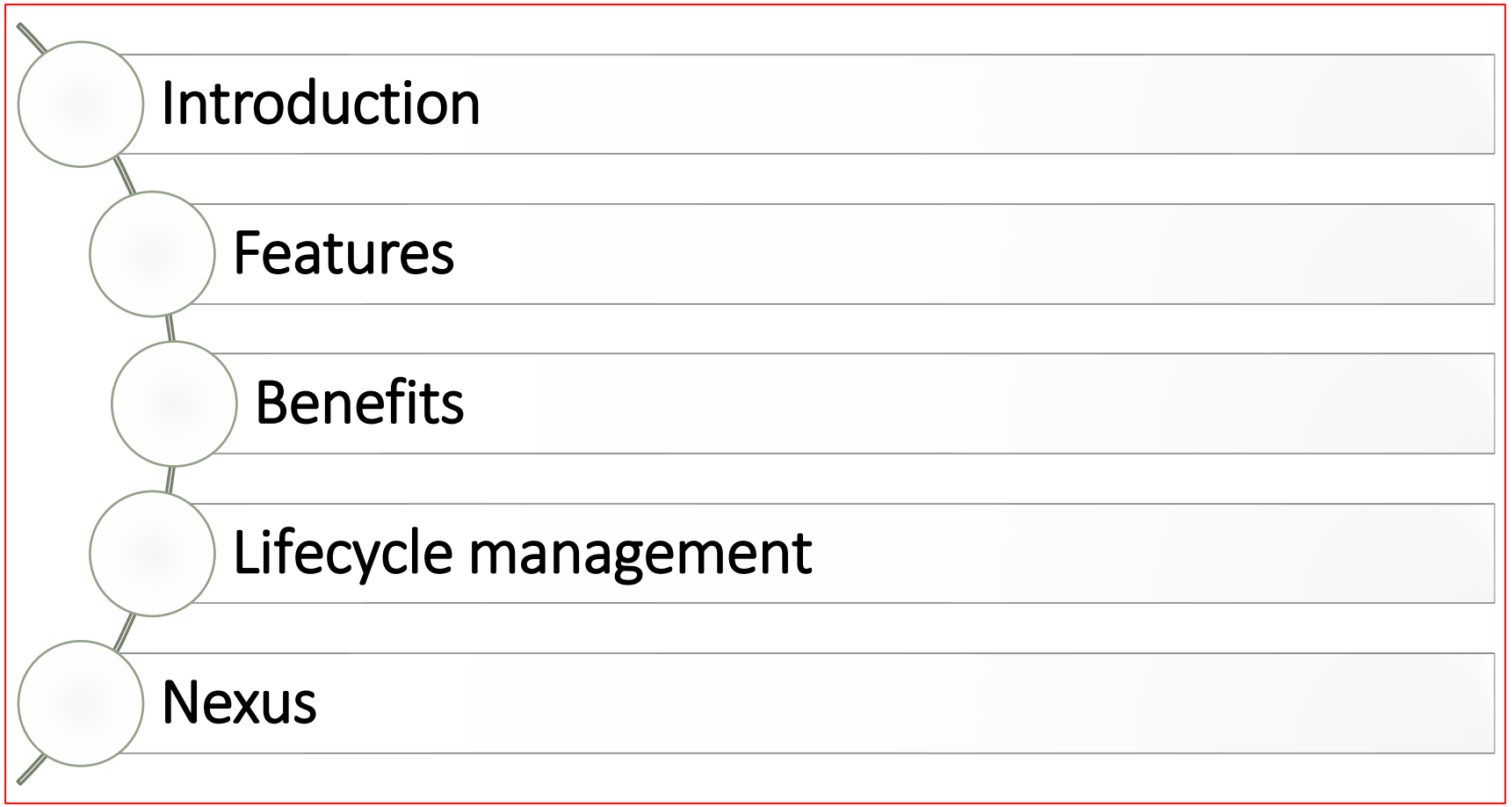
iPMAC
NET YOUR WORK

DEVSECOPS COURSE

ARTEFACT MANAGEMENT

TRAINER: TRAN HUU HOA

AGENDA


Introduction
Features
Benefits
Lifecycle management
Nexus

INTRODUCTION

Artifact management refers to the process of storing, versioning, and tracking binary files or “artifacts” produced during software development.

Having an effective artifact management strategy is crucial for scalability, efficiency, and collaboration in software development pipelines. It ensures that organizations can manage and control the artifacts used across their projects while speeding up software delivery

INTRODUCTION

Artifacts can take various forms, including:

- Source code
- Compiled code
- Libraries and dependencies
- Configuration files
- Documentation
- Executable files
- Database schemas and scripts

FEATURES

- Repository management
- Versioning
- Dependency management
- Security and access control
- Artifact life cycle management
- Caching and proxying
- Integration with CI/CD pipelines
- Auditing and logging
- Retention policies

BENEFITS

- Reproducibility: By storing and versioning artifacts, you can recreate builds consistently. This is crucial for debugging, testing, and ensuring that production deployments match development environments.
- Collaboration: Teams can share artifacts across projects, reducing duplication of effort. Developers can easily access libraries, dependencies, and other resources.
- Efficiency: Automated deployment and retrieval streamline workflows. Developers spend less time searching for dependencies or resolving conflicts.
- Security: Centralized repositories allow you to control access and scan artifacts for vulnerabilities. You can enforce policies and ensure only approved artifacts are used.
- Scalability: As your projects grow, artifact management ensures scalability. It handles large volumes of binaries efficiently.
- Auditability: Detailed metadata and version history provide transparency. You can track who published which artifact and when.

LIFE CYCLE MANAGEMENT

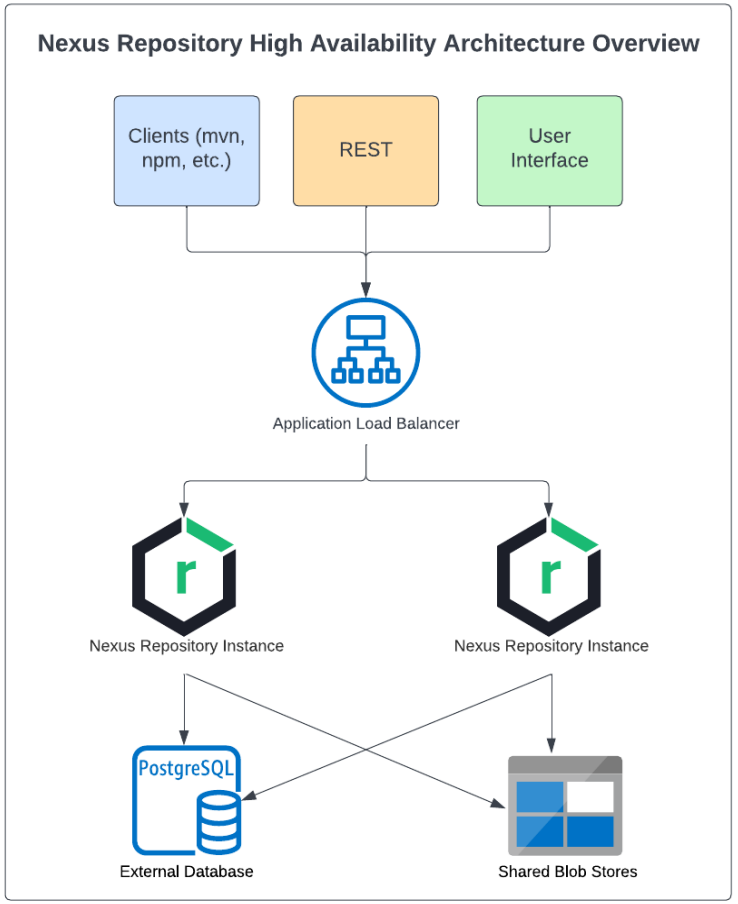
Artefact life cycle management refers to the process of handling and organizing various artifacts (such as compiled code, libraries, documentation, and other files) produced during the software development life cycle. The goal is to ensure that these artifacts are managed efficiently from initial development through to testing, staging, and ultimately production deployment

NEXUS

Nexus Repository by Sonatype is a powerful artifact management tool. It helps developers store, organize, and distribute artifacts (such as binaries, build artifacts, and Docker containers) needed for development

- **Centralized Storage**: Nexus Repository provides a single source of truth for all components, making it easier to manage and distribute artifacts.
- **Build Performance**: It optimizes build performance and storage costs by caching artifacts.
- **Support for Multiple Formats**: Nexus supports up to 18 package formats, including Maven, npm, Docker, and more.
- **High Availability**: It offers highly available clusters, edge nodes, and test servers without per-node charges.
- **Integration**: Nexus integrates seamlessly with popular CI/CD tools like Jenkins, CircleCI, and Maven.

NEXUS


Nexus Repository High Availability Architecture Overview
Clients (mvn, npm, etc.)
REST
User Interface
Application Load Balancer
Nexus Repository Instance
Nexus Repository Instance
PostgreSQL
External Database
Shared Blob Stores