

Configuration as Code of Jenkins (for Kubernetes)

Nicolaj Græsholt - @figaw on Twitter
DevOps Consultant at Eficode Praqma

eficode

PRAQMA
Now part of Eficode

About: nicolaj græsholt 😊

<DevOps Consultant>

eficode

PRAQMA
Now part of Eficode



♥ Good Documentation, Examples and Proof of Concepts

DMs are **OPEN**, please be nice :) → @figaw

PRAQMA
Now part of Eficode

about:talk

DON'T PANIC

(Large, friendly letters)

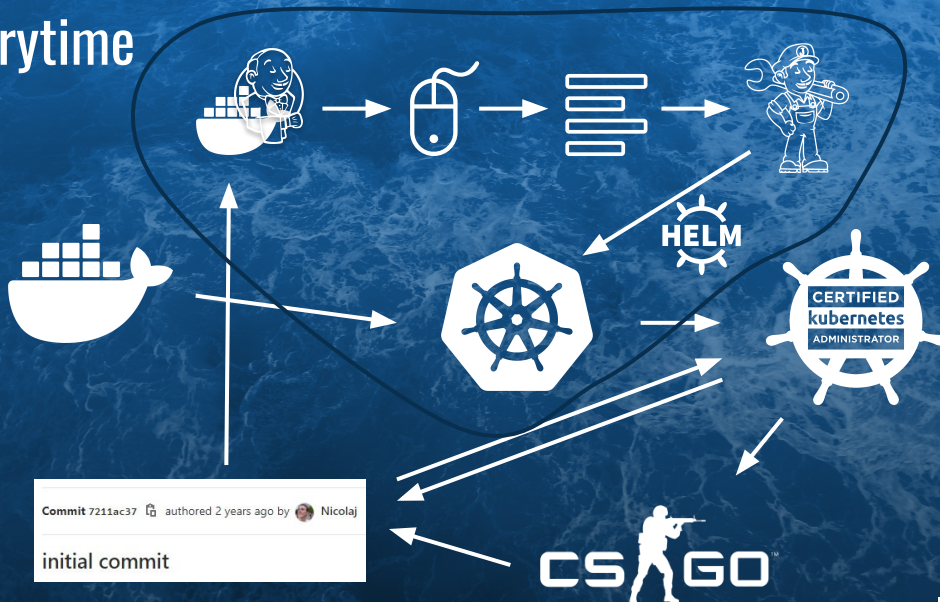
Everything You See Is What You Get

<https://github.com/figaw/configuration-as-code-jenkins-k8s>

@figaw

PRAQMA
Now part of Elicode

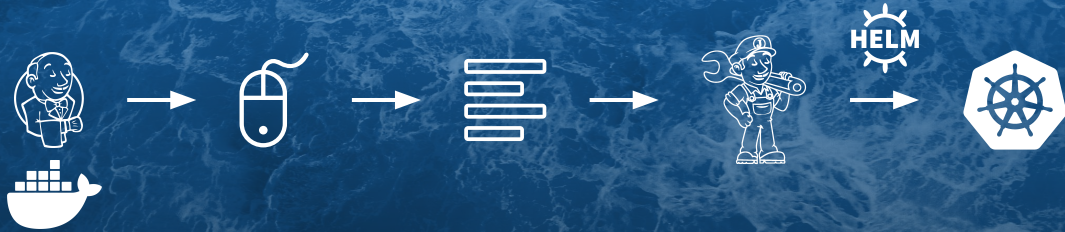
Storytime



@figaw

PRAQMA
Now part of Elicode









Focused Storytime



@figaw

PRAQMA
Now part of Eficode

Agenda

- I.  → , 
- II.  → 
- III.  → , 

@figaw

PRAQMA
Now part of Eficode



@figaw

PRAQMA
Now part of Eficode

I.  →  **Why?**

Easy Reconfiguration

Trivial to recreate jobs

Light Backups

Faster backups, cheaper storage

Configuration under Version Control

Who changed **What**, **When**, from **Which Value**?

@figaw

PRAQMA
Now part of Eficode

I. → How?

JobDSL ← *instant* conversion

Old and well supported

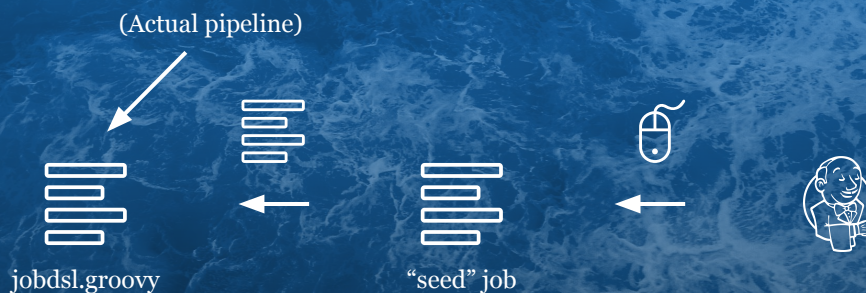
Declarative or Scripted Pipelines

“The Future,” resume on master restart, has DSL or Groovy

@figaw

PRAQMA
Now part of Eficode

I. A Job in JobDSL



- it gets smart, promise!

@figaw

PRAQMA
Now part of Eficode

I. Creating a Job

By JobDSL

1. Create a new file and call it `jobdsl.groovy`
2. Write JobDSL specification¹
3. Publish it to internal Git

`jobdsl.groovy:`

```
job('hello-world') {  
  steps {  
    echo "Hello, World!"  
  }  
}
```

1. Assume writing JobDSL is trivial

@figaw

PRAQMA
Now part of Elicode

I. Creating a Seed Job

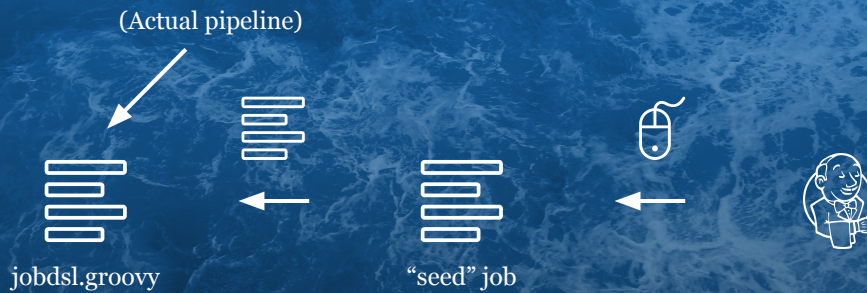
Creating the seed.groovy Job

1. Go to Jenkins UI
2. Create a new **Freestyle Job**
3. Add a JobDSL build-step, for the `jobdsl.groovy`
4. Click **Save** and **Run it**

@figaw

PRAQMA
Now part of Elicode

I. A Job in JobDSL

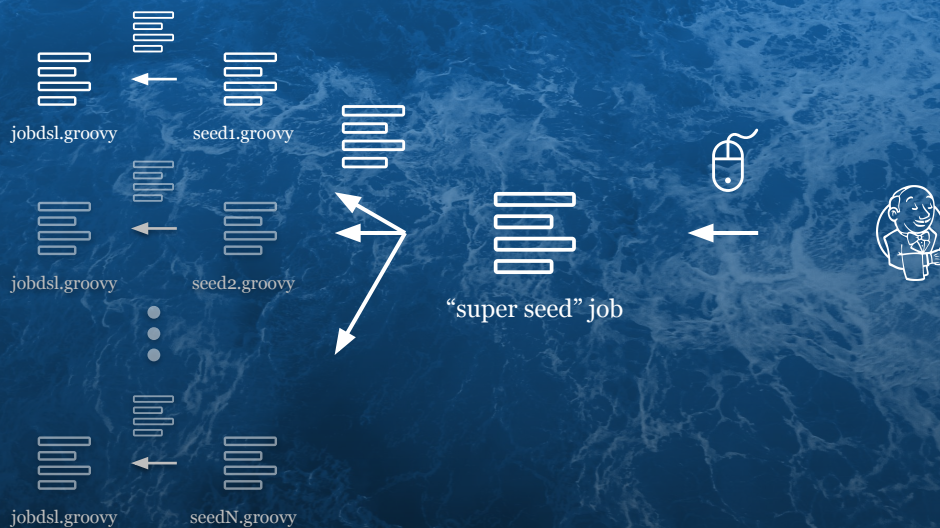


- it gets smart, promise!

@figaw

PRAQMA
Now part of Eficode

I. Bootstrapping all the Jobs



@figaw

PRAQMA
Now part of Eficode

I. JobDSL

The Configure Block:

The Coolest Feature¹

“.. used inside the Job DSL to give direct access to underlying XML of the Jenkins **config.xml**”²

```
job('example') {
  scm {
    git {
      remote {
        url('git@server:account/repol.git')
      }
      configure { node ->
        // node represents <hudson.plugins.git.GitSCM>
      }
    }
  }
}
```

1. Apart from being **Groovy**

2. <https://github.com/jenkinsci/job-dsl-plugin/wiki/The-Configure-Block>

@figaw

PRAQMA
Now part of Elicode

I. config.xml

Did You Know

A Jenkins job is available as XML,
through the “/**config.xml**”-endpoint?

```
view-source:localhost:8080/job/my-job/config.xml
1 <?xml version="1.1" encoding="UTF-8"?>
2 <project>
3   <description></description>
4   <keepDependencies>false</keepDependencies>
5   <properties>
6     <scm class="hudson.scm.NullSCM"/>
7     <canRoam>true</canRoam>
8     <disabled>false</disabled>
9     <blockBuildWhenDownstreamBuilding>false</blockBuildWhenDownstreamBuilding>
10    <blockBuildWhenUpstreamBuilding>false</blockBuildWhenUpstreamBuilding>
11    <triggers/>
12    <concurrentBuild>false</concurrentBuild>
13    <builders>
14      <hudson.tasks.Shell>
15        <command>echo &quot;Hello, World!&quot;</command>
16      </hudson.tasks.Shell>
17    </builders>
18    <publishers/>
19    <buildWrappers/>
20  </project>
```

@figaw

PRAQMA
Now part of Elicode

I. Demo Time!

Will Show:

- *A Jenkins job is already XML*
- *JobDSL to wrap XML*
- *JobDSL bootstrapping, for sanity check*



Source: <https://www.artstation.com/artwork/1XmOe>

@figaw

PRAQMA
Now part of Elicode

I. Is this cheating?

No:

- *Change job through the UI*
- *Persist changes with the generated `config.xml`, through Git*
- *Job configuration is under Version Control*
- *Easy reconfiguration with `super-seed.groovy`*

Next steps?

Declarative or Scripted Pipelines for the complete “as code” workflow

@figaw

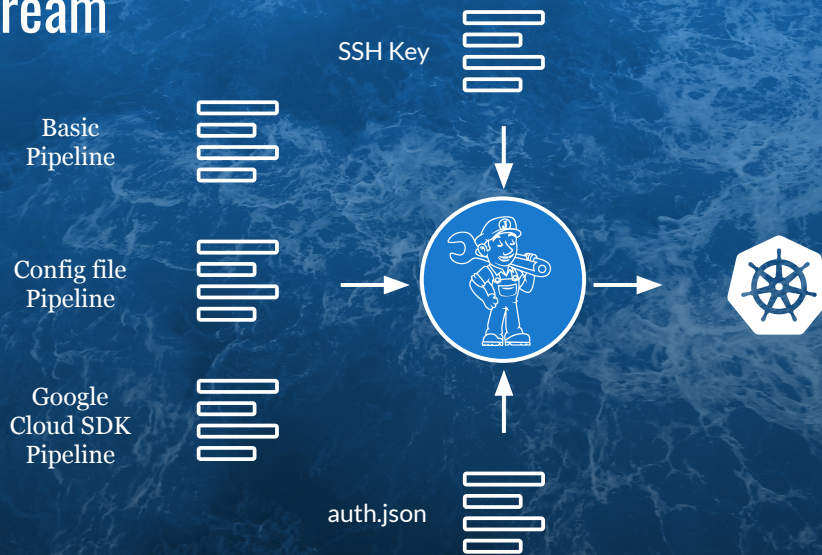
PRAQMA
Now part of Elicode



@figaw

PRAQMA
Now part of Eficode

My Dream



@figaw

PRAQMA
Now part of Eficode

II. → Why?

Easy Reconfiguration

Trivial to recreate last configuration, easy to create test setups

hint: *great for containers!*

Light Backups

Faster backups, cheaper storage

Configuration under Version Control

Who changed **What**, **When**, from **Which Value**?

@figaw

PRAQMA
Now part of Elicode

II. → How?

Step by Step

One plugin / setting at a time

Great Examples

<https://github.com/jenkinsci/configuration-as-code-plugin/tree/master/demos>

Switch Backup Strategy when Done

Save `jenkins.yaml` instead of `JENKINS_HOME`

@figaw

PRAQMA
Now part of Elicode

II. → Step 1?

Basic JCasC in Docker

- JCasC plugin

@figaw

PRAQMA
Now part of Eficode

II. → Step 1?

Basic JCasC in Docker

- JCasC plugin, preinstalling

plugins.txt

```
configuration-as-code:1.39
```

jenkins.yaml

```
jenkins:  
  systemMessage: "Configured as Code!"
```

Dockerfile:

```
FROM jenkins/jenkins:2.222.1  
COPY plugins.txt /usr/share/jenkins/ref/plugins.txt 1  
RUN /usr/local/bin/install-plugins.sh < /usr/share/jenkins/ref/plugins.txt  
  
COPY jenkins.yaml /var/jenkins_home/casc_configs/ 2  
ENV CASC_JENKINS_CONFIG=/var/jenkins_home/casc_configs/jenkins.yaml
```

1. <https://github.com/jenkinsci/docker/#script-usage>
2. <https://github.com/jenkinsci/configuration-as-code-plugin#getting-started>

@figaw

PRAQMA
Now part of Eficode

II. → Demo time!

Trust me, it's that simple. Will Show:

- Starting the simplest JCasC in Docker



Source: <https://www.artstation.com/artwork/1XmOe>

@figaw

PRAQMA
Now part of Elicode

II. → Demo time! Recap

Trust me, it's that simple. Will Show:

- Starting the simplest JCasC in Docker

Caveats:

Overwriting your JENKINS_HOME with -v

- Missing Configuration as Code plugin
- Missing /var/jenkins_home/casc_configs/jenkins.yaml



Source: <https://www.artstation.com/artwork/1XmOe>

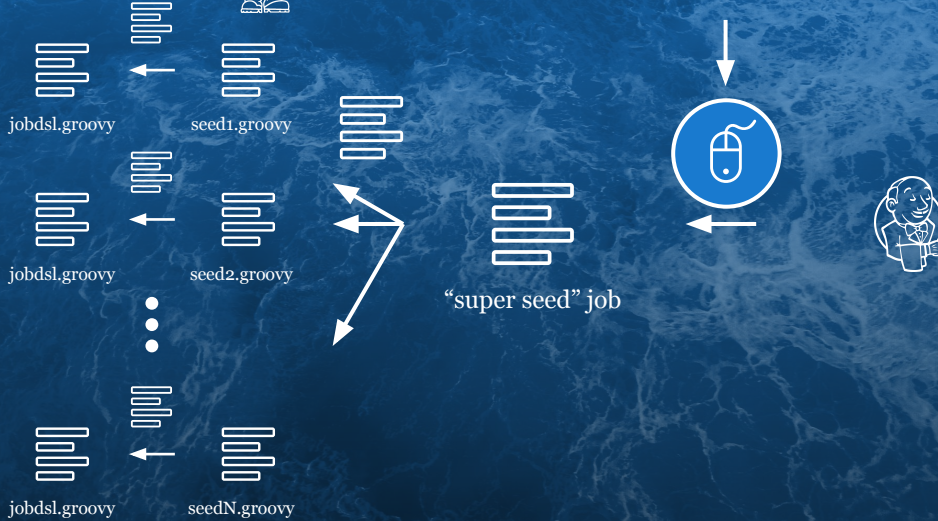
@figaw

PRAQMA
Now part of Elicode

II.



Step 2?



II.



Bootstrapping as Code

Adding a Job to jenkins.yaml

```
jenkins:
  systemMessage: "Configured as Code!"

jobs:
  - script: >
    job('hello-world') {
      steps {
        echo "Hello, World!"
      }
    }
  }
```

@figaw

PRAQMA
Now part of Elicode

II. → Bootstrapping as Code

Adding our super-seed.groovy to jenkins.yaml

```
...
jobs:
  - script: >
    job('super-seed') {
      scm {
        git {
          remote {
            url
            ('git@github.com:figaw/configuration-as-code-jenkins-k8s-jobdsl.git')
            credentials('jenkins-github-ssh')
          }
        }
      }
      steps {
        dsl {
          external('jobdsl/**/*.groovy')
          removeAction('DELETE')
        }
      }
    }
  }
```

@figaw

PRAQMA
Now part of Eficode

II. → Missing Pieces?

Authenticating

“Credentials” plugin

plugins.txt

```
configuration-as-code:1.39
credentials-binding:1.21
```

jenkins.yaml

```
...
credentials:
  system:
    domainCredentials:
      - credentials:
          - basicSSHUserPrivateKey:
              scope: GLOBAL
              id: jenkins-github-ssh
              username:
              passphrase:
              description: "SSH key for GitHub"
              privateKeySource:
                directEntry:
                  privateKey: ${JENKINS_GITHUB_SSH}
...

```

Environment Variable

@figaw

PRAQMA
Now part of Eficode

II. → Missing Pieces?

And..

plugins.txt

```
configuration-as-code:1.39
credentials-binding:1.21

job-dsl:1.77
git:4.2.2
ssh:2.6.1
workflow-aggregator:2.6
```

super-seed.groovy

```
jobs:
- script: >
  job('super-seed') {
    scm {
      git {
        remote {
          url ('git@github.com:figaw/[...].git')
          credentials('jenkins-github-ssh')
        }
      }
    }
    steps {
      dsl {
        external('jobdsl/**/*.groovy')
        removeAction('DELETE')
      }
    }
  }
}
```

@figaw

PRAQMA
Now part of Eficode

II. → Demo time?

Will Show:

- Starting a Jenkins
- With the super-seed job, bootstrapped



Source: <https://www.artstation.com/artwork/1XmOe>

@figaw

PRAQMA
Now part of Eficode

II. → Caveats

Preinstalling Plugins

- Recommendation is to create a **base-image** with the plugins
- Version of “Workflow Aggregator”-plugin
- See also: <https://github.com/jenkinsci/custom-war-packager> for ways of packaging Jenkins

@figaw

PRAQMA
Now part of Eficode

II. → More Missing Pieces?

Auto-bootstrapping

Our `super-seed.groovy` *must* run once

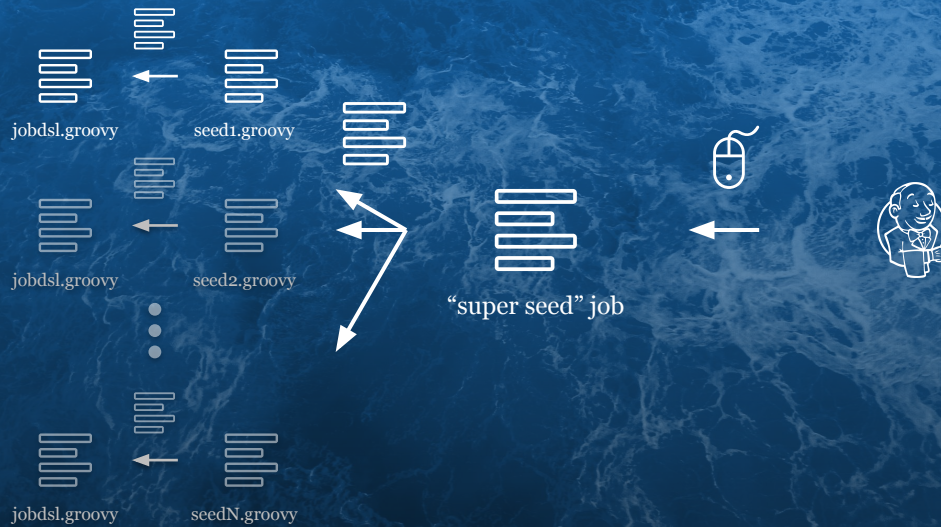
jenkins.yml:

```
jobs:
- script: >
  job('super-seed') {
    triggers {
      cron('H/2 * * * *')
    }
    ...
  }
```

@figaw

PRAQMA
Now part of Eficode

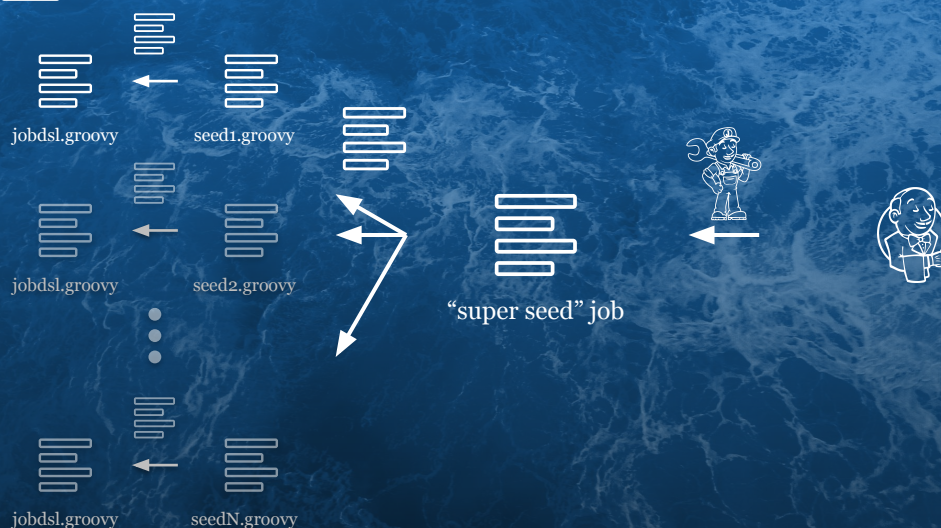
II. Bootstrapping all the Jobs



@figaw

PRAQMA
Now part of Elicode

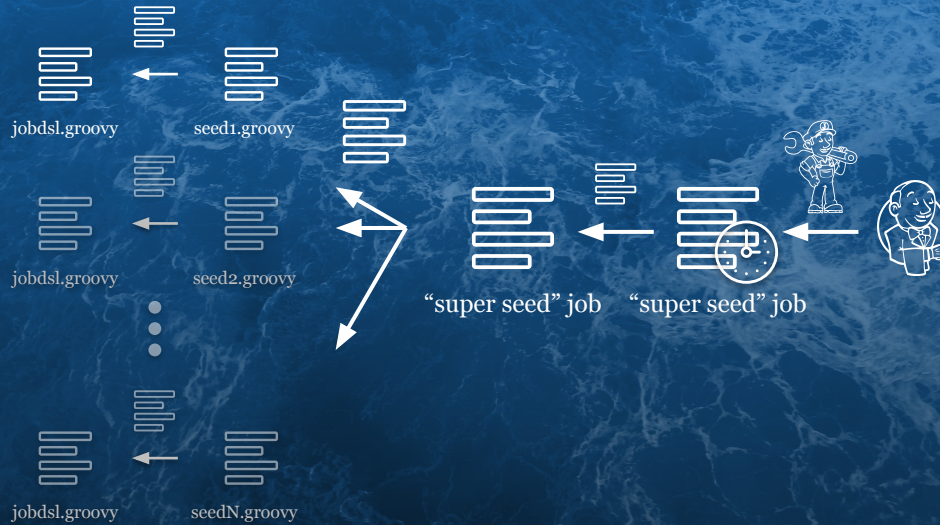
II. Bootstrapping all the Jobs



@figaw

PRAQMA
Now part of Elicode

II. Auto-Bootstrapping



@figaw

PRAQMA
Now part of Elicode

II. → Demo time?

No demo, talk about caveats.



Source: <https://www.artstation.com/artwork/1XmOe>

@figaw

PRAQMA
Now part of Elicode



@figaw

PRAQMA
Now part of Eficode



Unified way of Managing Platform

Scaling with Kubernetes build Agents

With custom images!

@figaw

PRAQMA
Now part of Eficode

III. → Caveats

Helm and Files in the Environment

What do we do in K8s? *Configmaps and Secrets!*

@figaw

PRAQMA
Now part of Elicode

III. → Demo Time!

Will Show:

- Installing our awesome Jenkins with Helm3
- Supply **secrets** for a Helm3 chart
- Configuring custom Agent Images

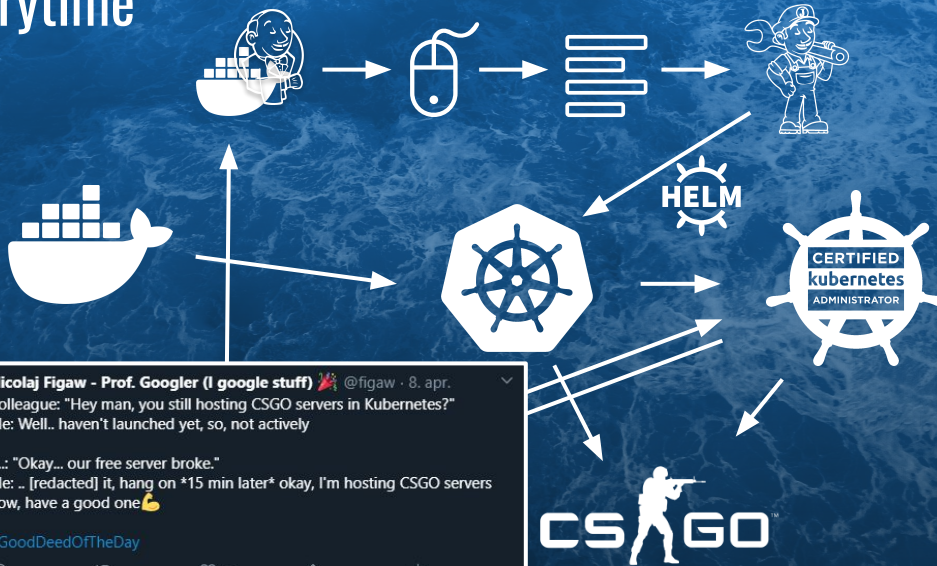


Source: <https://www.artstation.com/artwork/1XmOe>

@figaw

PRAQMA
Now part of Elicode

Storytime



<https://twitter.com/figaw/status/1247997695014436866>

@figaw

PRAQMA
Now part of Eficode

THANK YOU!

Questions?

*Nicolaj Græsholt - @figaw on Twitter
DevOps Consultant at Eficode Praqma*

eficode

PRAQMA
Now part of Eficode