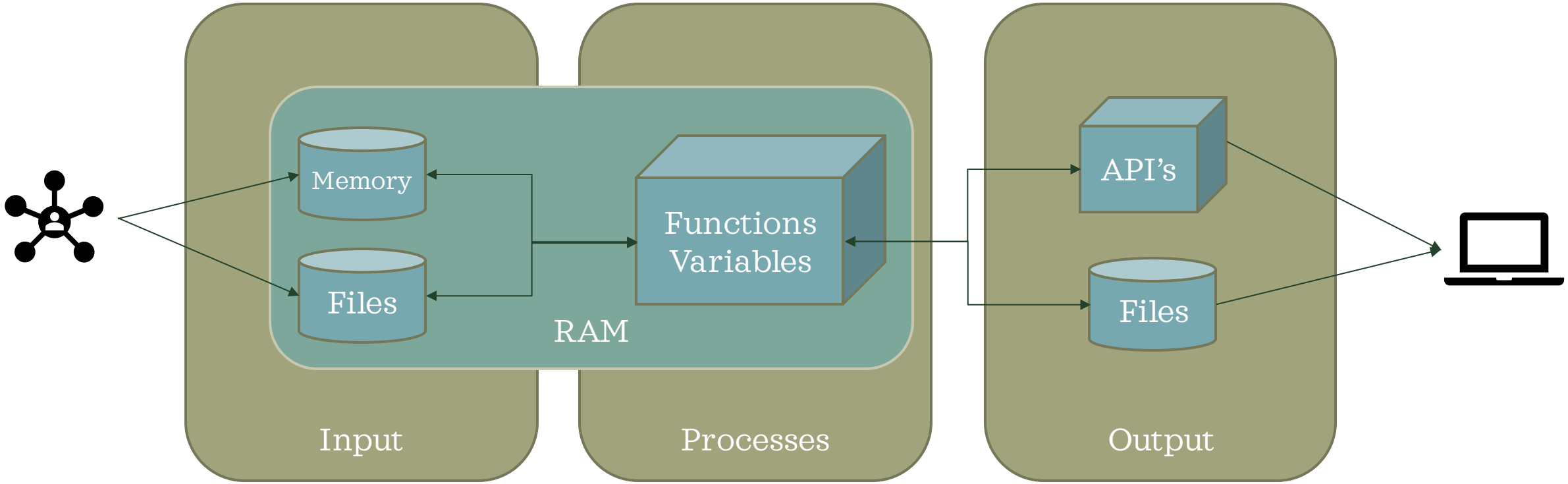
Qualities of a data pipeline

Where python matters

My goal is help you to start to planning great data driven products.

Anatomy of a data product



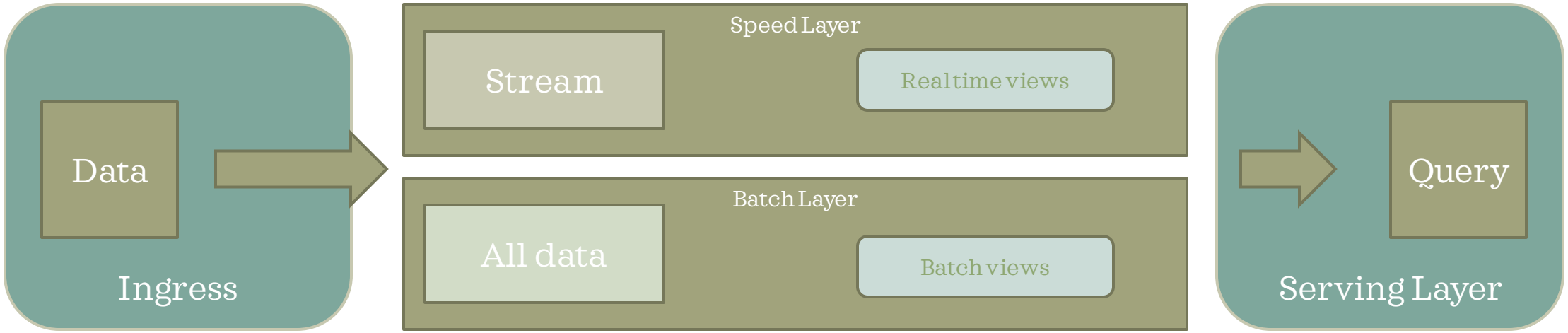


SAME AS A COMPUTER PROGRAM

Lambda and Kappa architecture

Λ vs **K**

Lambda


Data
Ingress
Speed Layer
Stream
Real time views
Batch Layer
All data
Batch views
Query
Serving Layer

Applications

System that requires permanent data stored.

User queries based on immutable data.

Users or Systems that requires huge amount of updates in the data and serves it in new datasets.

Pros

Reliable and safe

Fault tolerant (you can re-processes everything from scratch)

Scalable

Manage all the historical data in a distributed file system (delta lake)

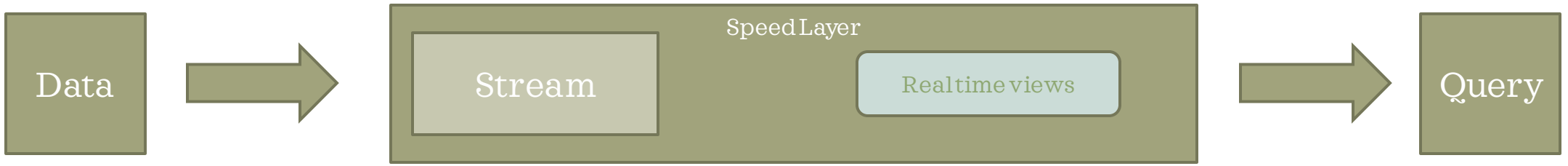
Cons

Premature data modelling, it's getting hard to migrate schemas or datasets.

Might be expensive due to volume of data you need to processes in each batch cycle.

Code can become complex due the separation of concerns between the layers.

Kappa



Pro tip: Unless you desperate for real time answers, stay in Batch Process

Applications

You do need a well-define event order and can interact with your dataset any time.

Systems that need a real time learning (Social Networks, Ads Platform, Fraud Detection)

Focus on the code changes

Pros

Use less resource than Lambda architecture

Leverage Machine Learning to real time basis

Horizontally scalable

You just need to reprocess the data when the code changes

Cons

Errors on data processing need a better exception manager

Might stop the pipeline to get bugs fixed

Qualities of a Pipeline

IT'S A COMPUTER PROGRAM :)

PROBLEMS ARE ALMOST THE SAME

If you see something that will get wrong in a software, probably it will get wrong on a data pipeline.

Security

Access levels to the data levels

Privacy over all layers

Use a common format

Separation of concerns

Avoid hard-coding / duplication

Automation

Versioning

Use the power of different tech platforms

CI/CD

Code Review / Lint

Monitoring

Let cloud to help you (cheap and fast)

Avoid vendor lock-in

Infrastructure monitoring

Testable and Traceable

Regression tests

Inputs must be deterministic

Focus in test the units of the pipeline (internal)

Test all the 3rd party components

After all implement an end-to-end test

Python plays well with all technologies

Where Python matters

- ELT
- Streaming
- Analysis
- Management & Scheduling
- Testing
- Validation

PySpark - Apache Spark Python API.

dask - A flexible parallel computing library for analytic computing.

luigi - A module that helps you build complex pipelines of batch jobs.

mrjob - Run MapReduce jobs on Hadoop or Amazon Web Services.

Ray - A system for parallel and distributed Python that unifies the machine learning ecosystem.

Where Python matters

- ELT
- Streaming
- Analysis
- Management & Scheduling
- Testing
- Validation

faust - A stream processing library, porting the ideas from Kafka Streams to Python.

streamparse - Run Python code against real-time streams of data via Apache Storm.

Where Python matters

- ELT
- Streaming
- Analysis
- Management & Scheduling
- Testing
- Validation

Pandas - A library providing high-performance, easy-to-use data structures and data analysis tools.

Blaze - NumPy and Pandas interface to Big Data.

Open Mining - Business Intelligence (BI) in Pandas interface.

Orange - Data mining, data visualization, analysis and machine learning through visual programming or scripts.

Optimus - Agile Data Science Workflows made easy with PySpark.

Where Python matters

- ELT
- Streaming
- Analysis
- Management & Scheduling
- Testing
- Validation

Airflow - Airflow is a platform to programmatically author, schedule and monitor workflows.

Where Python matters

- ELT
- Streaming
- Analysis
- Management & Scheduling
- Testing
- Validation

pytest - A mature full-featured Python testing tool.

mimesis - is a Python library that help you generate fake data.

fake2db - Fake database generator.

https://github.com/holdenk/spark-testing-base - a python framework to implemente pyspark tests

Where Python matters

- ELT
- Streaming
- Analysis
- Management & Scheduling
- Testing
- Validation

Cerberus - A lightweight and extensible data validation library.

schema - A library for validating Python data structures.

voluptuous - A Python data validation library.

Obrigado
Thank you
dankie
shukran
do jeh
xie xie
dêkuji
tak
kiitos
merci
danke
efharisto
toda
sukria
terima kasih
grazie
arigato
kamsa hamnida
takk
salamat po
dziekuje
spasibo
gracias
istutiy
asante
tack
kawp-kun krap/ka'
tesekkür ederim

QUESTIONS?

HELLO@BSAO.ME