DevOps per Machine Learning: training e inference pipelines con MLOps

Clemente Giorio – Deltatre – R&D Senior SW Engineer
Vito Flavio Lorusso – Microsoft – Program Manager
Gianni Rosa Gallina – Deltatre – R&D Senior SW Engineer

{codemotion}
Online Tech Conference
- Italian edition -

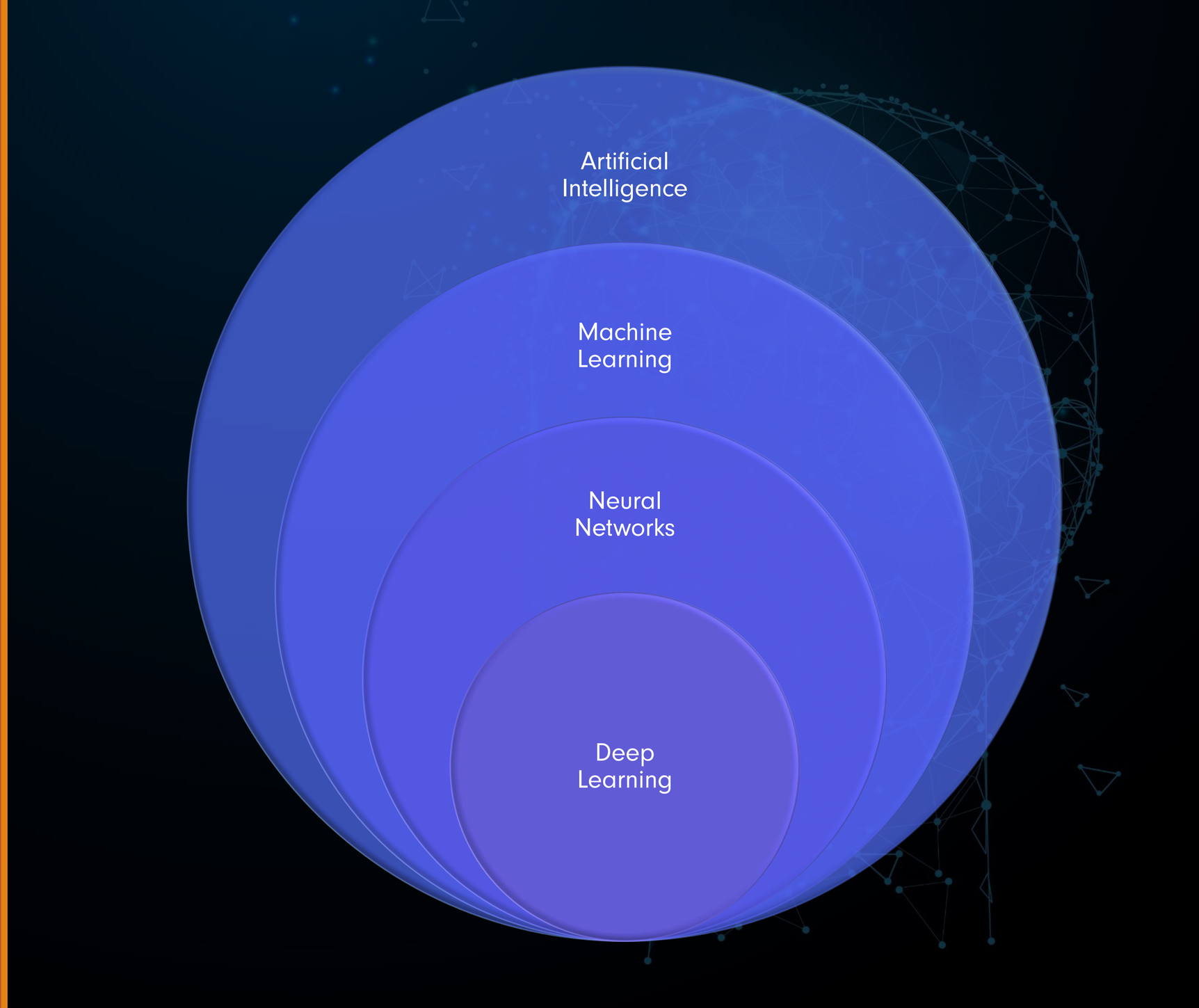
Deltatre Innovation Lab

Microsoft

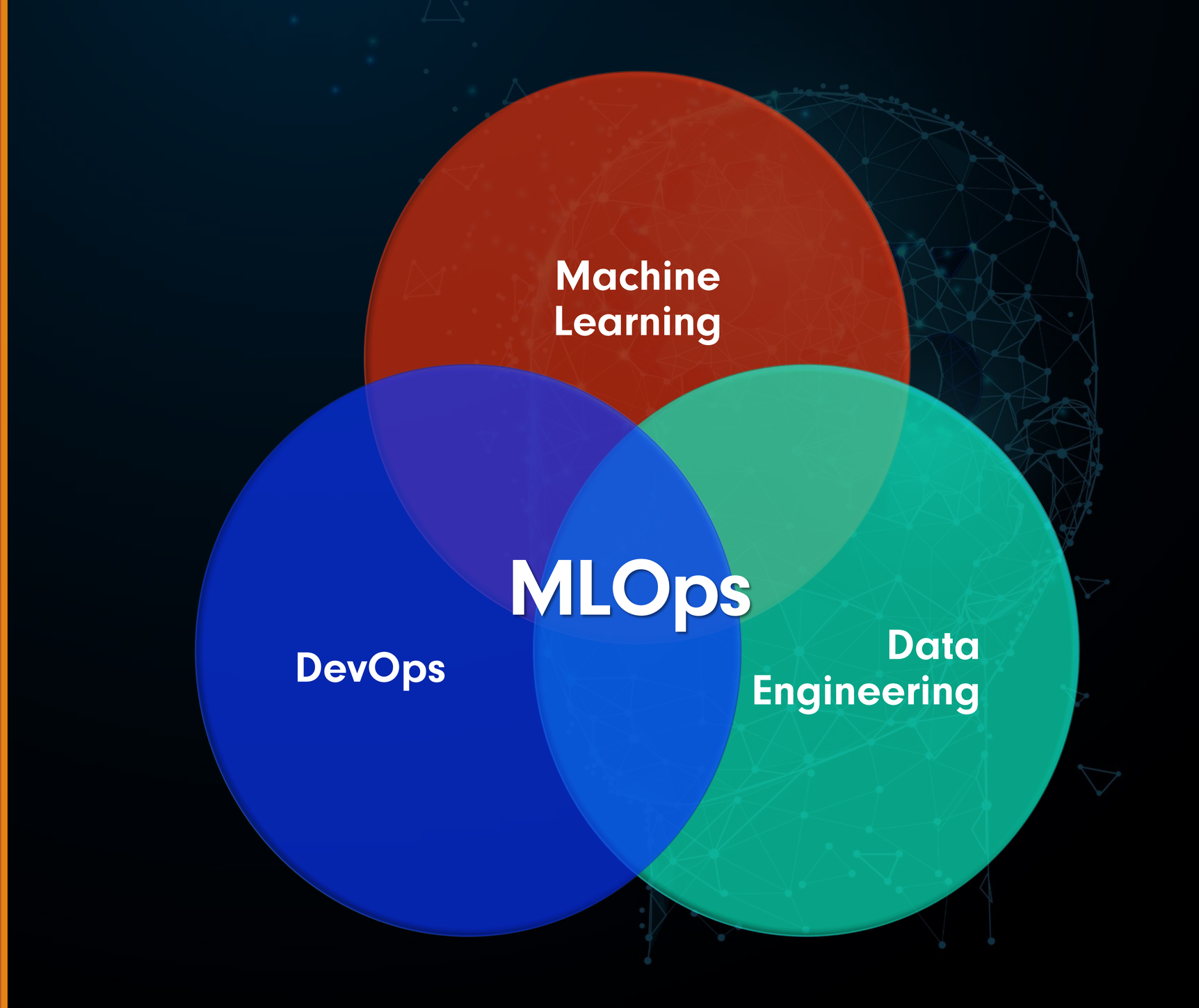
23-24-25 Marzo, 2021


AGENDA
AI & MLOps Overview
ML/DL Workflows
Machine Learning Pipelines

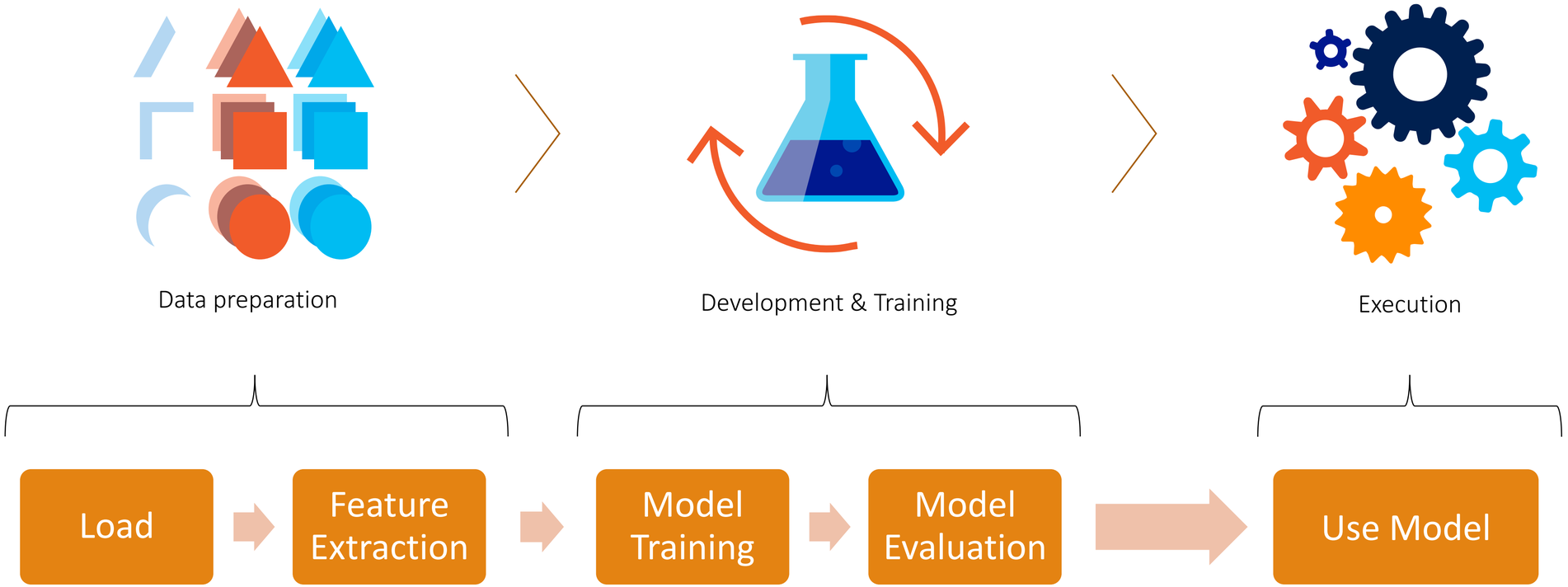
Machine Learning and MLOps overview



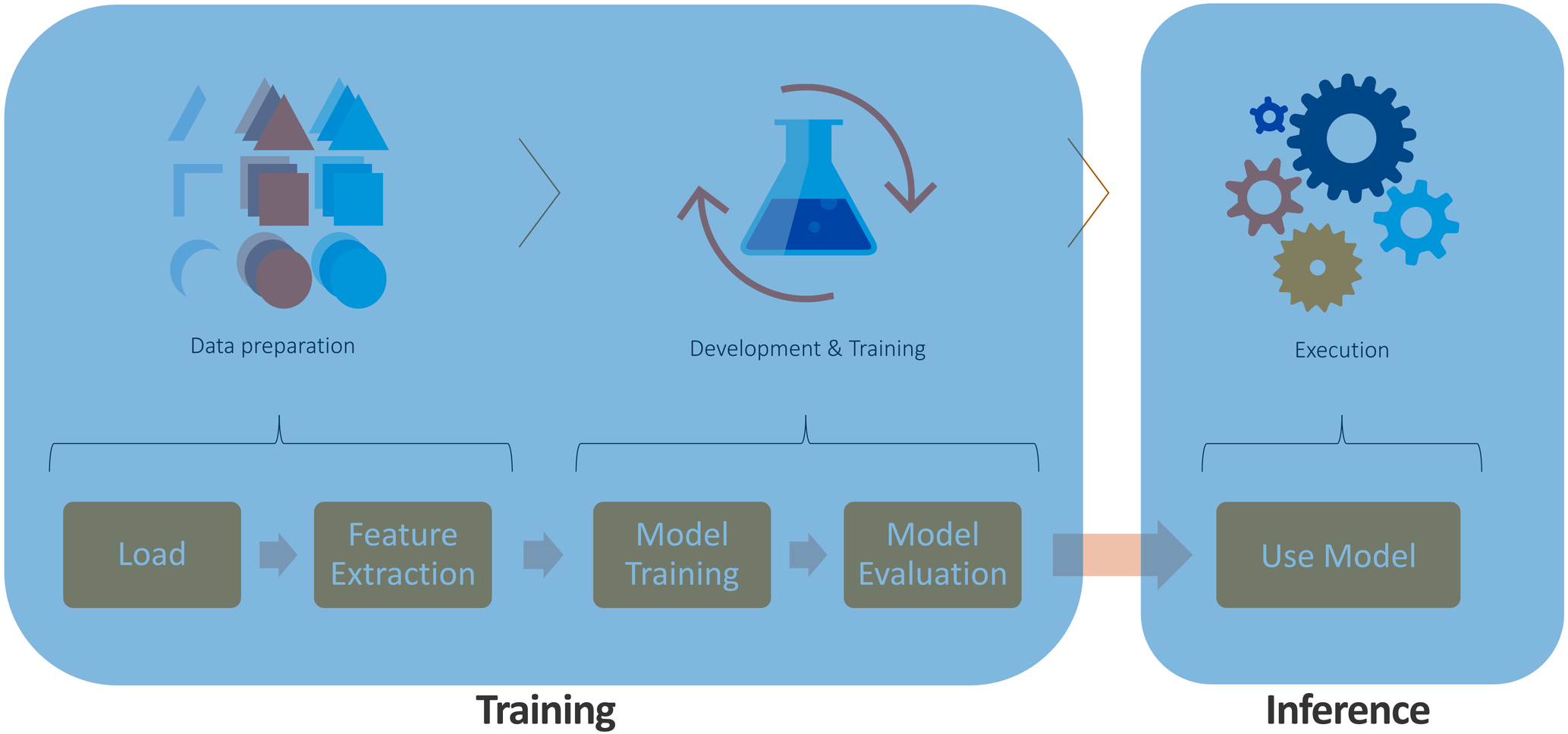
Machine Learning and MLOps overview



Typical ML Workflow



Training vs Inference/Scoring



Training

“Development” phase of a **Machine Learning project**

Usually **lead** and executed by **Data Scientists**
from start to end, **through experiments** and **trial & error**

Iterative process of **variable duration and results**
(until **specified target metrics are achieved**)

Typically, **resource & time intensive**
usually done on (lots of) CPUs and GPUs or AI-accelerators (TPU, FPGA)

Scalability ➔ reduce training time or **improve quality**

size of datasets and/or models + parallel processing = hardware/storage/bandwidth

Inference

“Production” phase of a **Machine Learning project**

Usually **lead** by **Software/AI Engineers** and **DevOps**

Data scientists’ role is mainly to **monitor** model **behavior** in the field

Different requirements**,** tools and frameworks

compared to training, more **similar** to **traditional development**

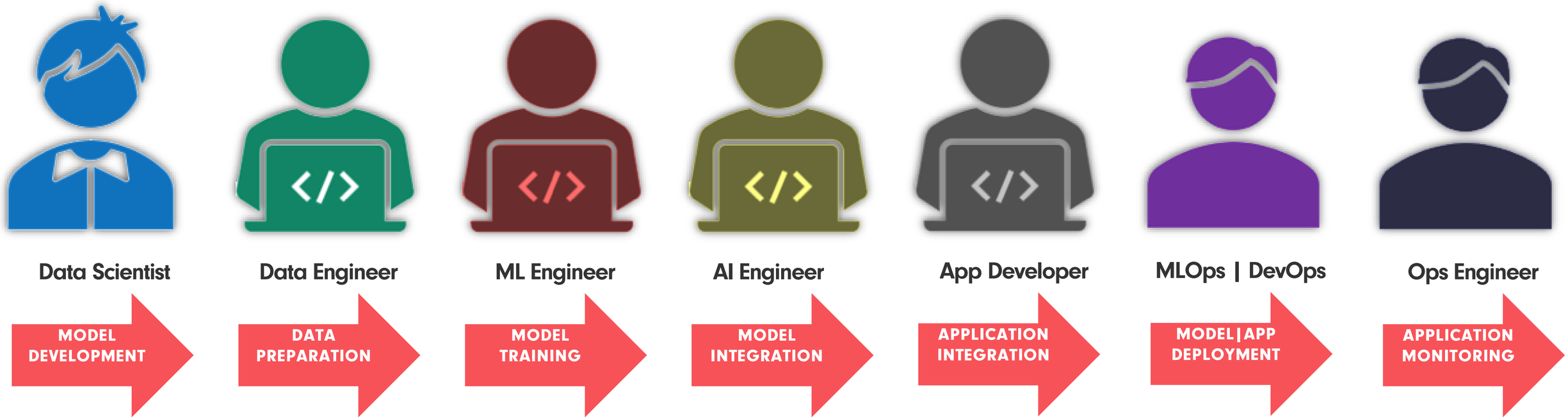
Deploy on **edge device** or **on-prem/cloud datacenter**

Scalability ➔ increase # of requests/sec

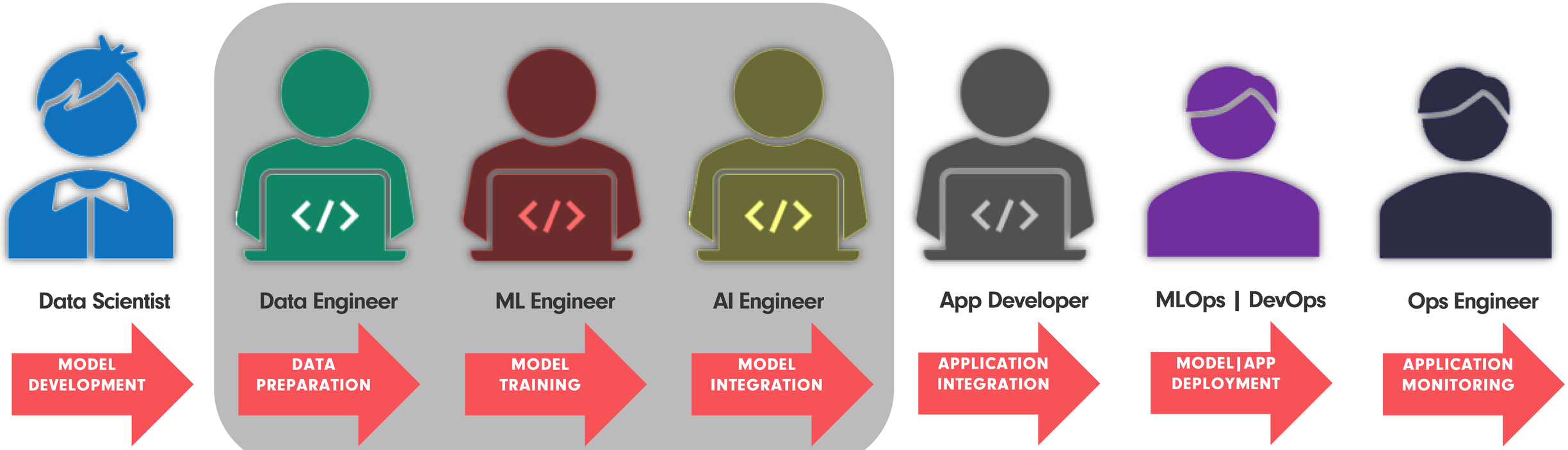
optimize models, parallel processing, increase/scale scoring instances

THE AI-TEAM

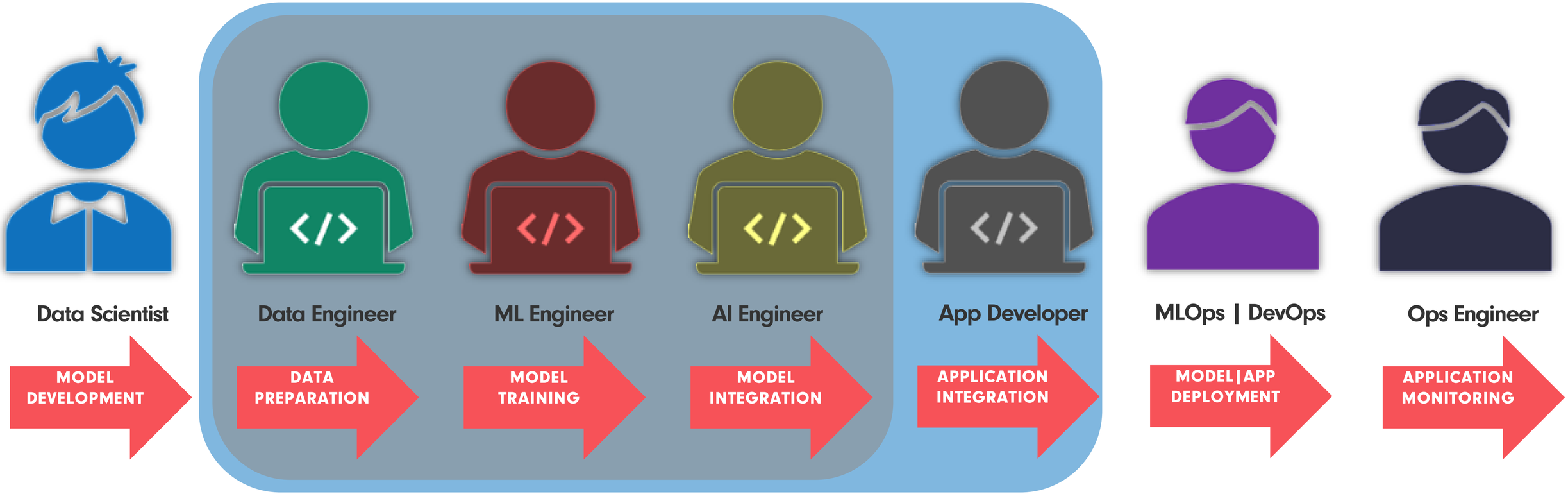
Typical ML Project Team



Typical ML Project Team



Typical ML Project Team



MLOps: DevOps on ML components

Model reproducibility & versioning

- Track, snapshot & manage assets used to create the model
- Enable collaboration and sharing of ML pipelines

Model auditability & explainability

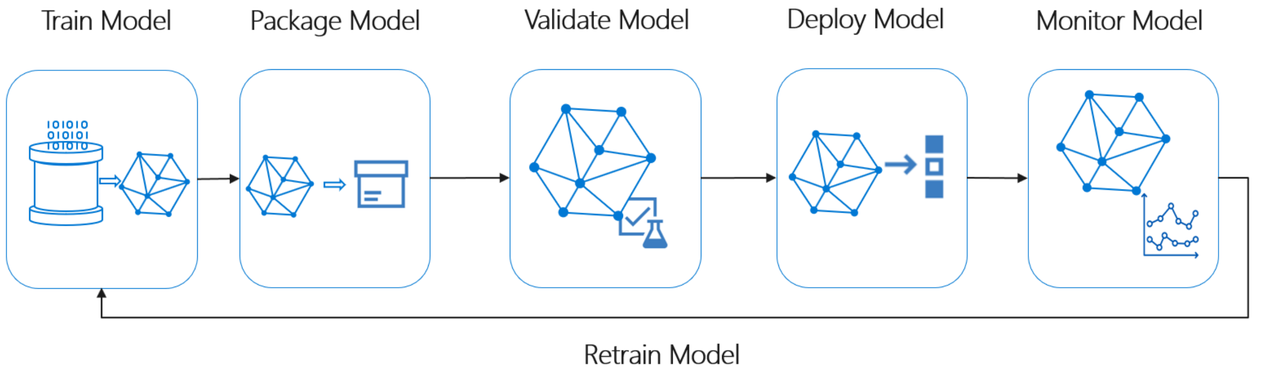
- Maintain asset integrity & persist access control logs
- Certify model behavior meets regulatory & adversarial standards

Model packaging & validation

- Support model portability across a variety of platforms
- Certify model performance meets functional and latency requirements

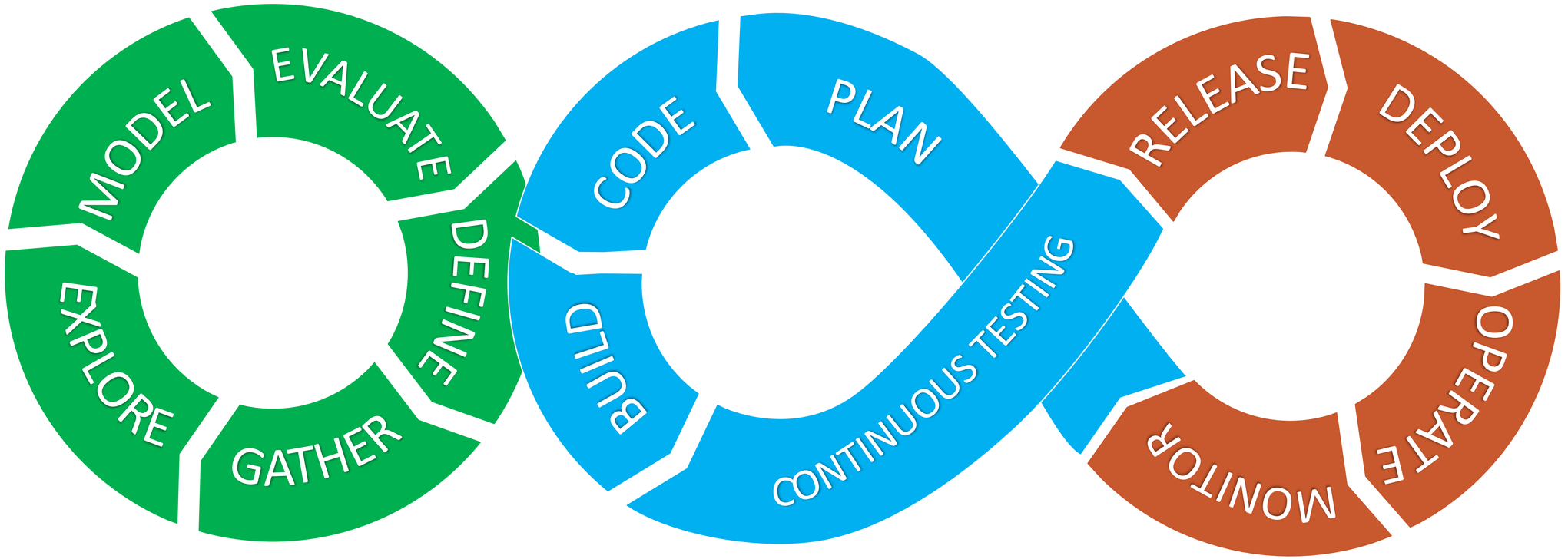
Model deployment & monitoring

- Release models with confidence
- Monitor & know when to retrain by analyzing signals such as data drift




HELPFUL
TIPS!

The forgotten exploration phase



Dataset management

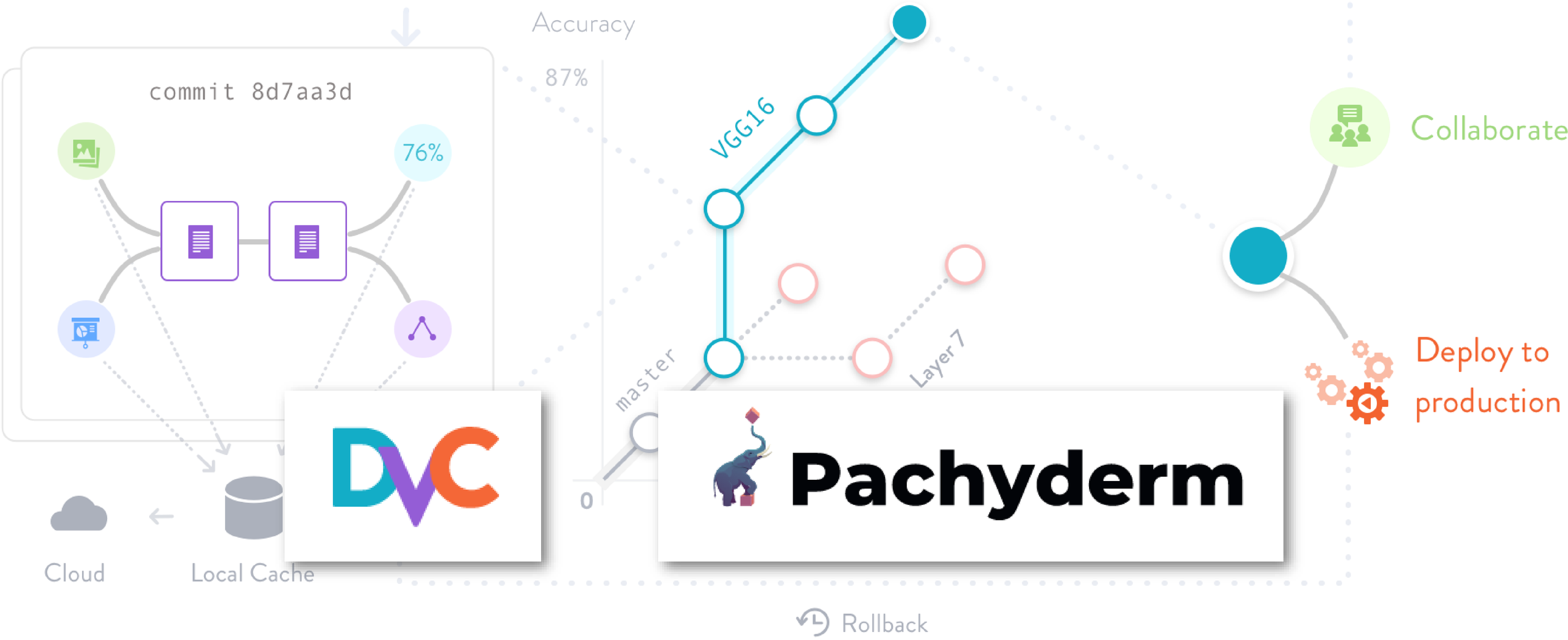


Image taken from DVC.org

Train, Validation and Test Split

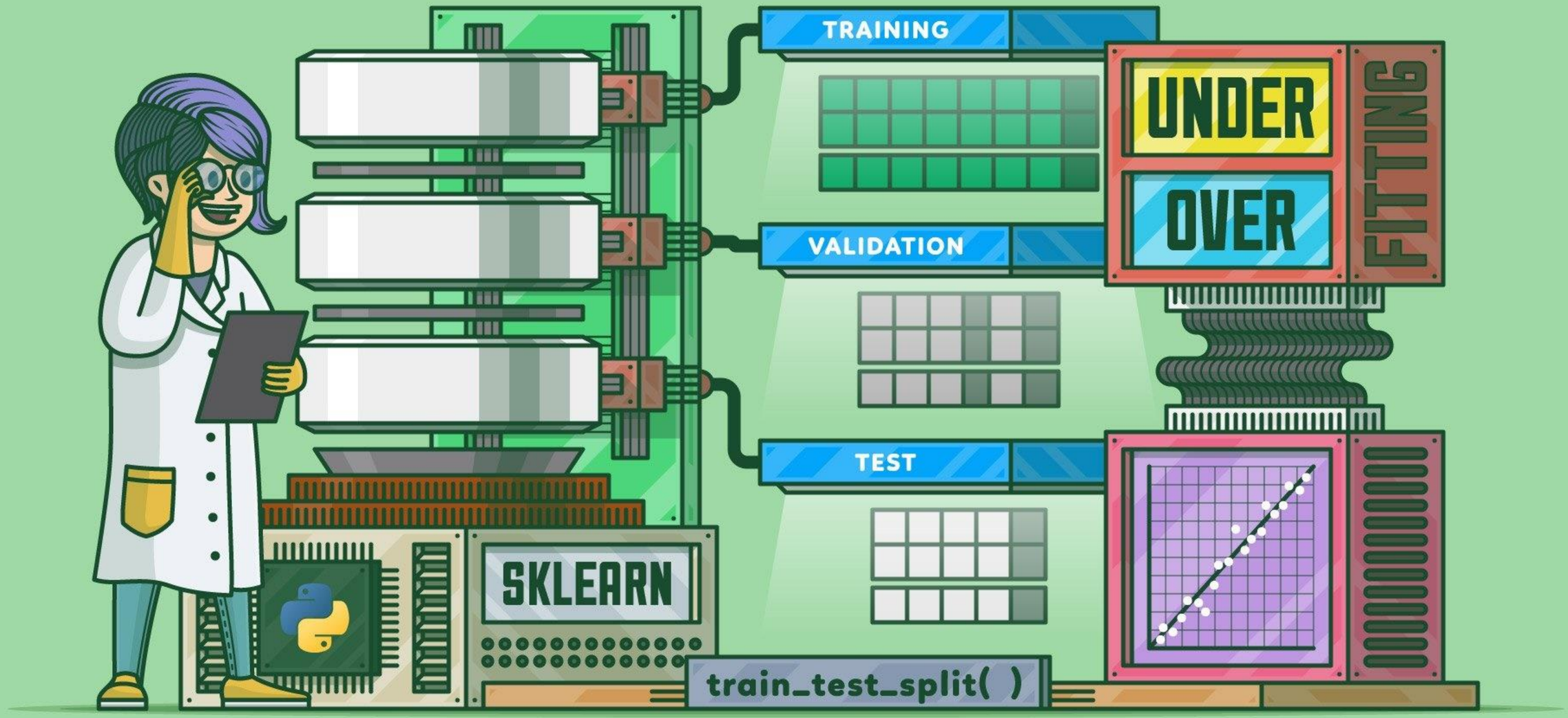
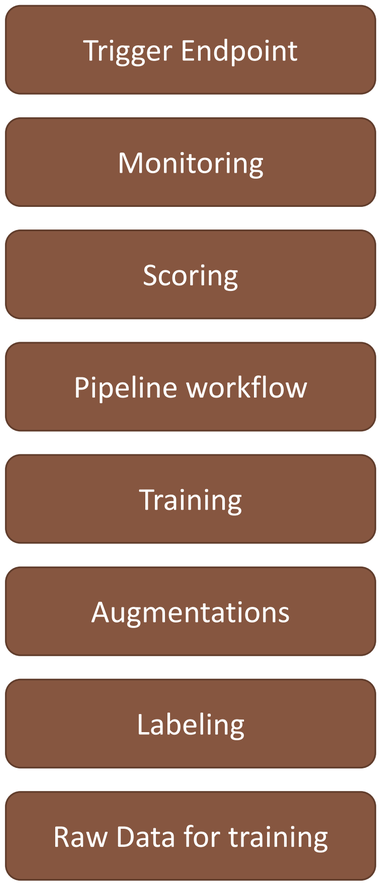


Image taken from Real Python

ML/DL Workflows

The building blocks of “usable” Machine Learning



First of all, you need **DATA** (possibly good data)

Most ML problems can’t be solved without **Labeling**

Augmentations or processing make data usable across different sources for our chosen algorithm

Most ML problems can’t be solved with just one ML model. **Pipelines** and **Workflows** are key to chain **transformations** and **model evaluations and results**

In order to use and maintain a ML pipeline proper **MLOps** and results **Monitoring** must be in place


AKS/Kubeflow with custom models
Azure Machine Learning with custom models
Customizable Cognitive services: Custom Vision, Custom Speech, Luis
Cognitive services: Vision, Speech, Language
Microsoft provides/manages
Engineers/DS provide/manage
Trigger Endpoint
Monitoring
Scoring
Pipeline workflow
Training
Augmentations
Labeling
Raw Data for training

Cognitive Services vs AML

Azure cognitive services

Cognitive Services are for developers without machine-learning experience

- A Cognitive Service provides a (general-purpose) trained model, made available using a SaaS REST API or an SDK.
- Services can be used and integrated within minutes, depending on the scenario.
- Cognitive Service readiness is ideal for who has **no AI knowledge and deals with general problems**.

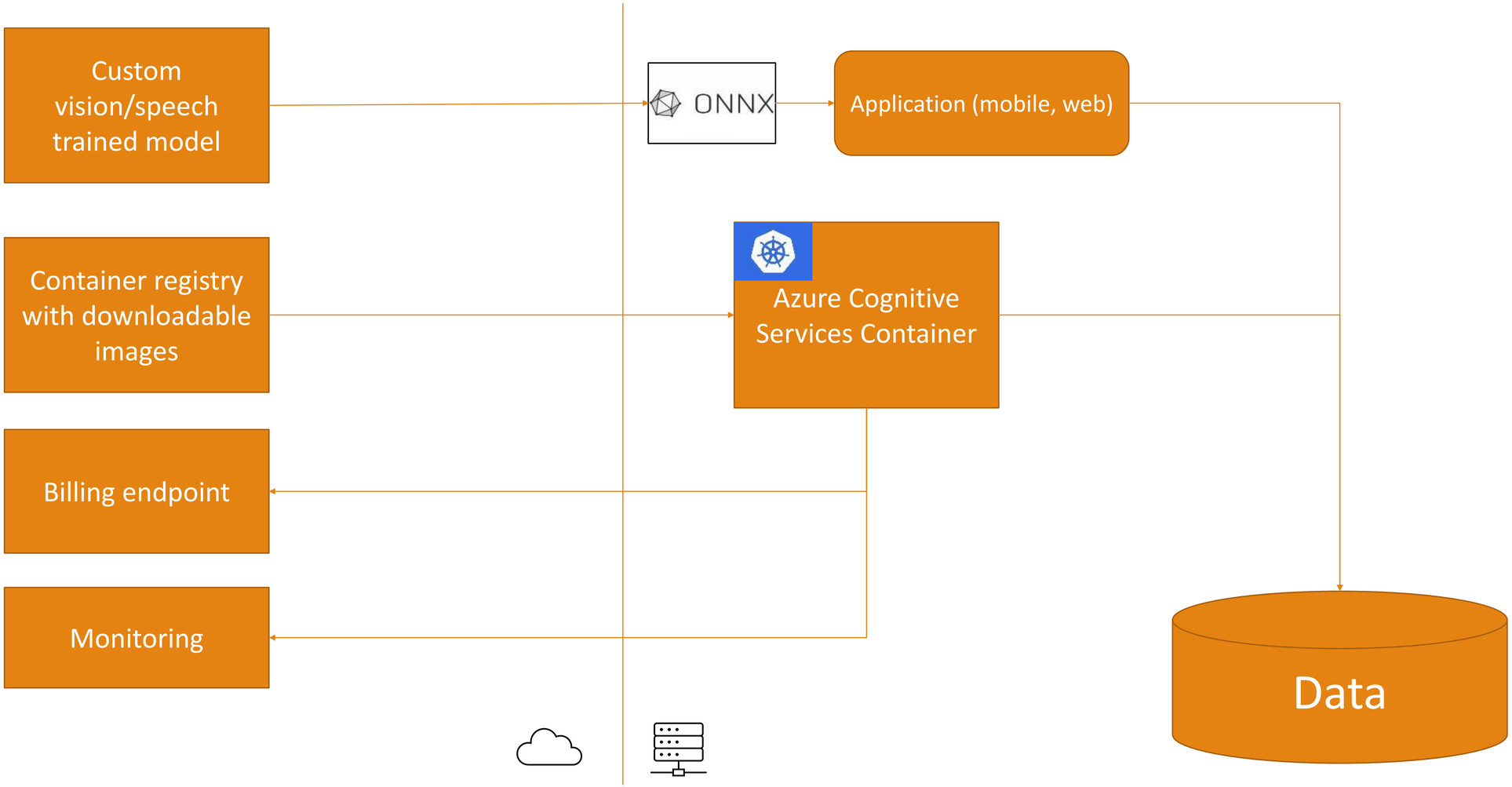
Azure machine learning

Azure Machine Learning is tailored for data scientists

- AML works for highly specialized or specific problems, often requires suiting all ML operations (data collections, cleaning, training, evaluating, …).
- Models implementation may require weeks, if not months, and engineering, maintaining and serving them requires infrastructure + software engineering + data science skills.

Familiarity and expertise with data science are required

Does it all need to be in the public Cloud?



DIY
Machine Learning

FRAMEWORKS/LIBRARIES

a collection of GPU-accelerated Data Science Libraries

JUPYTER

provides an interactive interface to cluster resources

KUBEFLOW

interfaces with Kubernetes simplifying the administration of Kubernetes services

KUBERNETES

acts as the cluster's operating system, keeping track of hardware resources and scheduling as needed

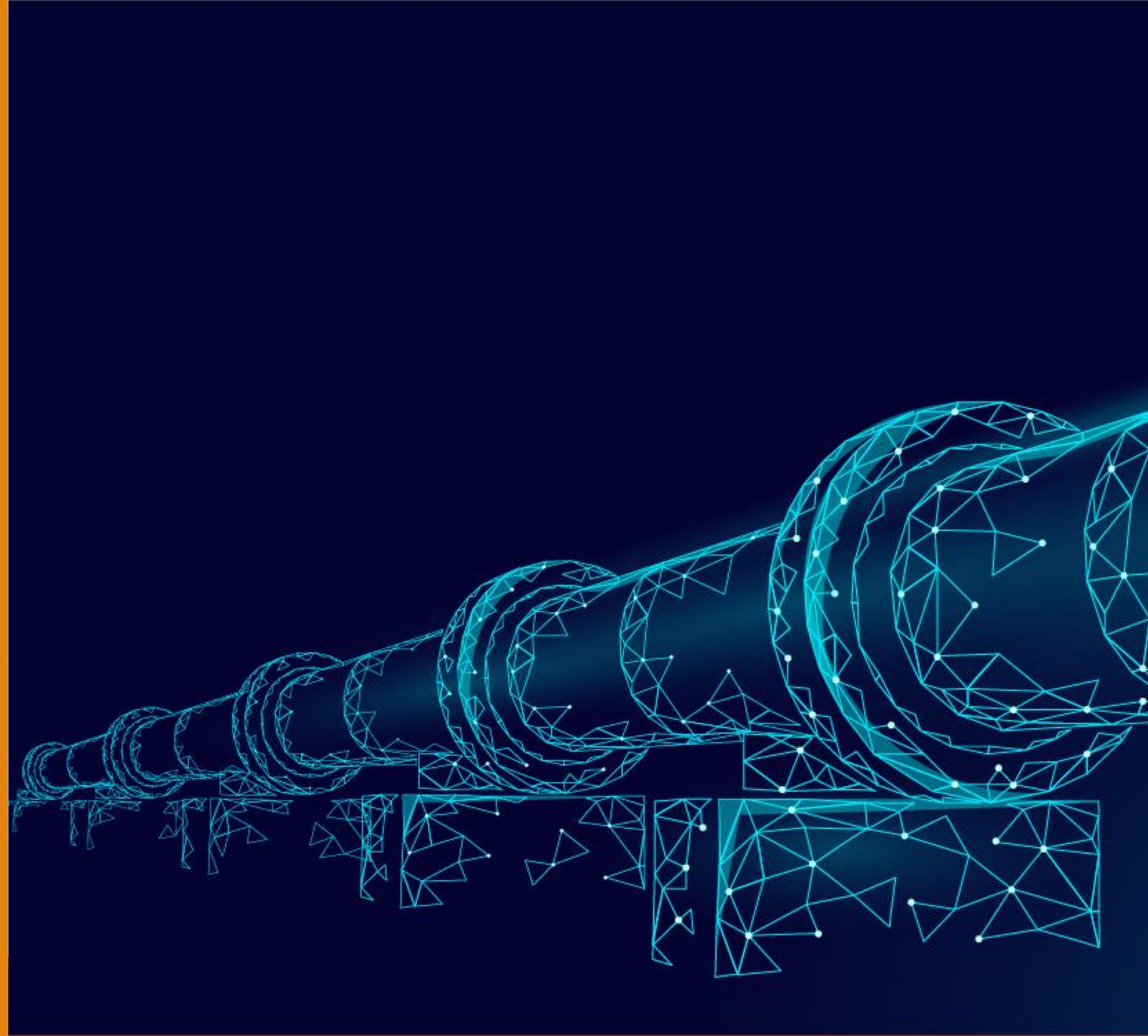
LINUX

tested and validated Linux operating system with NVIDIA drivers and CUDA libraries for optimal performance

GPUs

available to users from the cluster

Machine Learning Pipelines



What are AML pipelines?

AML Pipeline is an independently executable workflow of a **complete ML task**. Subtasks are encapsulated as a **series of steps** within the pipeline.

Step: discrete processing action designed for a specific task with its own resources and environment

AML **automatically orchestrates** all the dependencies between pipeline steps. This may include:

- Spinning up and down Docker images
- Attaching and detaching compute resources
- Moving data between steps consistently and automatically

Pipeline benefits

Simplicity

Speed

Repeatability

Flexibility

Modularity

Versioning and tracking

Quality assurance

Cost control

Controlled Runtime Environment

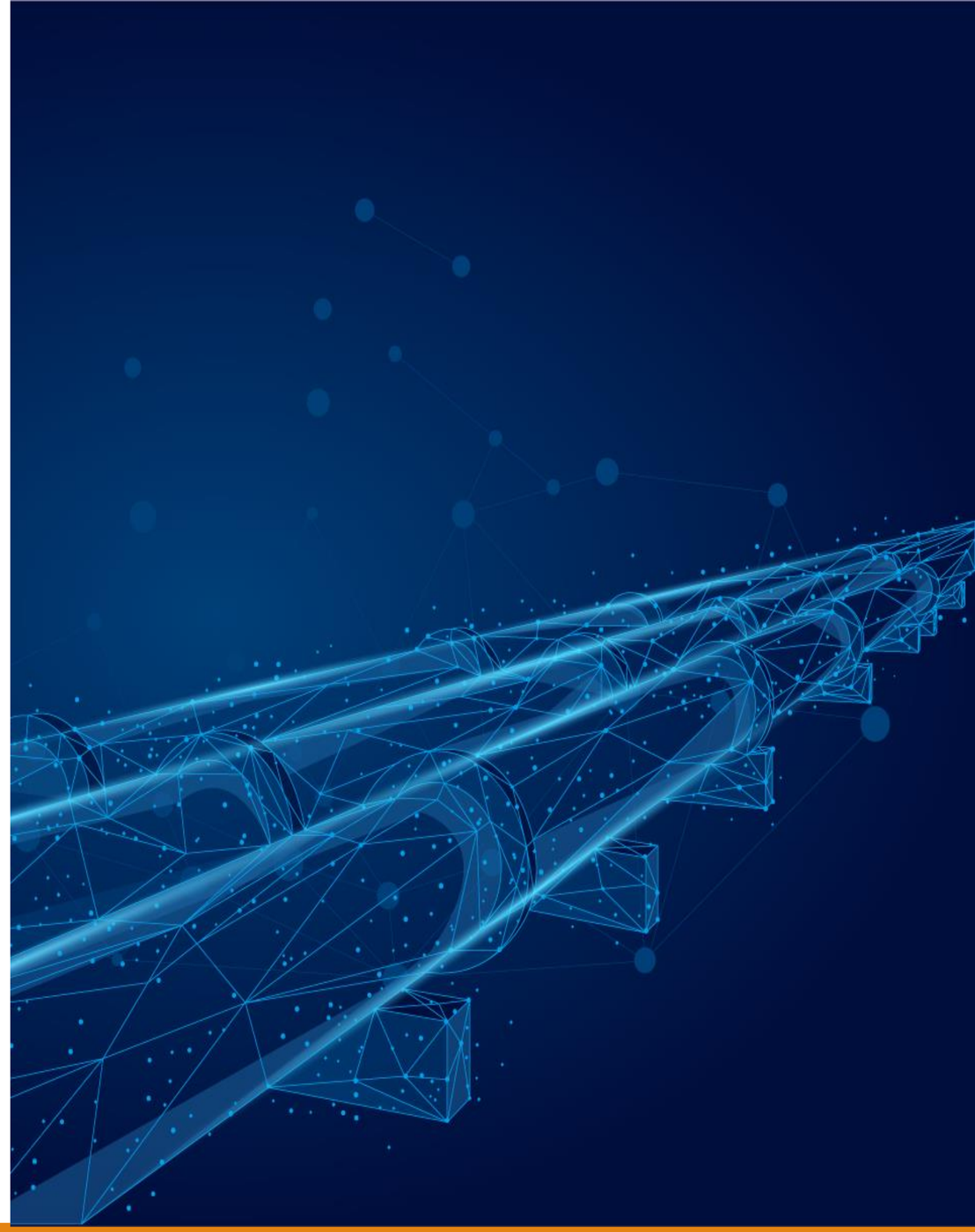
What type of pipelines?

DevOps CI/CD Pipelines
Related to implement CI/CD processes. They check:

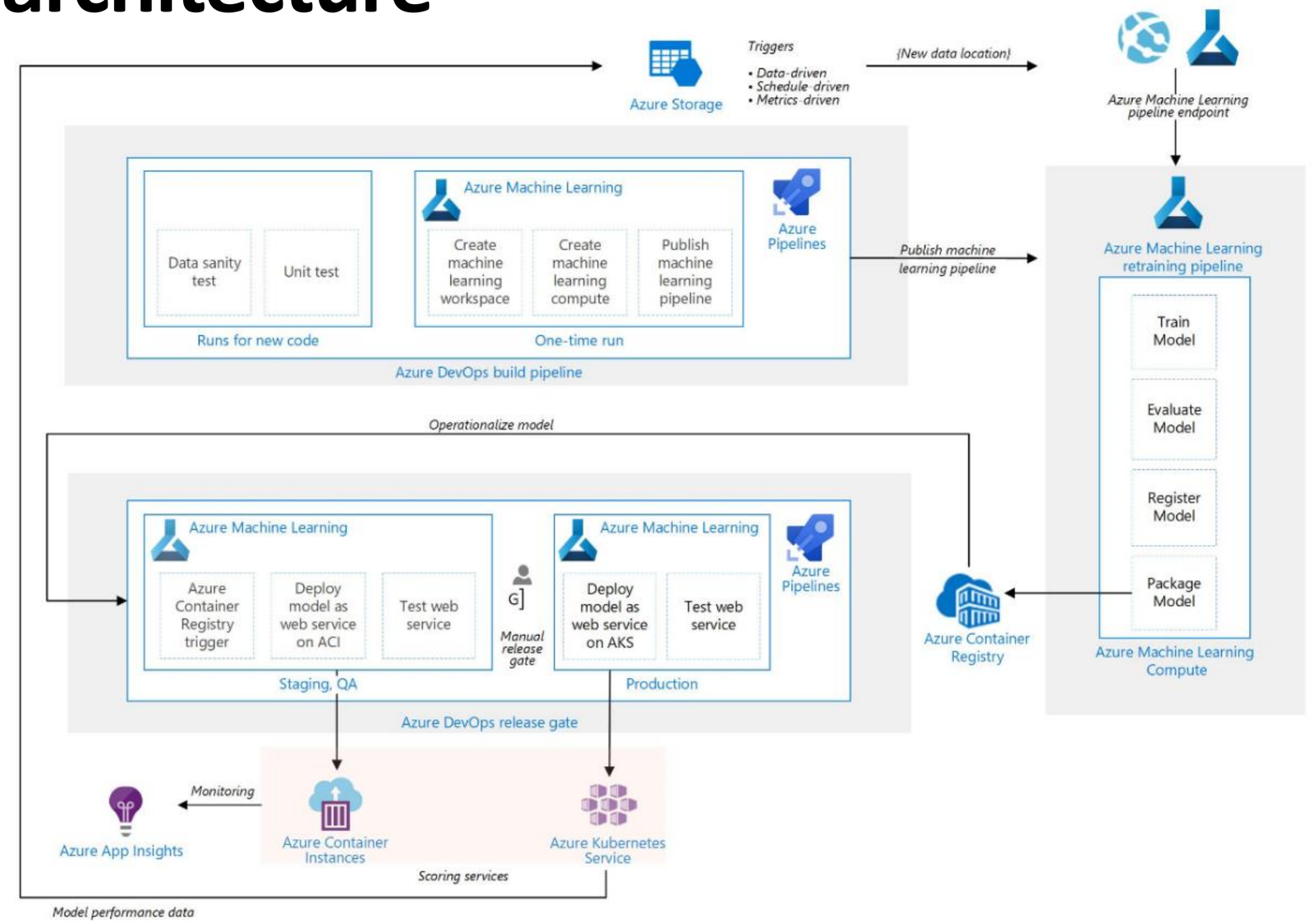
- Code quality
- Makes sure that any changes in the repository are deployed to the AML Service once they are committed/tested/approved, etc.

Azure Machine Learning Pipelines
Related to entities in the AML service such as

- Training models
- Scoring models
- Data augmentation pipelines



DevOps architecture



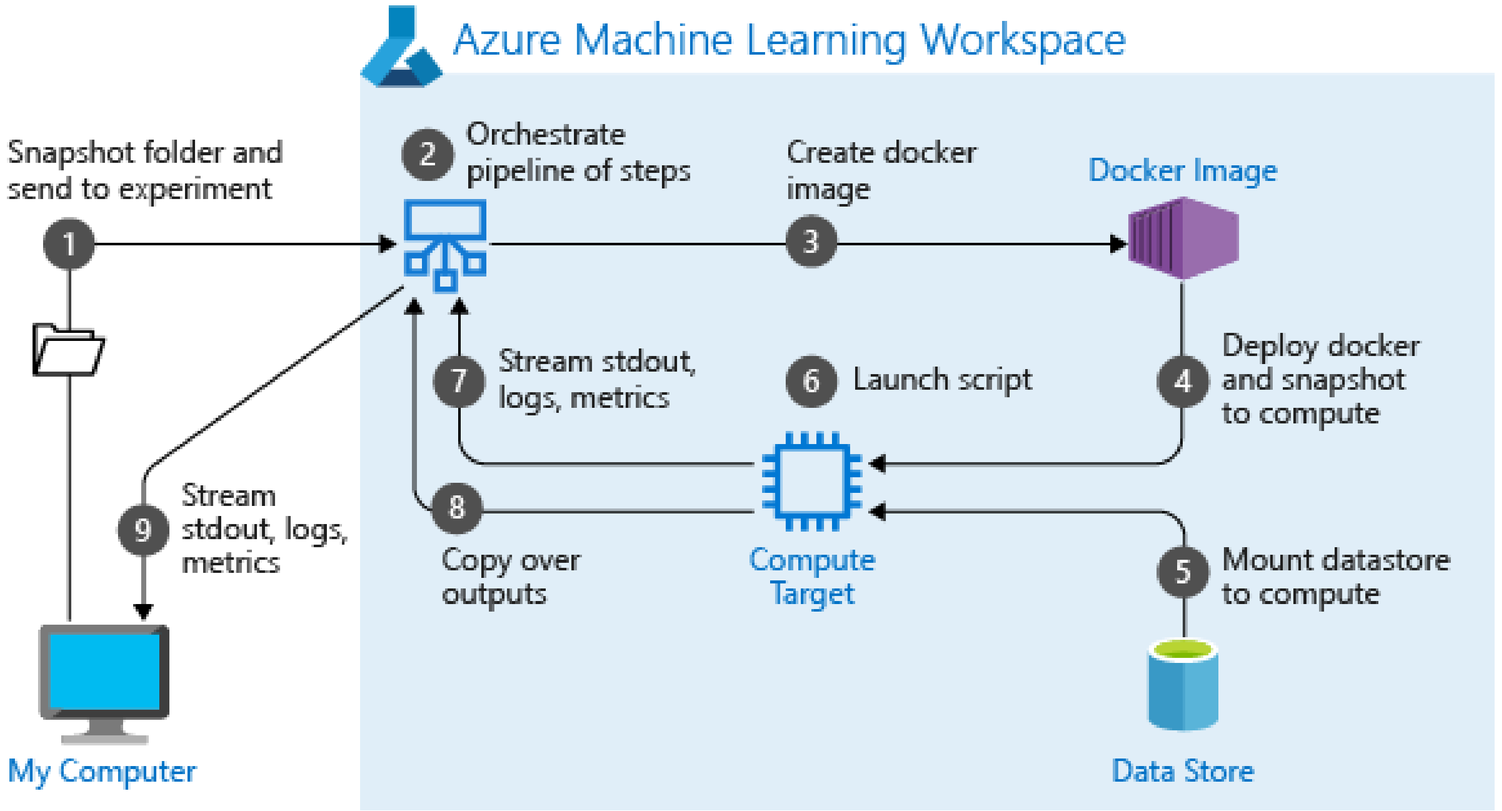
https://docs.microsoft.com/azure/architecture/reference-architectures/ai/mlops-python

AML Pipeline phases

Build

Publish

Run



Designer - Microsoft Azure Mach

https://ml.azure.com/visualinterface?

Microsoft Azure Machine Learning

Designer

New
Home
Author
Notebooks
Automated ML
Designer
Assets
Datasets
Experiments
Pipelines
Models
Endpoints
Manage
Compute
Datastores
Data Labeling
Linked Services

Designer

New pipeline

Show more samples

Easy-to-use prebuilt modules

Image Classification using DenseNet

Binary Classification using Vowpal Wabbit Model - Adu...

Wide & Deep based Recommendation - Restaur...

Regression - Automobile Price Prediction (Basic)

Pipelines

Pipeline drafts | Pipeline runs

No pipeline drafts found

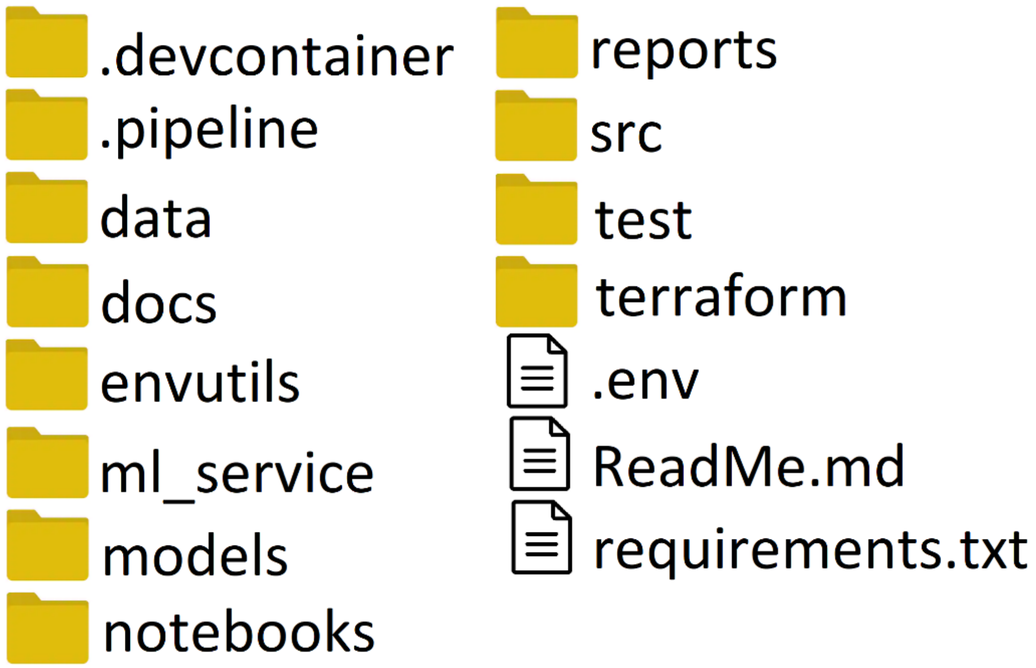
Create a new pipeline or start from a sample

Findings and best-practices

Project structure

Early stages of an ML project: Jupyter notebooks used for **explorations**.

Developing the project: complexity increase. CI/CD, single py scripts for each step and phase of the workflow.



Project strengths

Cookie cutter: a cross-platform logical, simple, but flexible project structure to carry out and share work

Documentation: strong, detailed and meaningful documentation for all the project phases

Python: cross-platform machine learning framework

Testing:

Unit and Integration tests are done using PyTest for Python.

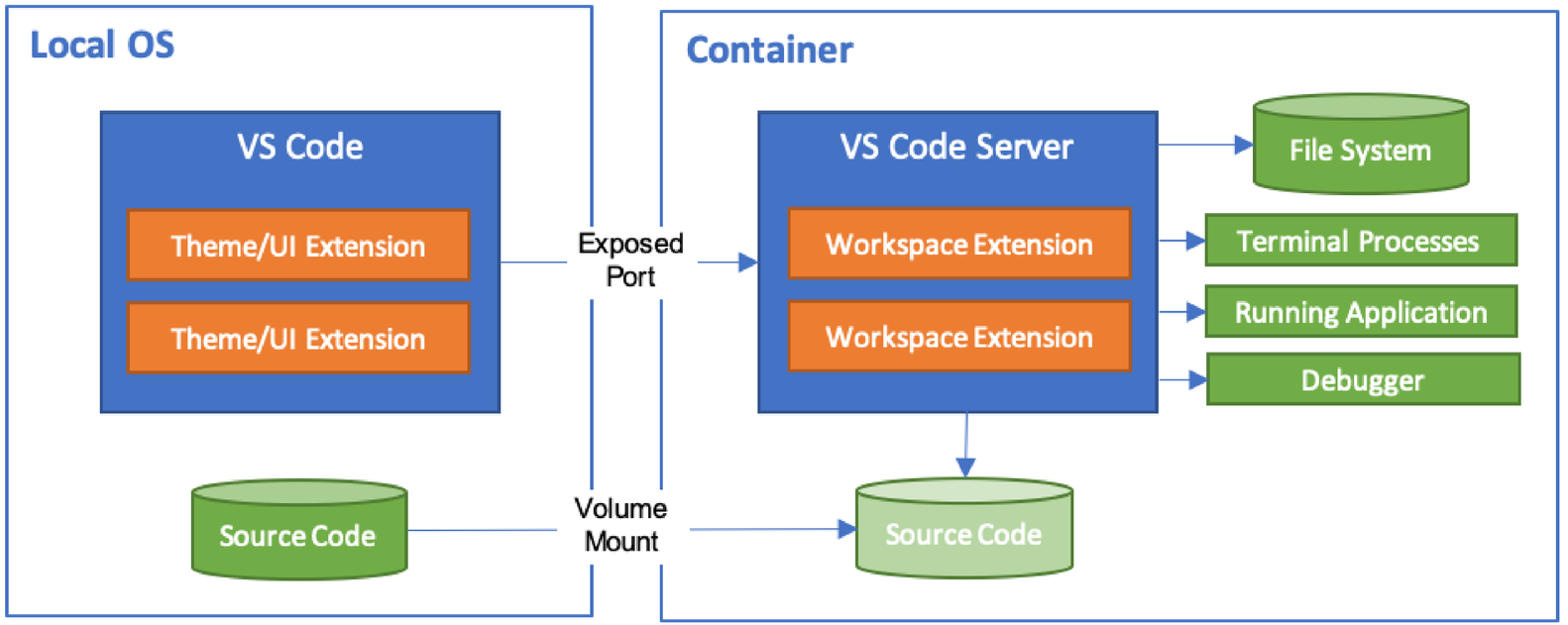
When possible, acceptance criteria should be verified with acceptance tests

Dev Environment in Containers

Remote Development extension pack for **Visual Studio Code**

→ Use a **Docker container** as a **full-featured development environment**

→ Use the **same Docker image** for all dev team (local machine, remote machine or cloud VMs)



https://code.visualstudio.com/docs/remote/containers
https://github.com/polatengin/project-standards/blob/master/DevContainers.md
https://channel9.msdn.com/Series/Beginners-Series-to-Dev-Containers

Real World:
In Production

Monitoring,
Infrastructure Security,
Data Security,
Infrastructure as Code,
ML Testing,
Deployment testing,
App/Service Integration

Thank You!

ευχαριστώ Salamat Po متشكرم شكراً Grazie

благодаря ありがとうございます Kiitos Teşekkürler 谢谢

ขอบคุณครับ Obrigado شكريه Terima Kasih Dziękuję

Hvala Köszönöm Tak Dank u wel дякую Tack

Mulţumesc спасибо Danke Cám ơn Gracias

多謝晒 Ďakujem תודה நன்றி Děkuji 감사합니다

QnA

About us

Clemente Giorio

R&D Senior Software Engineer @ **Deltatre**

@tinux80

- Augmented/Mixed/Virtual Reality
- Artificial Intelligence, Machine Learning, Deep Learning
- Internet of Things
- Embedded Apps
- Multimodal Tracking

dotNET{podcast}

MVP Microsoft Most Valuable Professional

intel Software INNOVATOR

[PACKT] PUBLISHING Author

FAB LAB NAPOLI

About us

Vito Flavio Lorusso

Program Manager @ **Microsoft**

@vflorusso

- Former web developer
- Former data engineer and developer
- Former “doing Database cluster installations in datacenters”
- Former Solutions Architect
- Former “cloud evangelist”
- Former Distributed systems engineer
- Constantly looking for my place in the digital world to help work get done

About us

Ing. Gianni ROSA GALLINA

R&D Senior Software Engineer @ **Deltatre**

@giannirg

- AI, Machine Learning, Deep Learning on multimedia content
- Virtual/Augmented/Mixed Reality
- Immersive video streaming & 3D graphics for sport events
- Cloud solutions, web backends, serverless, video workflows
- Mobile apps dev (Windows / Android / Xamarin)
- End-to-end solutions with Microsoft Azure

MVP Microsoft Most Valuable Professional

Microsoft Specialist
Programming in C#
Programming in HTML5 with JavaScript & CSS3

Microsoft CERTIFIED Solutions Developer
Windows Store Apps Using C#
Web Applications

Microsoft CERTIFIED AZURE AI ENGINEER ASSOCIATE

Microsoft CERTIFIED AZURE DATA SCIENTIST ASSOCIATE

unity Certified Programmer

PLURALSIGHT Author

https://gianni.rosagallina.com/en/

References

References (1/2)

Tools and IDE

https://www.python.org/
https://jupyter.org/
https://code.visualstudio.com/

ML/Deep Learning Services & Frameworks

https://azure.microsoft.com/services/machine-learning/
https://azure.microsoft.com/services/cognitive-services/
https://www.tensorflow.org/
https://keras.io/
https://pytorch.org/
https://fast.ai/

References (2/2)

MLOps

https://mlflow.org/
https://www.kubeflow.org
https://dvc.org/
https://www.pachyderm.com/
https://github.com/microsoft/MLOps
https://docs.microsoft.com/azure/machine-learning/team-data-science-process
https://docs.microsoft.com/azure/architecture/reference-architectures/ai/mlops-python
https://docs.microsoft.com/azure/architecture/reference-architectures/#ai-and-machine-learning

VSCode DevContainers

https://github.com/polatengin/project-standards/blob/master/DevContainers.md
https://code.visualstudio.com/docs/remote/containers
https://channel9.msdn.com/Series/Beginners-Series-to-Dev-Containers

QnA