Bootcamp
Bringing ML Models into Production

Lesson 1: Intro

pyladies amsterdam

DEXTER

Alyona Galyeva

Agenda

- About us
- Bootcamp Setup
- MLOps
- Home assignment

ABOUT US

PyLadies Amsterdam Team

Where to find us?

Website
https://amsterdam.pyladies.com

Materials
https://github.com/pyladiesams
https://www.youtube.com/pyladiesamsterdam

Events
https://www.meetup.com/PyLadiesAMS

Updates
https://www.linkedin.com/company/pyladies-amsterdam
https://twitter.com/PyLadiesAMS

Bootcamp Setup

DEXTER

FORECASTING ENERGY USING AI.

Dexter-PyLadies energy case

July 2021

DEXTER

cluster	imba/h [€/MWh]	imba irr. [€/MWh]	imba/pte [€/MWh]	imba/pte [GWh]	total abs load [GWh]	NMAE [%]	total cost [€]
B2C solar	2.613	0.721	2.016	0.034	0.069	49.45	180
wind	1.197	0.483	0.949	0.072	0.391	18.34	468
B2B solar	2.566	0.558	2.110	0.077	0.448	17.18	1,150
profile	0.817	0.239	0.638	0.058	0.866	6.64	708
telemetry	-0.100	0.078	-0.139	0.047	0.954	4.89	-95
total	0.858	0.265	0.658	0.167	2.729	6.14	2,342

Impact of Dexter

Forecast renewable energy to:

- Make renewable energy more profitable and push fossil fuel energy out of the market
- Balance the demand and generation of electricity to prevent strongly polluting emergency solutions

DEXTER

SERVICES

We offer forecasting and portfolio optimisation services enabling clients to turn challenges in the energy transition into opportunities

Load Forecasting

Flex asset optimisation

Portfolio optimisation

DEXTER

About Dexter

- Founded in 2016, based in Amsterdam
- Focused on short-term power trading optimisation
- Team:
  22 FTE - Energy market experts |
  Computer & Data Scientists |
  Weather model experts
- Countries:
  Benelux, Germany, Austria & UK

DEXTER

Working at Dexter

- Dynamic and young start-up
- Challenging software development & data science
- Contributing to the transition to renewable energy
- Fun team

Website: https://dexterenergy.ai/
Feel free to contact us with any questions

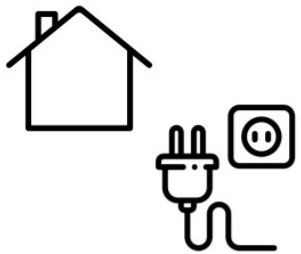
DEXTER

DEXTER

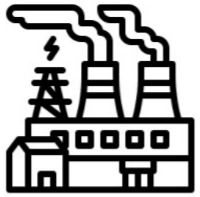
Dexter - PyLadies case

OLD APPROACH

Determine
energy consumption

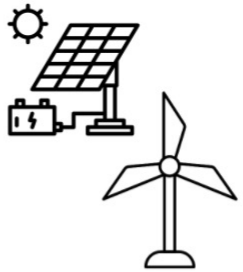


Adjust
energy generation

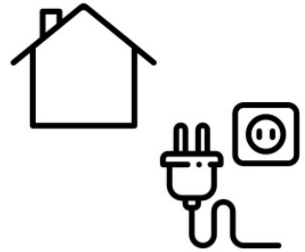


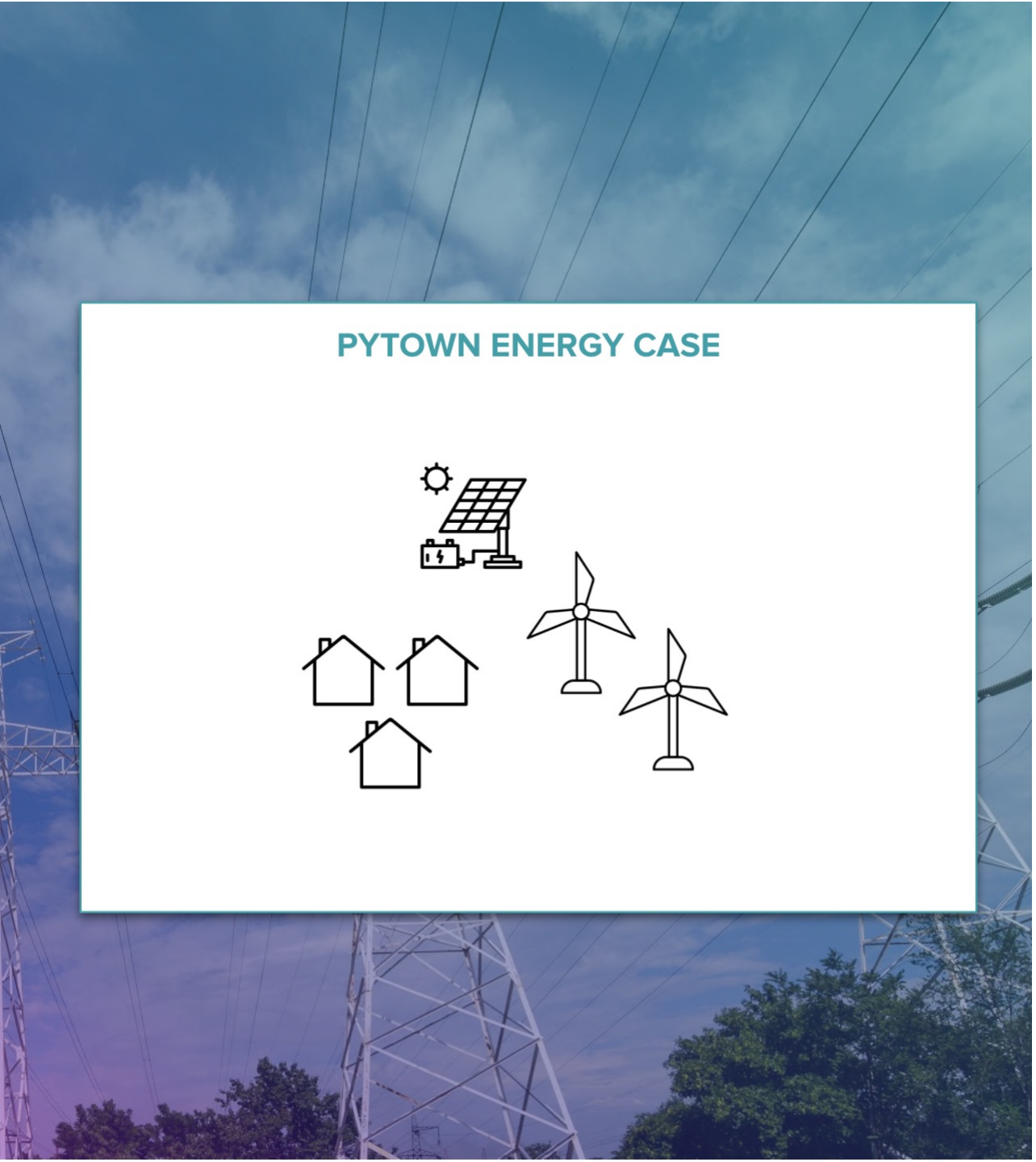
NEW APPROACH

Determine
energy generation



Adjust
energy consumption





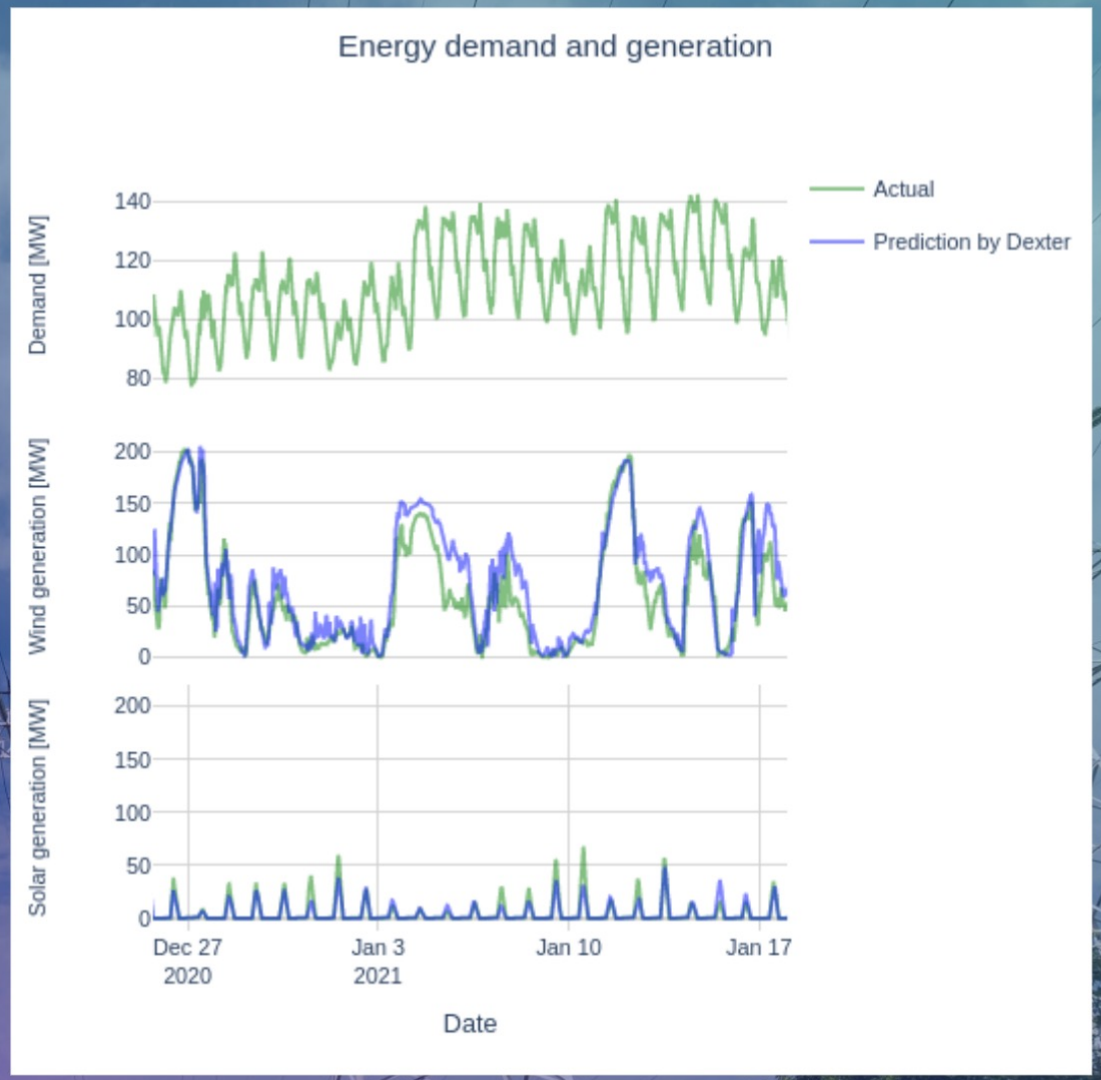
Dexter-Pyladies energy case

Pytown is a village in the North of the Netherlands that wants to become fully self-sufficient with 100% renewable energy.

✓ The village has built sufficient solar panels & wind turbines to produce their own energy demand.

✗ The timing of the energy generation does not match the timing of demand. This raises a lot of problems for Pytown. Frequent blackouts are the order of the day. Can you help Pytown to adjust their demand to the energy generation?

DEXTER



Dexter-Pyladies energy case

To prevent blackouts energy generation & demand have to be matched:

- The solar & wind forecast you can acquire from an external company
- Can you deploy a energy demand prediction model to solve the problem of Pytown?

DEXTER

Alyona Galyeva

- AI & Data Engineering Lead at **LINKIT**
- **PyLadies Amsterdam** Organizer
- AI Mentor at **WAI Accelerate**
- Microsoft **AI MVP**

Bootcamp

- Schedule and Learning materials
- Slack
- Mentor support
- Capstone

MLOps = ML + DEV + OPS

MLOps

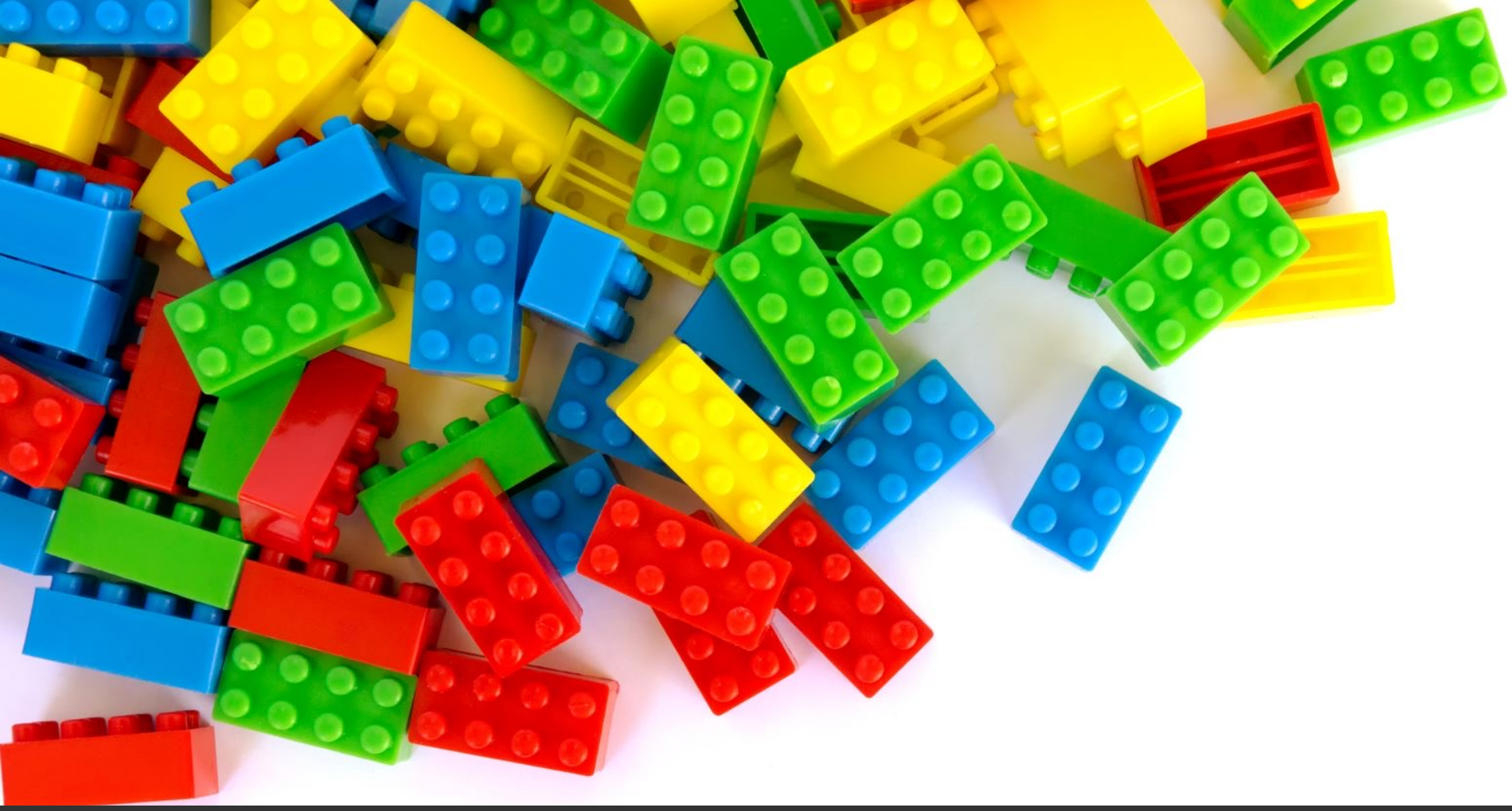
MLOps

MLOps is the standardization and streamlining of machine learning life cycle management.

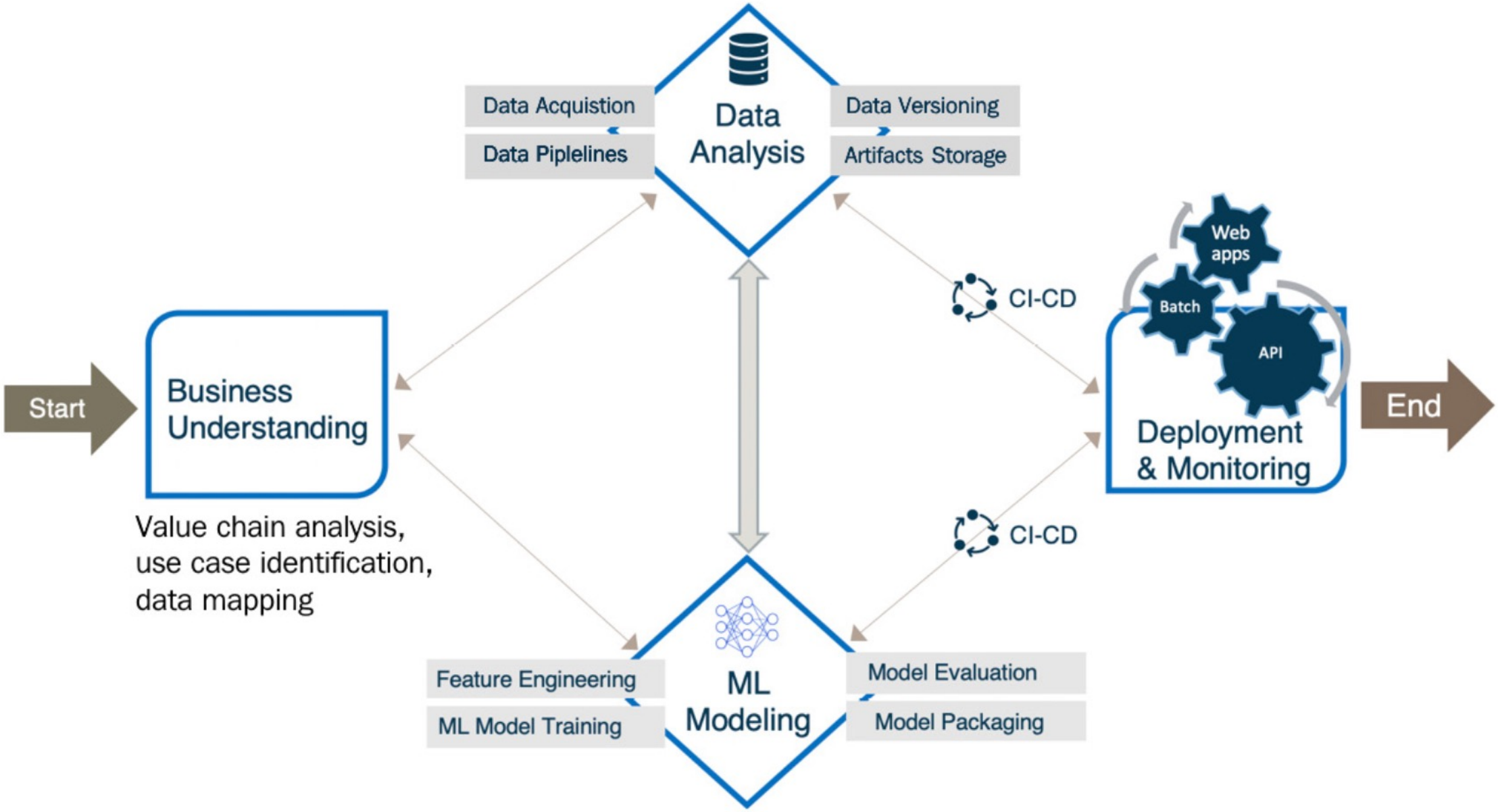
There are three key reasons that managing machine learning life cycles at scale is challenging:

- There are many dependencies
- Not everyone speaks the same language
- Lack of software engineers with ML knowledge

Source: Du Phan and others. Introducing MLOps. Chapter 1.



ML lifecycle building blocks



Source: Emmanuel Raj. Engineering MLOps. Chapter 2. Figure 2.1 – ML solution development process

Common ML challenges

- Data quality – “garbage in, garbage out” (accuracy, completeness, consistency, timeliness)
- Reproducibility – inherent element of randomness
- Data and Model Drift – data and target relationships change over time
- Scale – infrastructure needs per each step
- Multiple Objectives – balanced optimizations

ML model deployment scenarios

	Research	**Production**
Data	Static	Dynamic (constantly changing)
Fairness	Recommended	Necessary
Interpretability	Recommended	Necessary
Performance	State of the art	Better than simpler models
Priority	Fast training	Fast inference

Source: Emmanuel Raj. Engineering MLOps. Chapter 6. Table 6.1 – ML in research and production

ML model deployment scenarios

The right approach depends on your needs, such as latency requirements, hardware, network and privacy concerns, and inference costs.

Server-Side Deployment:

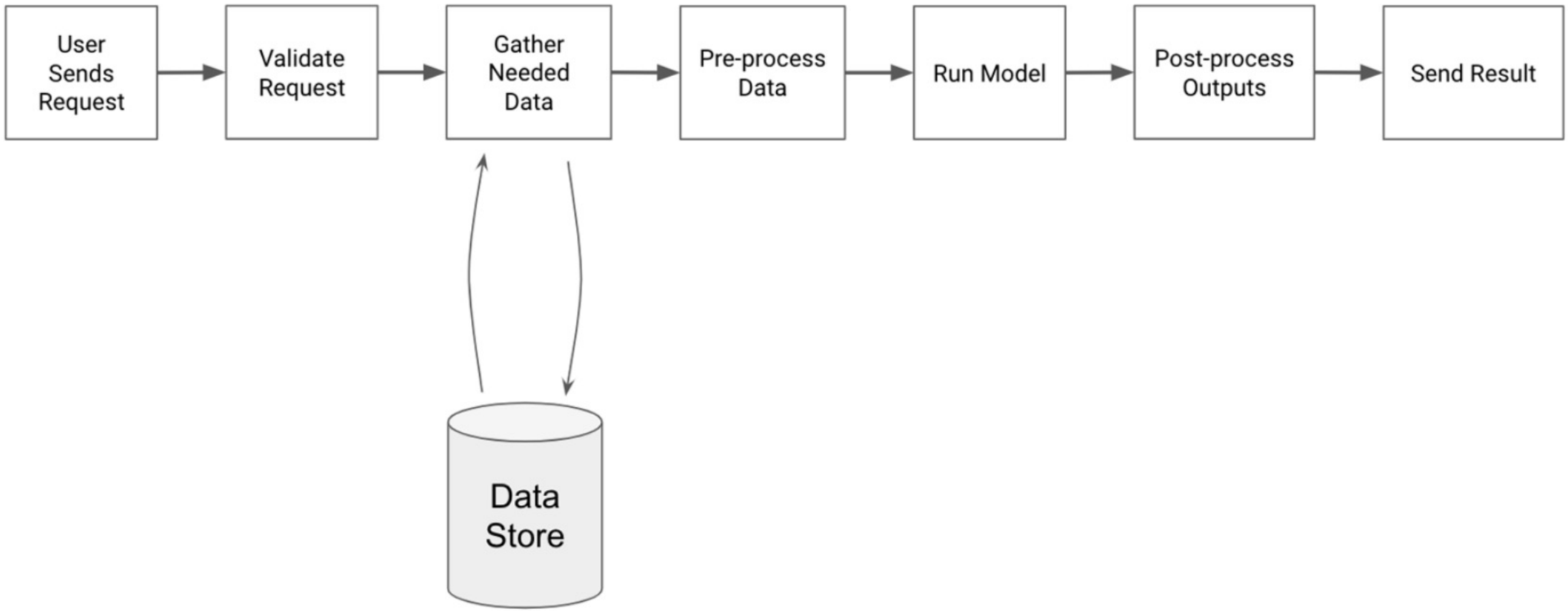
- streaming workflow
- batch workflow

Client-Side Deployment:

- on device
- browser side

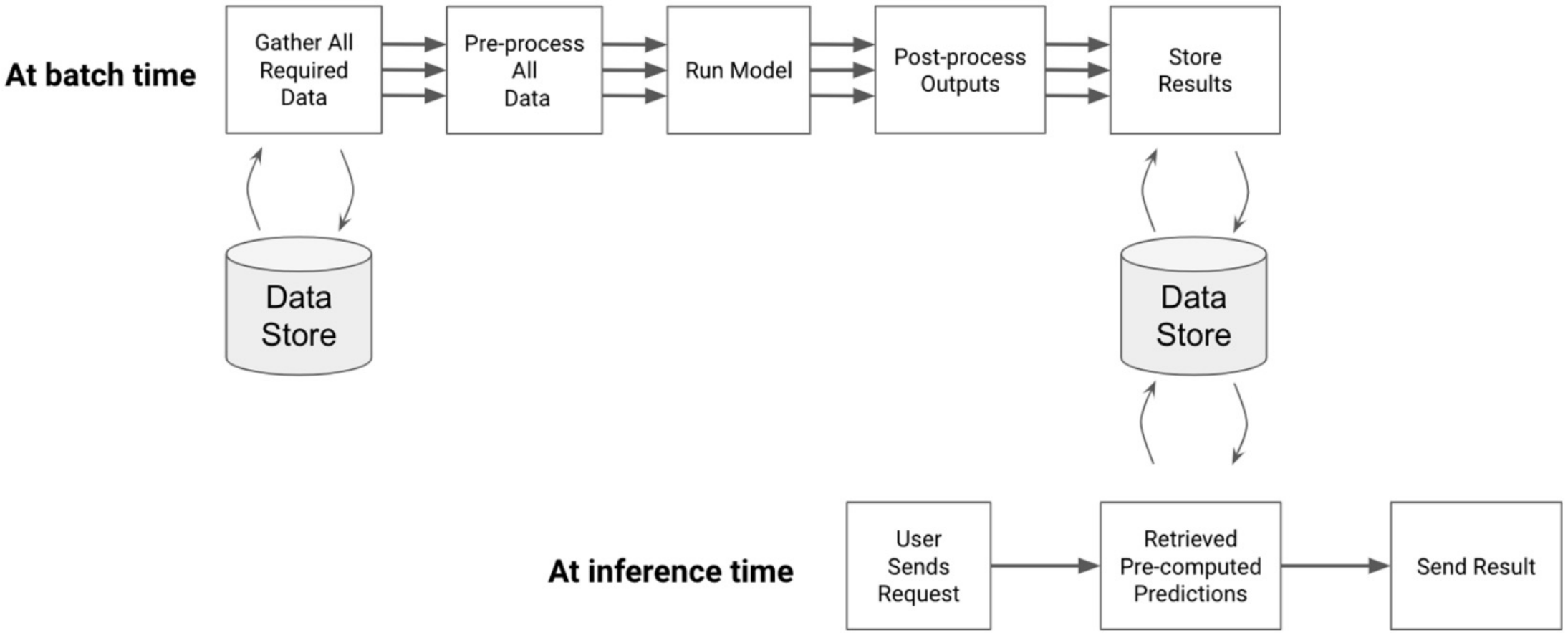
Federated Learning (A Hybrid Approach)

Streaming API Workflow



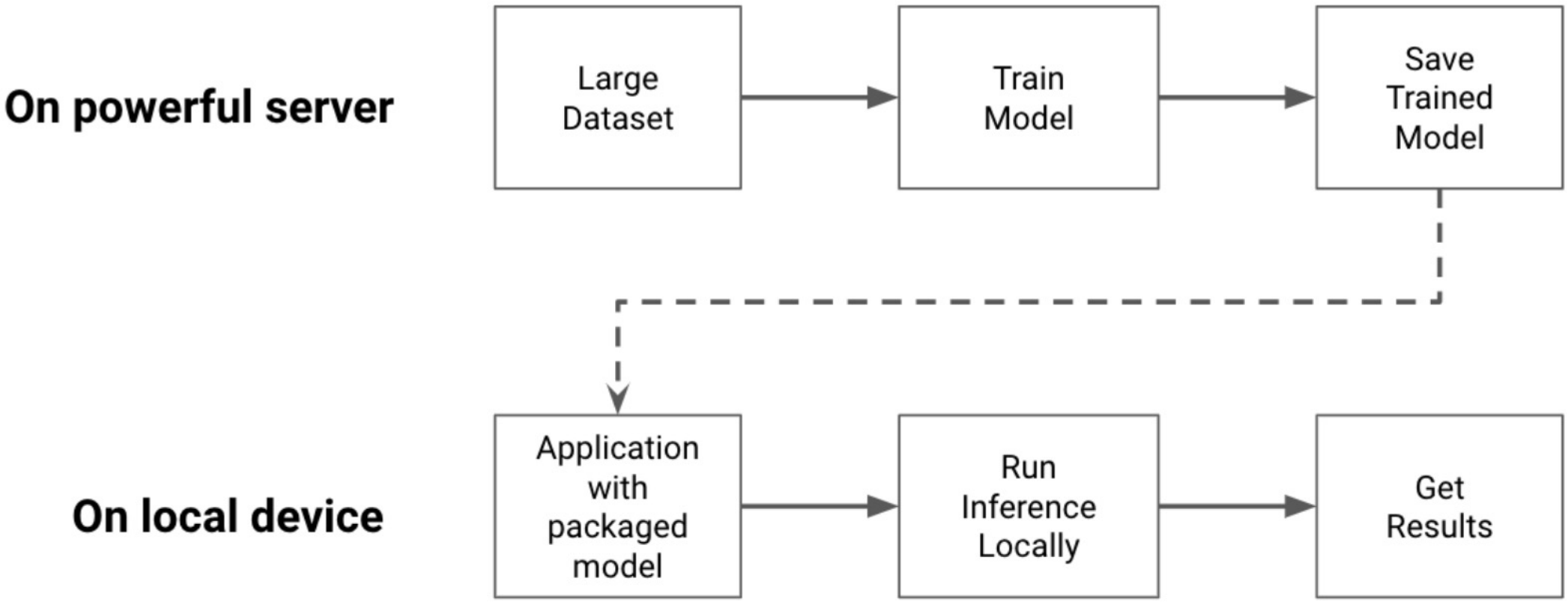
Source: Emmanuel Ameisen. Building Machine Learning Powered Applications. Chapter 9. Figure 9-1. Streaming API workflow

Batch Workflow



Source: Emmanuel Ameisen. Building Machine Learning Powered Applications. Chapter 9. Figure 9-3. Example of batch workflow

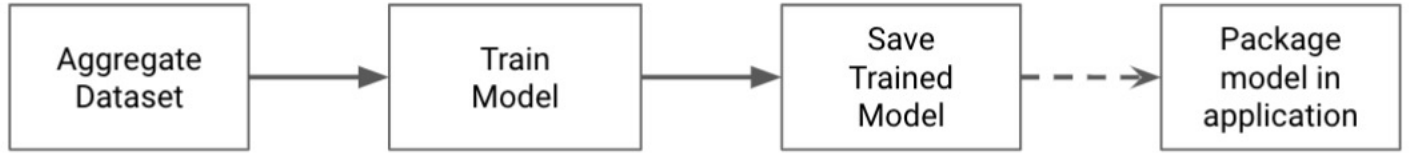
Device workflow



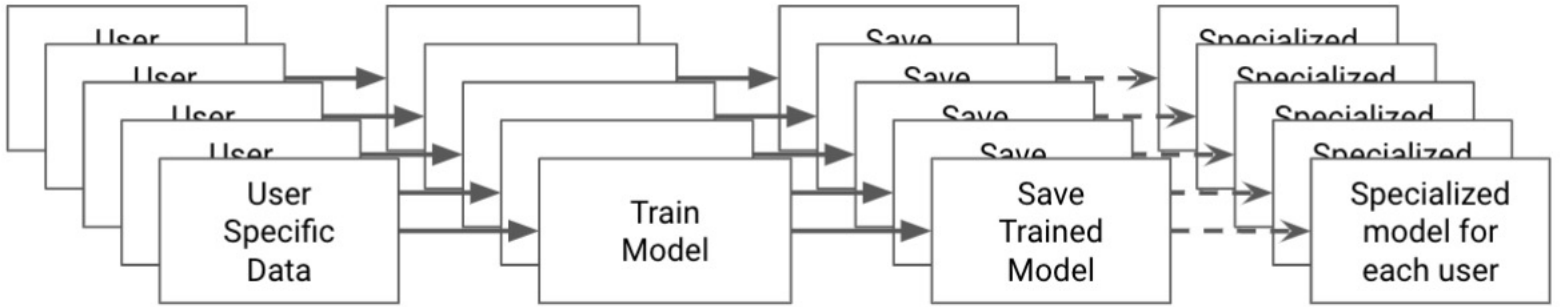
Source: Emmanuel Ameisen. Building Machine Learning Powered Applications. Chapter 9. Figure 9-4. A model running inference on device

Federated Learning

One model for all users

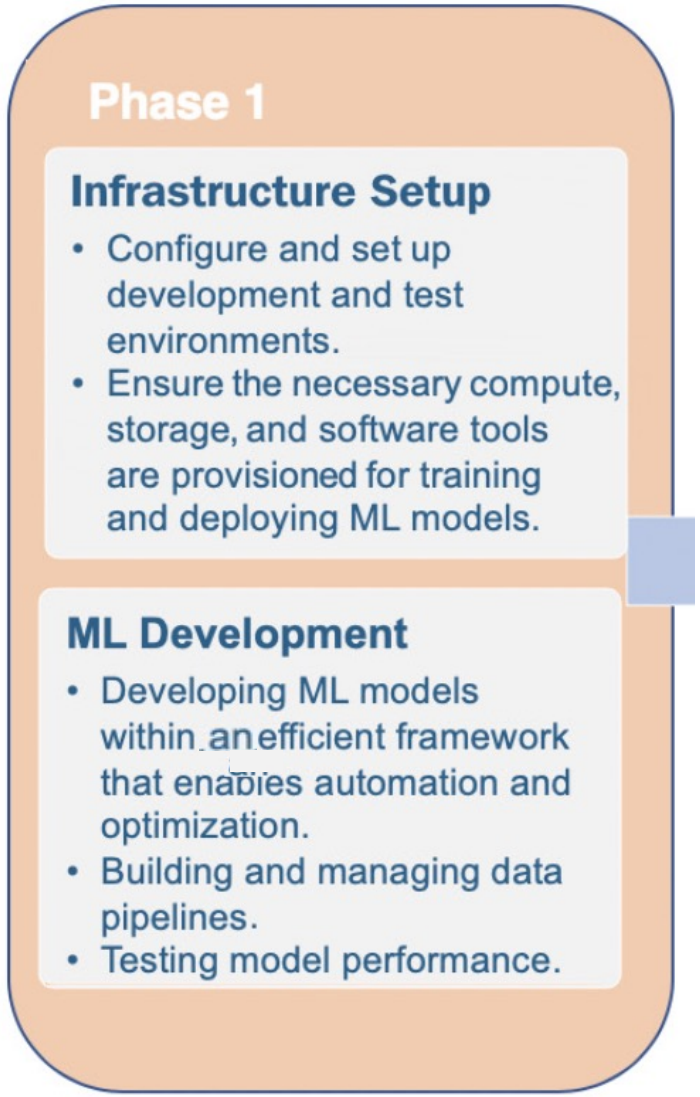


Each user has a model

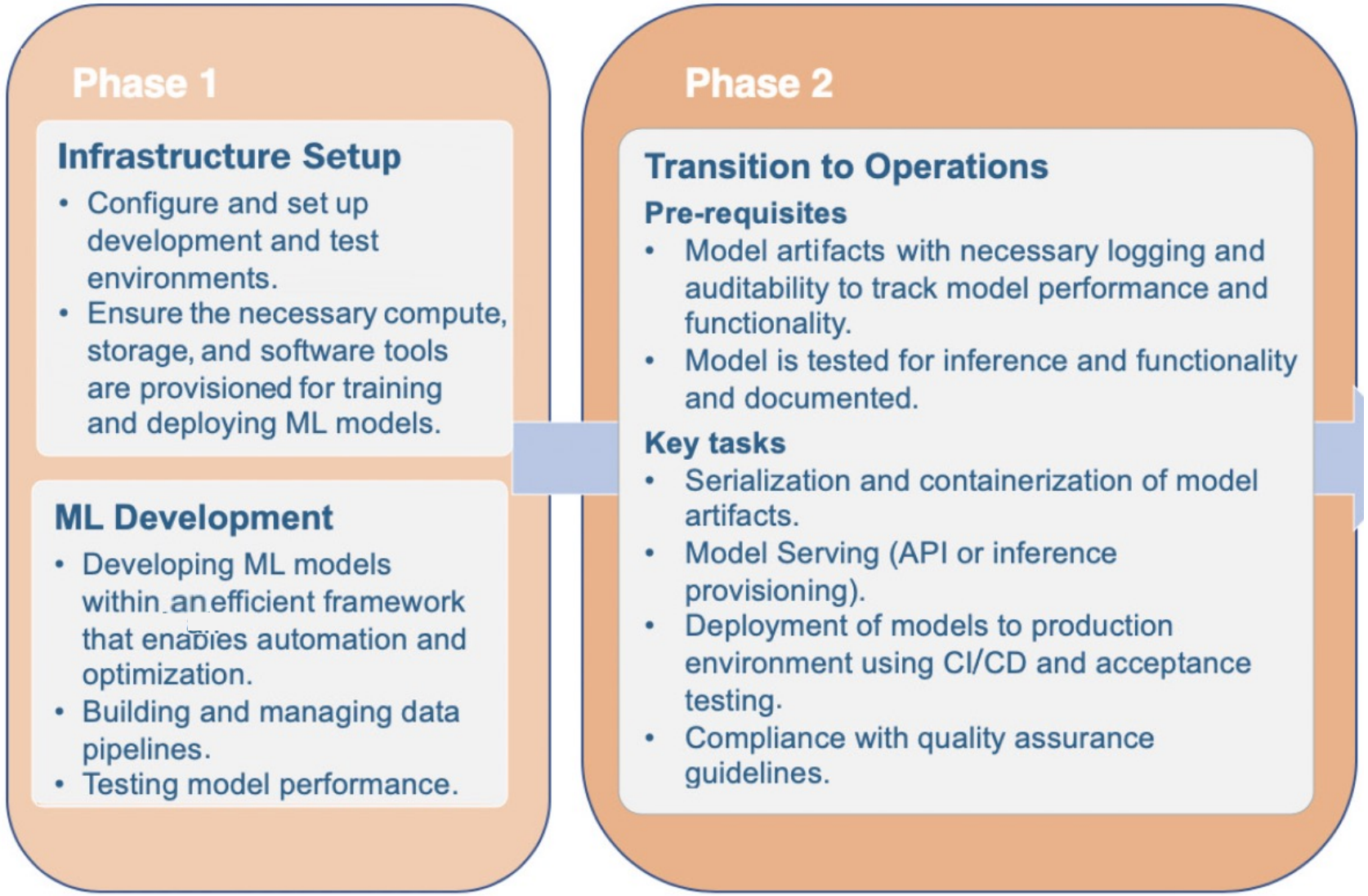


Source: Emmanuel Ameisen. Building Machine Learning Powered Applications. Chapter 9. Figure 9-6. One big model or many individual ones

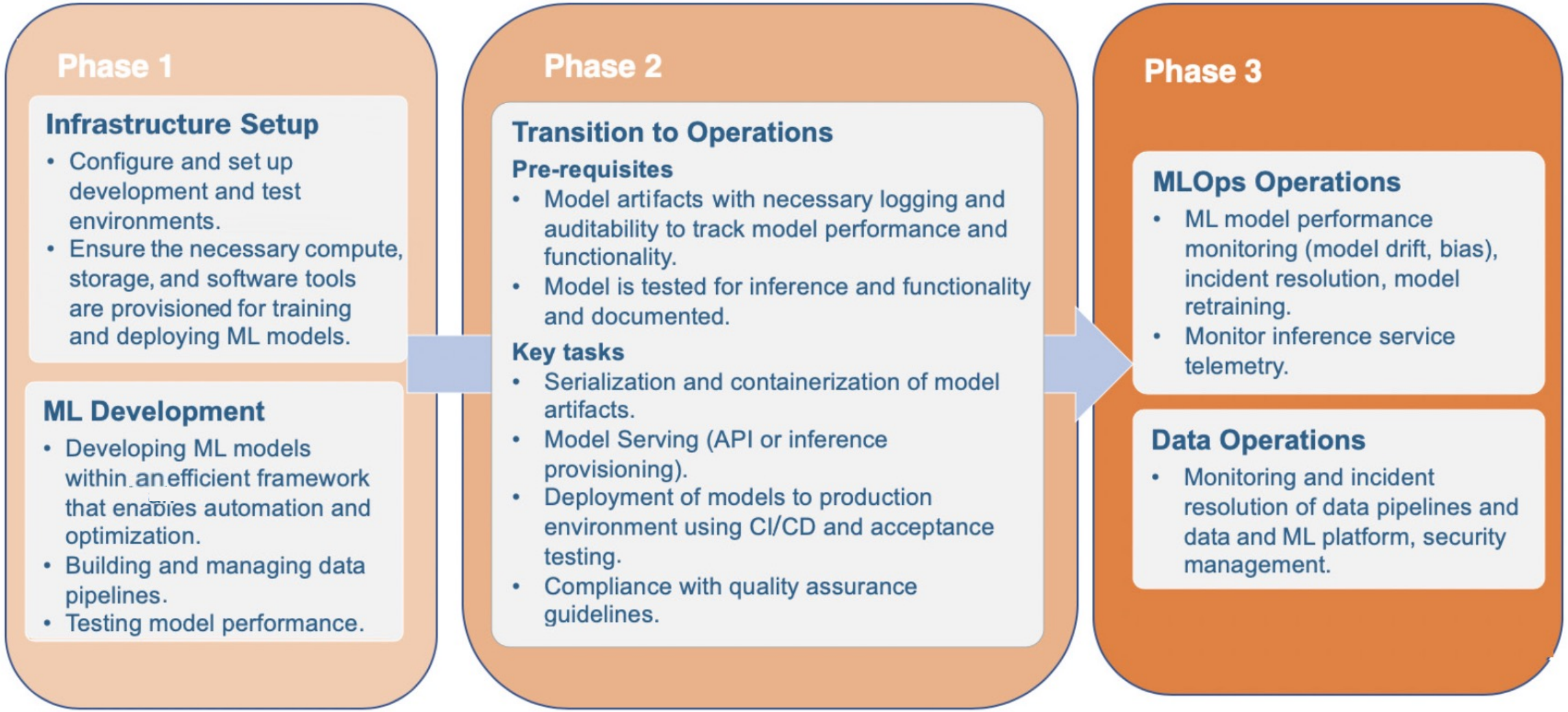
Steps to prepare model for deployment



Source: Emmanuel Raj. Engineering MLOps. Chapter 2. Figure 2.10 – Implementation roadmap for an MLOps-based solution



Source: Emmanuel Raj. Engineering MLOps. Chapter 2. Figure 2.10 – Implementation roadmap for an MLOps-based solution



Source: Emmanuel Raj. Engineering MLOps. Chapter 2. Figure 2.10 – Implementation roadmap for an MLOps-based solution

Home assignment

Home assignment – Lesson 1

https://github.com/pyladiesams/bootcamp-bringing-ML-models-into-production-intermediary-jun-aug2021/blob/master/bootcamp/lesson1/lesson1_tasks.md

print(f“{user_name} thanks for watching”)