FORGE Service Lab Deployment pipeline

Pasi Kivikangas

FORGE Service Lab

DIGILE

Deployment pipeline

Get started

Select and install your tools

- Version control system
- CI
- Deployment
- Test automation

Define your deployment pipeline

Write your application

Write your application deployment scripts

- Ansible, Puppet, Chef, Capistrano...

Create automated test cases for your application

Configure CI to build, test and deploy your application

Tools

FORGE has the selected set of tools for it's deployment pipeline

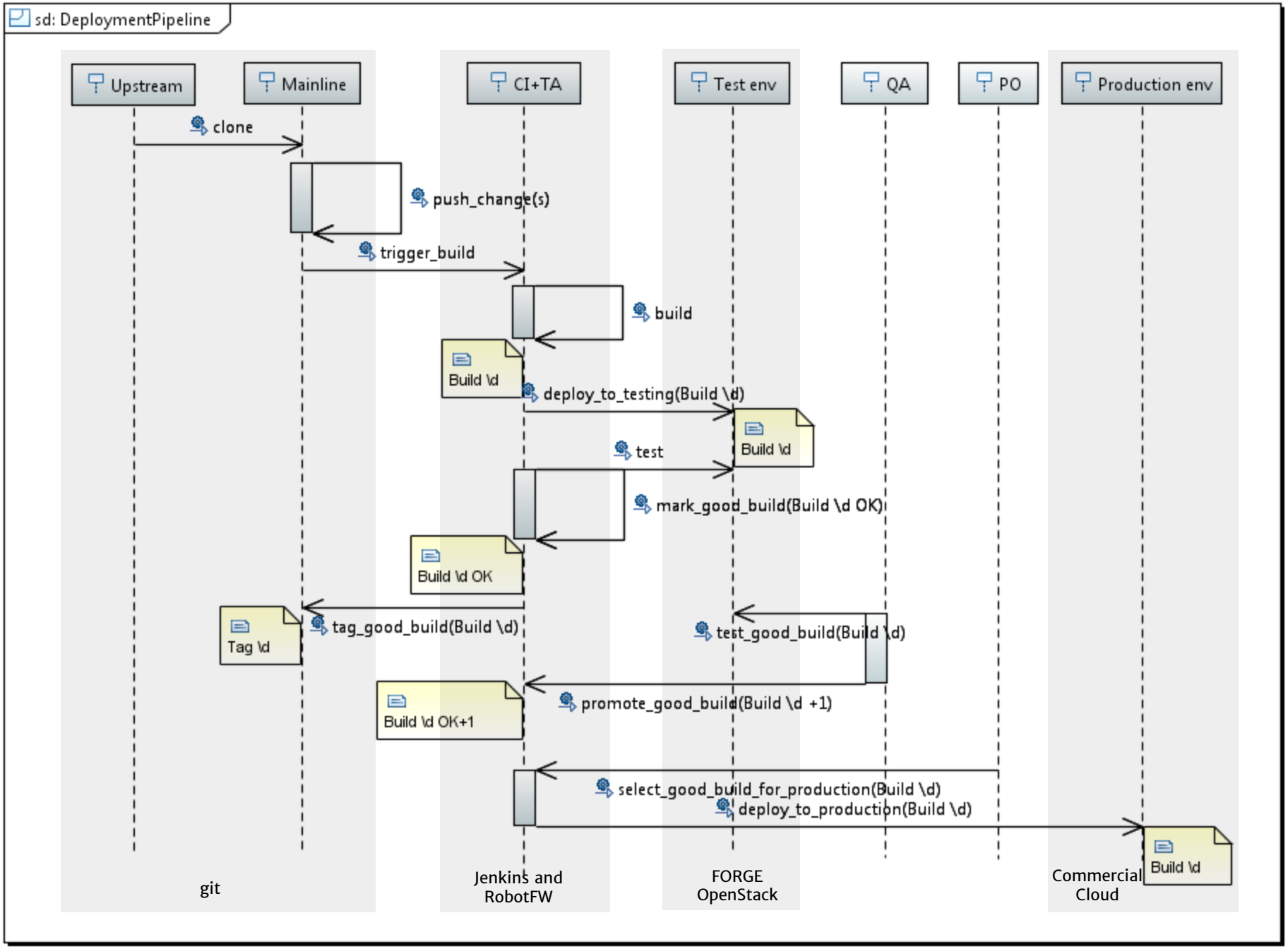
- **GitLab** for hosting source codes
- **Jenkins** as a CI server
- **Ansible** for deployments

  Ansible is an open-source software platform for configuring and managing computers. It combines multi-node software deployment, ad hoc task execution, and configuration management. It manages nodes over SSH and does not require any additional remote software (except Python 2.4 or later) to be installed on them.

  FORGE has many Ansible playbooks available e.g. Jenkins
- **Selenium and Robot Framework** for test automation

  Robot Framework is a generic test automation framework for acceptance testing and acceptance test-driven development (ATDD).
- **FORGE OpenStack cloud** as development and test environment

Note! Jenkins is able to use Git, Ansible and RobotFW gracefully


sd: DeploymentPipeline
Upstream
Mainline
CI+TA
Test env
QA
PO
Production env
clone
push_change(s)
trigger_build
build
Build \d
deploy_to_testing(Build \d)
Build \d
test
mark_good_build(Build \d OK)
Build \d OK
Tag \d
tag_good_build(Build \d)
test_good_build(Build \d)
Build \d OK+1
promote_good_build(Build \d +1)
select_good_build_for_production(Build \d)
deploy_to_production(Build \d)
Build \d
git
Jenkins and RobotFW
FORGE OpenStack
Commercial Cloud

Jenkins jobs

Consider the separating the Jenkins jobs based on the concern and according to how you have defined your deployment pipeline

E.g. GitLab

1. Build job (triggered automatically by the GitLab web hook)

2. Deploy to testing job (triggered by previous job)

3. Test job (triggered by previous job)

4. Deploy to production job (triggered manually)

Jenkins jobs – Build

- Build job (triggered automatically by the GitLab web hook)
  - Clone the code from GitLab release/master branch
  - Run unit tests
  - Check coding style
  - Initiate the subsequent "Deploy to testing job"
  - In case subsequent jobs are ok, then tag the build in GitLab
  - Note! you might want to have a web hook from GitLab to notify this job in case of a commit and push in GitLab. That way each push would automatically trigger a build job.

Jenkins jobs – Deploy to testing

- Deploy to testing job (triggered automatically)
  - Clone Ansible playbook from GitLab release/master branch
  - Run Ansible playbook to make a deployment to testing environment
  - Prepare to revert testing environment back to previous known good in case of a failure
  - Initiate the subsequent "Test job"

Jenkins jobs – Test

- Test job (triggered automatically)
  - Clone your test asset from Gitlab release/master
  - Run automated tests e.g. Robot framework tests
  - Tag good builds in GitLab
  - Note! Run manual tests against testing environment and promote the build if it's good

Jenkins jobs – Deploy to production

- Deploy to production job (manual step)
  - Clone Ansible playbook from GitLab release/master branch
  - Run this job manually to deploy desired (and promoted) build to production env
  - Prepare to revert production environment back to previous known good in case of a failure

Jenkins can execute desired jobs in desired order

All | **BI Reports** | GitLab | Plaza | Plaza-Devel | Redmine

S	W	Name ↓	Last Success	Last Failure	Last Duration	Robot Results
		BI-Reports-1.build	19 hr - #76	N/A	59 sec	
		BI-Reports-2.deploy-to-testing	19 hr - #89	N/A	26 sec	
		BI-Reports-3.test	19 hr - #54	7 days 19 hr - #50	31 sec	2 / 2 passed
		BI-Reports-4.deploy-to-production	1 mo 2 days - #15	1 mo 3 days - #14	41 sec	

Icon: S M L

Legend RSS for all RSS for failures RSS for just latest builds

Jenkins can execute frequent and continuous performance testing too

All | BI Reports | GitLab | **Plaza** | Plaza-Devel | Redmine

S	W	Name ↓	Last Success	Last Failure	Last Duration	Robot Results
		Deploy-Plaza-Devel	4 days 20 hr - #117	2 mo 1 day - #78	1 min 33 sec	
		Deploy-Plaza-Production	4 days 21 hr - #38	1 mo 23 days - #29	6 min 54 sec	68 / 73 passed
		Deploy-Plaza-Testing	6 days 21 hr - #149	1 mo 15 days - #138	9 min 24 sec	179 / 195 passed
		Plaza	6 days 21 hr - #83	1 mo 16 days - #78	9 min 28 sec	179 / 195 passed
		Plaza-build-performance	1 hr 33 min - #127	N/A	1 min 41 sec	
		Plaza-Development	4 days 20 hr - #104	2 mo 9 days - #61	1 min 34 sec	
		Plaza-Performance-testing-Continuously-DEVEL	3 hr 0 min - #384	N/A	1 min 3 sec	
		Plaza-Performance-testing-Continuously-PRODUCTION	2 hr 53 min - #556	N/A	39 sec	
		Plaza-Performance-testing-Continuously-TESTING	3 hr 0 min - #388	N/A	1 min 3 sec	
		Plaza-Robot-Testing	1 hr 38 min - #325	1 mo 2 days - #274	6 min 29 sec	220 / 235 passed

Icon: S M L

Legend | RSS for all | RSS for failures | RSS for just latest builds

DIGILE

FORGE Service Lab

DIGILE in a Nutshell

- **DIGILE** is the Center for Science, Technology and Innovation (SHOK) focusing on Internet economy and related technologies and business
- **Mission:** DIGILE creates Internet economy competencies to enable new global business and job growth for DIGILE's stakeholders and partners
- **Three main services**:
  - **Research:** Cooperative national and international research programs to create new technological and business innovations
  - **Solutions:** Facilitation of business ecosystems and lead solution creation to explore new global business opportunities
  - **Digital service creation:** FORGE Service Lab for fast digital service creation and competence scaling
- **Core enablers**:
  - International networking
  - Operative excellence
  - Co-creation leadership

FORGE Service Lab
WHAT, WHY, WHO, FOR WHOM

- **WHAT:** FORGE Service Lab is a laboratory for creating digital services in the Internet-era. It is intended as a tool to accelerate the creation of digital services in Finland - *from an idea to a scalable implementation*.

- **WHY:** Internet economy will grow stronger and digitalisation spreads across all industries. Most of the value is being created via digital services. As a result, digital services know-how needs to become one of the nation's core competencies.

- **WHO:** DIGILE, CSC-IT Center for Science, Kainuun Etu Oy with the Ministry of Traffic and Communication, the financing partner for the ramp-up

- **FOR WHOM**: To all who are interested in developing digital services e.g. businesses, educational institutions, business development teams, the public sector – all industries and government sectors are included.

FORGE Service Lab – Offering

Legal & Contract framework for each stakeholder: service developers and partners

Offers wide development framework for service projects where multiple stakeholders and partners can share openly the knowledge and develop efficiently globally recognisible successfull services

Partner network from multidisciplinary perspectivie: eg. Business development, Service Design, Technical development

Reference model for the creation of digital services, from the idea to the scalable implementation

Crowdsourcing methods ad tools which enables to create as meaningful and successfull service as possible from the end users perspective

Cloud computing platform for agile and fast ways to develop and test the services

Guidance and support for the project during the service creation path in order to manage the big picture

More information

- Documentation
  - https://support.forgeservicelab.fi/redmine/projects/forge-support/wiki
- Support tickets
  - https://support.forgeservicelab.fi/redmine/projects/forge-support/issues/new
- XMPP
  - forge-support@xmpp.forgeservicelab.fi
- Email
  - support@forgeservicelab.fi

THANK YOU