Entwicklungsumgebungen

https://job-dsl.herokuapp.com/	Playground zur Jobs DSL

Plugins - Build Trigger

* https://plugins.jenkins.io/selfie-trigger-plugin
*

Pipeline Trigger Job

```
# Build Trigger - Zeitgesteuert
TZ=Eur ope /Berlin
@midnight
# Pipeline - Definition Pipeli nscript
node {
    sta ge( " myp ock etm od"){
        try {
            build 'mypoc ket mod /ma ster'
        } catch (err) {
            echo " Caught: ${err} "
        }
        try {
            build 'rades -an not ations'
        } catch (err) {
            echo " Caught: ${err} "
        }
        try {
            build 'localmod'
        } catch (err) {
            echo " Caught: ${err} "
        }
    }
}
```

Monitoring API

Simple Job

```
Builds tatus URL: jenkin sur l/j ob/ job Nam e/l -
ast Bui ld/ api /json
Beispiel: http:/ /st ach el: 809 0/j ob/ rad es- -
ann ota tio ns/ las tBu ild /ap i/json
```

Multib ranch Pipeline Plugin

```
Builds tatus URL: jenkin sur l/j ob/ rep oNa me/ -
job /Br anc hNa me/ las tBu ild /ap i/json
Beispiel: http:/ /st ach el: 809 0/j ob/ myp ock -
etm od/ job /ma ste r/l ast Bui ld/ api /json
```

Monitoring API (cont)

```
> Im Response sind die wichtigen Felder:
* response.fullDisplayName
* response.result
{
  "_class": "hudson.model.FreeStyleBuild",
  "actions": [
    {
      "_class": "hudson.model.CauseAction",
      "causes": [
        {
          "_class": "hudson.model.Cause$UpstreamCause",
          "shortDescription": "Gestartet durch vorgelagertes Projekt
„NighlyTrigger", Build 6",
          "upstreamBuild": 6,
          "upstreamProject": "NighlyTrigger",
          "upstreamUrl": "job/NighlyTrigger/"
        }
      ]
    },
    {},
    {
      "_class": "hudson.plugins.git.util.BuildData",
      "buildsByBranchName": {
        "refs/remotes/origin/master": {
          "_class": "hudson.plugins.git.util.Build",
          "buildNumber": 6,
          "buildResult": null,
          "marked": {
            "SHA1": "d68e6e7cbc7155a549e480e7354a6e644933-
82fa",
            "branch": [
              {
                "SHA1": "d68e6e7cbc7155a549e480e7354a6e644-
93382fa",
                "name": "refs/remotes/origin/master"
              }
            ]
          },
          "revision": {
```

Monitoring API (cont)

```
>          "SHA1": "d68e6e7cbc7155a549e480e7354a6e644933-
82fa",
           "branch": [
             {
               "SHA1": "d68e6e7cbc7155a549e480e7354a6e644-
93382fa",
               "name": "refs/remotes/origin/master"
             }
           ]
         }
       },
       "lastBuiltRevision": {
         "SHA1": "d68e6e7cbc7155a549e480e7354a6e64493382fa",
         "branch": [
           {
             "SHA1": "d68e6e7cbc7155a549e480e7354a6e644933-
82fa",
             "name": "refs/remotes/origin/master"
           }
         ]
       },
       "remoteUrls": [
         "https://github.com/FunThomas424242/rades-annotations.git"
       ],
       "scmName": ""
     },
     {
       "_class": "hudson.plugins.git.GitTagAction"
     },
     {},
     {},
     {}
  ],
  "artifacts": [],
  "building": false,
  "description": null,
  "displayName": "#6",
  "duration": 28582,
```

Monitoring API (cont)

```
>    "estimatedDuration": 47475,
  "executor": null,
  "fullDisplayName": "rades-annotations #6",
  "id": "6",
  "keepLog": false,
  "number": 6,
  "queueId": 39,
  "result": "SUCCESS",
  "timestamp": 1568557708973,
  "url": "http://stachel:8090/job/rades-annotations/6/",
  "builtOn": "",
  "changeSet": {
    "_class": "hudson.plugins.git.GitChangeSetList",
    "items": [],
    "kind": "git"
  },
  "culprits": []
}
```

Quelle: https://serverfault.com/questions/309848/how-do-i-check-the-build-status-of-a-jenkins-build-from-the-command-line

Spielejenkins im Internet bereitstellen

Lässt sich über Docker für max. 3h realisieren:

1. Aufruf der Dockerspielwiese: https://labs.play-with-docker.com
2. Add new Instanz
3. docker pull jenkins
4. docker run -p 8080:8080 -p 50000:50000 jenkins
5. Ablesen der IP und Ports der Umgebung
6. Öffnen der Jenkinsseite über den ausgelesenen 8080 z.B.
http://ip172-18-0-7-bmh2hhlfhn20008g21v0-8080.direct.labs.play--with-docker.com/login?from=%2F
7. In der Console nach oben scrollen und das Jenkins admin Passwort herauskopieren. Sieht etwa so aus: fd87362a76514aa997-3631c99257ecb4
8. Im geöffneten Jenkins anmelden (mit dem Passwort)

Manchmal machte die Installation mit default Plugins Probleme, daher besser erstmal keine zu installieren und dann manuell später nachholen

Jenkins - Monitore

https://github.com/gg-irou/yet-another-jenkins-notifier	Chrome Extension	Nur auf Jobebene bei aktiver Authentifizierung
https://github.com/zi-onyx/jenkins-tray	Natives Binary	Multi Jenkins Support mit Tray aber die Liste öffnet im extra Fenster
========	======	======
HudsonTracker		Unterstützt die Jenkins Authorisierung nicht
http://fieryjoy.com/kato	Natives Binary	Benötigt Download

Zeitgesteuerte Ausführung

Pipeline mit Jenkinsfile

```
pipeline {
    agent any
    triggers {
        cron('H /4  * 1-5')
    }
    stages {
        stage( 'Ex ample') {
            steps {
                echo 'Hello World'
            }
        }
    }
}
`
String cron_string = BRANCH_NAME == "master" ? "@hourly" : ""
pipeline {
agent none
triggers { cron(cron_string) }
stages {
// do something
}
```

Zeitgesteuerte Ausführung (cont)

```
}
`
```

Quellen:
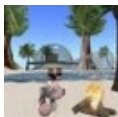* https://stackoverflow.com/questions/32028761/jenkins-build-pipe-line-scheduled-trigger
* https://stackoverflow.com/questions/39168861/build-periodically-with-a-multi-branch-pipeline-in-jenkins/44902622#44902622