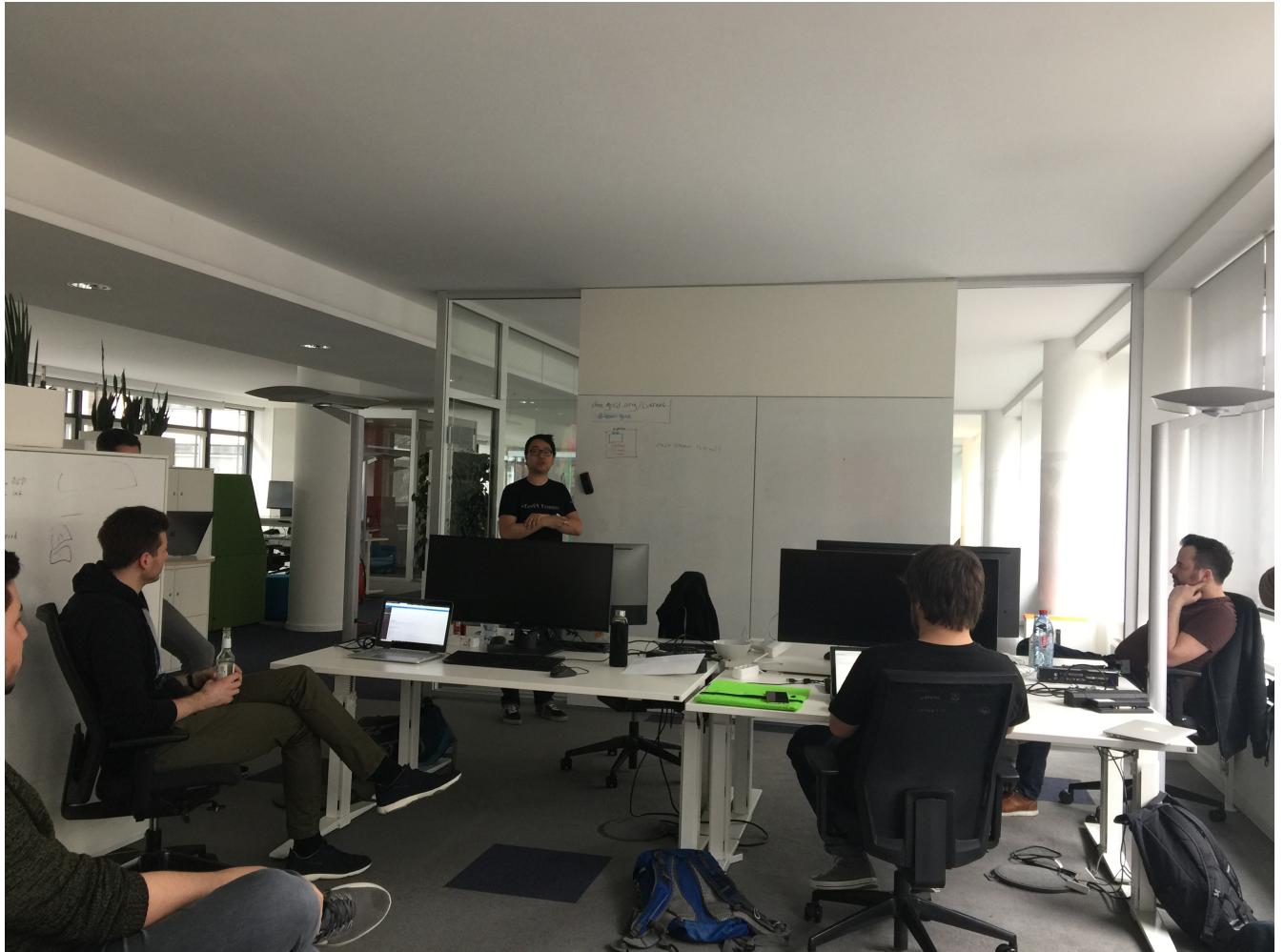


Hackathon CI/CD

Table of Contents

Introduction to Continous Integration	1
GoCD Introduction	1
Goals for Today	2
GoCD Architecture	2
Setup GoCD	3
On AWS	3
On my Mac with Docker	3
My first pipeline	4



Introduction to Continuous Integration

[Current documentation on GoCD](#)

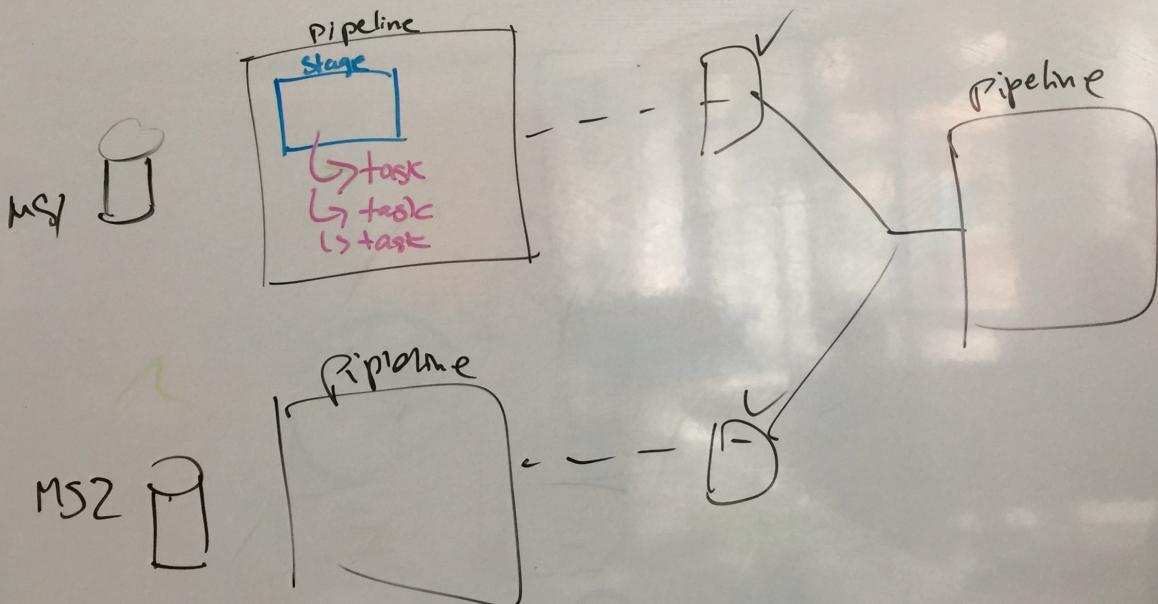
CI = Continuous Integration

CD = Continuous D = Deployment D= Delivery

GoCD Introduction

docs.gocd.org/current

#team-gocd



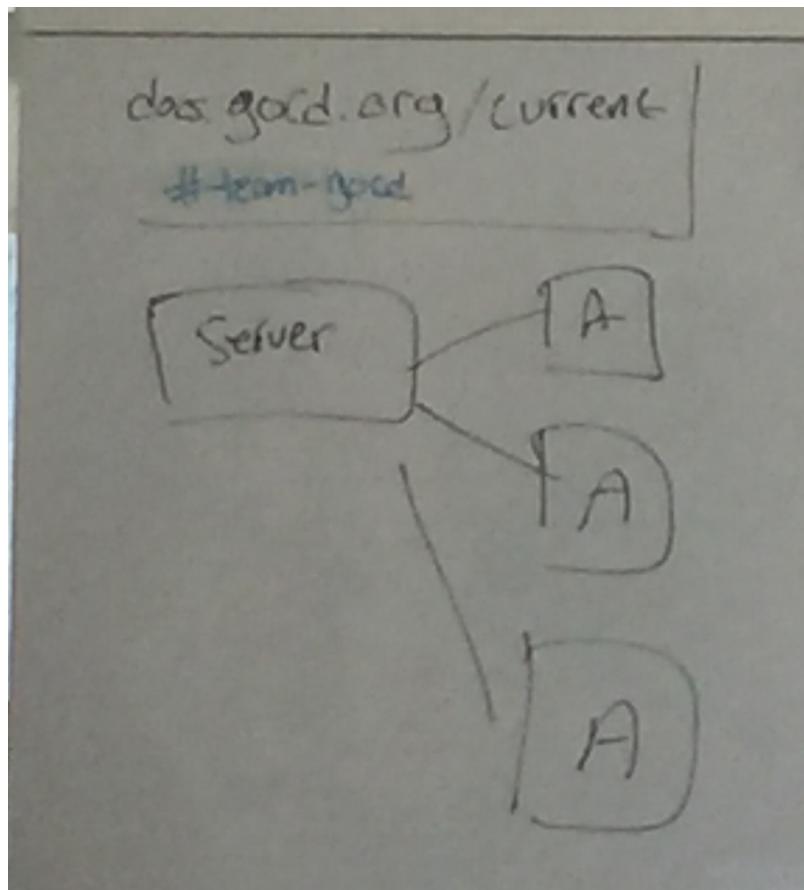
Pipeline as a first class citizen

Pipeline contains Stages Stage is a group of Operations. Can be run in parallel. Tasks - a task is one command line. i.e. install, build,

Goals for Today

- Setup GoCD
- myfirst pipeline as gui
- myfirst pipeline as code
- serialized build
- cached dependencies
- scaling
- value map presentation
- fan in, fan out, deploy on aws

GoCD Architecture



- One Server
- Multiple Agents
- An Agent is running a Stage

Setup GoCD

On AWS

There is already a ready to run AMI at AWS.

On my Mac with Docker

I want to setup GoCD on my Macintosh with an Agent with which i can build a small Spring Boot application. I need a java compiler and maven on the Agent.

Welcome to the world of pain!

First we start an actual GoCD Server.



Why the heck is there no docker image tagged with **latest**?

```
docker run -d -p8153:8153 -p8154:8154 gocd/gocd-server:v18.2.0
```

Now i went through some pain trying to build my own docker immage with an agent based on an official gocd/go-agent image. Unfortunately the gocd-agents do NOT contain a full JDK, but only a JRE. Luckily there is a bit heavy but working agent thanks to Ashwanth Kumar! The git repo for that agent is [here](#).

```
docker run -d --net="host" -e GO_SERVER_URL=https://localhost:8154/go ashwanthkumar/gocd-agent-jdk-aws-maven-sbt:latest
```

After starting the agent you have to go to the GoCD Server frontend to the Agents section and **activate** the agent.

The screenshot shows a web browser window with a tab bar containing various links like 'allegro', 'cloudflare', 'Jitsi', 'gocd/g', 'Agents', 'herrwile', 'johnnew', 'Maven', 'docker', 'gocd/c', 'gocd/c', 'How to', 'Use bri', 'nginx', and 'ashwanth'. The main content area is titled 'Agents' and includes a search bar with 'Filter Agents'. Below the search bar, a green banner says 'Deleted 1 agent'. A table lists one agent:

Agent Name	Sandbox	OS	IP Address	Status	Free Space	Resources	Environments
unknown-host-250	/var/lib/go-agent	Alpine Linux v3.3	127.0.0.1	Idle	48.04 GB	none specified	none specified

Now lets setup a simple pipeline!

My first pipeline

My project is located [here](#)

I setup a simple pipeline in GoCD with the web frontend.

localhost:8080

Add Pipeline - Go

herwiegger/springbo

herwiegger

AsciIDoc Writer's Go

Maven - Settings Re

Maven - Settings Re

Asciidoc PDF | A

PIPELINES ENVIRONMENTS AGENTS ADMIN

Add Pipeline

Step 1: Basic Settings Step 2: Materials Step 3: Stage/Job

Basic Settings

Pipeline Name*

aSimpleMavenBuild

Pipeline Group Name

defaultGroup

* indicates a required field

CANCEL NEXT

Copyright © 2018 ThoughtWorks, Inc. Licensed under Apache License, Version 2.0.
Go includes third-party software. Go Version: 18.2.0 (6228-fbb379a451871b29aa47ae907a78dc64379233).

In the second step, the connection to source code repo got configured. They use the rather strange sounding name material for dealing with that.

localhost:8080

Add Pipeline - Go

herwiegger/springbo

herwiegger

AsciIDoc Writer's Go

Maven - Settings Re

Maven - Settings Re

Asciidoc PDF | A

PIPELINES ENVIRONMENTS AGENTS ADMIN

Add Pipeline

Step 1: Basic Settings Step 2: Materials Step 3: Stage/Job

Materials

Material Type*

Git

URL*

https://github.com/herwiegger/spring

Branch

master

Destination Directory

Poll for new changes

Shallow clone (recommended for large repositories)

Blacklist

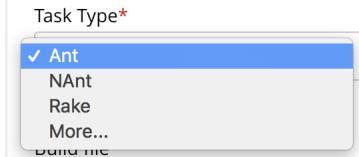
Enter the paths to be excluded. Separate multiple entries with a comma.

Invert the file filter, e.g. a Blacklist becomes a Whitelist instead.

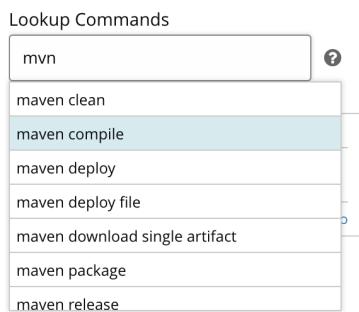
CHECK CONNECTION

Connection OK.

Now its time to setup the first build step. Out of the box GoCD does not support Maven that much. Supported tools are: Ant, Nant, Rake. If you do not use one of these tools you can invoke an arbitrary command line. The only prerequisite is, that the build agent running this task needs the respective tool locally installed.

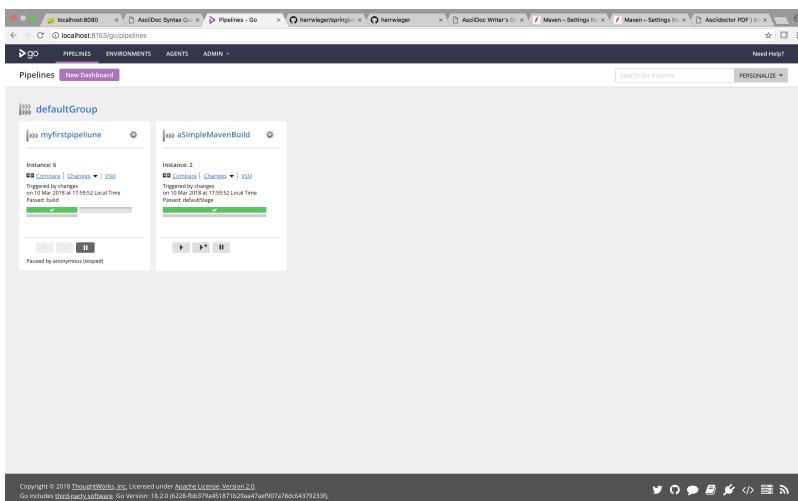


Support for mvn is rather limited. But at least you can lookup typical mvn invocations.



Finishing the three step wizard you get a simple pipeline, which can compile a mvn project.

Going back on the pipeline tab in the web frontend you get an overview over all available pipelines.



Now you can start the pipeline by pressing the pause symbol (to unpause) and then the start button.

The screenshot shows the Go Pipeline Activity page for a pipeline named 'ASimpleMavenBuild'. A single execution instance is listed under the 'defaultStage' tab. The instance is labeled '1 VSM' and has a yellow progress bar. The status bar information includes:

- revision: dfba5f4fec06...
- 10 Mar 2018 at 17:57:13
- Local Time
- Triggered by changes

At the bottom of the page, there is a copyright notice: "Copyright © 2018 ThoughtWorks, Inc. Licensed under Apache License, Version 2.0. Go includes third-party software. Go Version: 18.2.0 (6228-fbb379a451871b29aa47ae907a78dc64379233)".

Clicking on the pipeline name shows you the execution history of the pipeline.

The screenshot shows the Go Pipeline Activity page for the same pipeline 'ASimpleMavenBuild'. Now, two execution instances are listed under the 'defaultStage' tab. The first instance is labeled '2 VSM' with a green progress bar, and the second is labeled '1 VSM' with a red progress bar. Both instances have the same status bar information as the first one.

At the bottom of the page, there is a copyright notice: "Copyright © 2018 ThoughtWorks, Inc. Licensed under Apache License, Version 2.0. Go includes third-party software. Go Version: 18.2.0 (6228-fbb379a451871b29aa47ae907a78dc64379233)".

Getting info on an execution run you have to go over the build status bar on the info icon and click.

Instance defaultStage

2 VSM

revision: 8eaf0eeef3afcb...
10 Mar 2018 at 17:59:52
Local Time
Triggered by changes

So now you get the stage details.

Run: 1 of 1 Passed | Automatically triggered on 10 Mar 2018 at 17:59:52 Local Time Duration: 00:00:24 | Compare

JOBS

- Failed: 0
- In Progress: 0
- Passed: 1
 - ✓ compileWithMaven

STAGE HISTORY

Status	Count
Passed	2
Pending	1

Copyright © 2018 ThoughtWorks, Inc. Licensed under Apache License, Version 2.0.
Go includes third-party software. Go Version: 1.8.2 (62f6b379a451871b29aa47ae907a78dc64379233).

And from here you can get the info on the jobs executed.

Run: 1 of 1 Passed | Automatically triggered on 10 Mar 2018 at 17:59:52 Local Time Duration: 00:00:24 | Compare

Name	Result	State	Duration	Agent
compileWithMaven	Passed	Completed	2 seconds	unknown-host:250 (127.0.0.1)

STAGE HISTORY

Status	Count
Success	2
Error	1

Clicking on the job name finally moves you to the console output of the job.

SCHEDULED ON: 10 Mar 2018 at 17:59:52 Local Time
COMPLETED ON: 10 Mar 2018 at 18:00:17 Local Time [more...](#)
DURATION: 2.0s

AGENT: unknown-host:250 (127.0.0.1)
BUILD CAUSE: modified by Thomas Wieger <thomas.wieger@eng-softwarelabs.de>

Console	Tests	Failures	Artifacts	Materials	Properties	Link to this tab	JOB HISTORY
---------	-------	----------	-----------	-----------	------------	------------------	-------------

```
[go] Job Started: 2018-03-10 17:00:13 GMT
[go] Start to prepare aSimpleMavenBuild/2/defaultStage/1/compileWithMaven on unknown-host:250 [/var/lib/go-agent]
[go] Start to build aSimpleMavenBuild/2/defaultStage/1/compileWithMaven on unknown-host:250 [/var/lib/go-agent]
[go] Task: mvn clean compile <took: 2.354s>
[go] Current job status: passed
[go] Start to create properties aSimpleMavenBuild/2/defaultStage/1/compileWithMaven on unknown-host:250 [/var/lib/go-agent]
[go] Start to upload aSimpleMavenBuild/2/defaultStage/1/compileWithMaven on unknown-host:250 [/var/lib/go-agent]
[go] Job completed aSimpleMavenBuild/2/defaultStage/1/compileWithMaven on unknown-host:250 [/var/lib/go-agent]
```