

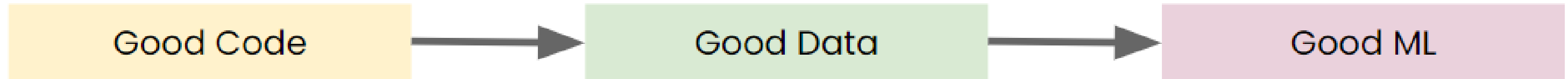
Ensuring Data Quality Standards with DevOps

INTRODUCTION TO DEVOPS



Cem Sakarya
DevOps Risk Advisor

DevOps ensures good code



- Most software handle data
- Good software drives high-quality data
- DevOps helps maintain high-quality in software

Data quality

- How trusted is the information?
- Most recent and accurate information
- Can be hard and costly to achieve
- Not all data requires the same quality
- There are many elements to Data Quality

Elements of data quality



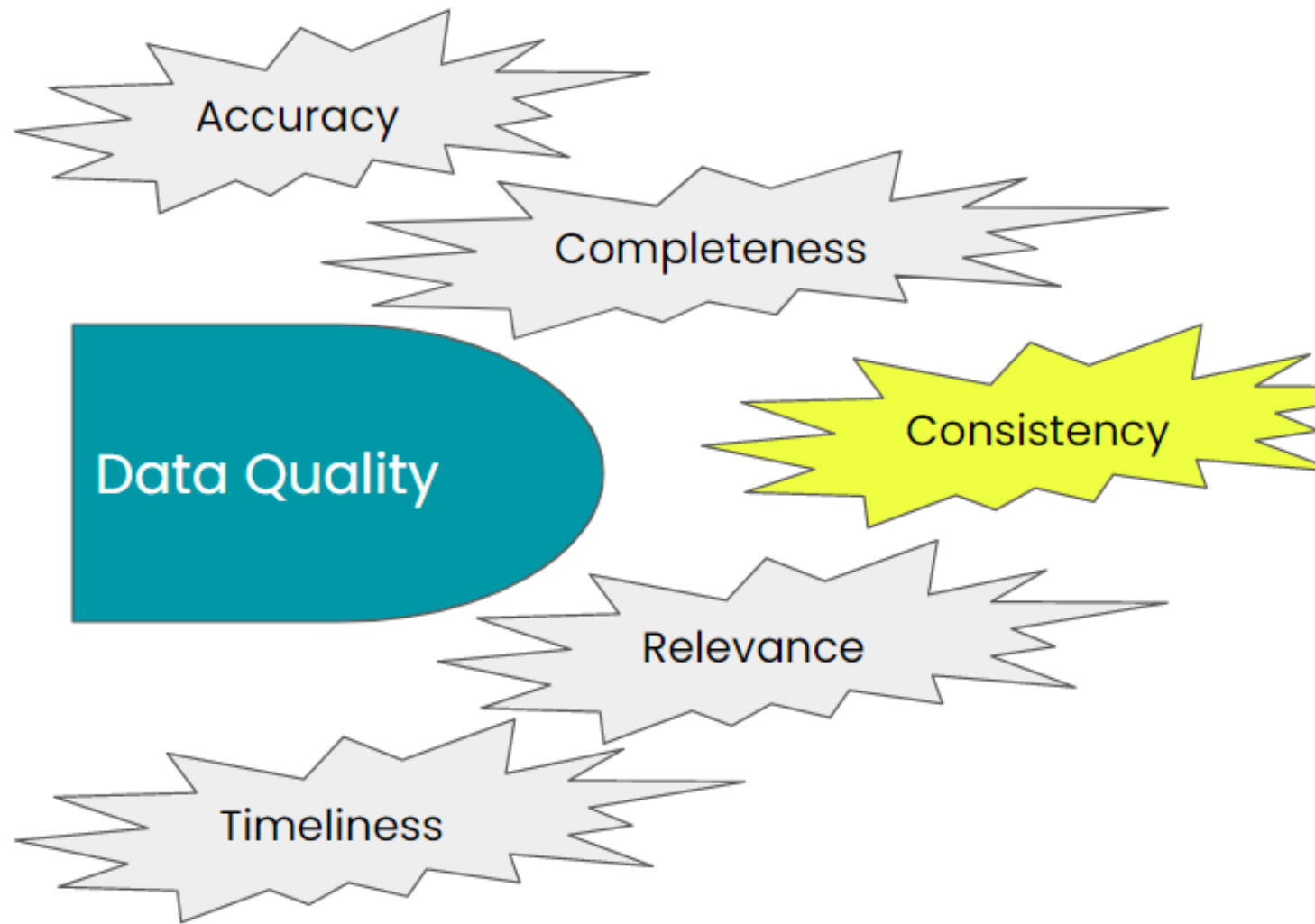
- Elements breaks down and define Data Quality
- They should be traced and standardized



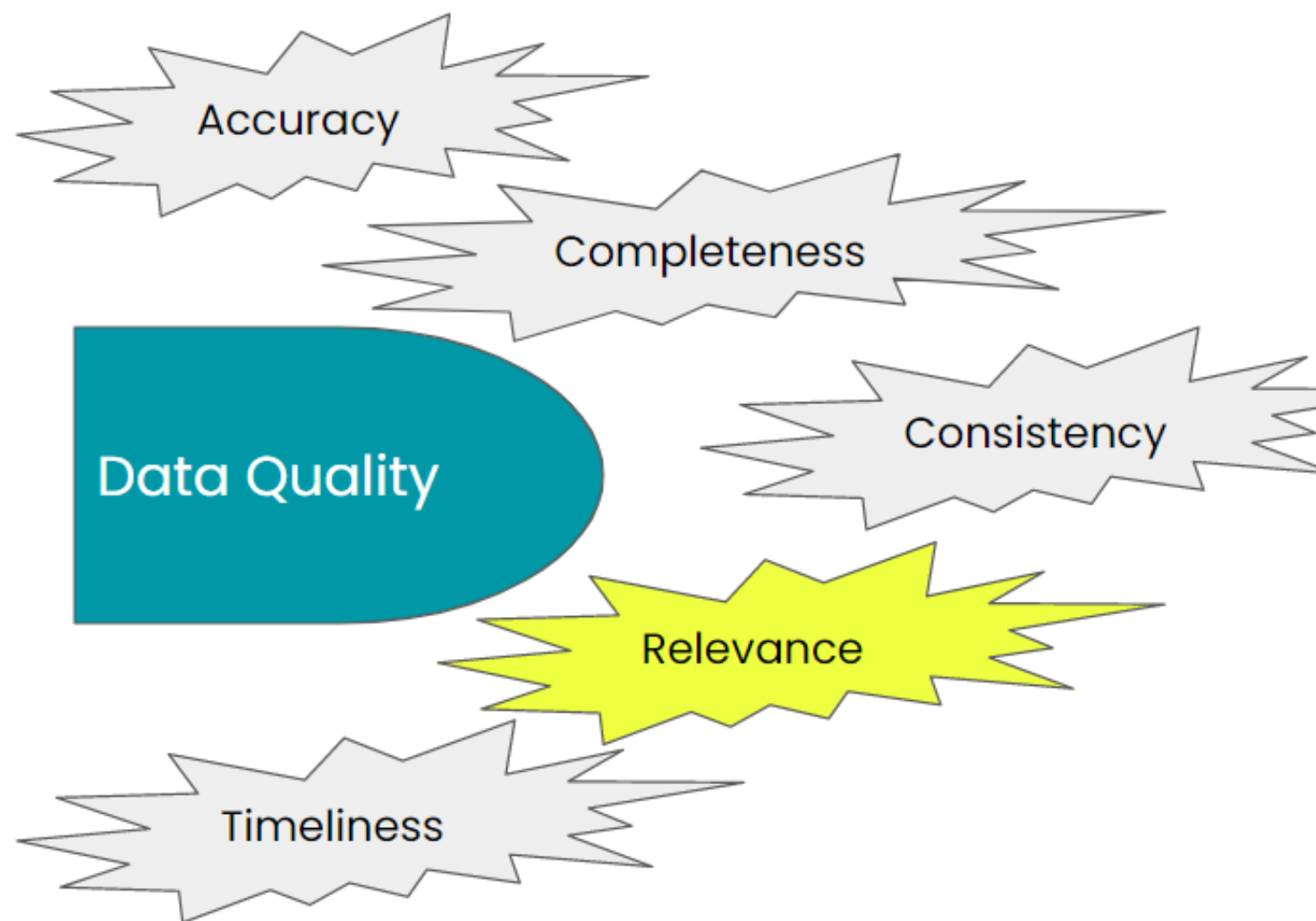
- Is the data correct in all details?
- Should be closely monitored



- How comprehensive is the data?
- Check if data is lost when it is stored in a database
- Check if data is lost when it is moved to different databases



- Consistency is about the reliability of the information
- It is also referred to as Data Integrity
- Should be closely monitored and tested



- Keeping and handling only the relevant information
- Irrelevant information wastes time and resources



- How up to date is the information?
- Old data could be replaced by more recent information
- Raw data within microservices is not used in data analysis
- Microservices talk to each other via API calls to get the most recent information
- The data within microservices should be replicated at an acceptable frequency

Let's practice!

INTRODUCTION TO DEVOPS

Observability and Reliability

INTRODUCTION TO DEVOPS



Cem Sakarya
DevOps Risk Advisor

Observability

Observing inner components of software

Components might consist of:

- Microservices
- APIs
- Databases
- Other components

Check they are working correctly.



¹ <https://www.istockphoto.com/nl/vector/het-lijnpictogram-van-de-verrekijker-oceanconcept-verrekijkerteken-op-witte-gm1245624425-363092800?phrase=field%20glass>

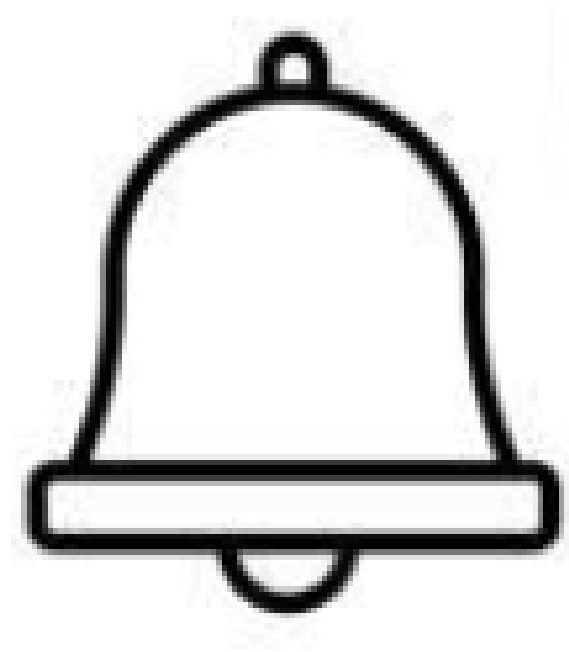
Why is observability needed?

- Many different parts tied to each other
- Spot specific problems and their root cause
- Monitoring systems effectively



Alerting

- Alerts help detect problems early
- Set alerts if software fails to run
- Setting up alerts enable teams to react fast when problems happen



¹ <https://www.istockphoto.com/nl/vector/notification-bells-icon-gm1364077374-435399123?phrase=alerting>

Why is reliability important?

Examples:

- Government services
- Defense systems
- Banking software

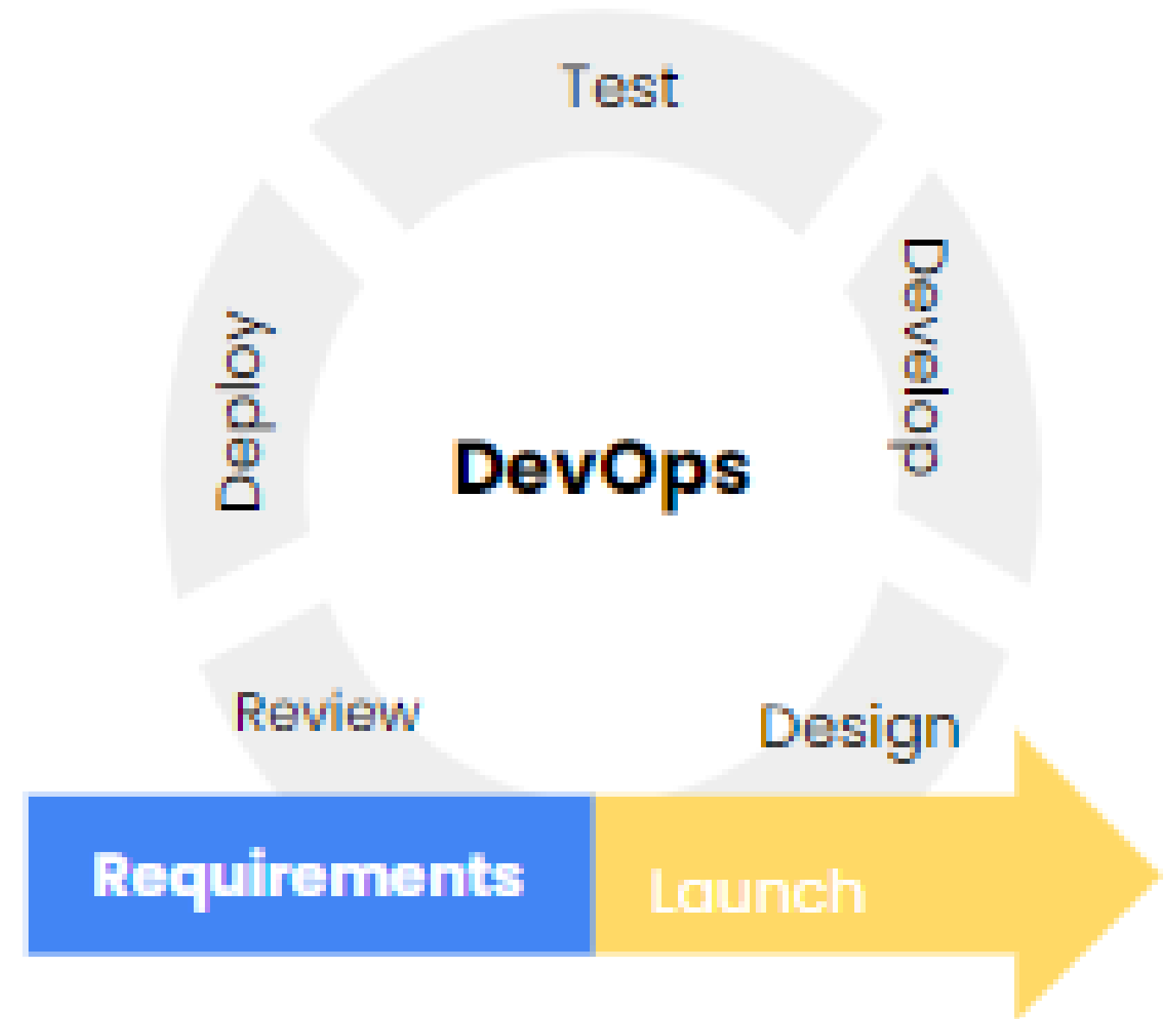
Many other software also require high reliability

How to increase reliability?

Two main principles:

- Testing
- Observability

DevOps automates testing, and enforces observability.



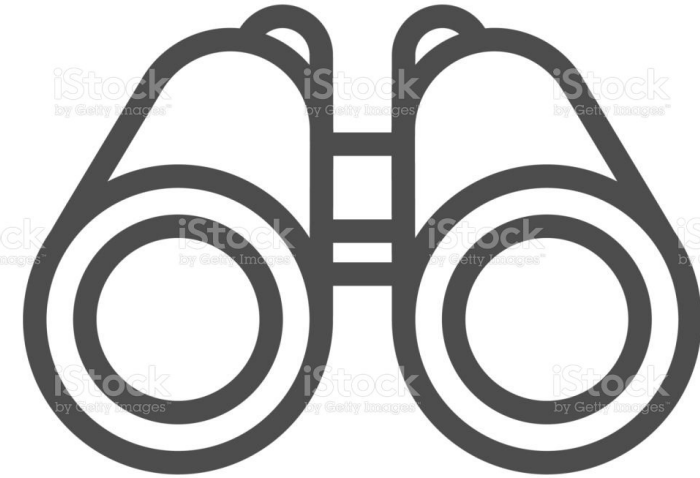
Testing



Testing is used to detect the problems before they go to market.

DevOps ensures automated testing.

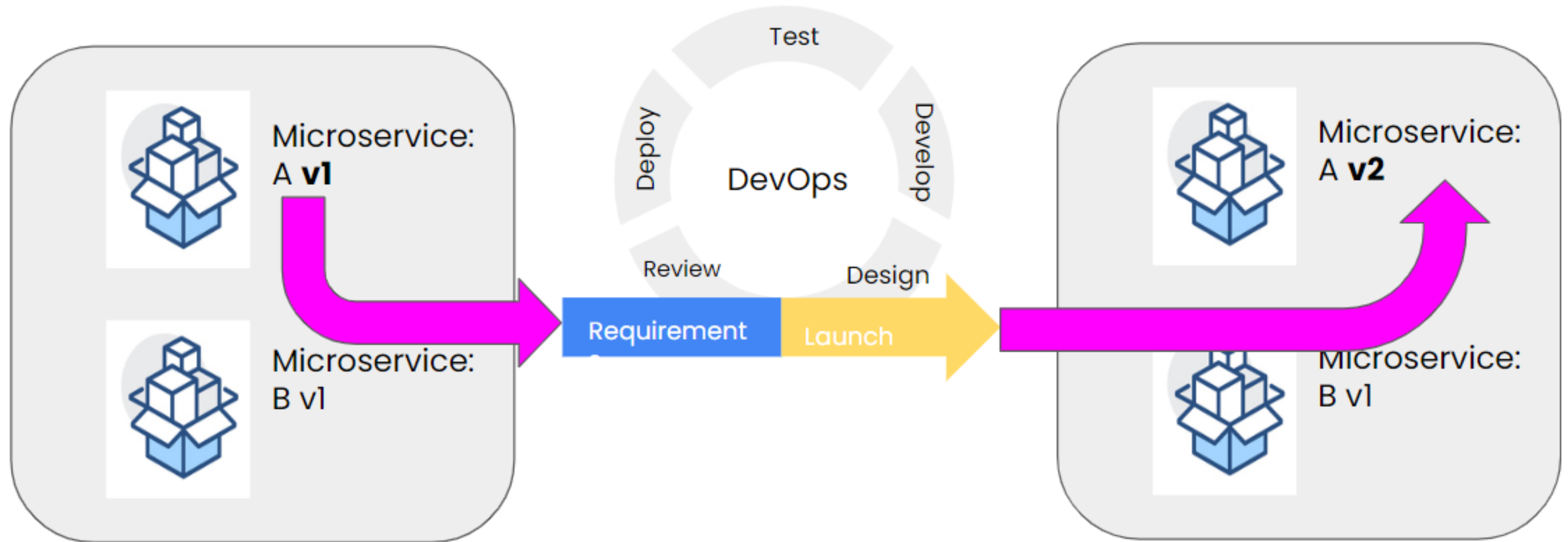
Observability



Observability is used to detect the problems as they happen in user interaction.

Observability is needed even when the software is tested enough.

Reliability in microservices



Let's practice!

INTRODUCTION TO DEVOPS

DevOps Cultural Concepts

INTRODUCTION TO DEVOPS



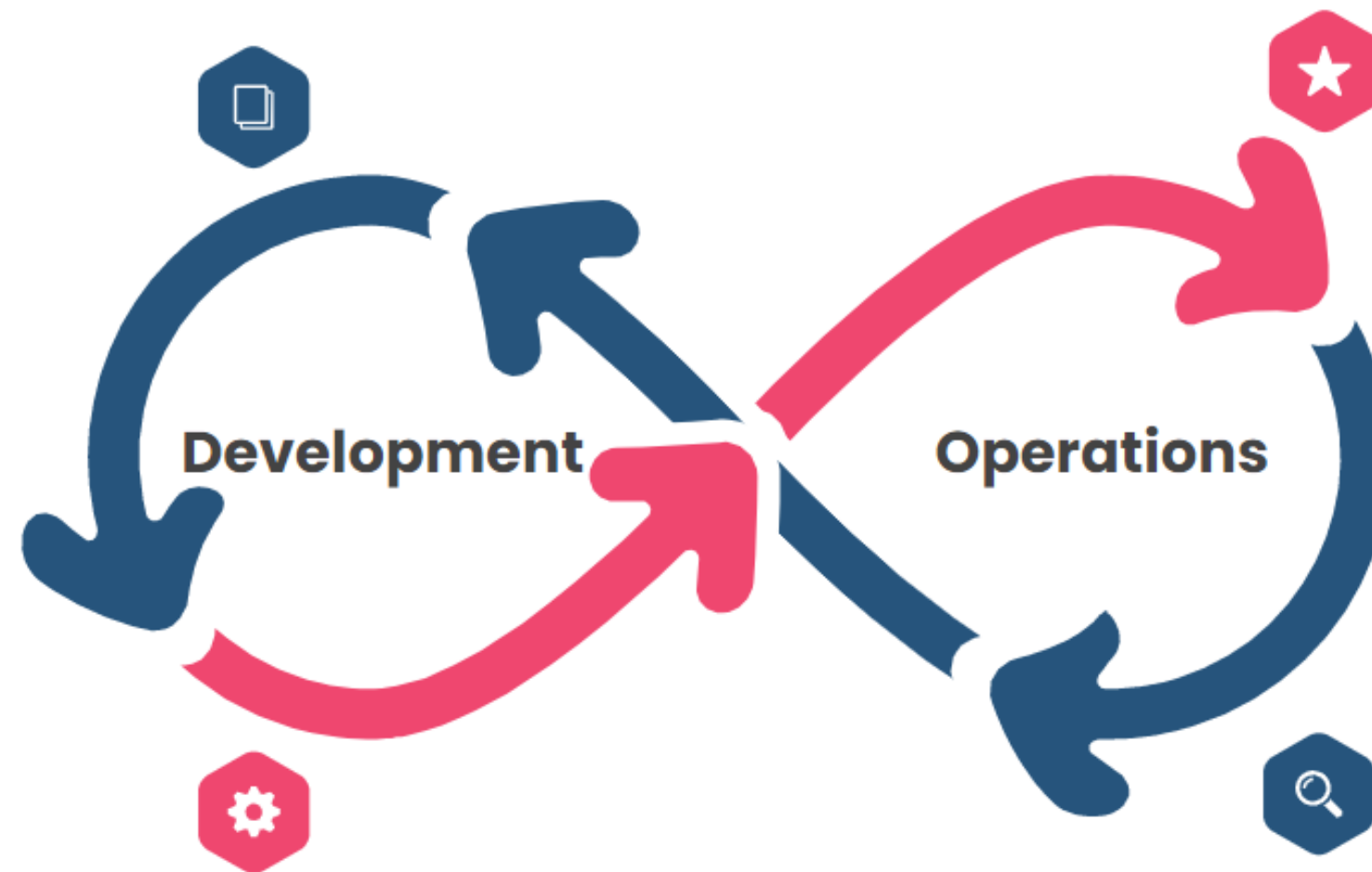
Cem Sakarya
DevOps Risk Advisor



Concepts

- Collaboration
- Autonomous Teams
- Shared Responsibilities
- Automation
- Post-Mortem

Collaboration



- Heart of DevOps
- Collaboration between Development and Operations

Autonomous teams

- Teams are diverse
- Self sufficient teams
- Teams have members who specialized in:
- Product Engineering
- Operations
- Data Engineering, etc.

Shared responsibilities

- Autonomous teams take care of specific parts of the software
- They are responsible for developing and running
- You build it, you run it

Automation

- CI/CD pipelines helps automate change management, testing, and deployment
- Automation is a time-saver
- Developers focus on creativity, not the simple tasks
- Without CI/CD pipelines, testing and deployment must be done manually

Post-mortem

All software is prone to errors

Something can always go wrong

When a software problem is identified:

- First solve it
- Then review what went wrong

This review is called Post-Mortem

No one is blamed in the Post-Mortem



How this issue could have been prevented?

Common Answers

- More testing
- More code review

BLAMELESS

How this issue could have been better handled?

More observability helps with:

- Earlier detection of issues
- Earlier detection of the root cause

BLAMELESS

Let's practice!

INTRODUCTION TO DEVOPS

Tying it all together

INTRODUCTION TO DEVOPS



Cem Sakarya
DevOps Risk Advisor

What is DevOps after all?

DevOps is a combination of:

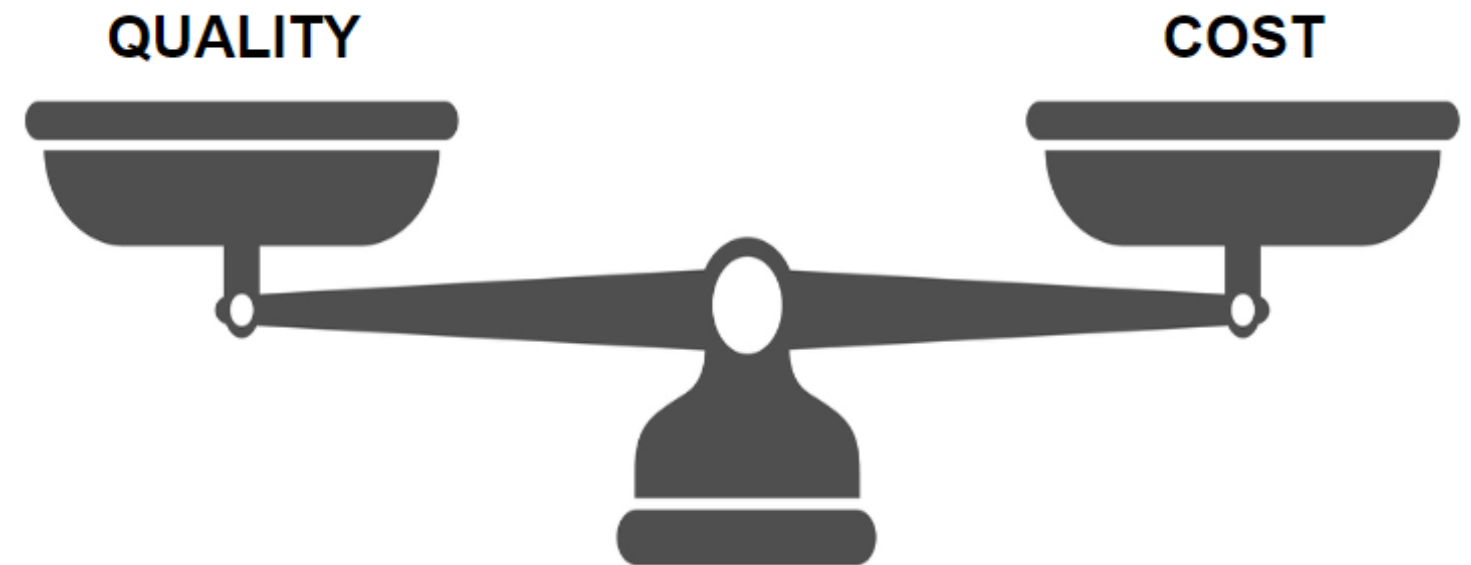
- tools
- technologies
- cultural elements

Software Development -> DevOps

Data Engineering -> DataOps

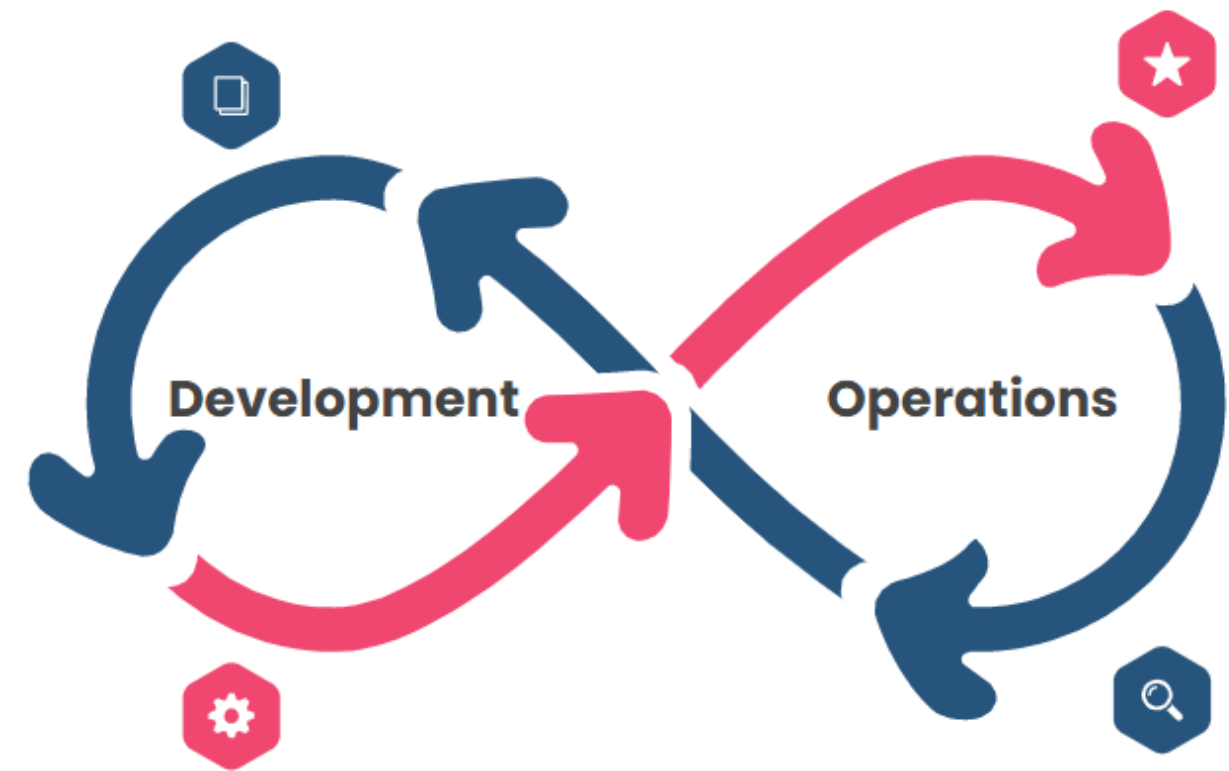
Machine Learning Engineering -> MLOps

It helps ensure high productivity, high quality, and managed costs.



What is DevOps about?

- Reorganizes into autonomous teams
- Collaboration of teammates with different backgrounds
- Teams being responsible for different parts of software
- Automating change management
- Testing
- Experimentation



Why is DevOps necessary?

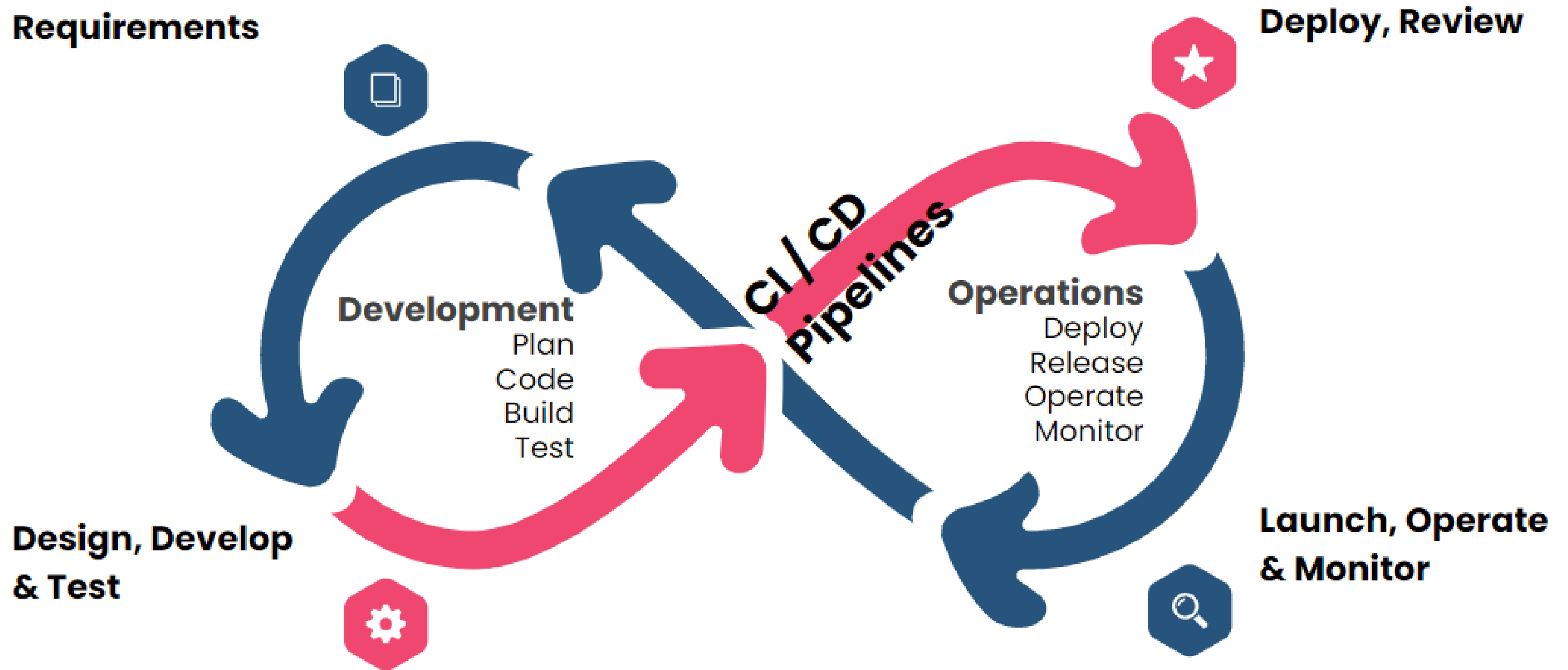
- It is very efficient compared to other systems.
- Automation and collaborations boost productivity
- Testing and observability is necessary for reliability
- Experimentation helps with product quality

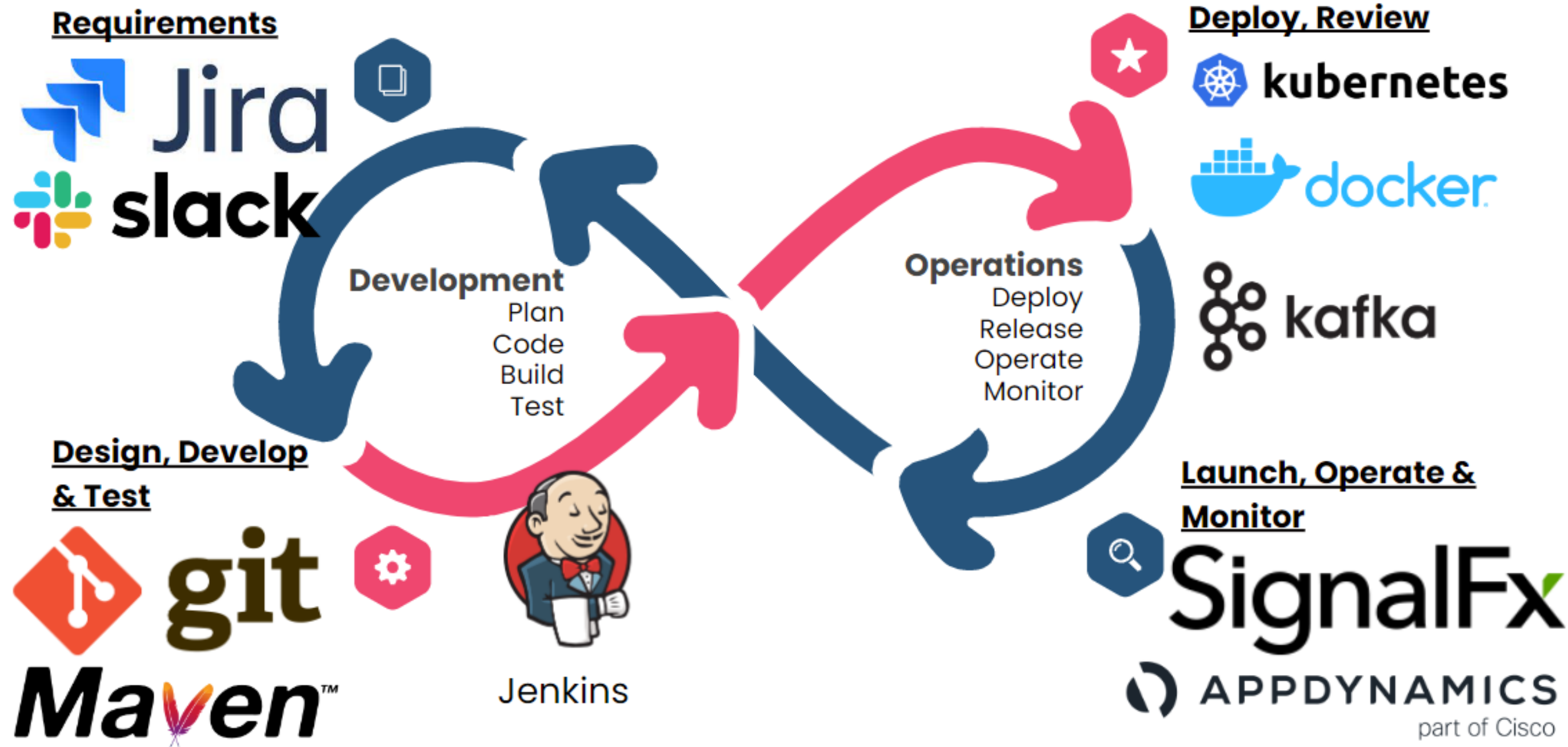
When is DevOps applicable?

- DevOps principles are applicable most of the time.
- It is necessary for large organizations.



What are the components of DevOps?





Let's practice!

INTRODUCTION TO DEVOPS

CONGRATS!

INTRODUCTION TO DEVOPS



Cem Sakarya
DevOps Risk Advisor

What did we learn?

1. Introducing ourselves to DevOps and its main components
2. Looking deeper into modern software architecture and infrastructure
3. Exploring the data engineering applications of DevOps and DevOps reporting
4. Ensuring reliability and data quality



You can use your skills to...

- navigate through software development
- for software design, development, and testing
- apply data engineering implications of DevOps

Time to celebrate!

INTRODUCTION TO DEVOPS