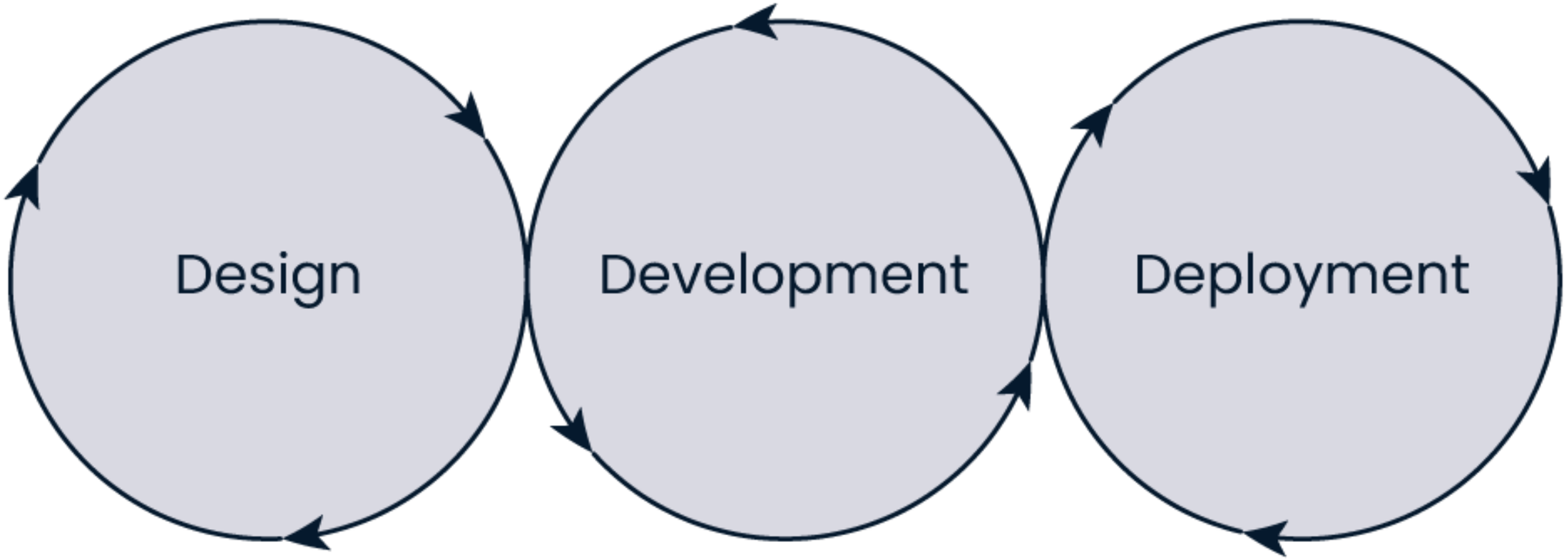
Monitoring machine learning models

MLOPS CONCEPTS

Monitoring & retraining

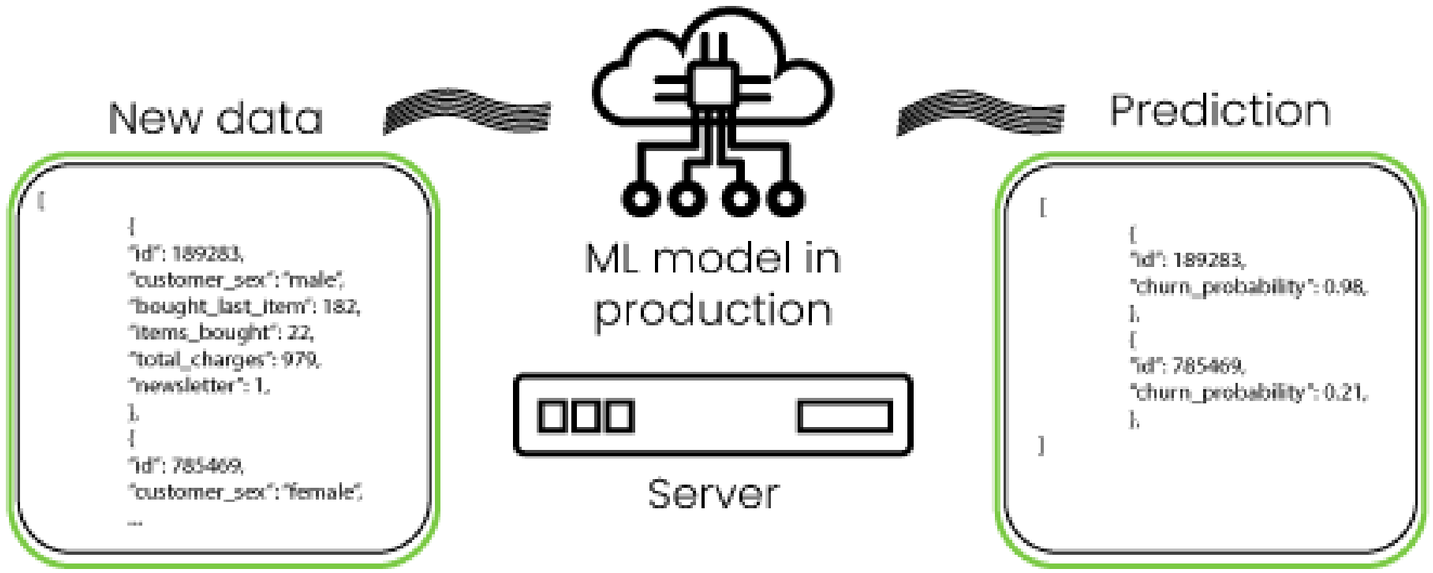


- Added value
- Business requirements
- Key metrics
- Data processing

- Feature engineering
- Experiment tracking
- Model training & evaluation

- Runtime environments
- Microservices architecture
- CI/CD pipeline
- Monitoring & retraining

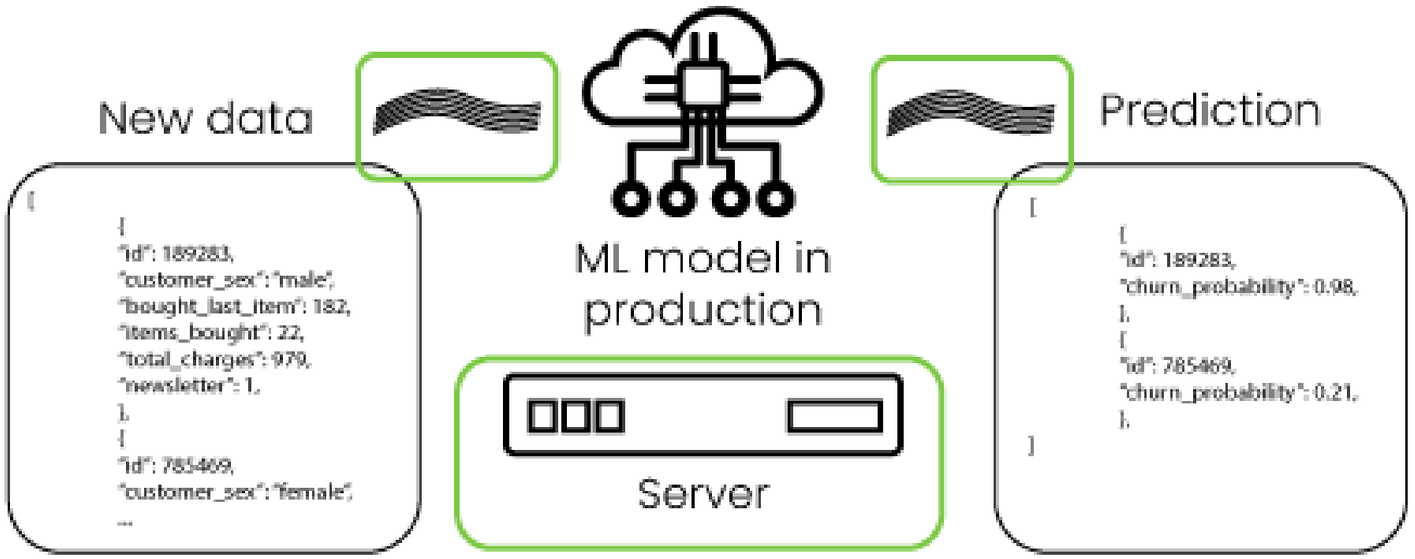
Types of monitoring



Statistical monitoring: focuses on the input and output data, including predictions

Examples: customer X has a 72% probability of churning, customer Y has a 31% probability of not churning

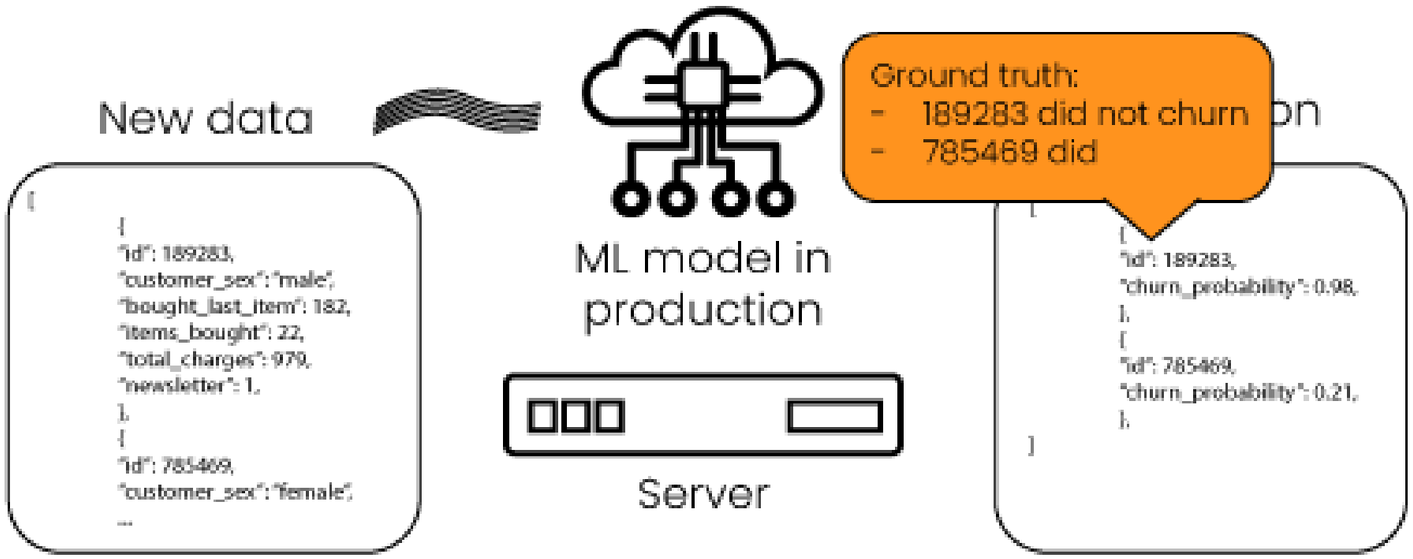
Types of monitoring



Computational monitoring: focuses on technical metrics

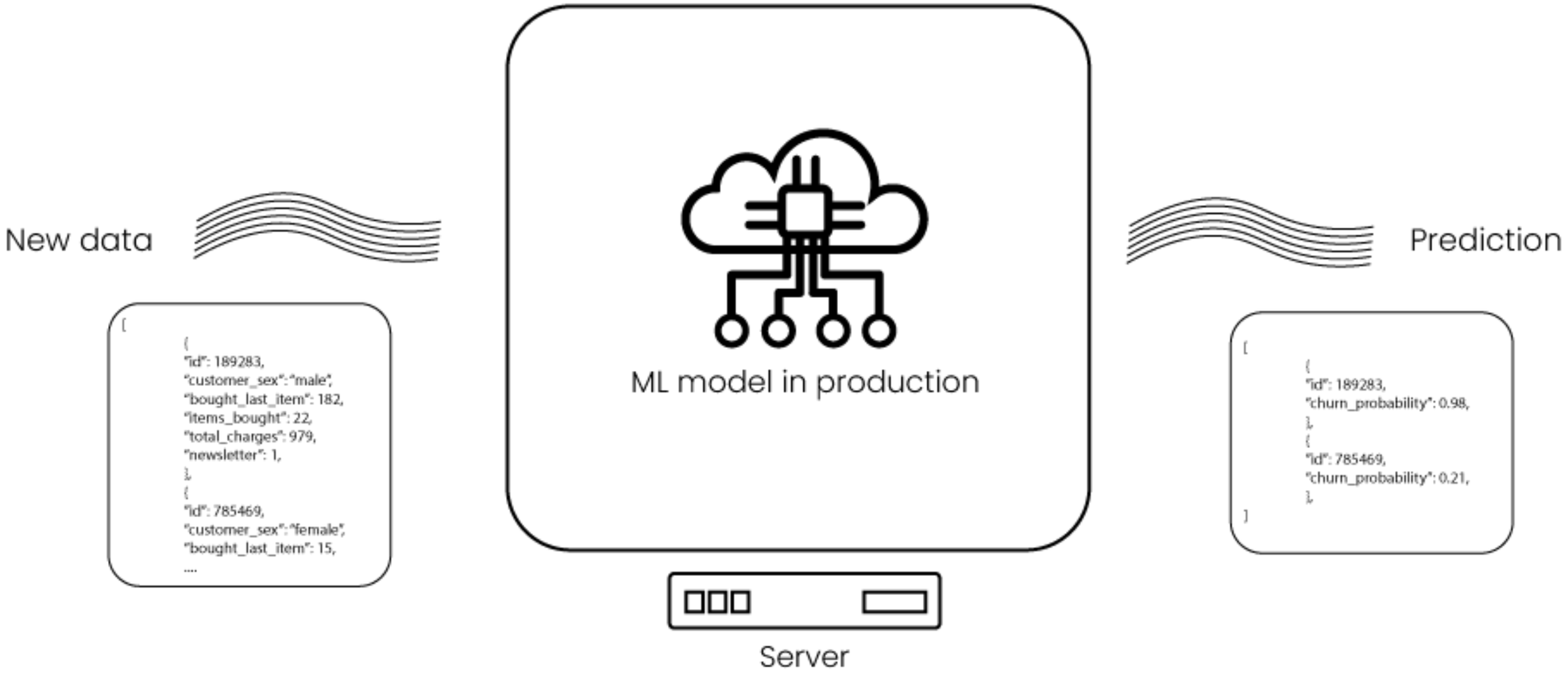
Examples: server CPU usage, number of incoming requests, number of predictions, downtime of server

Feedback loop



Feedback loop: the process through which the ground truth is used to improve the machine learning model

Monitoring in production



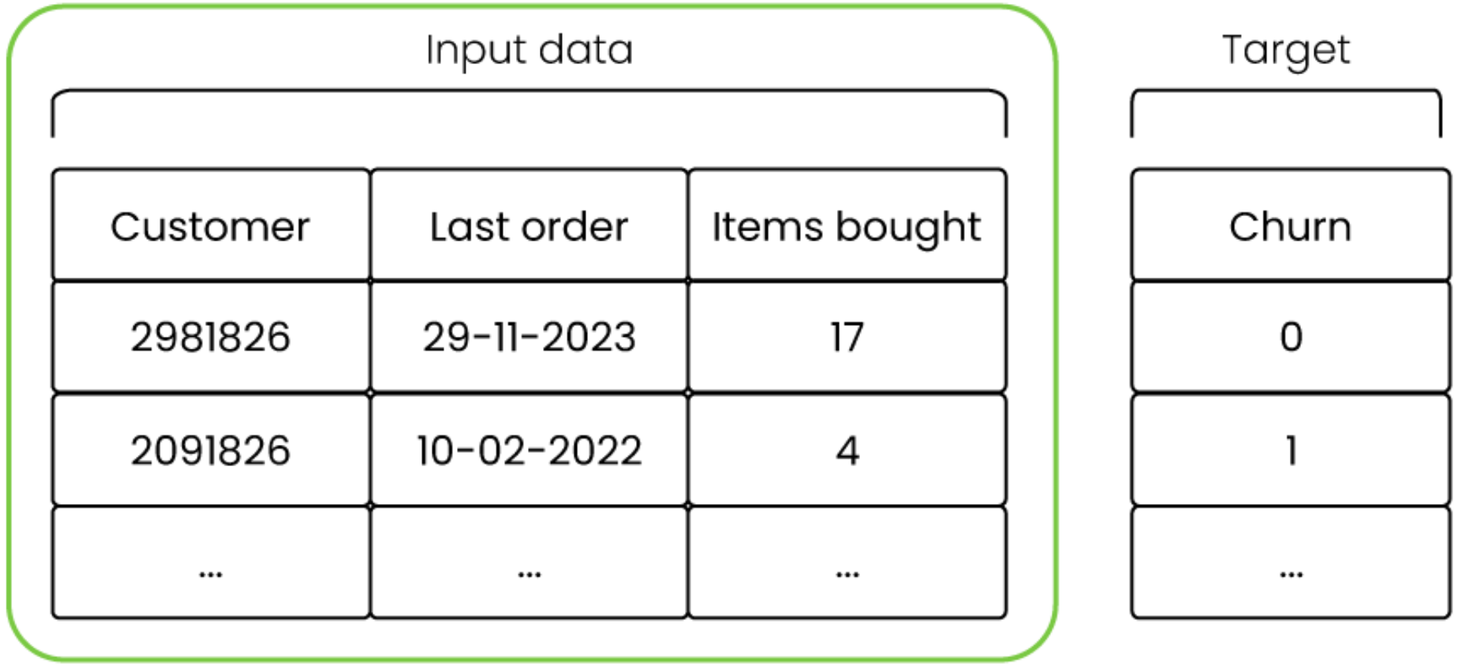
Retraining a machine learning model

MLOPS CONCEPTS

Retraining after changes

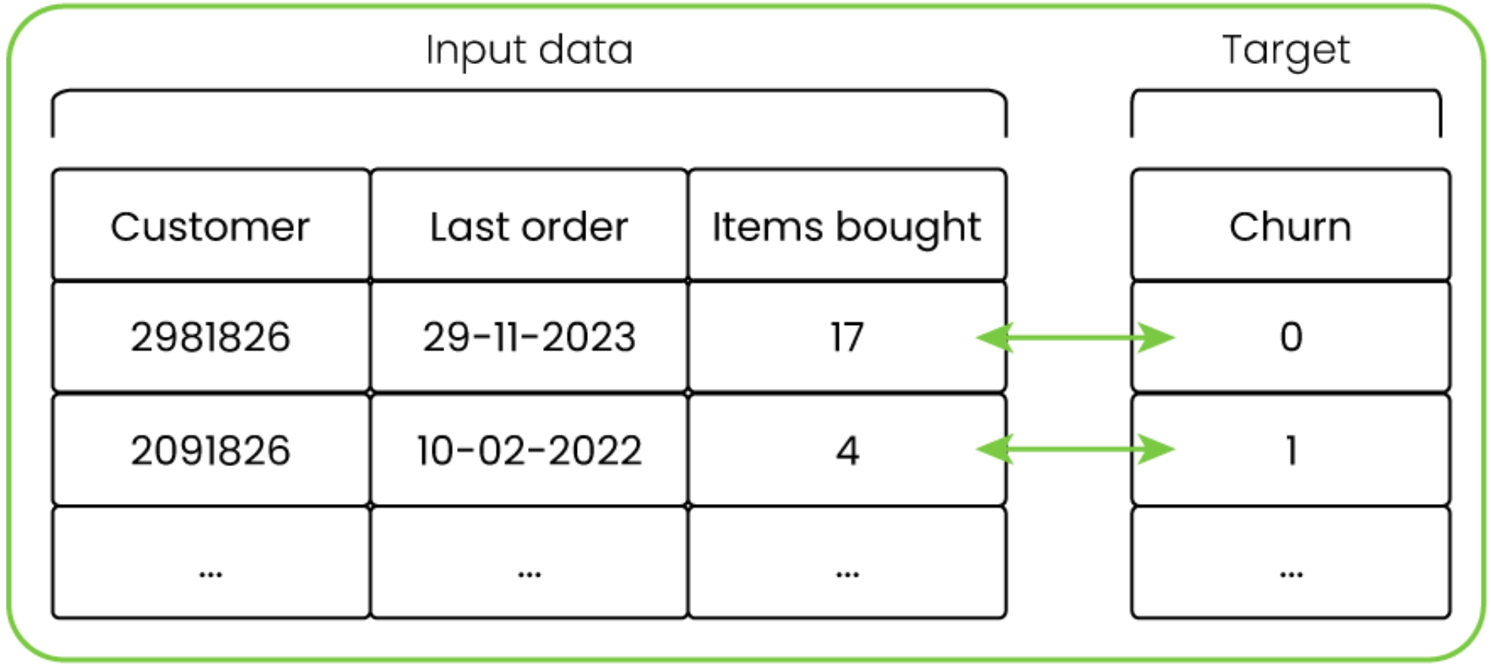
Retraining: use new data to develop a fresh version of the machine learning model

Data drift



Data drift: changes in the input data

Concept drift



Concept drift: changes in the relationship between input and output data

How often to retrain?

- **Business environment**: how volatile is the data?
- **Cost**: how much does it cost to retrain?
- **Business requirements**: what is the required model performance?

Retraining methods

Old data

Customer	Last order	Items bought
2981826	29-11-2023	17
2091826	10-02-2022	4
...	...	...

Churn
0
1
...

Original model

New data

Customer	Last order	Items bought
3029712	12-01-2024	17
4900298	18-04-2024	81
...	...	...

Churn
1
0
...

New model

Retraining methods

Old data

Customer	Last order	Items bought
2981826	29-11-2023	17
2091826	10-02-2022	4
...	...	...

Churn
0
1
...

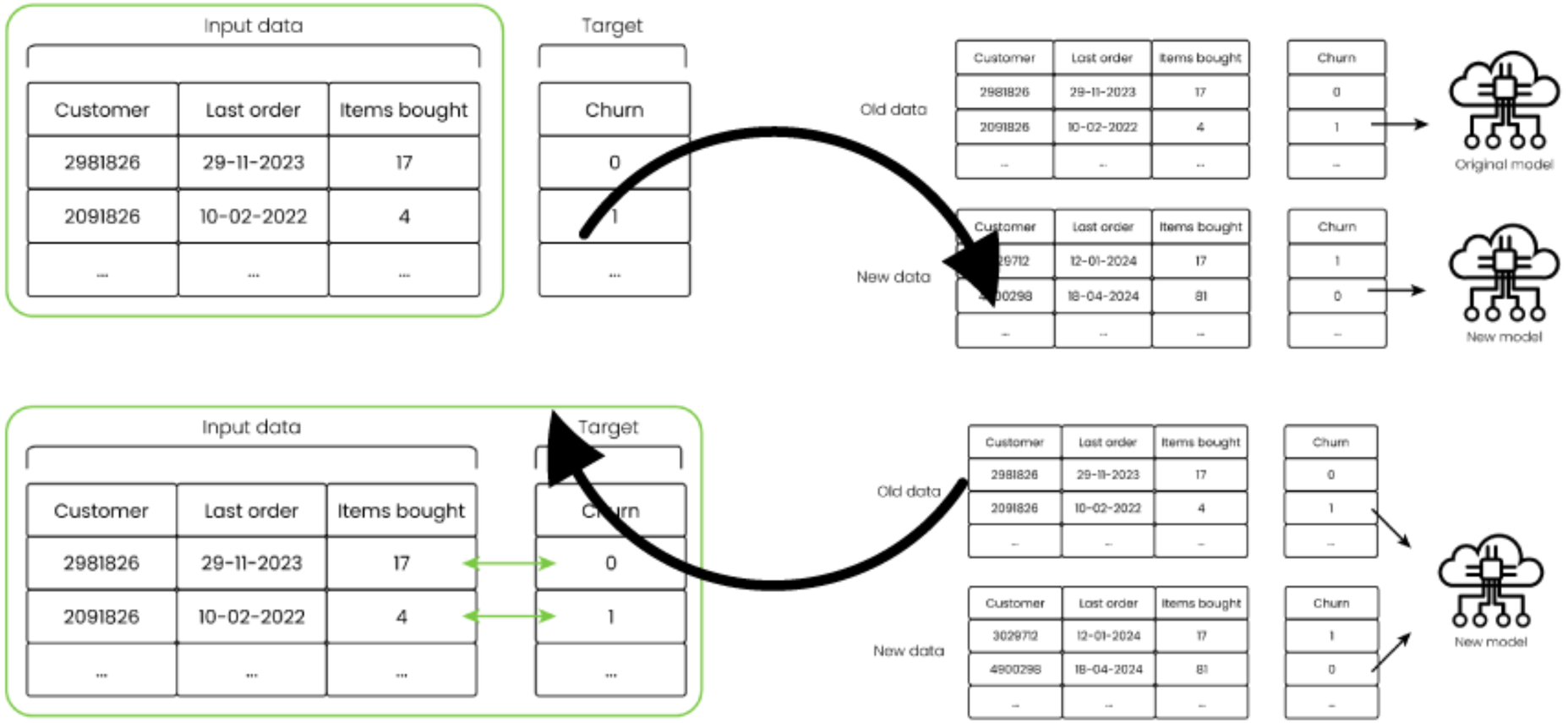
New data

Customer	Last order	Items bought
3029712	12-01-2024	17
4900298	18-04-2024	81
...	...	...

Churn
1
0
...

New model

Automatic retraining



Levels of MLOps maturity

MLOPS CONCEPTS

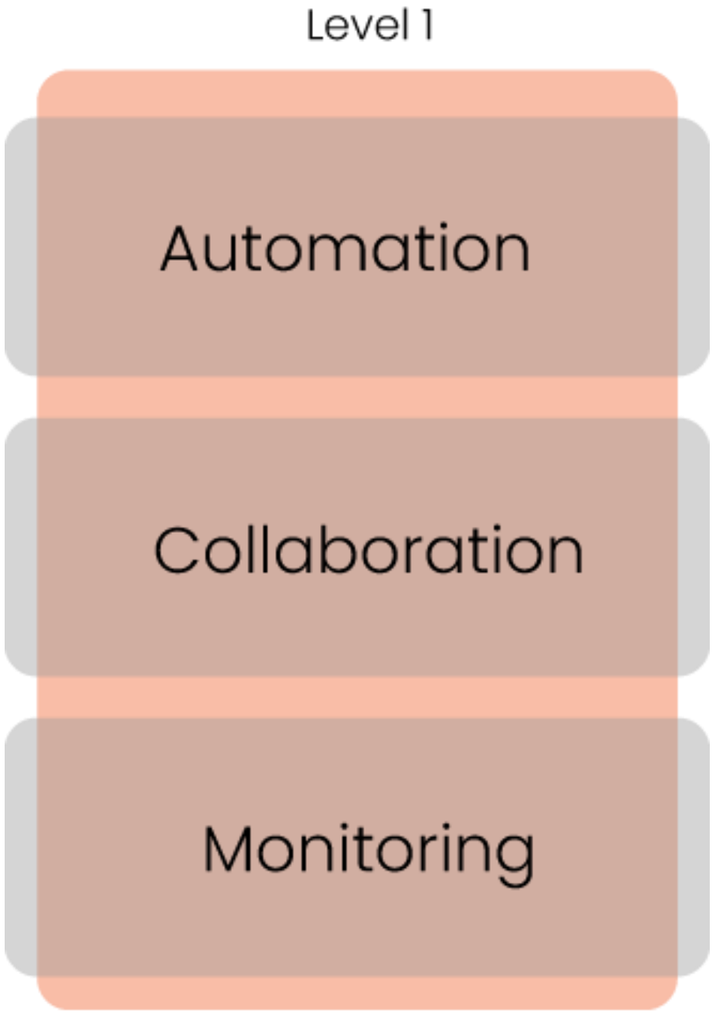
MLOps maturity

- Level of **automation**, **collaboration**, and **monitoring** within MLOps processes
- Higher level is not necessarily better
- Focus on development and deployment phase

Levels of MLOps maturity

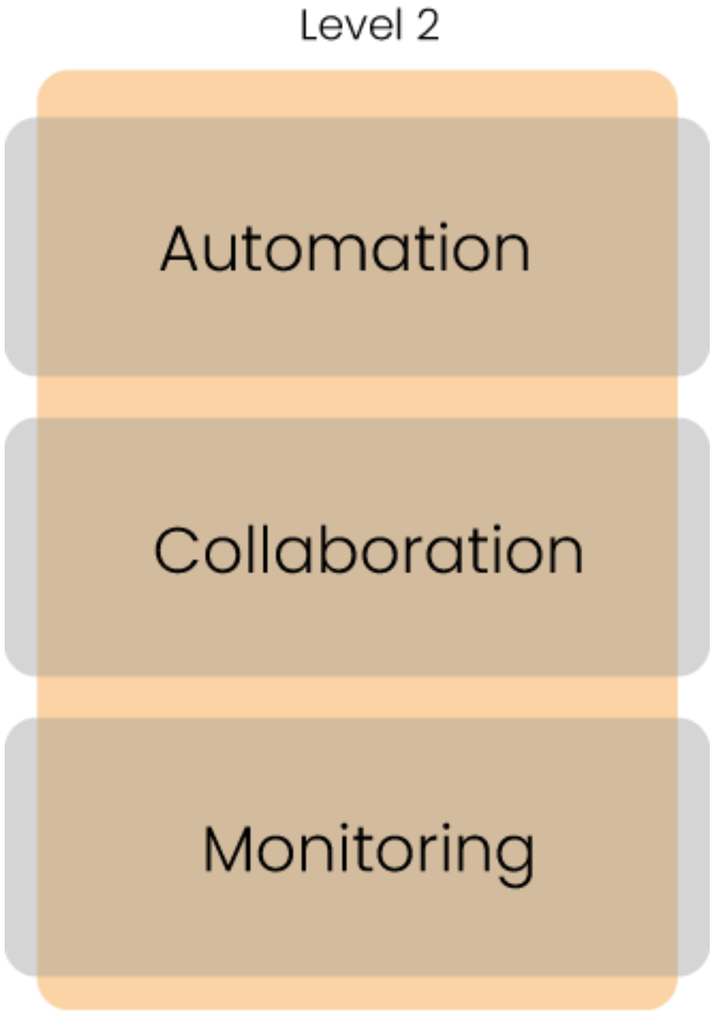
	Level 1	Level 2	Level 3
Automation	Manual processes	Automated development (CI)	Full automation
Collaboration	Distinction machine learning and operations	Collaboration during handover from development	Close collaboration
Monitoring	No monitoring	Development tracking (experiments, feature store)	Full monitoring

Level 1: Manual processes



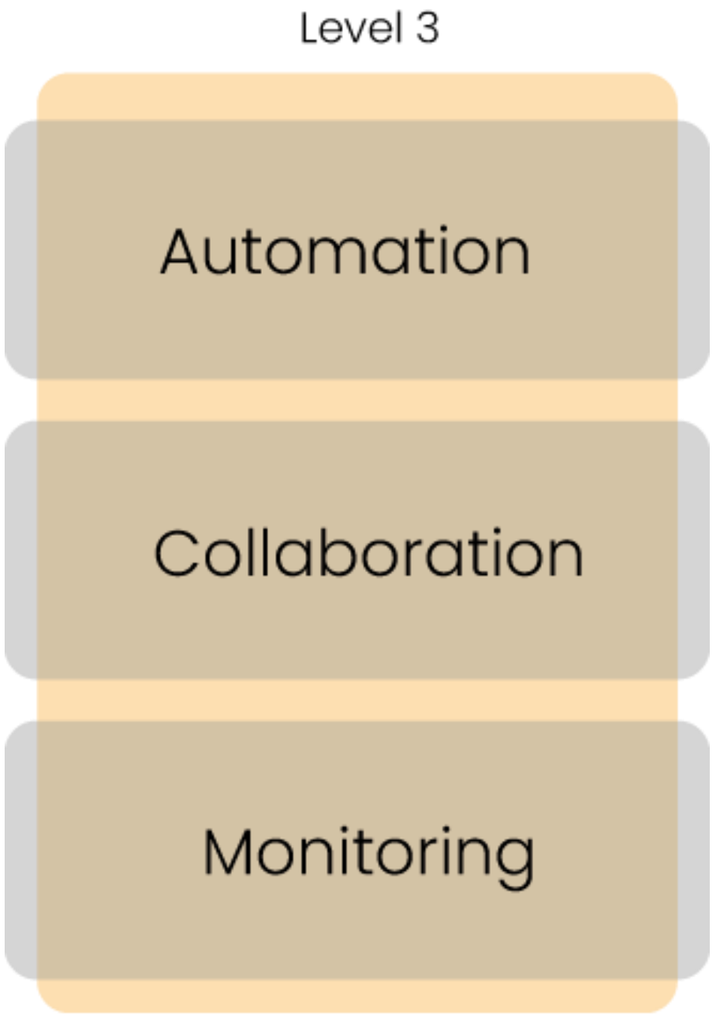
- Manual process for development
- Manual process for deployment
- No collaboration between ML and operations
- Teams work in isolation
- No tracking of development
- No monitoring after deployment

Level 2: Automated development



- Automated development pipeline (Continuous integration)
- Manual process for deployment
- After development teams will collaborate to deploy model
- Tracking of ML experiments and features
- Little monitoring after deployment

Level 3: Automated development and deployment

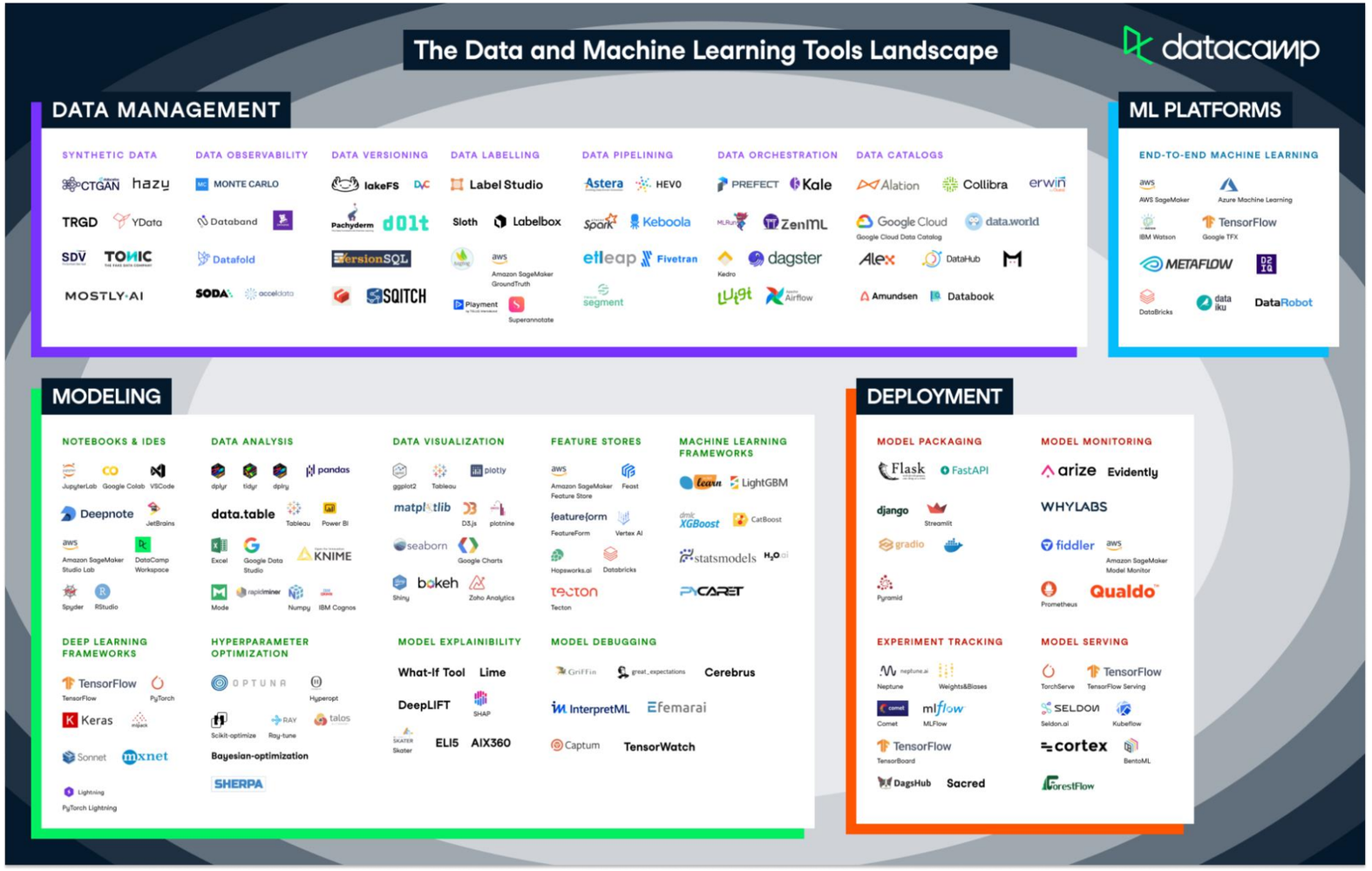


- Automated development pipeline (CI)
- Automated deployment pipeline (CD)
- Close collaboration between teams
- Monitoring of development and deployment
- Potentially automatically triggering retraining

MLOps tools

MLOPS CONCEPTS

MLOps tools



[1] https://www.datacamp.com/blog/infographic-data-and-machine-learning-tools-landscape

Feature store

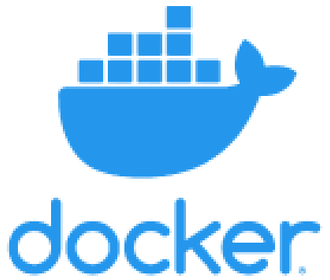
- Both open-source
- **Feast**: self-managed
- **Hopsworks**: part of larger platform

Experiment tracking

- **MLFlow and ClearML**: full machine learning lifecycle tools
- **Weights and Biases**: tracking and visualizing experiments

Containerization

- **Docker**: containerizing applications
- **Kubernetes**: running containerized applications
- **Cloud providers**: provides Kubernetes-like services



Amazon EKS

Azure Kubernetes Service (AKS)

Google Kubernetes Engine

CI/CD pipeline

- **Jenkins**: open-source continuous integration tool
- **GitLab**: code sharing and version control through repositories

Jenkins

GitLab

Monitoring

- **Fiddler**: machine learning model monitoring
- **Great expectations**: data monitoring

MLOps platforms

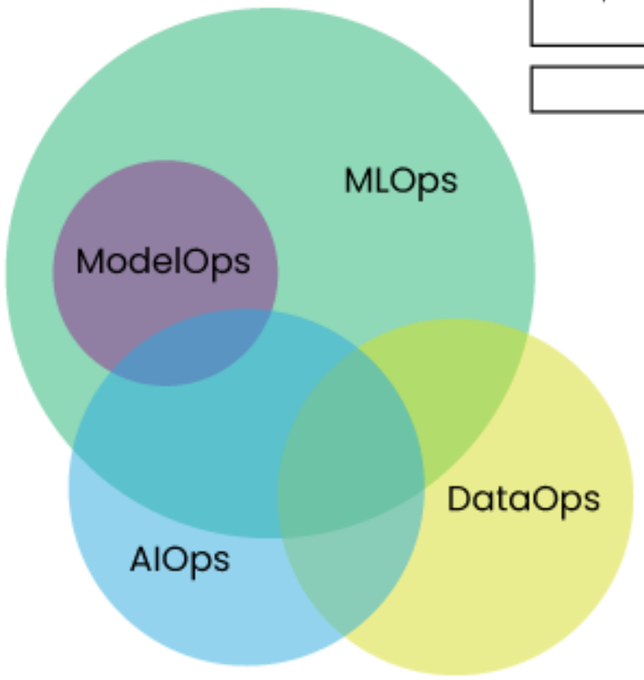
Tools for full machine learning lifecycle

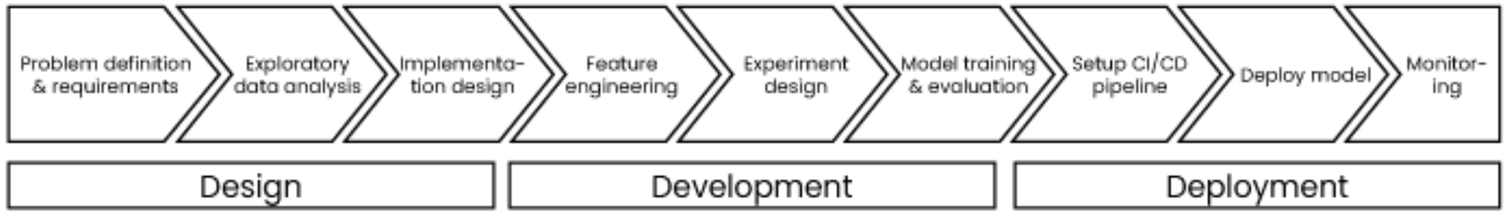
- **AWS Sagemaker**
- **Azure Machine Learning**
- **Google Cloud AI platform**

Recap: MLOps concepts

MLOPS CONCEPTS

What is MLOps?



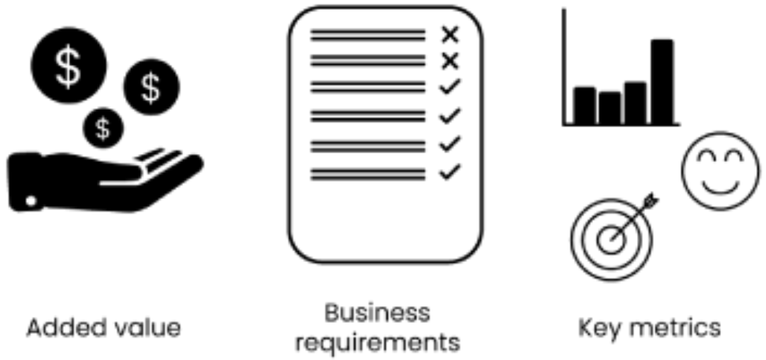


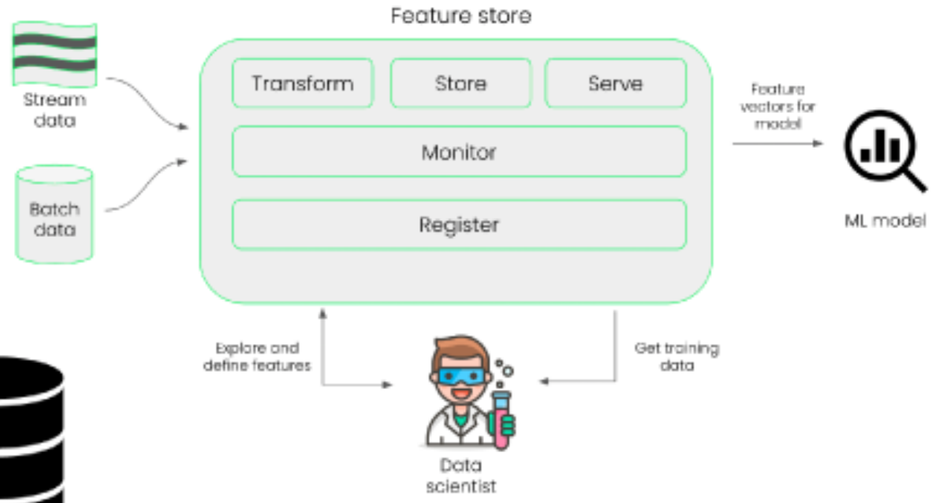
Data scientist

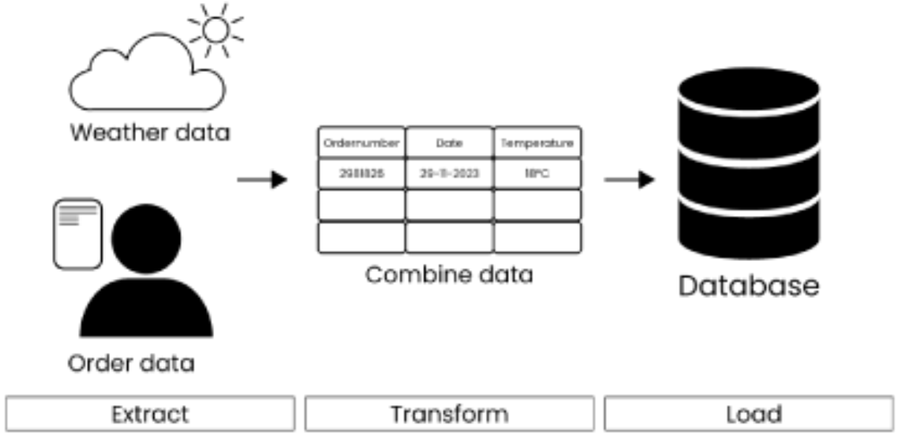
Data engineer

Business stakeholder

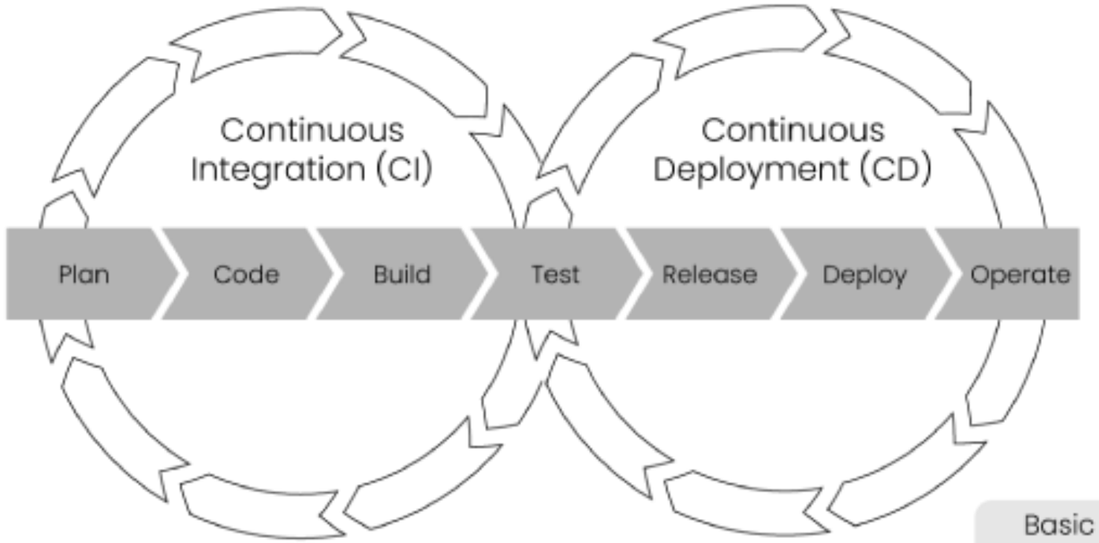
Design and development

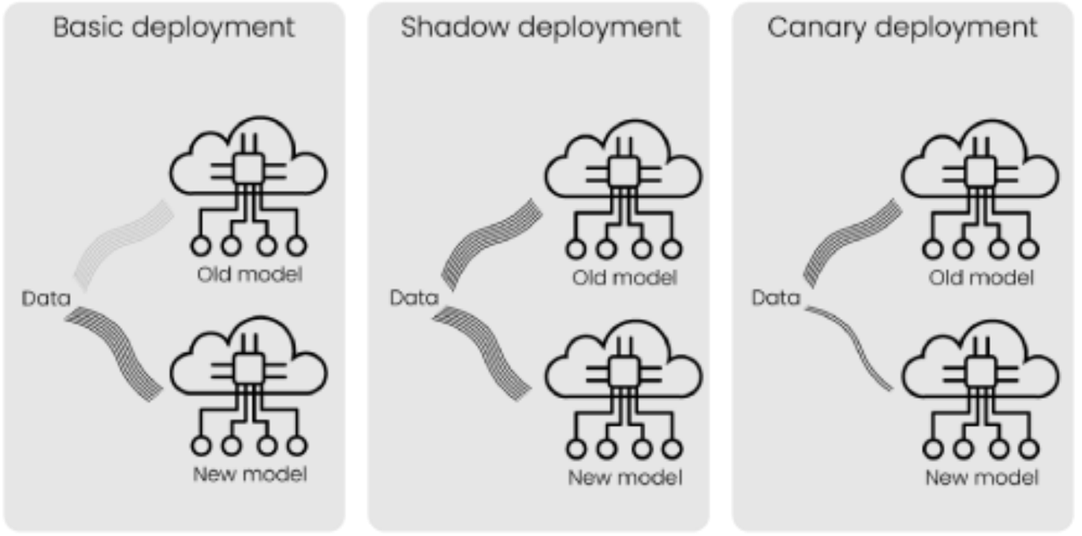






Deployment


Continuous Integration (CI)
Continuous Deployment (CD)
Plan
Code
Build
Test
Release
Deploy
Operate


Basic deployment
Data
Old model
New model
Shadow deployment
Data
Old model
New model
Canary deployment
Data
Old model
New model

Maintaining machine learning

	Level 1	Level 2	Level 3
Automation	Manual processes	Automated development (CI)	Full automation
Collaboration	Distinction machine learning and operations	Collaboration during handover from development	Close collaboration
Monitoring	No monitoring	Development tracking (experiments, feature store)	Full monitoring