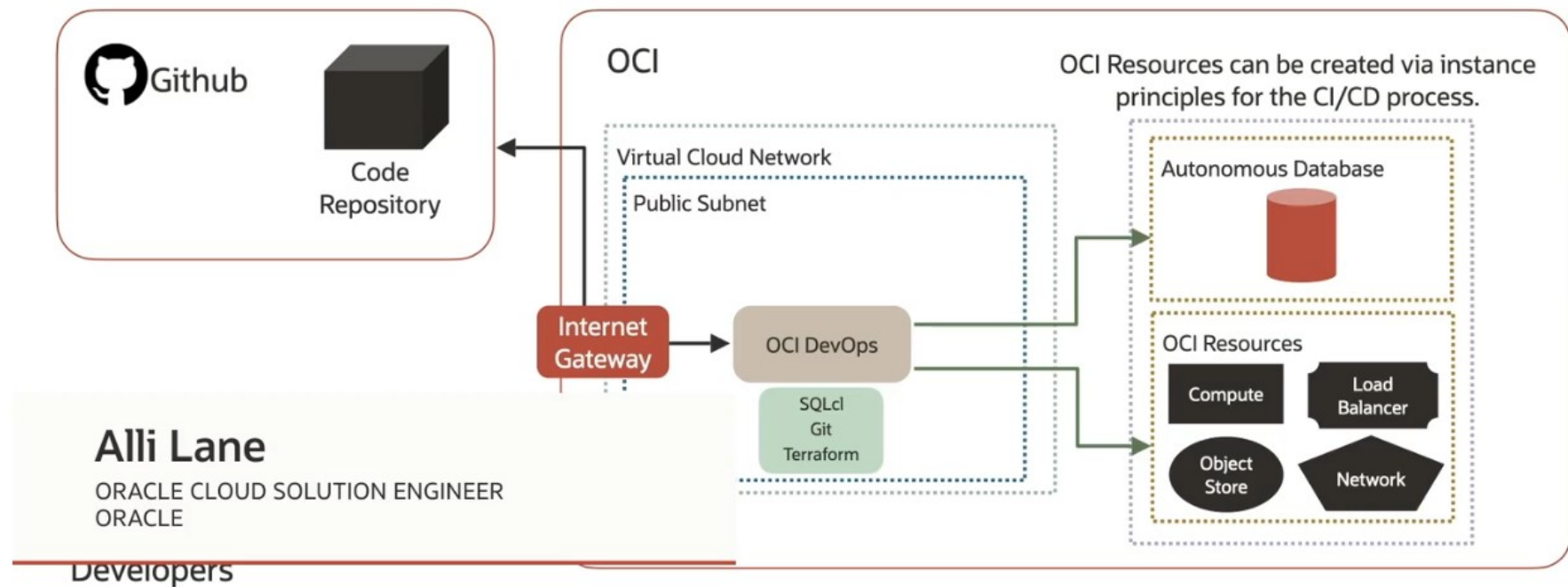


Sample Environment

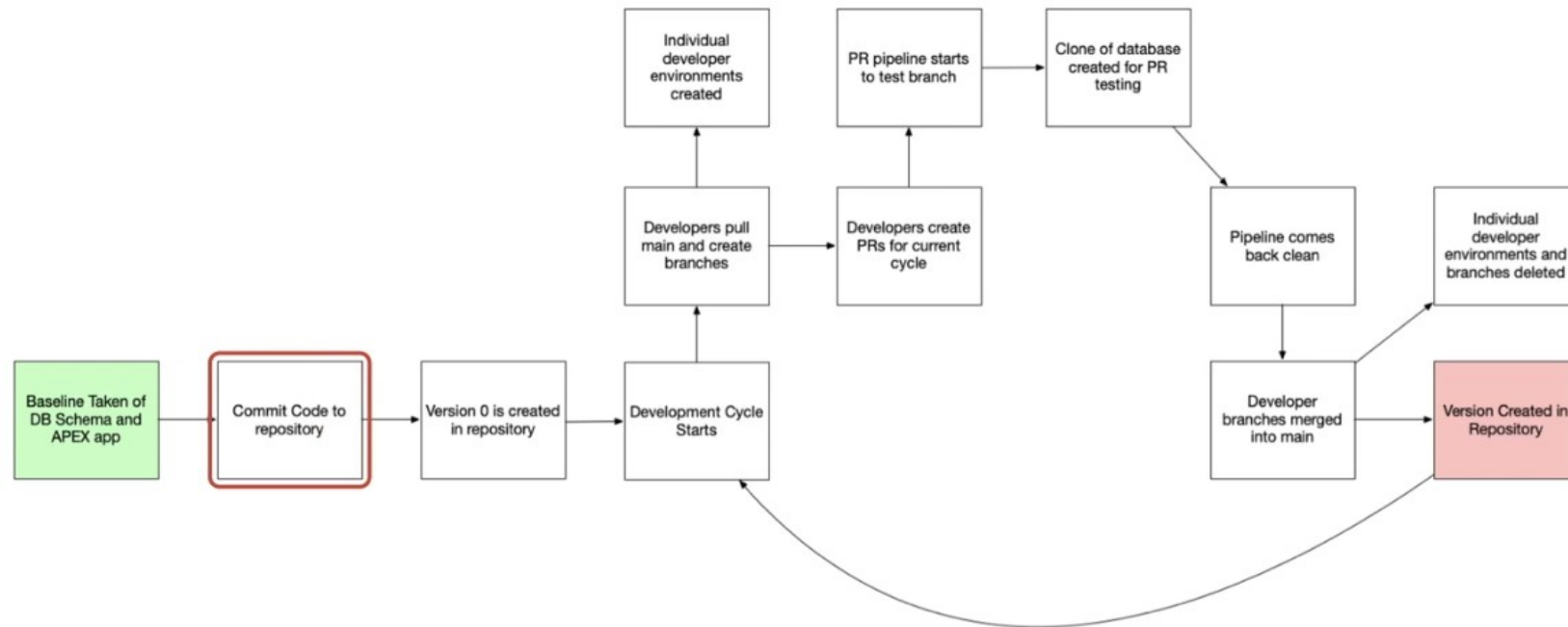
What does a sample environment look like for CI/CD with the ADB and OCI?



High-Level Flow



Development



Individual Environments and Creation

- Individual environments need:
 - APEX (Apps and workspace)
 - DB Schema(s)
 - Code from the repository (latest) or deployment
- Automate the process with APIs:
 - OCI CLI/PLSQL SDK
 - ORDS and the SQL Endpoint
 - Terraform

Individual Environments and Creation

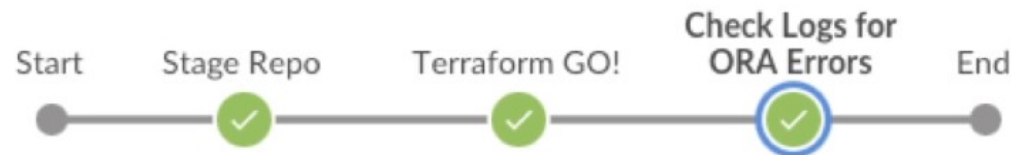
- Autonomous Database Clones
- OCI DB VMs
 - Clone a VM DB
 - Create a new DB VM from a backup
- Using OCI ExaCS
 - Sparse cloning
- Using Multitenancy
 - DB-APIs via ORDS
 - Using REST-enabled SQL service
- Reusable Instance
 - Guaranteed restore points/Flashback Database
 - RMAN duplicate/clone
 - Data Pump
- Docker/Virtual Machines
- ACFS/gDBClone

CI Process for Code Pushes

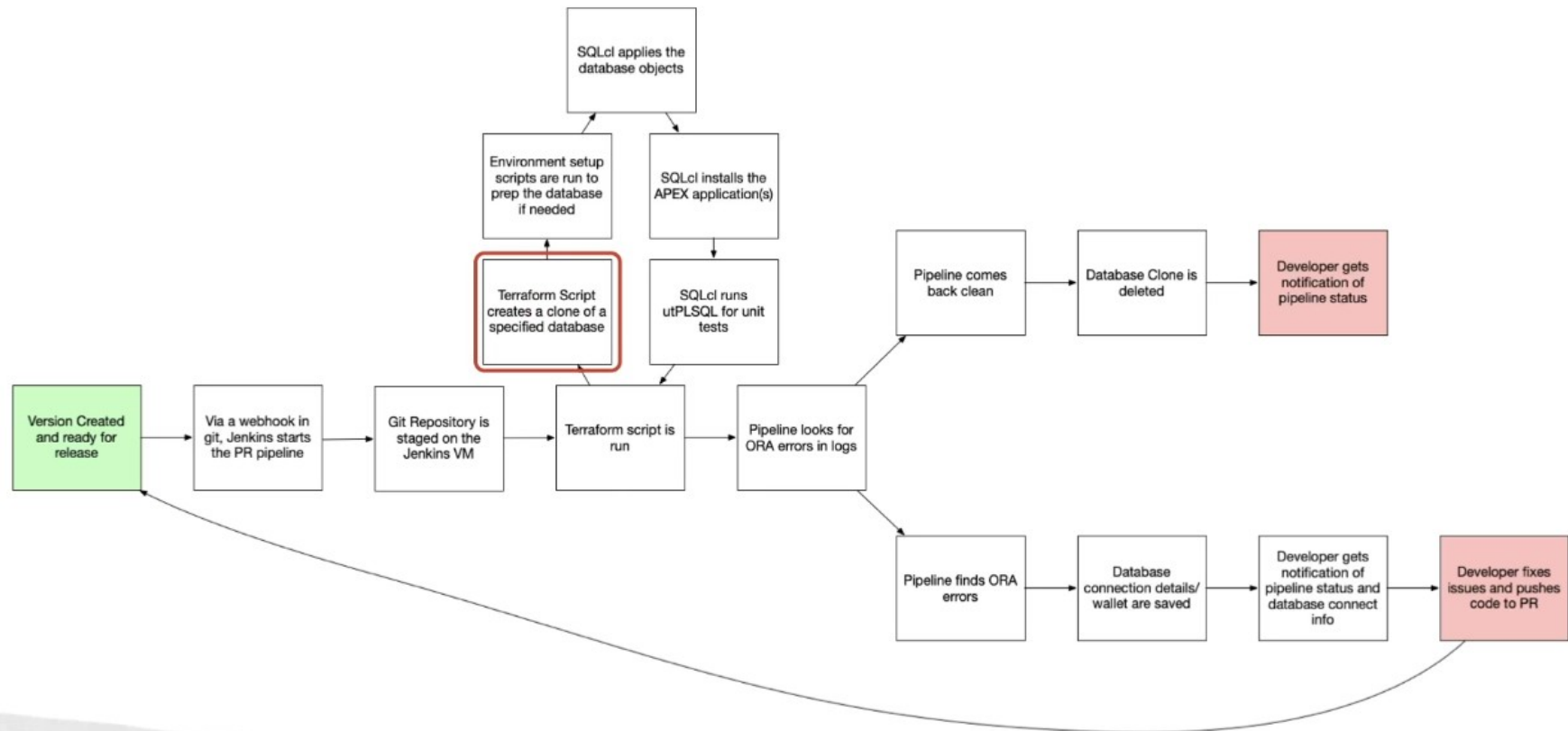
Every code pull request/merge (maybe push) should spawn a CI/CD pipeline

The pipeline should:

- Clone from an environment with the latest version of main
- Apply the developers branch/merged main into that database
- Run unit tests
- Report back on the status of the pipeline
- Destroy the environment if it is successful/keep it running if not successful



Pipeline Process



Deployment Targets and Automating the Process

- Pipelines can also be scheduled or started manually
- Choose deployment target with pipelines

Jenkins creates the artefacts and stores

Download and apply these artefacts onto the environment

Deployment Targets and Automating the Process

- Pipelines can also be scheduled or started manually
- Choose deployment target with pipelines

Automation

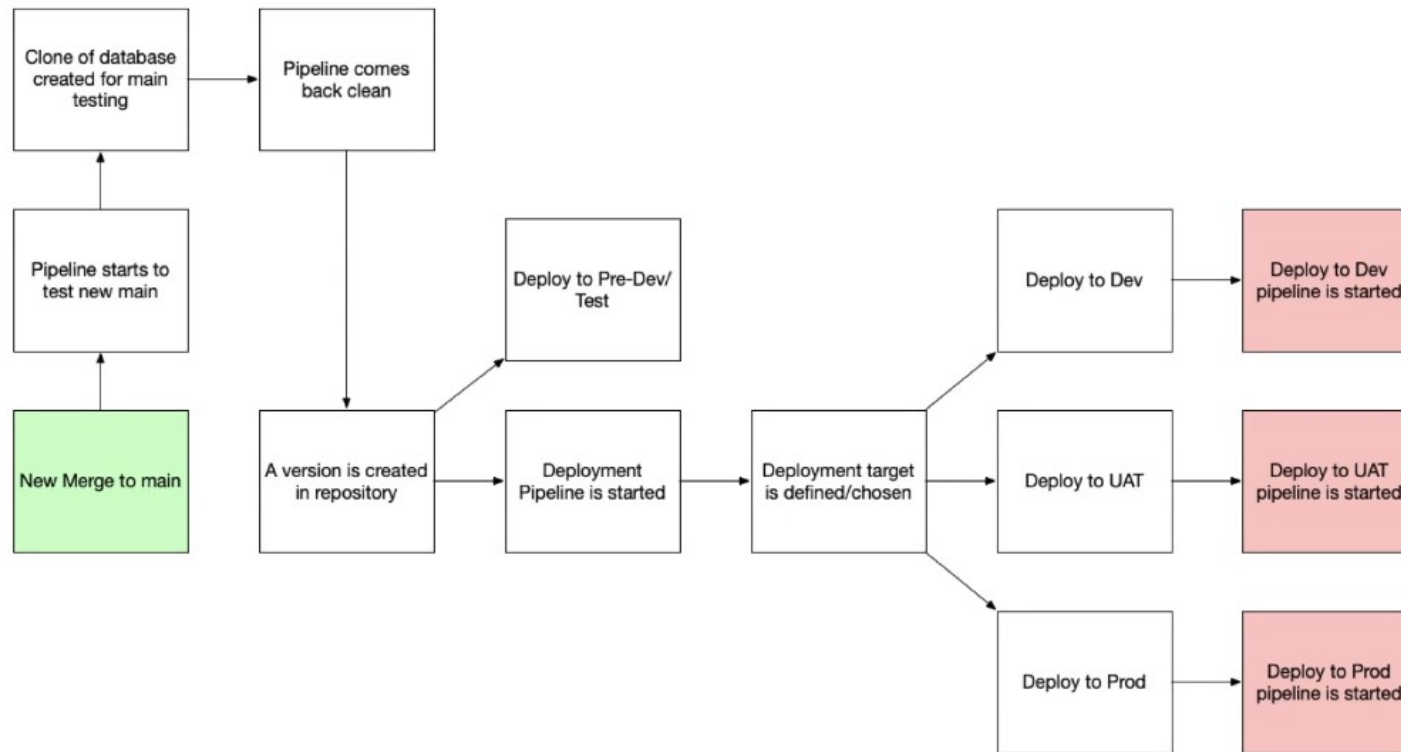
Pull request

Merge request

Deployment Targets and Automating the Process

- Pipelines can also be scheduled or started manually
- Choose deployment target with pipelines
- Eliminate surprises by having a logical progression of environments:
 - Dev
 - UAT
 - Production
- Deploy manually
 - Zip up the repo and deploy when you want
- Use ADB/DB clones to practice deployments manually or via pipelines

Deployment





Rollback

Pluggable database:

- Rollback the entire database
- Flashback the database
- Restore to a guaranteed restore point



Rollback

Not using multi-tenancy??? - We got you covered

- Rollback database objects via Liquibase
- Stage last versions of APEX apps from repository
- Install last version APEX app
- Flashback tables to restore data (metadata changes)

Rollback

Or Roll Forward?

- In some cases, it may be easier/better to fix the issue in production and bring the fix back to the main code line
- Constant testing/automation should help with these incidents

