# Release Management

## What is it

- A core component of DevOps, release management is the process of controlling the deployment of software into its execution environments. Release management ensures that software changes are delivered to users in a reliable manner.

## Key Benefits / Why is this important

- In most organisations there are multiple execution environments, including testing and live production. Release Management ensures that code is promoted through these environments in a controlled way. Example: Developer Test Internal QA External QA Live
- Release management requires:
  - Version control system. Branches or tags are created to mark code that's ready to be deployed. Semantic versioning is often used in this process.
  - CI/CD pipelines automate code compilation, testing and feature toggling.
  - Infrastructure as Code. This ensure the delivery environment is configured correctly to receive the code.
- Effective release management includes planning for failures. Automated rollback strategies allow for quick reversal of changes where failures are detected. One such strategy is blue/green deployment.
- Blue/green deployment uses two separate live environments (slots) where one is serving user traffic and the other is inactive and can accept new versions of the software. When new versions are made live, rolling back is as simple as swapping back to the old slot.
- Auditing capabilities, often built into release management tools, enable organizations to track who deployed what and when, aiding in both compliance and debugging.

## Techniques supporting this practice

- Source Control
- CI/CD
- Automated testing
- DevOps
- Semantic Versioning
- Blue/Green deployment
- Auditing

---