

DATA ENGINEERING

Building the infrastructure for a large-scale application using DataOps and micro-services



January 17, 2024

SUBJECT: Data Engineering

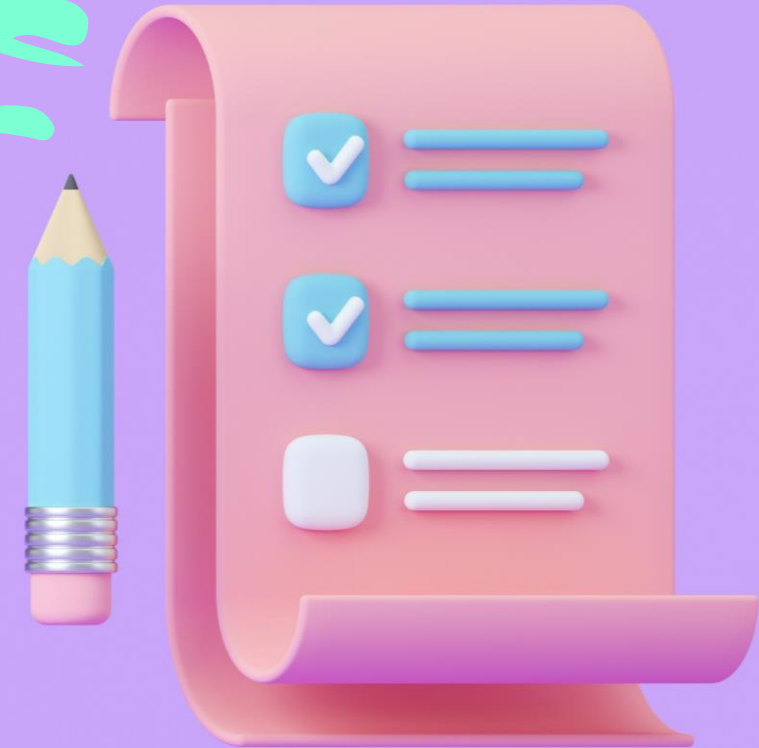
STUDENT NAME: DADA NANJESHA GOUDA SHANBOG

COURSE ID: DLMDSEDE01

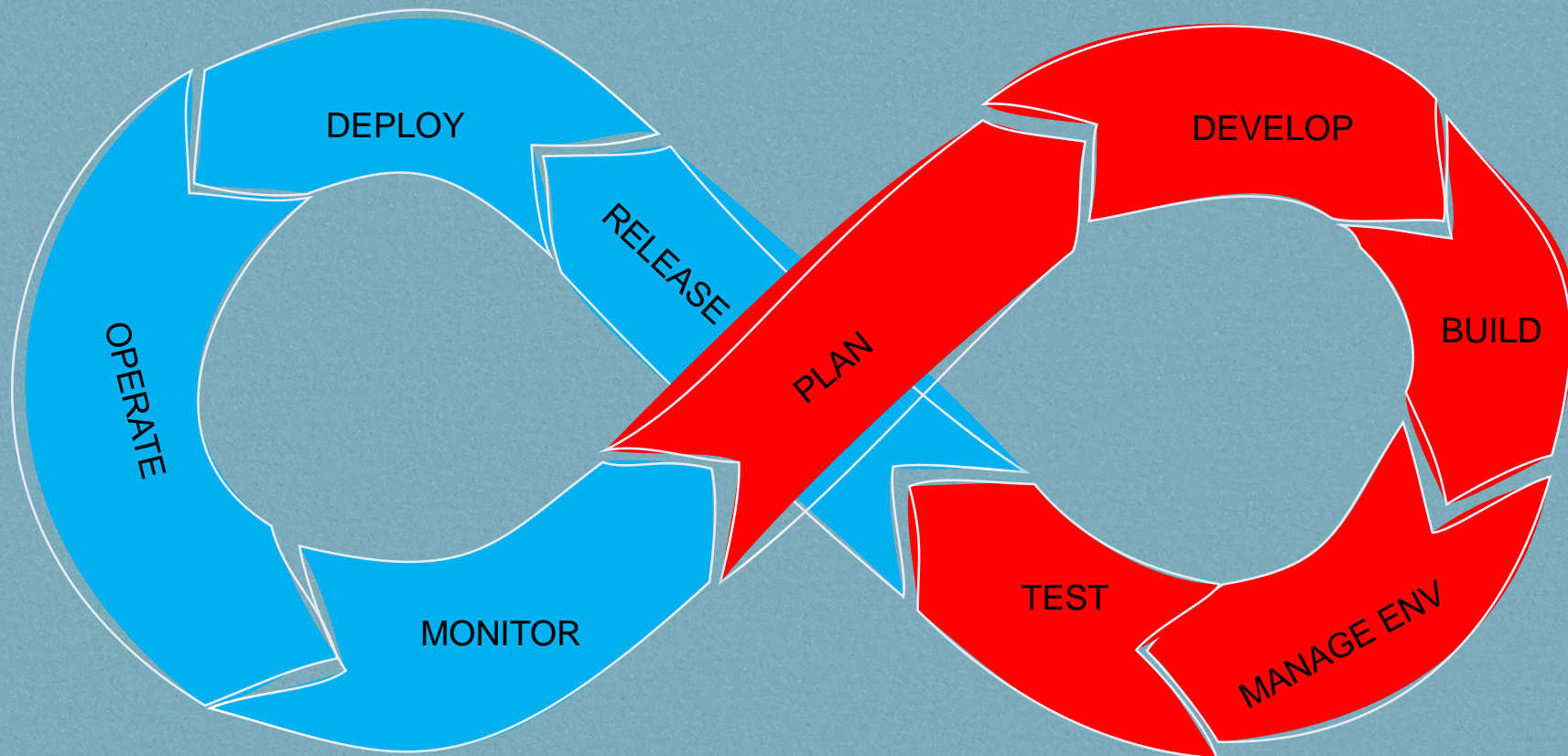
MATRICULATION NUMBER: 10220129

Agenda:

- Introduction
- Main principles of DataOps.
- Practical application to the use case
- Advantages and disadvantages of different application frameworks
- Comparison of DevOps/DataOps platforms



Introduction: DataOps



DataOps Principles:

Highly automated

Open-source

Take advantage of best-of-breed tools

Have layered interfaces

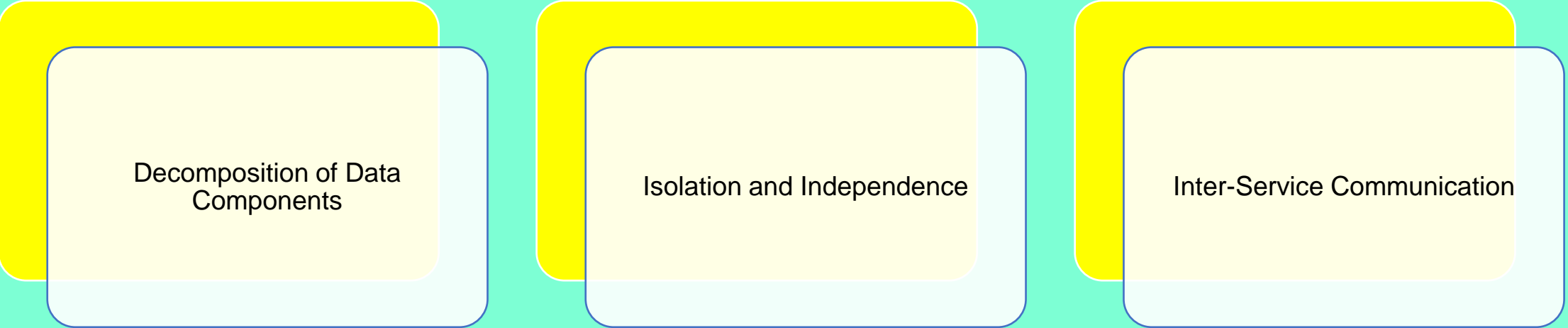
Track data lineage

Feature deterministic, probabilistic, and humanistic data integration

Combine both aggregated and federated methods of storage and access

Process data in both batch and streaming modes

DataOps with respect to Microservice:



Decomposition of Data
Components

Isolation and Independence

Inter-Service Communication

DataOps with respect to Infrastructure as Code (IaC):



Consistent Environment
Management



Scalability and Flexibility



Version Control for Infrastructure

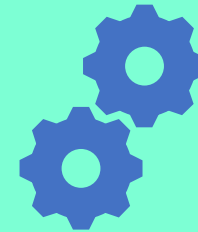
DataOps with respect to Continuous Integration/Continuous Deployment (CI/CD):



Automated Data Pipelines



Rapid Iterations and Feedback



Deployment Automation

Transferring DataOps Concepts to Data-Intensive Applications:



DATA PROCESSING
MICROSERVICES



IaC FOR DATA INFRASTRUCTURE



CI/CD FOR DATA WORKFLOWS

DataOps principles for Continuous Integration/Continuous Deployment (CI/CD):



Source Code Management



Automated Testing



CI/CD Pipeline



Artifact Repository



Automated Deployment

DataOps principles for Infrastructure as Code (IaC):



Define Infrastructure
Components

Environment
Consistency

Automated Provisioning

Version Control for IaC

DataOps principles for Microservices (MS) Architecture:

Identify Microservices

Decompose the Monolith

APIs for Communication

Isolation and Independence

Security Measures

Advantages and disadvantages of different application frameworks:

Spring Boot

Advantages:

- Microservices Support
- Built-in CI/CD Integration
- Strong Community and Documentation

Disadvantages:

- Being Java-Based

Django

Advantages:

- Rapid Development
- RESTful Microservices Support
- Well-Suited for Python Developers

Disadvantages:

- Less Modular

Compare between DevOps/DataOps platforms:

Jenkins

Advantages:

- Extensive Plugin Support
- Versatility and Wide Adoption
- Good Support for Infrastructure as Code(IaC)

Disadvantages:

- Complex Configuration

GitLab CI/CD

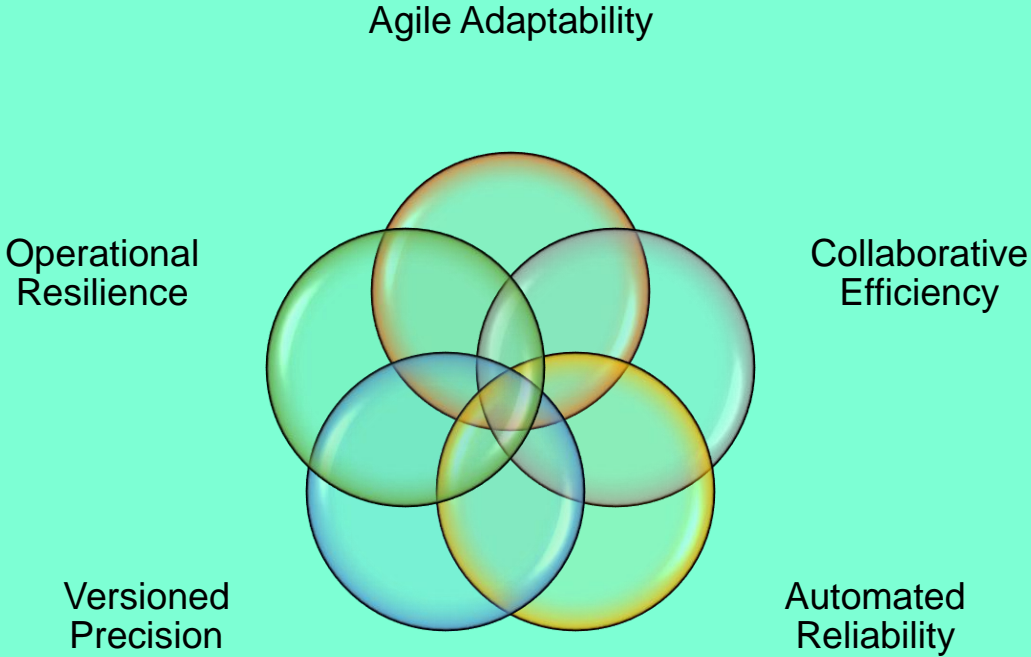
Advantages:

- Integrated Source Code Repository and CI/CD
- Simplified Configuration
- Good Support for Containerization

Disadvantages:

- Learning Curve

Conclusion: Implementing DataOps in projects yields transformative benefits:





Bibliography :

- Kleppmann, M. (2017). *Designing data-intensive applications: The big ideas behind reliable, scalable, and maintainable systems*. O'Reilly.
- Farcic, V. (2016). *The DevOps 2.0 toolkit: Automating the continuous deployment pipeline with containerized microservices*. CreateSpace Independent Publishing Platform.
- *What is DataOps? how does it work?*. Qlik. (n.d.). <https://www.qlik.com/us/dataops>

Thank you!