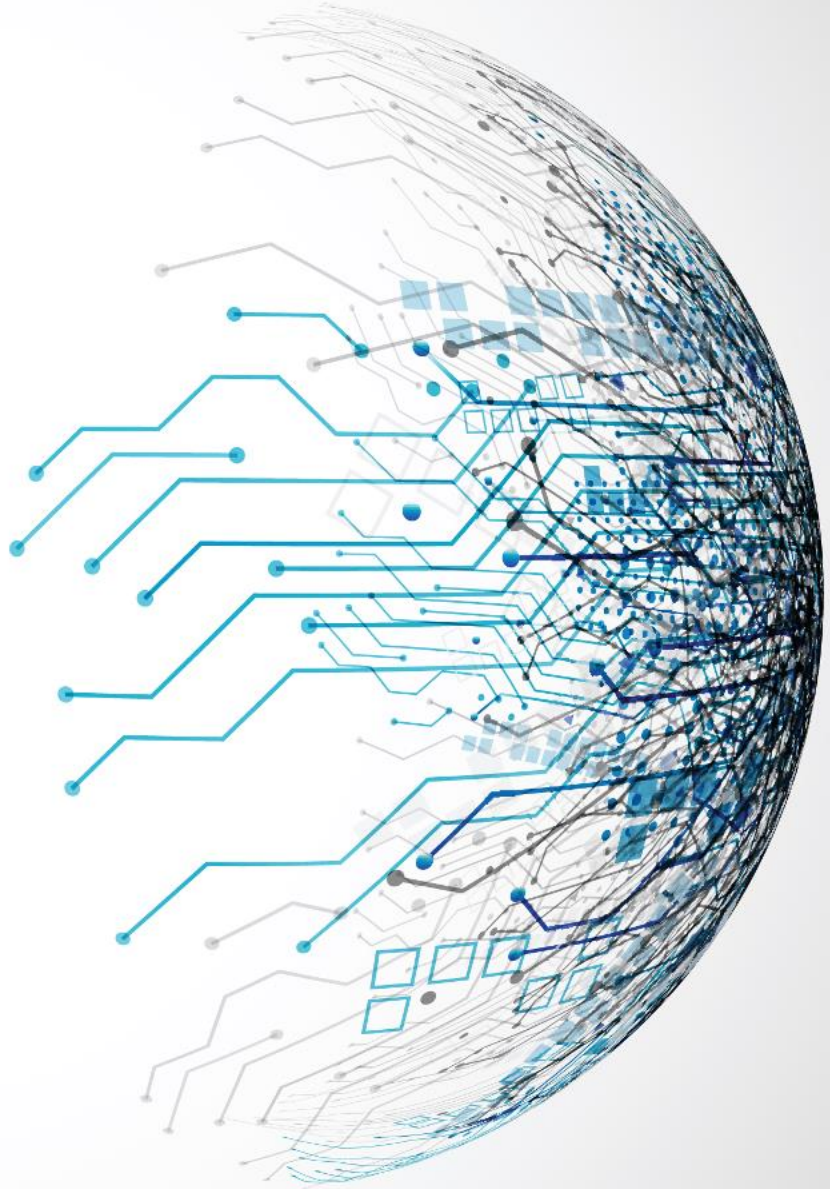
Deep Learning

INTRODUCTION AND SOFTWARE STACK



Dr. Mohammed Salah Al-Radhi

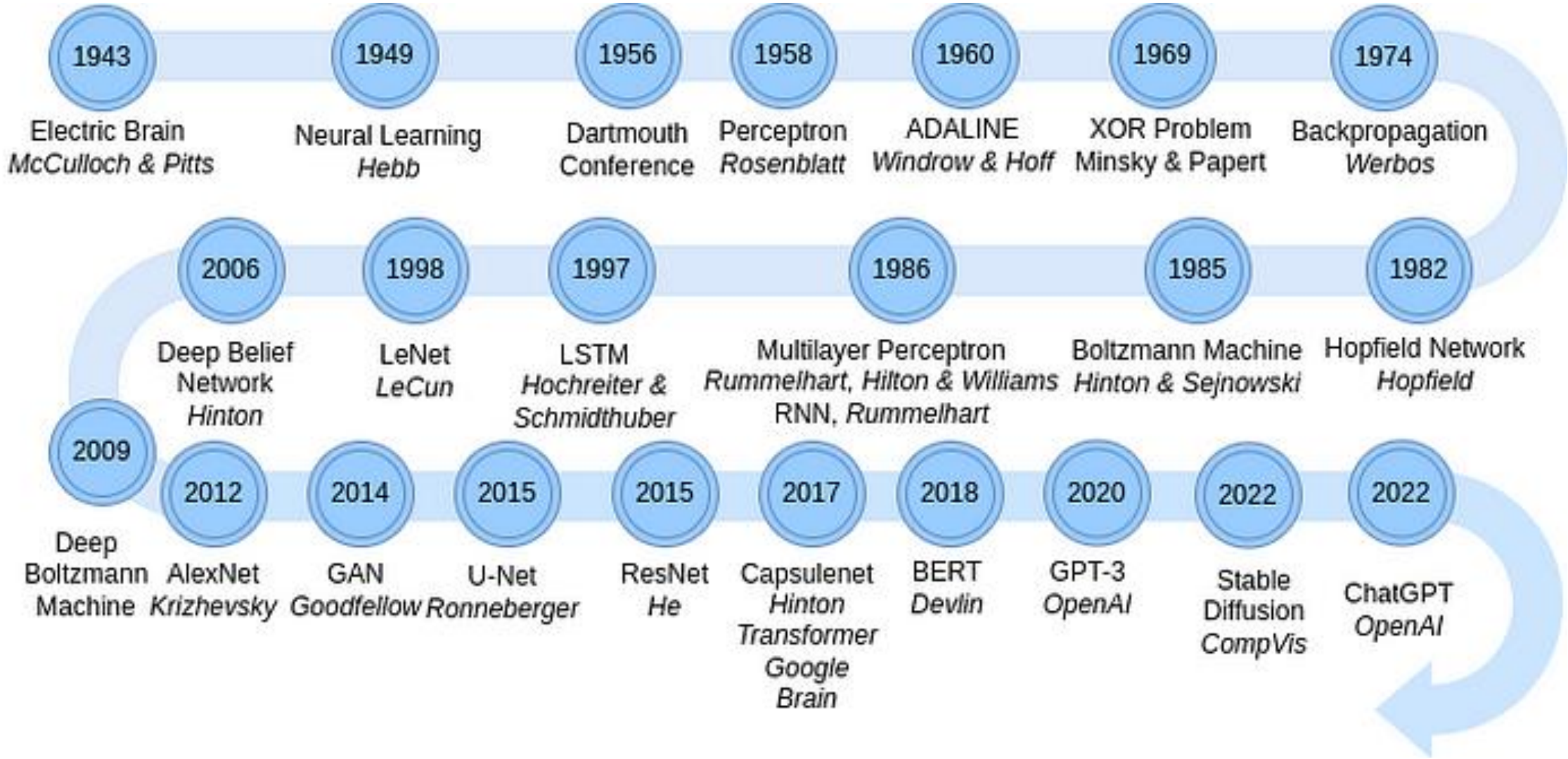
(slides by: Dr. Bálint Gyires-Tóth)

Copyright

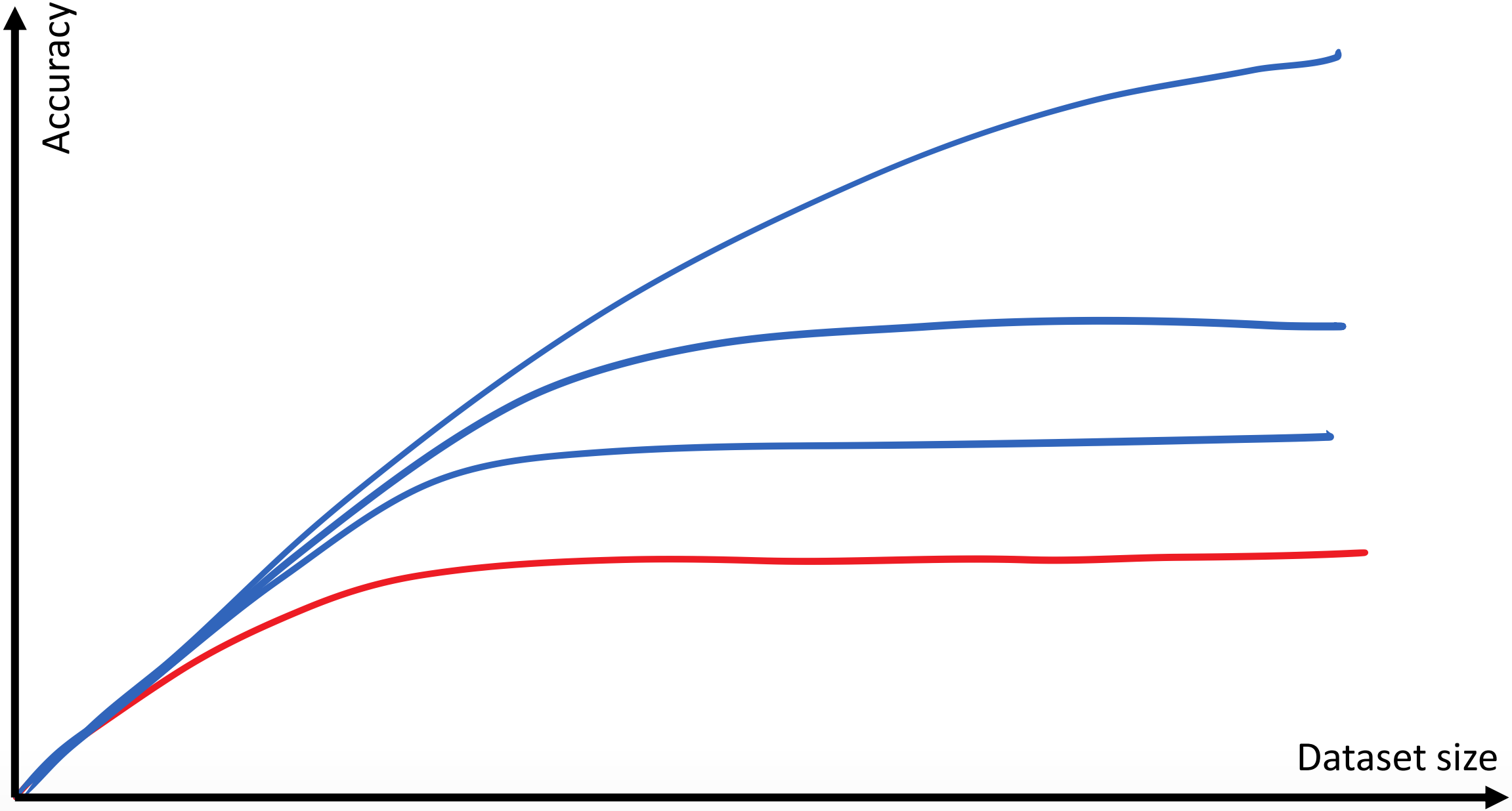
Outline

1. Brief history of deep learning
2. CRISP-DM for deep learning
3. Deep learning roles
4. Basic software components
5. Advanced software components

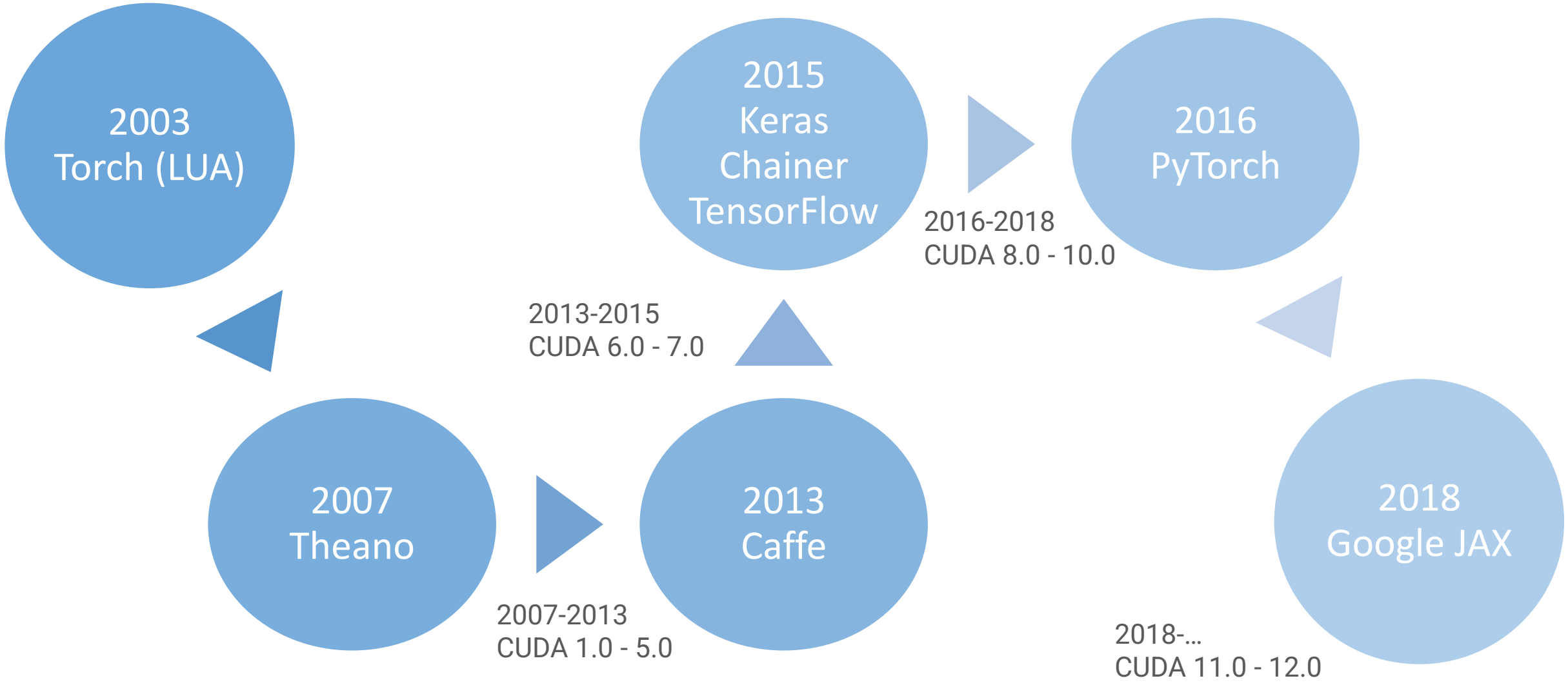
Brief history of deep learning



Source: https://pub.towardsai.net/a-brief-history-of-neural-nets-472107bc2c9c


Accuracy
Dataset size

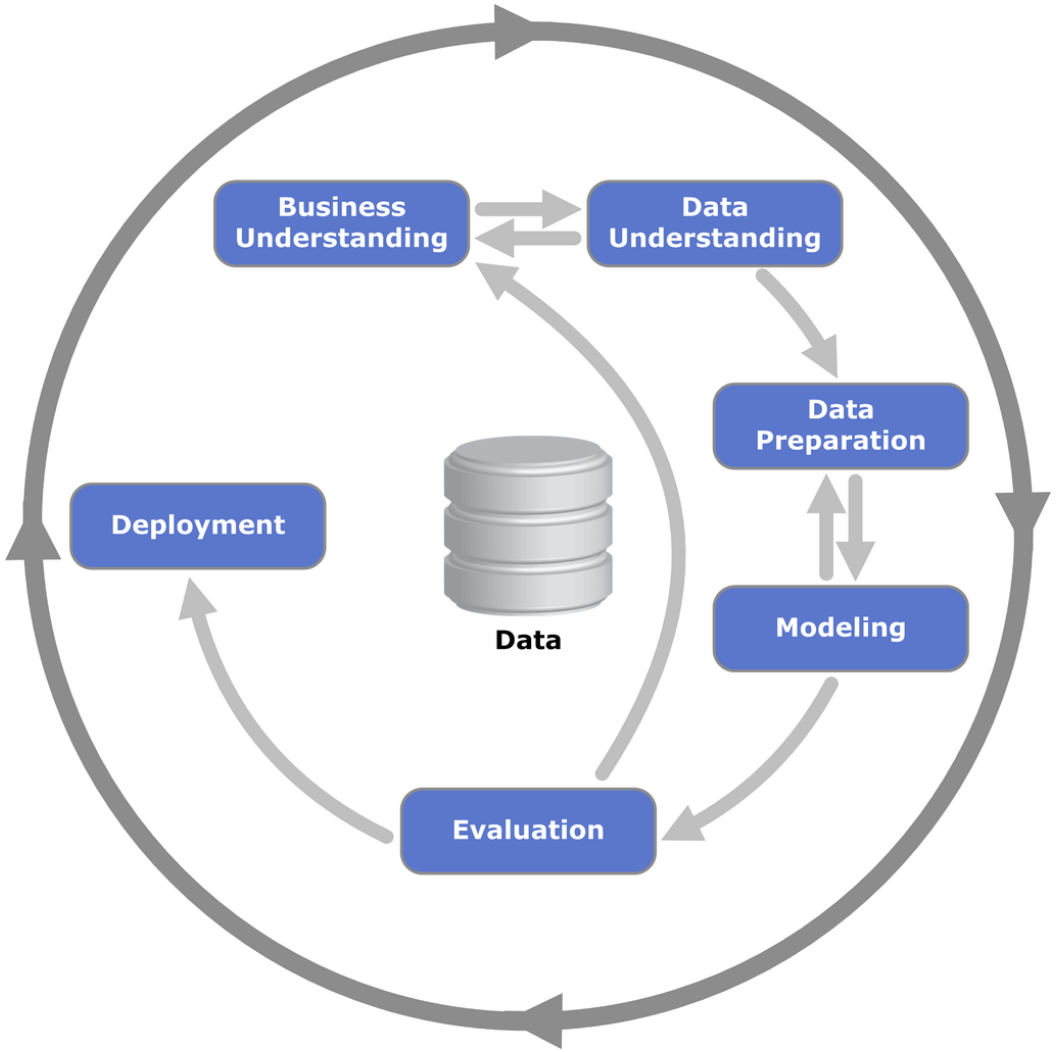
Brief history of deep learning framework



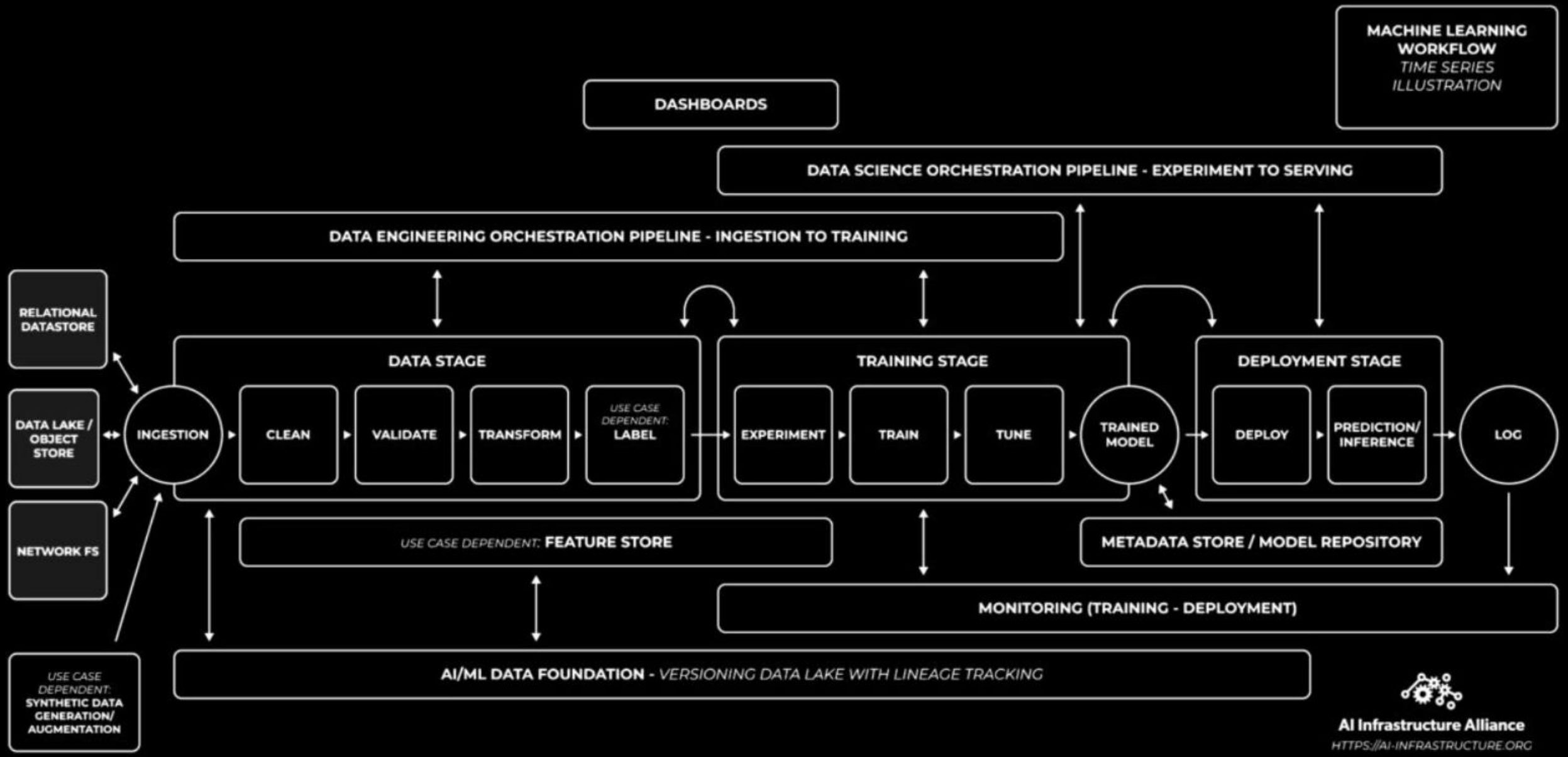
CRISP-DM for deep learning

Cross Industry Standard Process for Data Mining

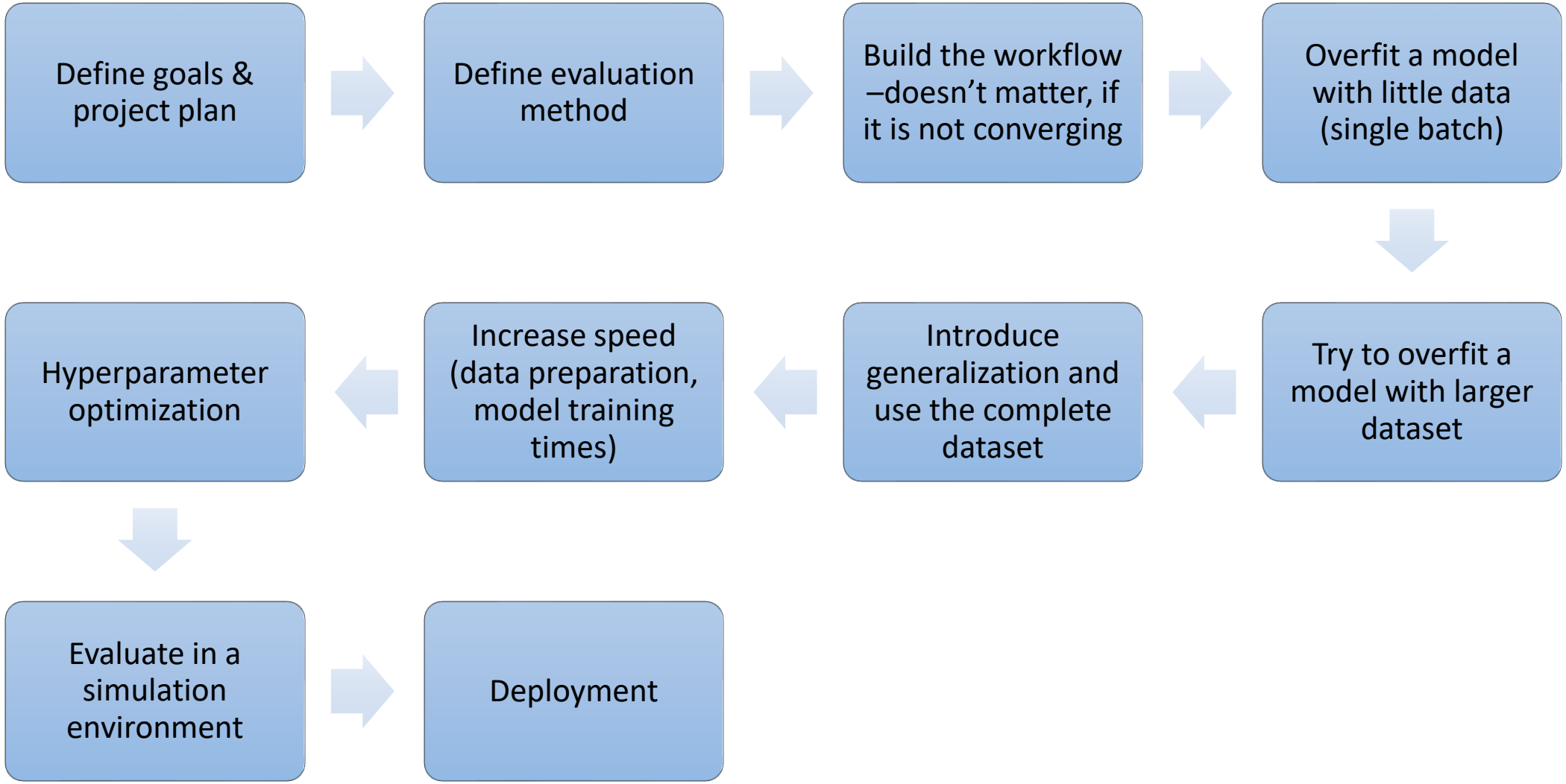
- *Business Understanding*
- *Data Understanding*
- *Data Preparation*
- *Modeling*
- *Evaluation*
- *Deployment*



Source: https://en.wikipedia.org/wiki/Cross_Industry_Standard_Process_for_Data_Mining


MACHINE LEARNING WORKFLOW
TIME SERIES ILLUSTRATION
DASHBOARDS
DATA SCIENCE ORCHESTRATION PIPELINE - EXPERIMENT TO SERVING
DATA ENGINEERING ORCHESTRATION PIPELINE - INGESTION TO TRAINING
RELATIONAL DATASTORE
DATA LAKE / OBJECT STORE
NETWORK FS
DATA STAGE
INGESTION
CLEAN
VALIDATE
TRANSFORM
USE CASE DEPENDENT: LABEL
TRAINING STAGE
EXPERIMENT
TRAIN
TUNE
TRAINED MODEL
DEPLOYMENT STAGE
DEPLOY
PREDICTION/ INFERENCE
LOG
USE CASE DEPENDENT: FEATURE STORE
METADATA STORE / MODEL REPOSITORY
MONITORING (TRAINING - DEPLOYMENT)
USE CASE DEPENDENT: SYNTHETIC DATA GENERATION/ AUGMENTATION
AI/ML DATA FOUNDATION - VERSIONING DATA LAKE WITH LINEAGE TRACKING
AI Infrastructure Alliance
HTTPS://AI-INFRASTRUCTURE.ORG

Machine learning project main steps



No free lunch theorem

Wolpert, D. H., & Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, *1*(1), 67-82.

AI/Deep learning roles

Data engineer

Data scientist

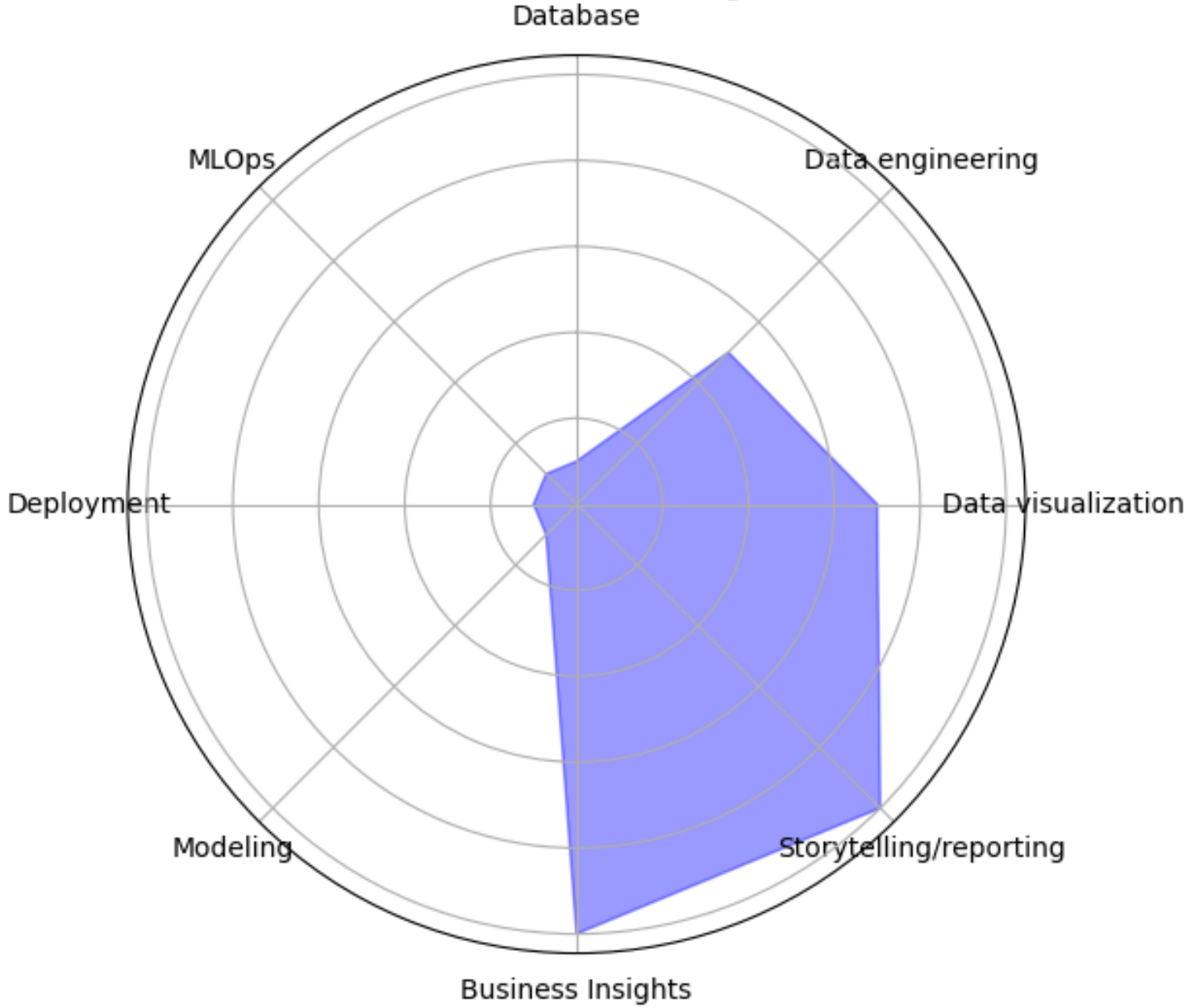
Business analyst

DL/ML Engineer

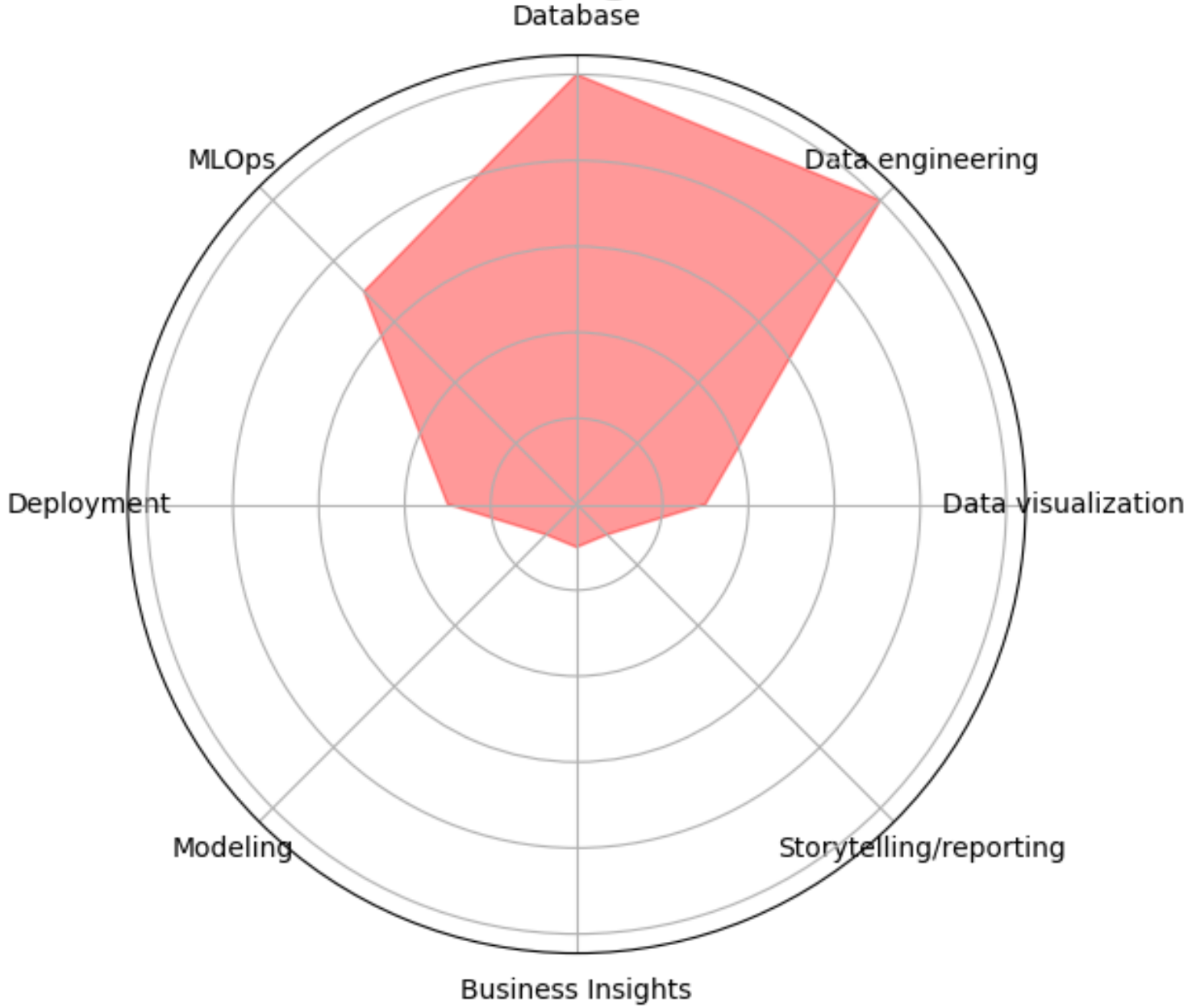
AI/Deep learning skills

Database	Data engineering	Data vizualization	Storytelling/reporting
Business Insights	Modeling	Deployment	MLOps

Business Analyst

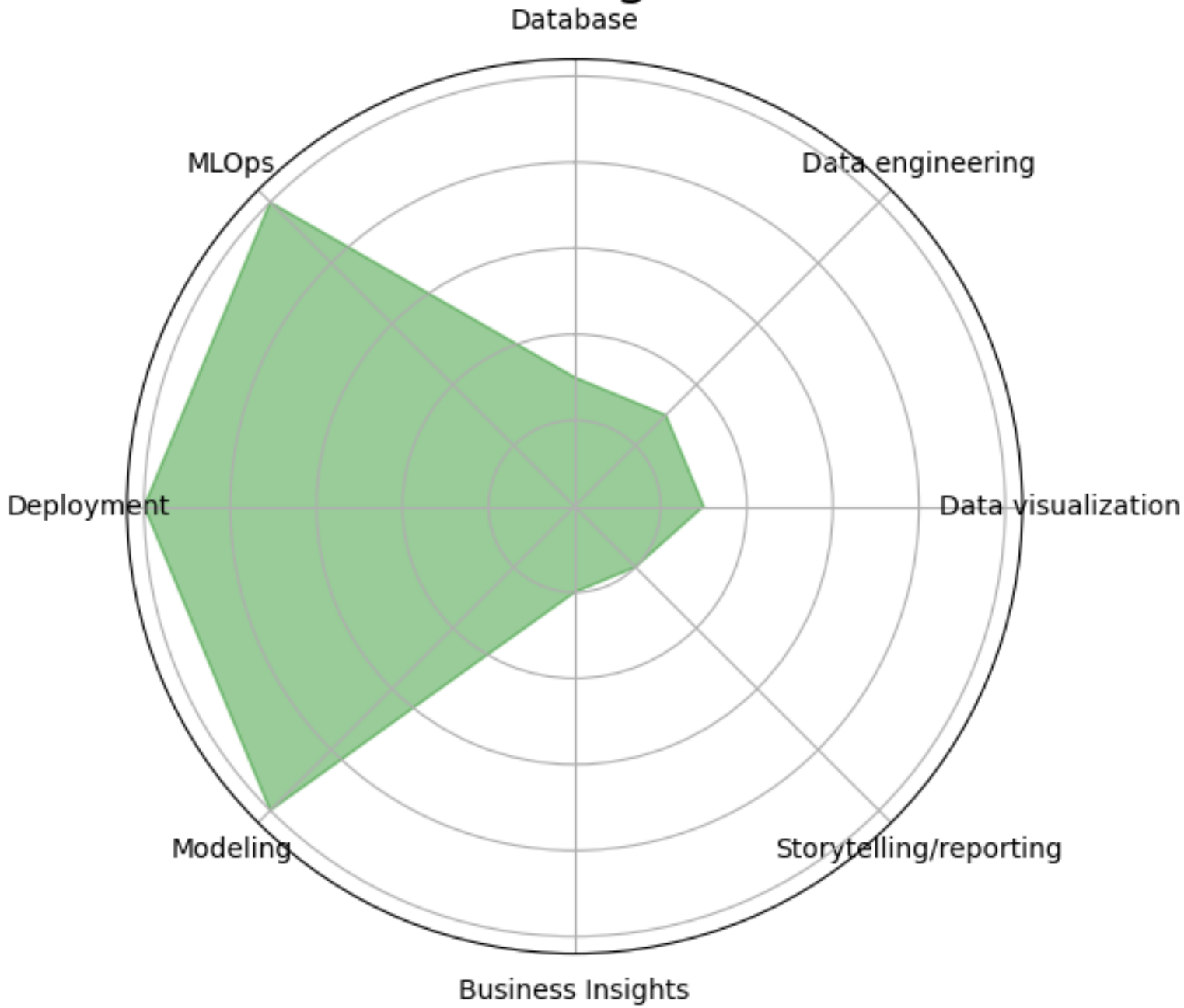


Data Engineer

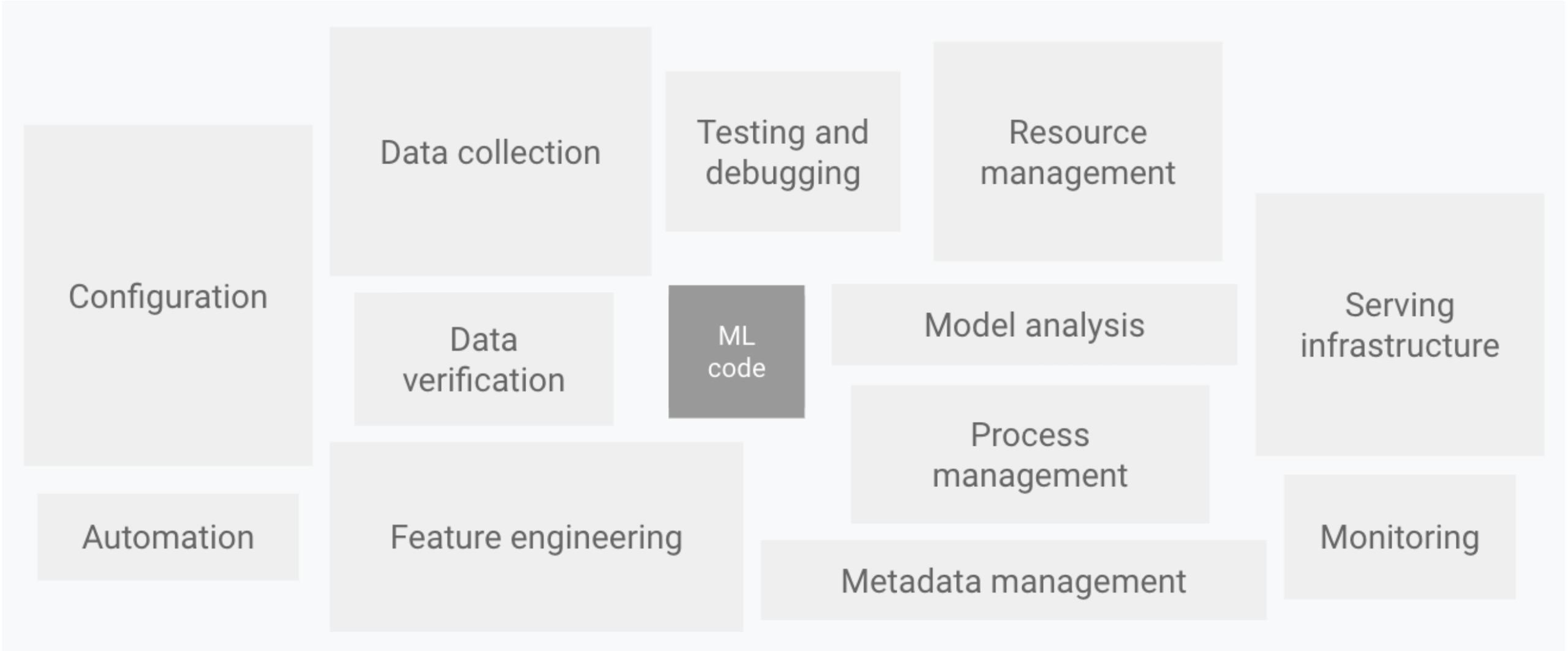


Data Scientist

ML/DL Engineer



General ML related tasks

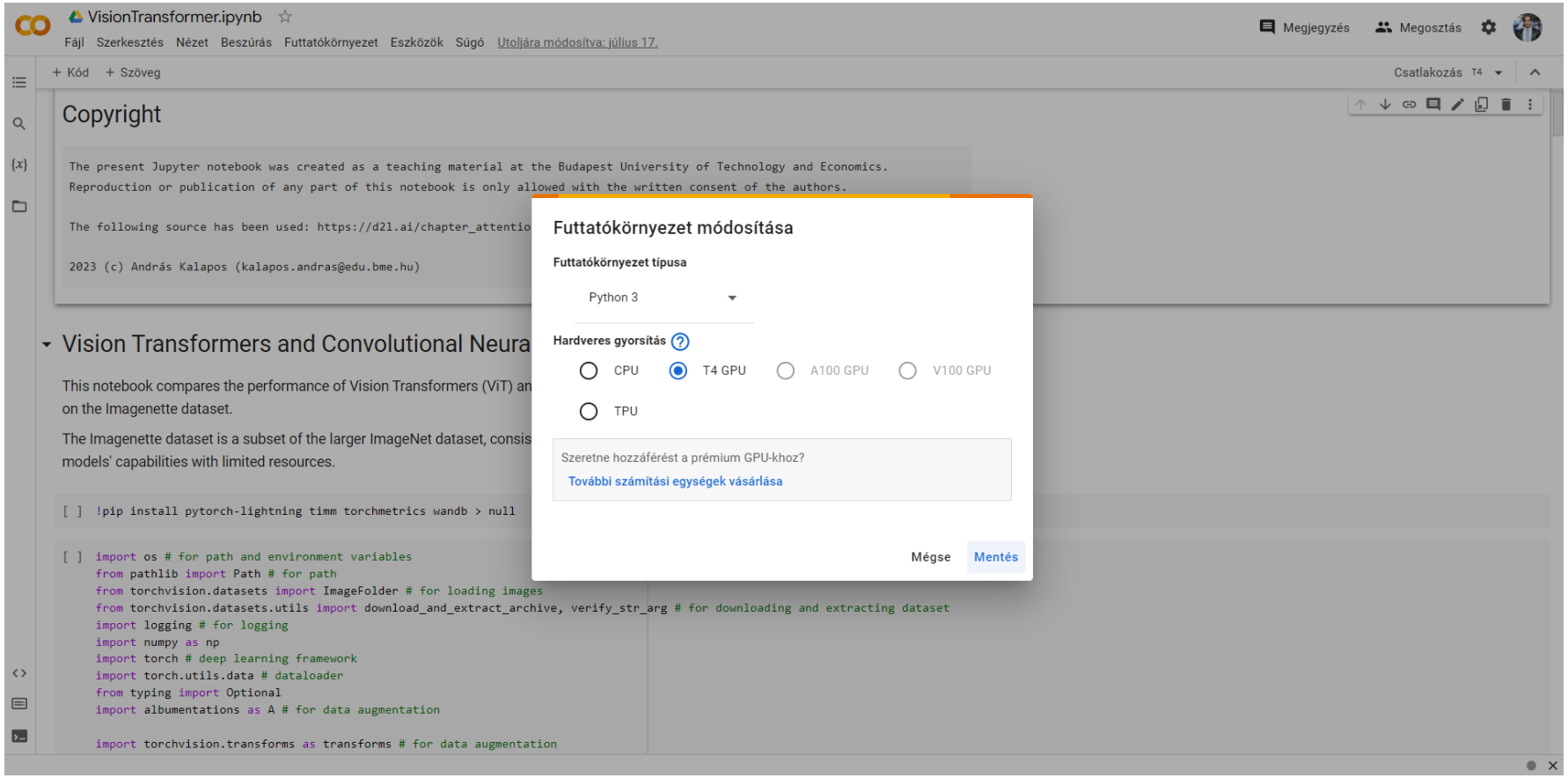


Source: https://cloud.google.com/architecture/mlops-continuous-delivery-and-automation-pipelines-in-machine-learning

Deep Learning software stack

Google Colab

https://colab.research.google.com/

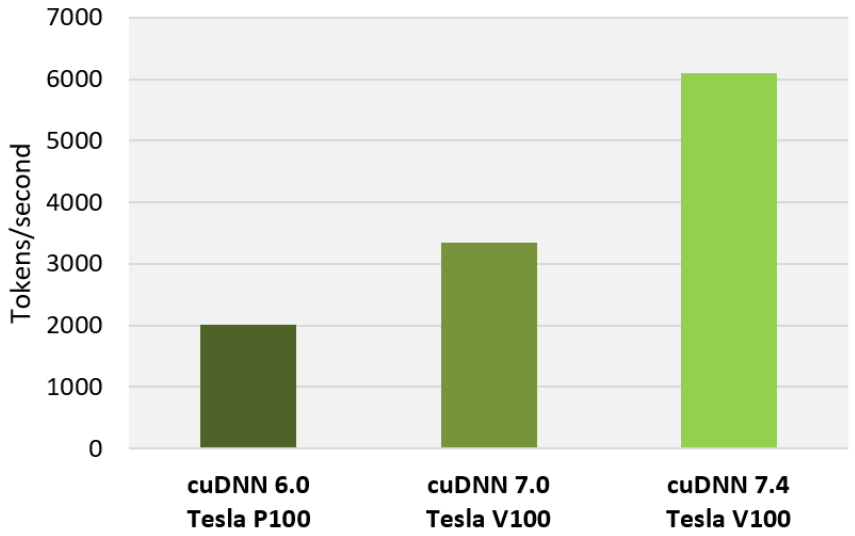



Grafana
PyTorch
Keras
TensorFlow
HOROVOD
ANSIBLE
Prometheus
docker
kubernetes
slurm
workload manager
NVIDIA
CUDA
cuDNN

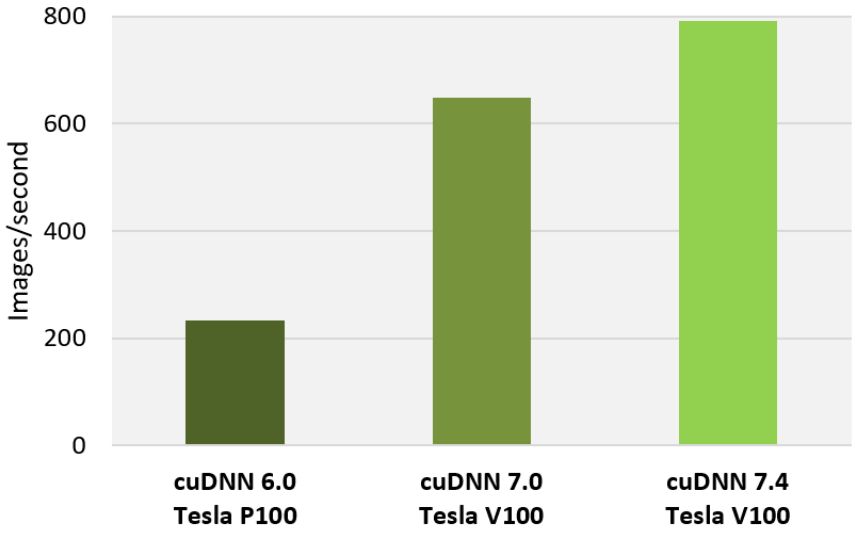
Basic software components: CUDA driver

Required: NVIDIA Driver

Main elements: cuBLAS, cuSPARSE, cuDNN, NCCL, NVVP, debugger, memcheck.



TensorFlow performance (tokens/sec), Tesla P100 + cuDNN 6 (FP32) on 17.12 NGC container, Tesla V100 + cuDNN 7.0 (Mixed) on 18.02 NGC container, Telsa V100 + cuDNN 7.4 (Mixed) on 18.10 NGC container, OpenSeq2Seq (GNMT), Batch Size: 64



TensorFlow performance (images/sec), Tesla P100 + cuDNN 6 (FP32) on 17.12 NGC container, Tesla V100 + cuDNN 7.0 (Mixed) on 18.02 NGC container, Telsa V100 + cuDNN 7.4 (Mixed) on 18.10 NGC container, ResNet-50, Batch Size: 128

CUDA version

`nvidia-smi`

```
Fri Jul 28 19:17:05 2023
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 525.125.06   Driver Version: 525.125.06   CUDA Version: 12.0     |
|-------------------------------+----------------------+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|                               |                      |               MIG M. |
|===============================+======================+======================|
|   0  NVIDIA A100-SXM...  On   | 00000000:07:00.0 Off |                    0 |
| N/A   53C    P0   349W / 400W |  76249MiB / 81920MiB |    100%      Default |
|                               |                      |             Disabled |
+-------------------------------+----------------------+----------------------+
|   1  NVIDIA A100-SXM...  On   | 00000000:0F:00.0 Off |                    0 |
| N/A   57C    P0   394W / 400W |  76873MiB / 81920MiB |    100%      Default |
|                               |                      |             Disabled |
+-------------------------------+----------------------+----------------------+
|   2  NVIDIA A100-SXM...  On   | 00000000:47:00.0 Off |                    0 |
| N/A   61C    P0   370W / 400W |  76615MiB / 81920MiB |     99%      Default |
|                               |                      |             Disabled |
+-------------------------------+----------------------+----------------------+
|   3  NVIDIA A100-SXM...  On   | 00000000:4E:00.0 Off |                    0 |
| N/A   59C    P0   404W / 400W |  76615MiB / 81920MiB |    100%      Default |
|                               |                      |             Disabled |
+-------------------------------+----------------------+----------------------+
|   4  NVIDIA A100-SXM...  On   | 00000000:87:00.0 Off |                    0 |
| N/A   74C    P0   397W / 400W |  76619MiB / 81920MiB |    100%      Default |
|                               |                      |             Disabled |
+-------------------------------+----------------------+----------------------+
|   5  NVIDIA A100-SXM...  On   | 00000000:90:00.0 Off |                    0 |
| N/A   67C    P0   353W / 400W |  76601MiB / 81920MiB |    100%      Default |
|                               |                      |             Disabled |
+-------------------------------+----------------------+----------------------+
|   6  NVIDIA A100-SXM...  On   | 00000000:B7:00.0 Off |                    0 |
| N/A   71C    P0   389W / 400W |  76631MiB / 81920MiB |     99%      Default |
|                               |                      |             Disabled |
+-------------------------------+----------------------+----------------------+
|   7  NVIDIA A100-SXM...  On   | 00000000:BD:00.0 Off |                    0 |
| N/A   71C    P0   336W / 400W |  76687MiB / 81920MiB |    100%      Default |
|                               |                      |             Disabled |
+-------------------------------+----------------------+----------------------+

+-----------------------------------------------------------------------------+
| Processes:                                                                  |
|  GPU   GI   CI        PID   Type   Process name                  GPU Memory |
|        ID   ID                                                   Usage      |
|=============================================================================|
|    0   N/A  N/A    342870      C   /opt/conda/bin/python3          76246MiB |
|    1   N/A  N/A    342871      C   /opt/conda/bin/python3          76870MiB |
|    2   N/A  N/A    342872      C   /opt/conda/bin/python3          76612MiB |
|    3   N/A  N/A    342873      C   /opt/conda/bin/python3          76612MiB |
|    4   N/A  N/A    342874      C   /opt/conda/bin/python3          76616MiB |
|    5   N/A  N/A    342875      C   /opt/conda/bin/python3          76598MiB |
|    6   N/A  N/A    342876      C   /opt/conda/bin/python3          76628MiB |
|    7   N/A  N/A    342877      C   /opt/conda/bin/python3          76684MiB |
+-----------------------------------------------------------------------------+
```

`nvcc --version`

```
nvcc: NVIDIA (R) Cuda compiler driver
Copyright (c) 2005-2022 NVIDIA Corporation
Built on Wed_Sep_21_10:33:58_PDT_2022
Cuda compilation tools, release 11.8, V11.8.89
Build cuda_11.8.r11.8/compiler.31833905_0
```

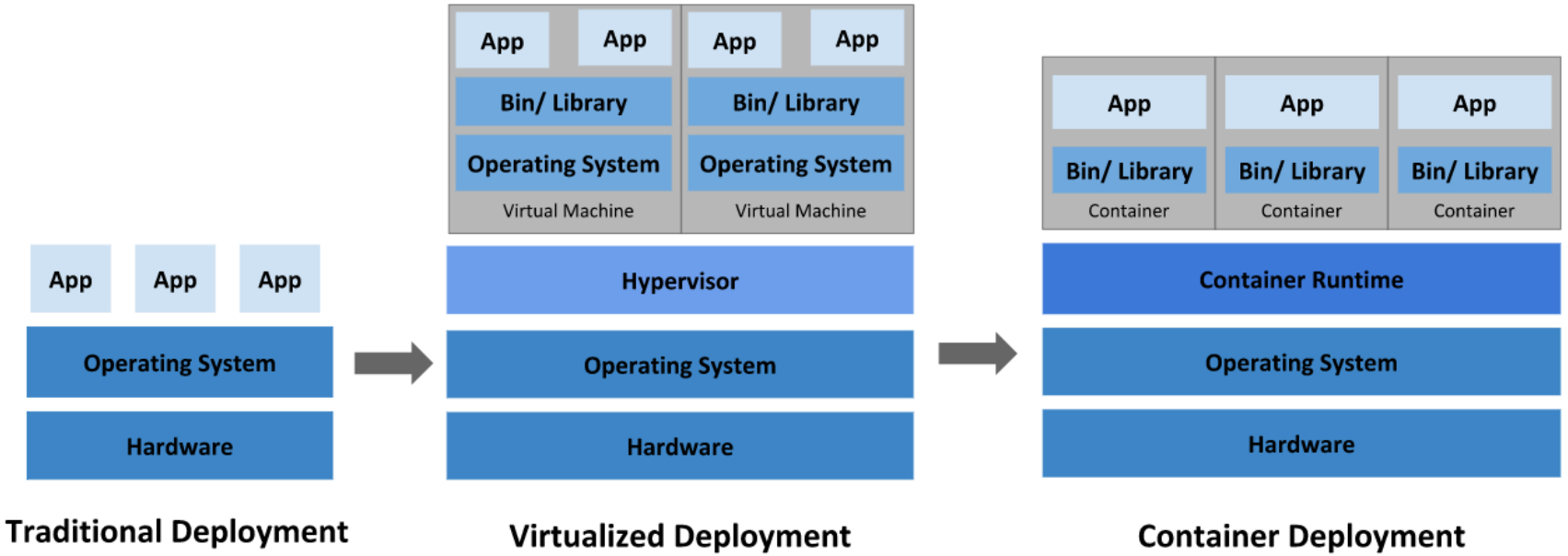
Basic software components: Containerization

Runs a virtual machine on the host and shares resources.

Encapsulations of system environments.

Advantages:

- Reproducibility
- Portability
- Isolation
- Integration



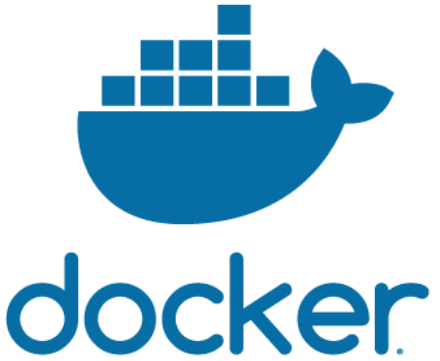
Source: https://kubernetes.io/docs/concepts/overview/what-is-kubernetes/

Basic software components: Containerization

- PaaS (Platform-as-a-Service) - Primarily for microservices
- OS and GPU lightweight virtualization
- Isolates Dev and Ops
- Docker images are stored in a local cache and can be interacted with by commands
- Large ecosystem, Linux, MacOS and Windows support

Difference compared to VM: e.g. the system is 1 GB

- 1000 VM ~ 1000 * 1 GB
- 1000 application container ~ 1GB
- Container is refreshed –> Everything is refreshed



Basic software components: DL frameworks

- TensorFlow and TensorFlow Keras (Google)
- PyTorch (Meta AI)
- JAX (Google)
- MXNet (Apache)
- Gluon (Amazon)
- Chainer
- PaddlePaddle

Depricated

- Sonnet (DeepMind)
- CNTK (Microsoft)

Advanced components: monitoring

Metrics logging tools is required:

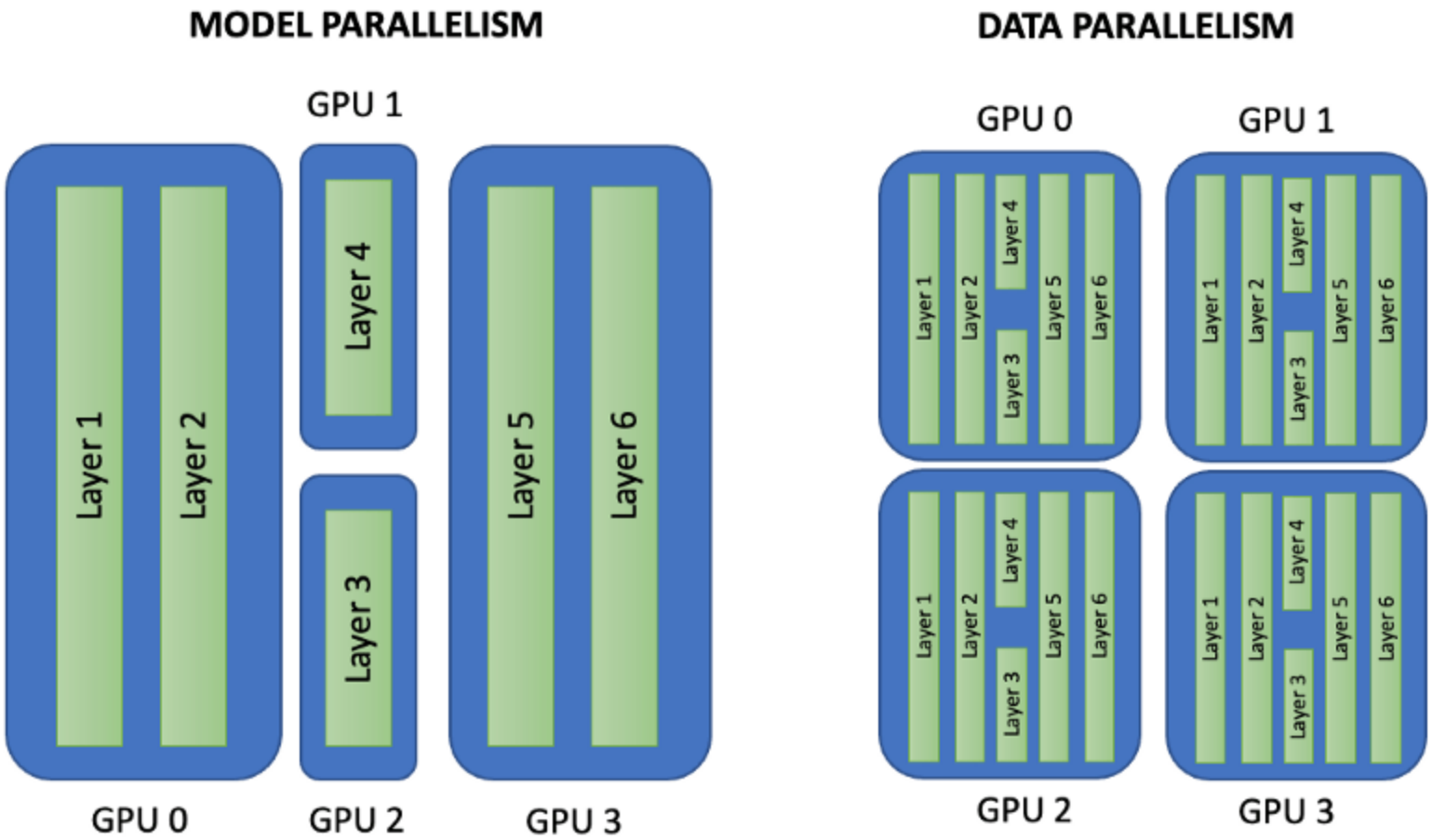
- nvidia-smi dmon
- Prometheus + NVML (NVIDIA Management Library)

Open source tools:

- Grafana
- Zabbix

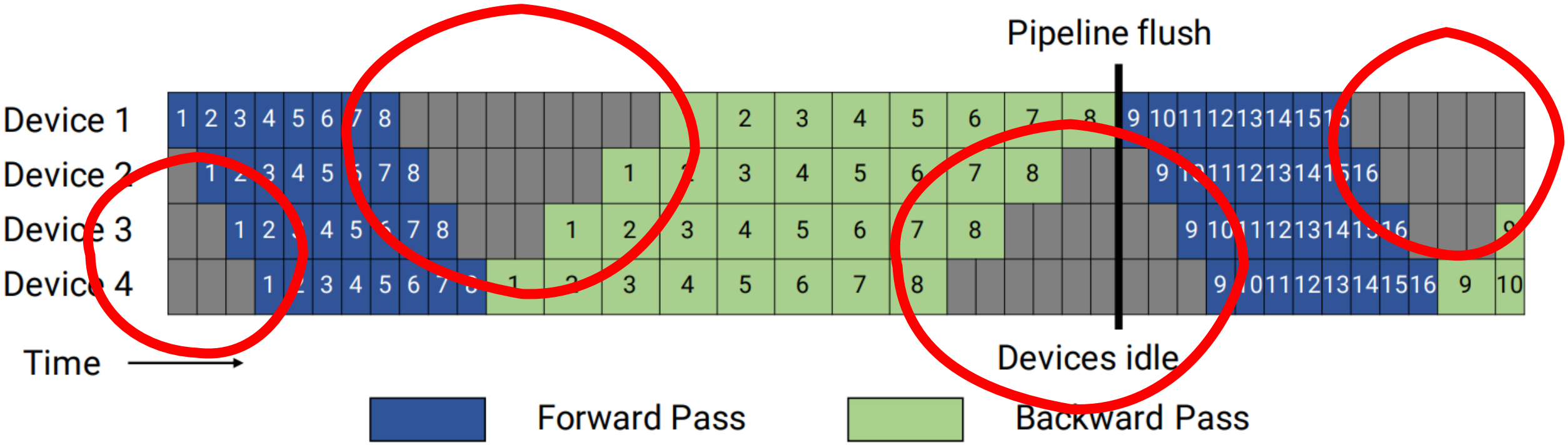
Source: https://grafana.com/grafana/dashboards/14574-nvidia-gpu-metrics/

Advanced components: multi GPU, multi node



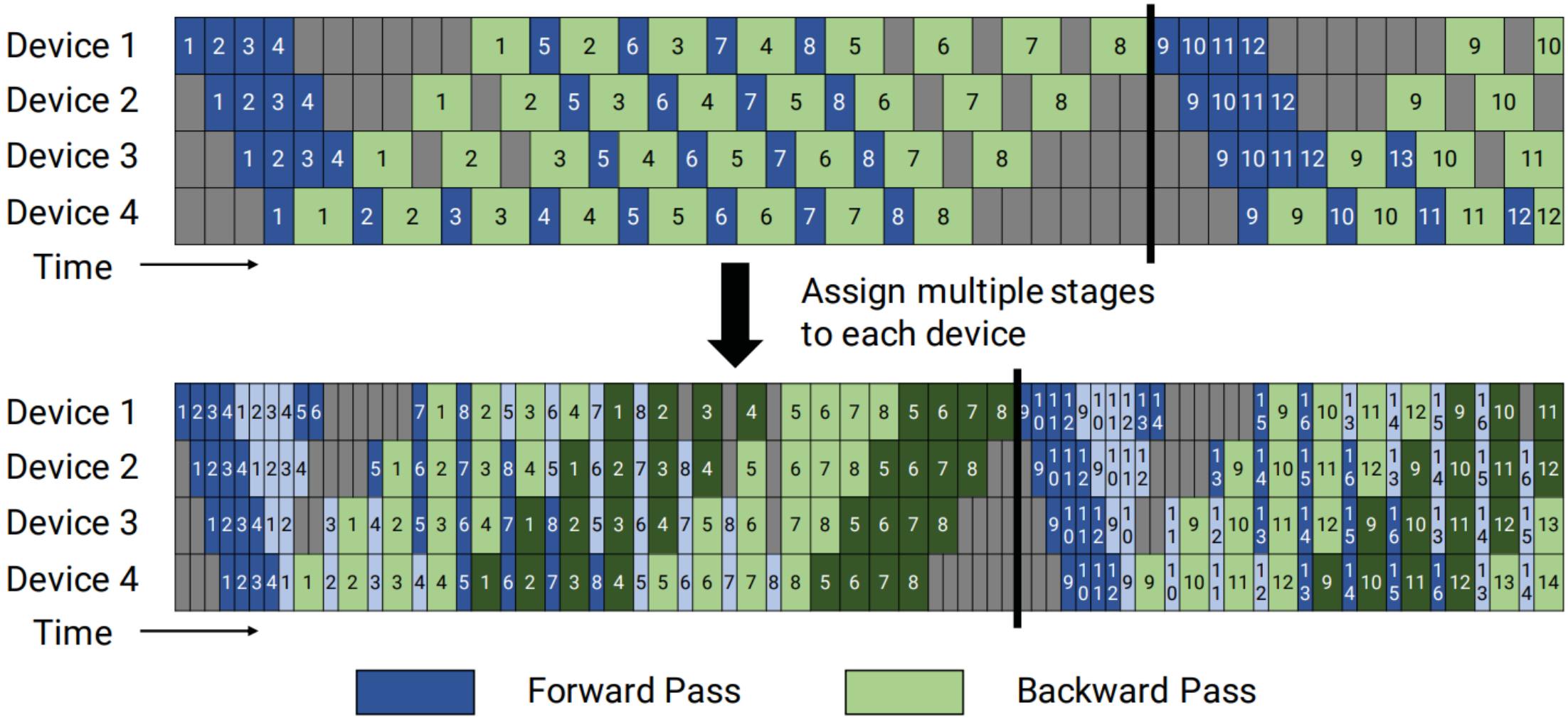
Source: https://neptune.ai/blog/distributed-training-frameworks-and-tools

Multi-GPU training (microbatching)



Source: Narayanan, Deepak, et al. "Efficient large-scale language model training on gpu clusters using megatron-lm." *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*. 2021.

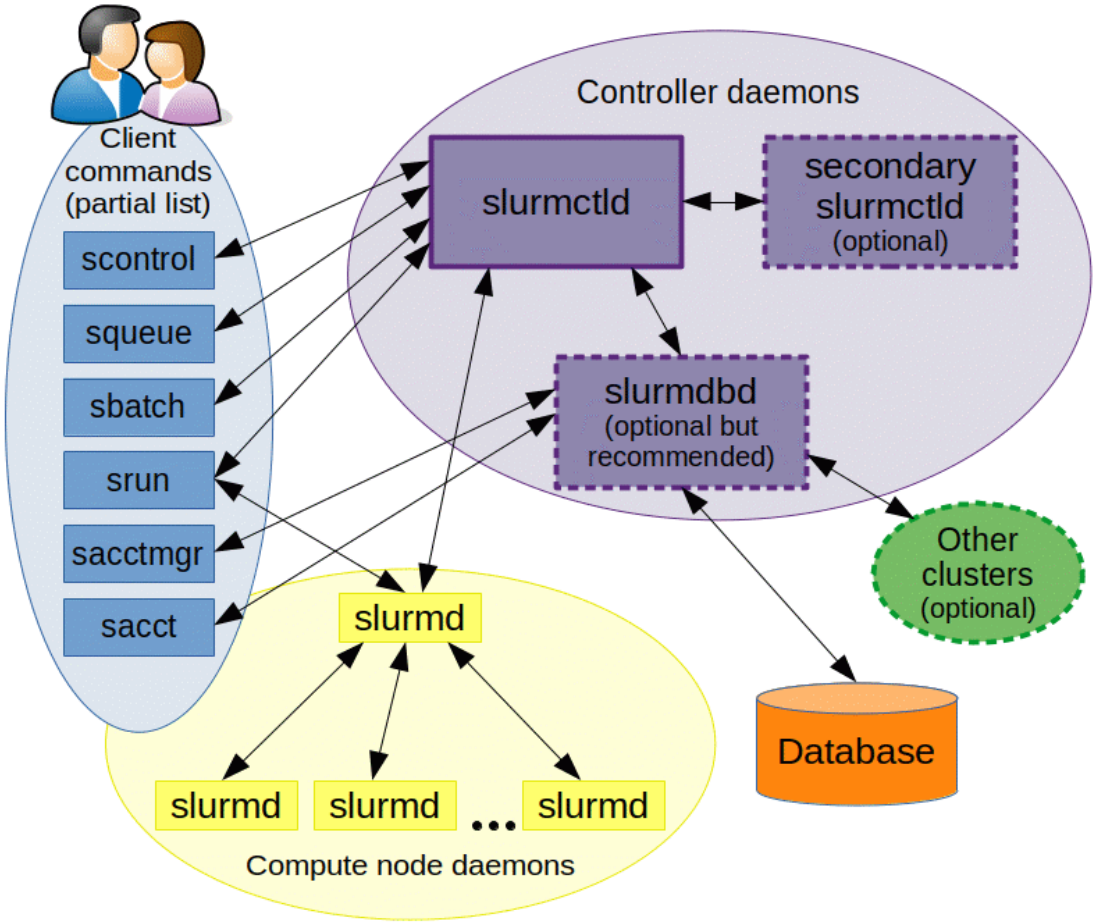
Multi-GPU training pipeline



Source: Narayanan, Deepak, et al. "Efficient large-scale language model training on gpu clusters using megatron-lm." *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*. 2021.

Advanced components: scheduler

- HPC/AI solution: SLURM Workload Manager



Source: https://slurm.schedmd.com/overview.html

SLURM batch script

```
#!/bin/bash

#SBATCH -J sample
#SBATCH -t 15:00
#SBATCH -N 2
#SBATCH -n 8
#SBATCH --gres=gpu:4
```

Job name

Requested time

Number of nodes

Number of CPUs

Number of GPUs

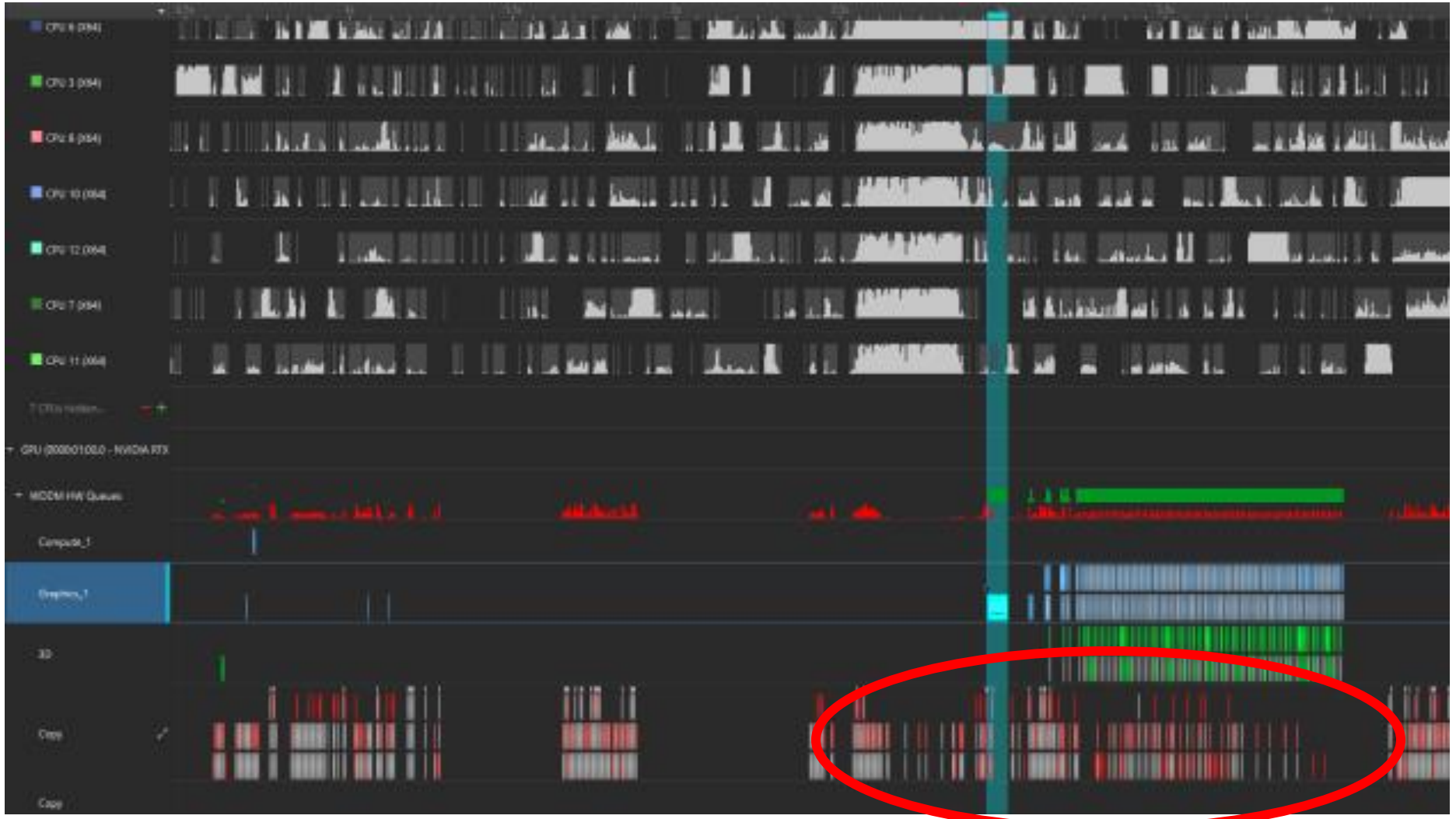
```
module load singularity OpenMPI/3.1.6-GCC-8.3.0

mpirun singularity run --nv horovod.sif python test.py
```

```
$ sbatch batchfile.sh
```

Advanced component: performance analytics

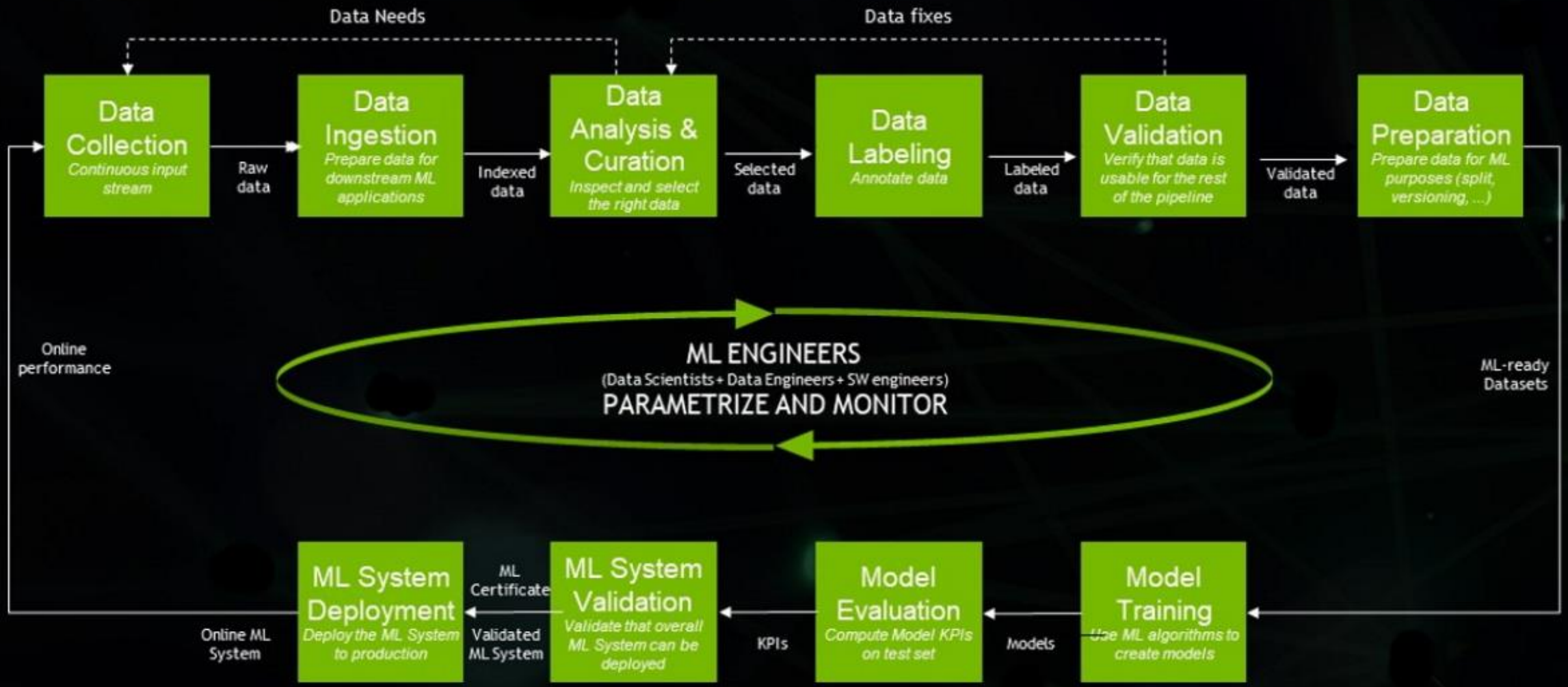
NVIDIA Nsight Systems:

- Monitor CPU and GPU usage
- Identify bottlenecks
- Get most out of the HW component



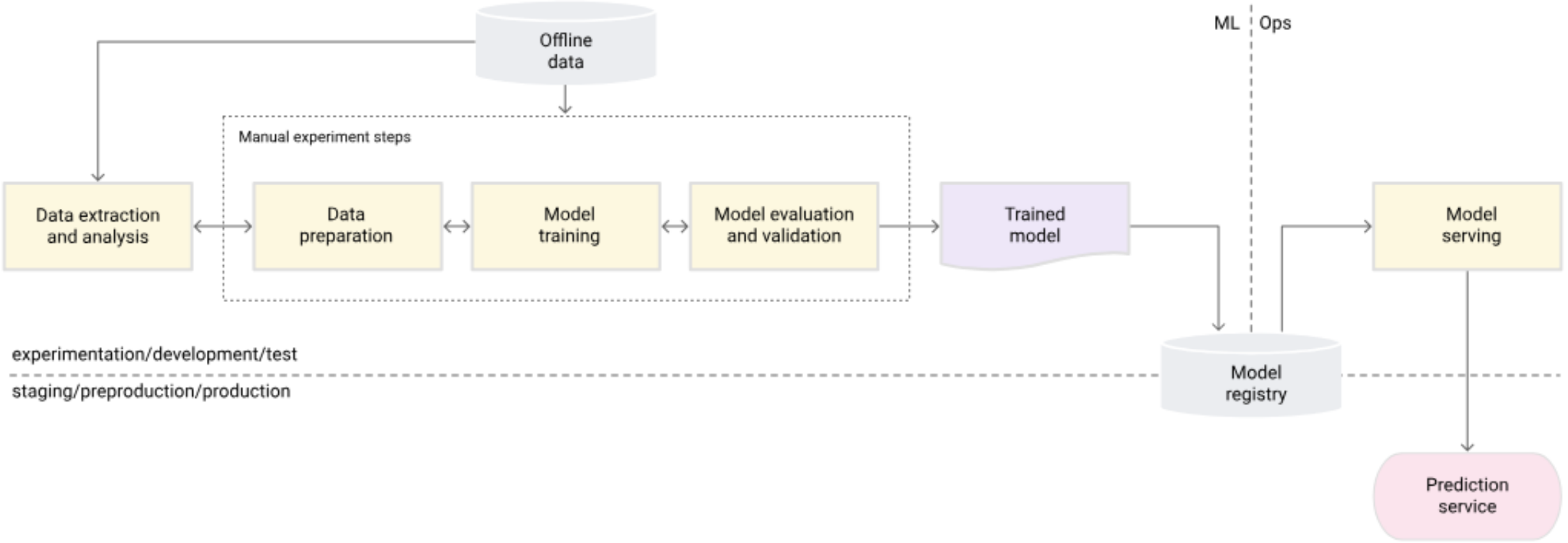
CPU-GPU interactions

MLOPS: THE AI LIFECYCLE FOR IT PRODUCTION



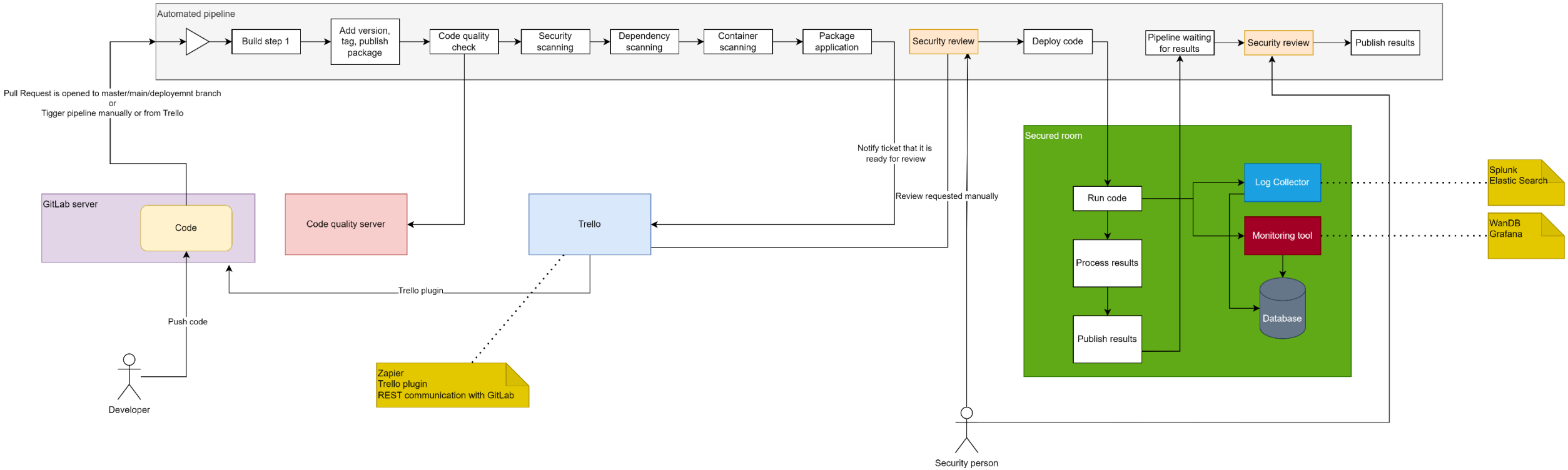
Source: https://blogs.nvidia.com/blog/2020/09/03/what-is-mlops/

MLOps pipeline: level 0 – manual process

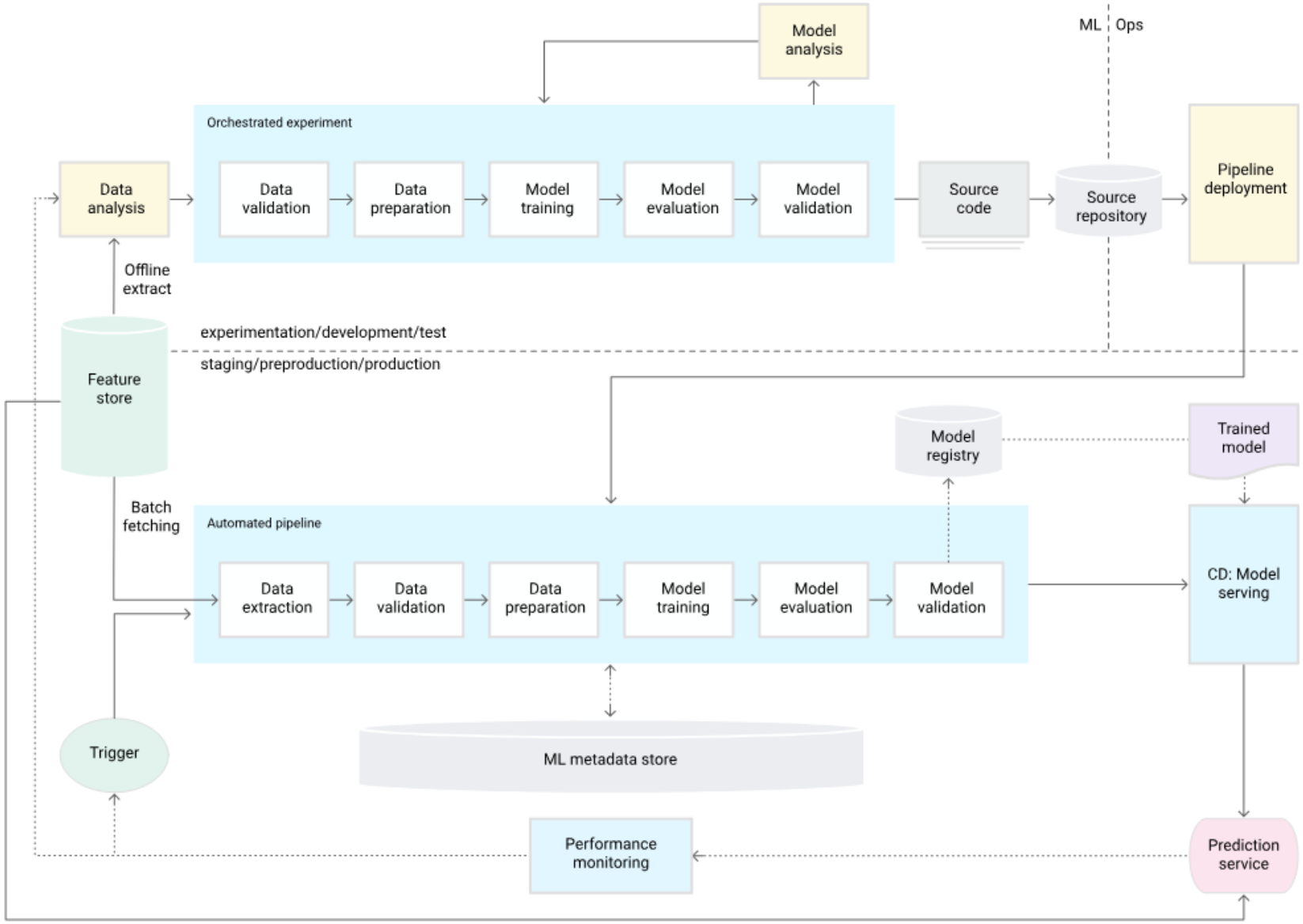


Source: https://cloud.google.com/architecture/mlops-continuous-delivery-and-automation-pipelines-in-machine-learning

MLOps pipeline: with manual intervention

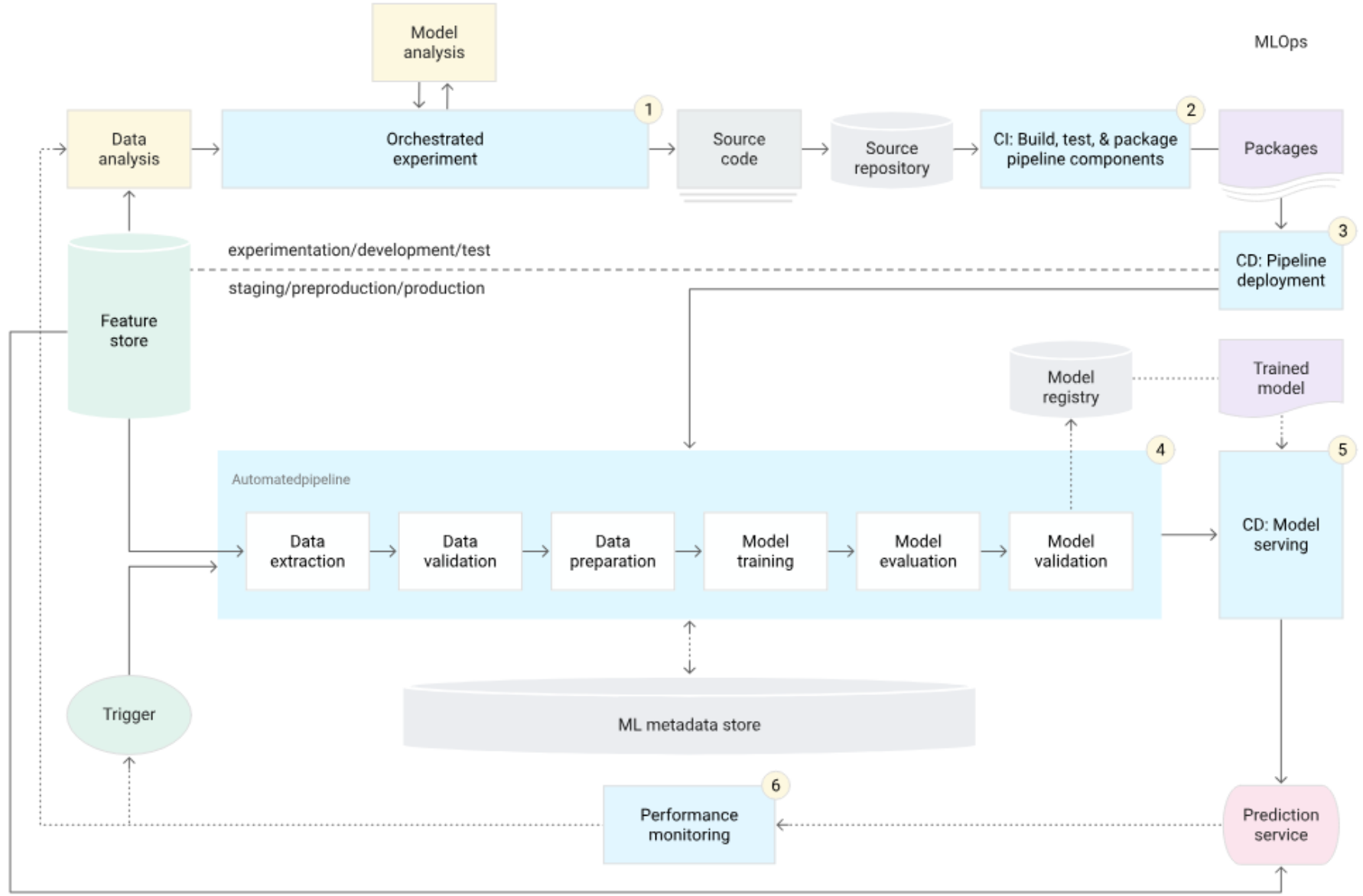


MLOps pipeline: level 1 – ML pipeline automation



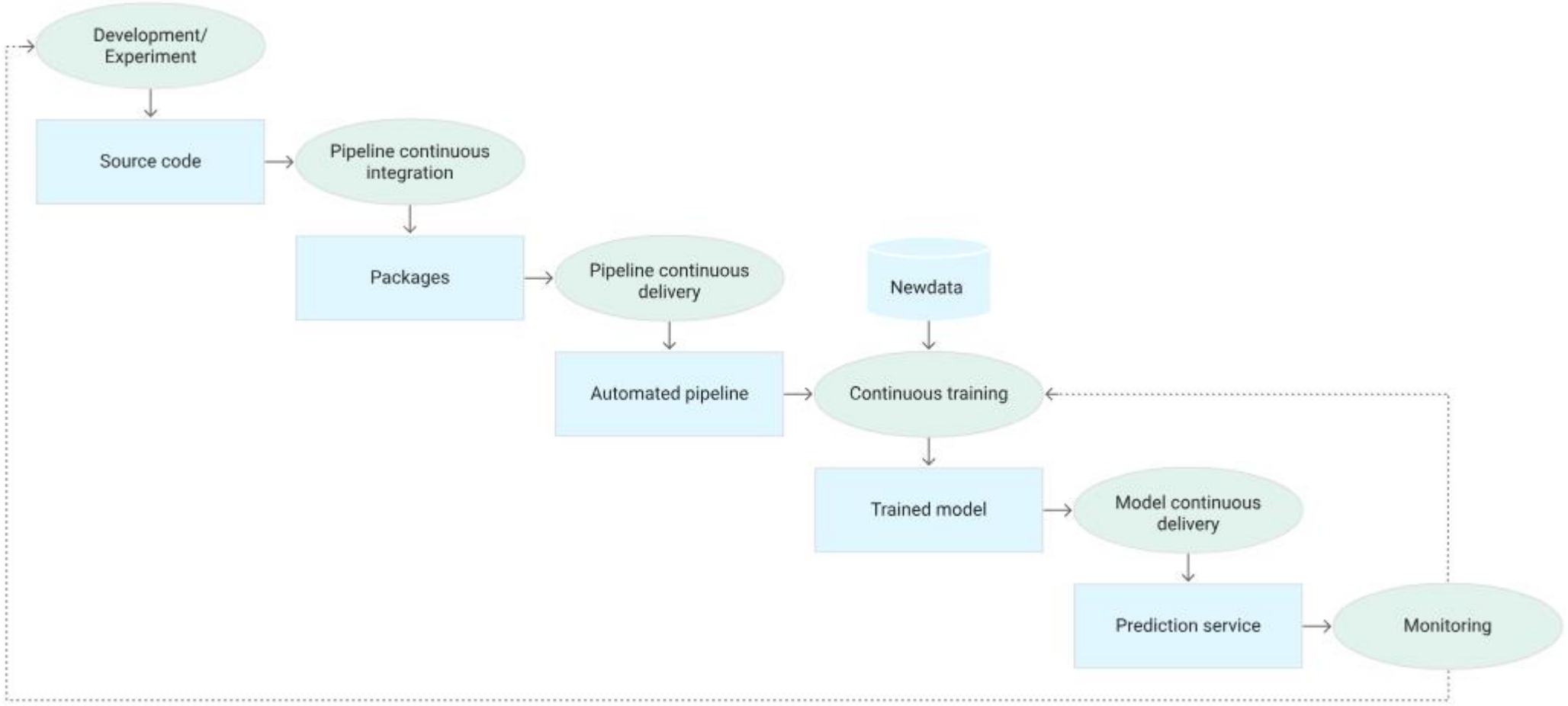
Source: https://cloud.google.com/architecture/mlops-continuous-delivery-and-automation-pipelines-in-machine-learning

MLOps pipeline: level 2 – CI/CD



Source: https://cloud.google.com/architecture/mlops-continuous-delivery-and-automation-pipelines-in-machine-learning

MLOps pipeline: level 2 – CI/CD



Source: https://cloud.google.com/architecture/mlops-continuous-delivery-and-automation-pipelines-in-machine-learning

References

- Google Colab: https://colab.research.google.com/
- MLOps pipeline: https://cloud.google.com/architecture/mlops-continuous-delivery-and-automation-pipelines-in-machine-learning

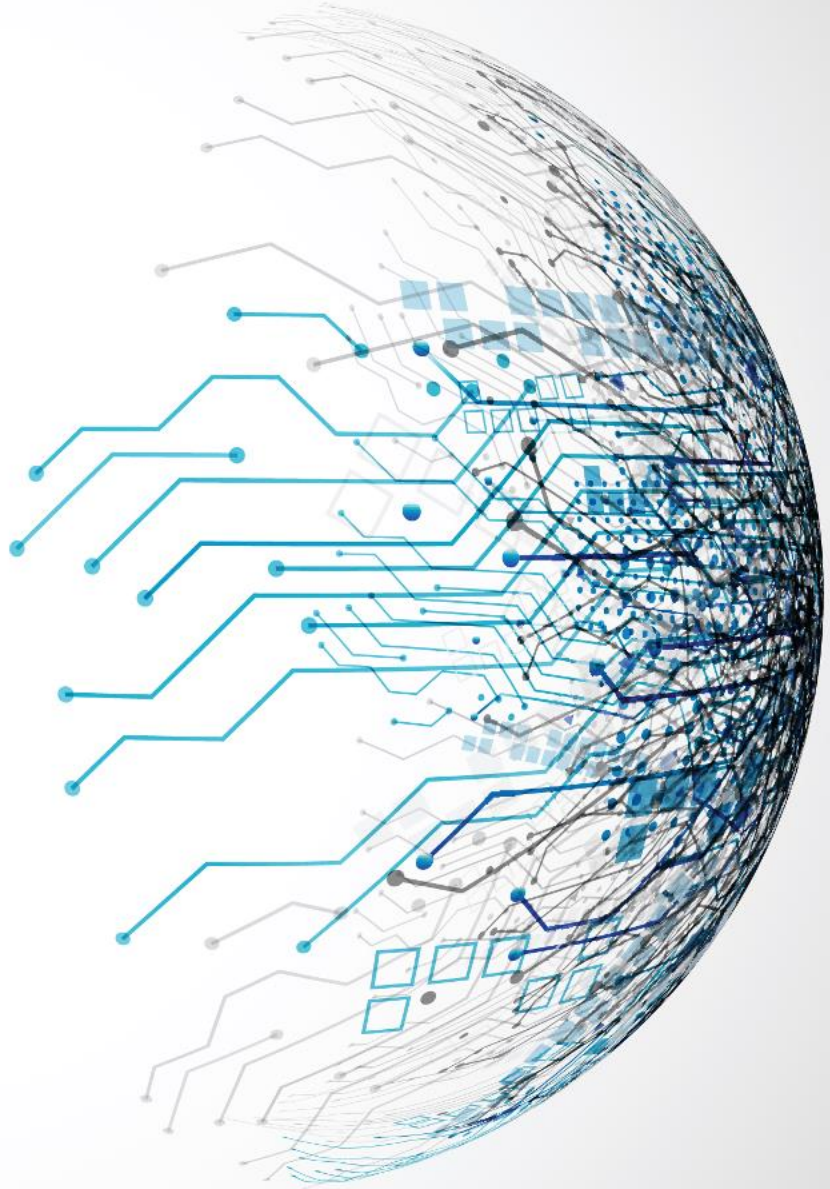
Please, don’t forget to send feedback:

https://bit.ly/bme-dl

Thank you for your attention

Dr. Mohammed Salah Al-Radhi
malradhi@tmit.bme.hu

(slides by: Dr. Bálint Gyires-Tóth)



03 September 2024