JENKINS SESSION 4

Deployment acceleration in Jenkins

Hrishikesh Mohan

MULTIBRANCH PIPELINES

- Homework discussion
  - Python application
  - Multi-staged pipeline vs a normal pipeline
  - Use cases

SCHEDULED TRIGGERS

Jenkins uses a cron syntax (five fields: minute, hour, day of month, month, day of week) to define build times, similar to Linux’s cron but with unique features for load balancing.

- Open the job’s configuration.
- Scroll to “Build Triggers.”
- Check the “Build periodically” box.
- Enter your schedule expression.
- Every minute: `* * * * *`
- Every hour: `H * * * *`
- Specific weekdays: `H * * * 1,3,5` (Monday, Wednesday, Friday)

PARALLEL STAGES, ENVIRONMENT VARS AND CREDENTIALS

- Parallel stages allow multiple stages to run at the same time, which speeds up build and test processes by handling tasks that don’t depend on each other concurrently.

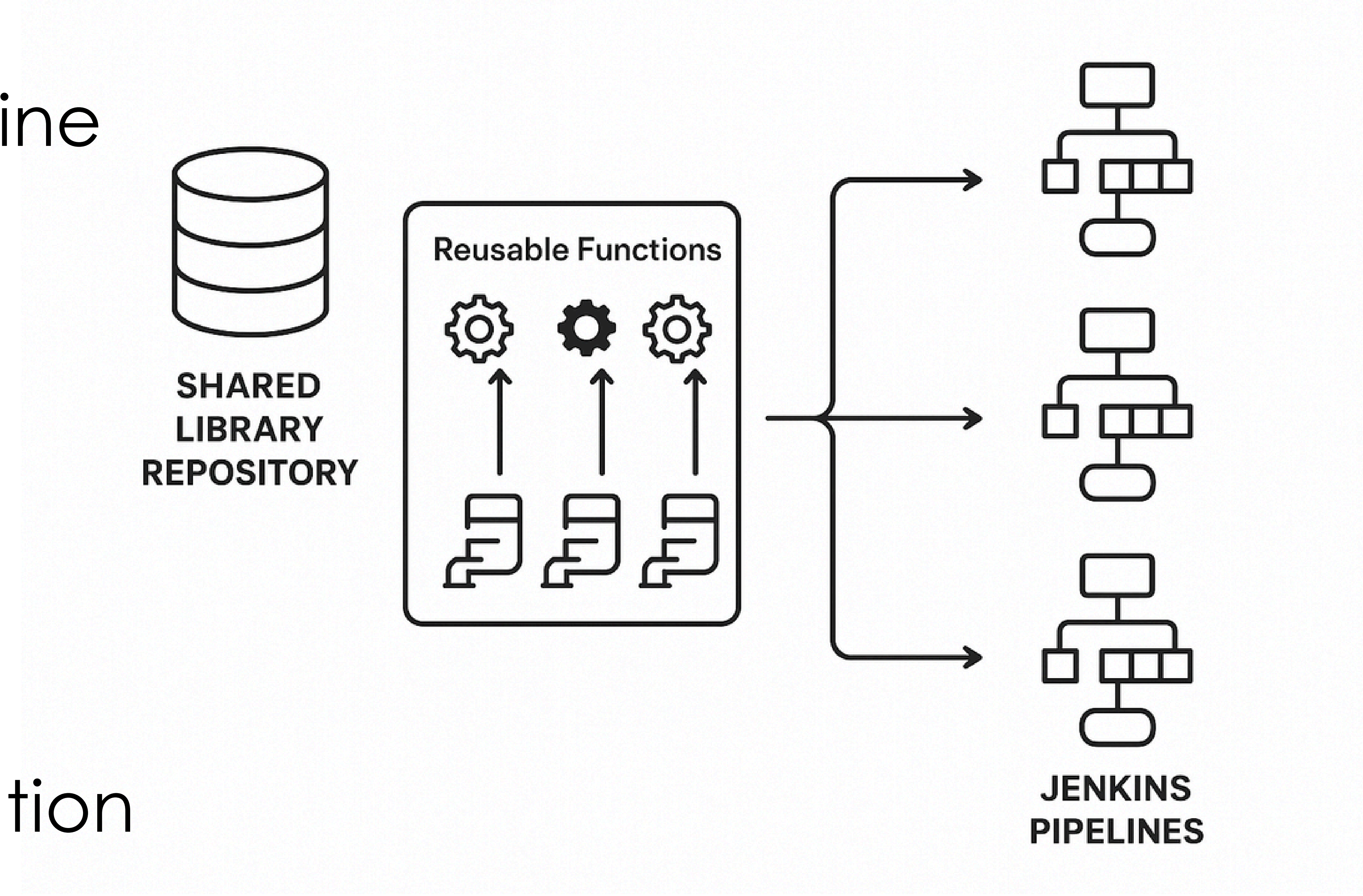
- Environment variables are key-value pairs used to configure pipeline behavior. Jenkins provides built-in environment variables, and users can define their own for application settings, paths, and deployment targets.

- Credentials are securely stored secrets in Jenkins (like passwords, API keys, or AWS keys). They are injected into pipelines and masked in logs, providing secure authentication for deployments and integrations.

```
pipeline {
    agent any
    environment {
        APP_NAME = 'demo-parallel-stages'
    }
    stages {
        stage('Build') {
            steps {
                       echo 'Building application...'
                       echo "Application Name: ${APP_NAME}"
             }
        }
        stage('Test') {
            parallel {
                stage('Unit Tests') {
                    steps { echo 'unit testing application...' }
                }
                stage('Integration Tests') {
                    steps { echo 'integration testing application...' }
                }
            }
        }
    }
}
```

INTRO TO SHARED LIBRARIES

- Jenkins can use functions in pipeline instead of writing the whole thing everytime
  - Central repository containing reusable pipeline functions
  - Multiple Jenkins pipelines consuming shared functions
  - Version control and collaboration on CI/CD code



THANK YOU