CSCS
Centro Svizzero di Calcolo Scientifico
Swiss National Supercomputing Centre

ETH zürich

Jenkins service for CI/CD

CSCS User Lab Day – Meet the Swiss National Supercomputing Center

Manitaras Theofilos-Ioannis, CSCS

September 2, 2022

Outline

- Continuous Integration & Jenkins in a Nutshell
- Accessing & using the CSCS CI Jenkins instance
- Best practices for Jenkins at CSCS
- Triggering Jenkins Jobs
- Conclusions

Continuous Integration & Jenkins in a Nutshell

Continuous Integration (CI) & Jenkins

- CI is a software development practice where code changes are continuously integrated in a shared code basis
- Each addition/change is tested via an automated pipeline
- Adoption of CI allows catching errors sooner
- Code conflicts are reduced due to the "continuous" nature of the practice
- Jenkins is an open source automation server which automates the non-human part of the software development process
- It offers a large number of plugins which enhance its capabilities.
- Compatible with all the popular software repositories, e.g GitHub, GitLab, BitBucket etc.

Typical CI Workflow

1. Each developer owns a separate fork of the project
2. A change to the code base is submitted via a pull/merge request to the main/develop repository branch
3. An automated CI pipeline is triggered to ensure that the introduced change passes the test suite
4. The result of the CI is reported, notifying the developers if needed
5. The proposed change is reviewed and polished
6. The change is accepted and merged to the main branch
7. The software is redeployed to production (optional)

Accessing & using the CSCS CI Jenkins instance

Gaining access to the CSCS CI

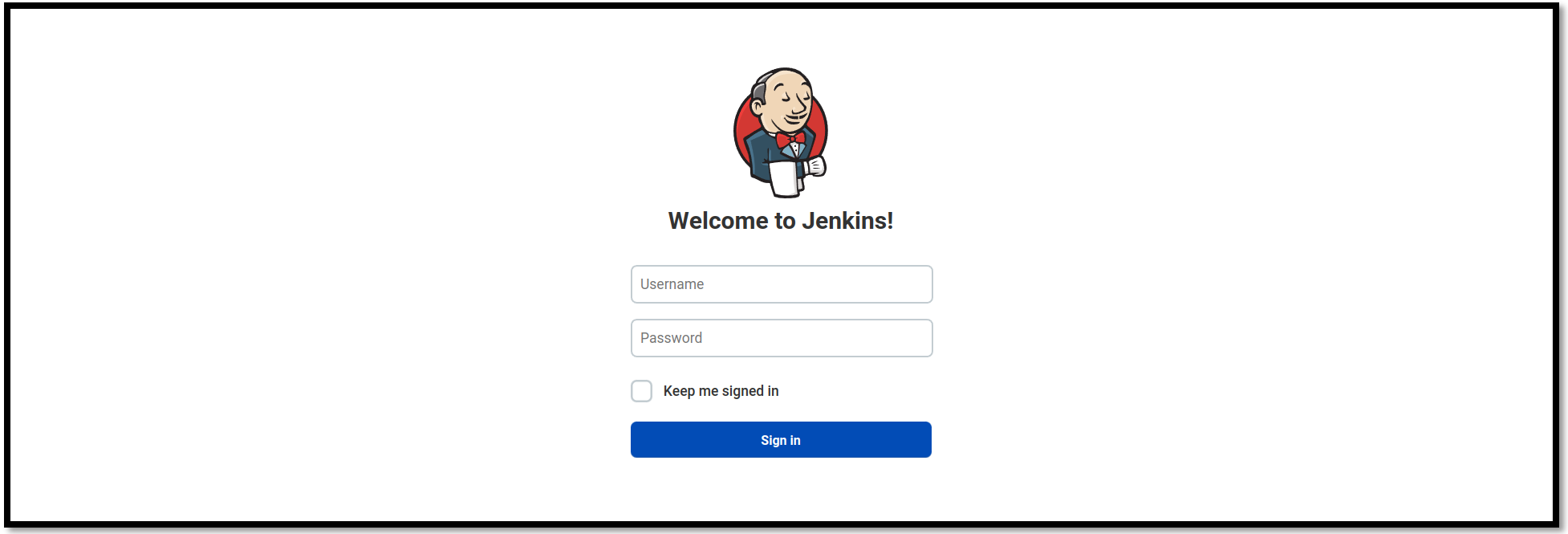
1. The PI of a project requests access to the service by opening an issue at help@cscs.ch
2. A Jenkins folder with the corresponding name is created
3. A new user belonging to the project is created
4. A Jenkins agent(node) is associated to the folder
5. The pipelines run on the target system (Daint) on behalf of the above user

Logging to the Jenkins Web interface (1/2)

- The CSCS Jenkins instance is not available in public Internet
- Local port forwarding has to be performed via ssh
- The end user has to forward a local port to **lisone.cscs.ch:443** via **ela.cscs.ch**
- For Linux/Mac users the forwarding can be perfomed from the shell:
  ssh -L 7000:lisone.cscs.ch:443 ela.cscs.ch
- The Jenkins web interface can now be accessed using a web browser through **https://ci.cscs.ch:7000/**
- The above local port (7000) is chosen by the end user

Logging to the Jenkins Web interface (2/2)

Use your CSCS credentials to login

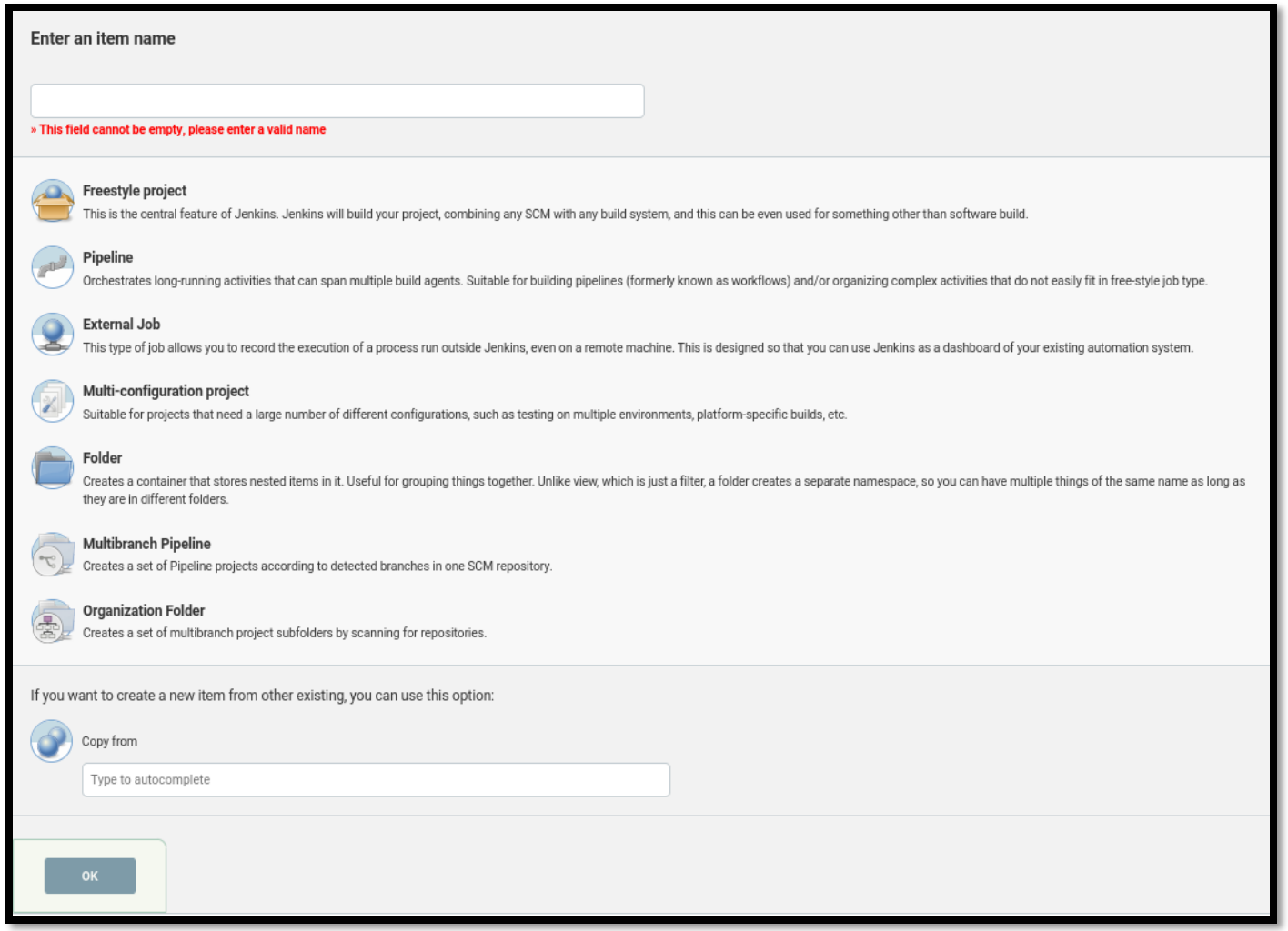


Dedicated Folde – Agents per Project

Dedicated folder and Jenkins agent

Creating a new Jenkins project

Creating a new project is straightforward by selecting **New Item**:



Pipeline as Code – The Jenkinsfile

The Jenkinsfile is at the core of the Jenkins pipeline-as-code approach serving a dual purpose:

1. Describes via a domain specific language (DSL) the actions performed when a build is triggered
2. Existence of a Jenkinsfile signifies that the project is using Jenkins for it's CI

Jenkinsfiles come in two alternative flavors:

1. **Scripted Pipeline**: a Groovy based script described the actions performed by the Jenkins build
2. **Declarative Pipeline**: an easier to follow declarative syntax based on simple constructs specifying Jenkins actions

Structure of a Jenkinsfile

The structure of a Jenkinsfile is different, depending on the syntax flavor:

Scripted

```
stage("Stage 1") {
   node("mynode1") {
      <step 1>
         .
         .
         .
      <step n>
   }
}

stage("Stage N") {
   node("mynode2") {
      <step 1>
         .
         .
         .
      <step n>
   }
}
```

Declarative

```
pipeline {
      stages {
       stage("Stage 1") {
          agent { node { label 'mynode1} }
          steps {
             <step 1>
                .
                .
             <step n>
          }
       }
       stage("Stage 2") {
           agent { node { label 'mynode2} }
          steps {
             <step 1>
                .
                .
             <step n>
          }
       }
   }
}
```

Nodes/Agents

- The Jenkins **Built-In-Node** is the system running the jenkins instance and is not intended to run any actual jobs
- A **Node** in Jenkins terms is any system capable of executing jobs, e.g a remote machine, a VM, a container.
- An **Agent** is the declarative pipeline equivalent of a node other than the build-in node, executing Jenkins Jobs
- In the CSCS setup, the nodes/agents consist of lightweight VMs which allow for batch job submission but have limited functinality otherwise

CSCS
Centro Svizzero di Calcolo Scientifico
Swiss National Supercomputing Centre

ETH zürich

Best practices for Jenkins at CSCS

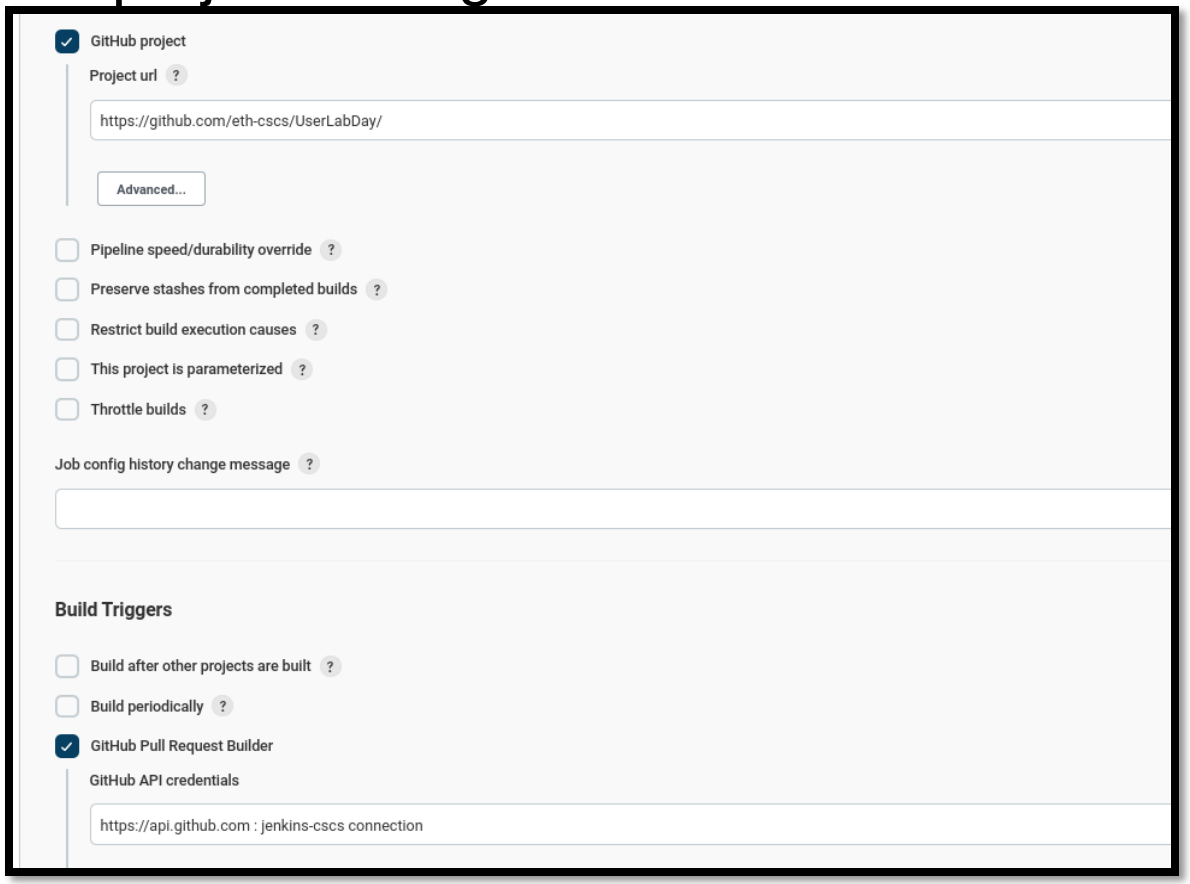
Best Practices

- Adopt the pipeline-as code modern approach of Jenkins 2 including the **Jenkinsfile** in your git repository
- Use the **--wait** option to submit batch jobs using **sbatch**:
  sbatch -wait <batch_script>
- For single node jobs use the **cscsci** partition offering higher priority and is suitable for ci
- Copy the build output/error on **SCRATCH** to be able to access via your user
- Make use of artifacts to store output/error and produced binaries

Triggering Jenkins Jobs

Enabling the ghprb for your project (1/2)

1. Invite the **jenkins-cscs** GitHub user in your project, setup by CSCS with both read & write privileges
2. Enable ghprb on your project configuration:



New PR triggers the build

Search or jump to… / Pull requests Issues Marketplace Explore

eth-cscs / UserLabDay

Unwatch 13 Star 0 Fork 0

Code Issues 0 Pull requests 1 Projects 0 Wiki Insights Settings

Add hostname command #4

Edit

Open teojgo wants to merge 1 commit into `master` from `feature/add_hostname`

Conversation 0 Commits 1 Checks 0 Files changed 1 +1 −0

teojgo commented 6 days ago Member

No description provided.

Add hostname command 22ffbd5

teojgo self-assigned this 6 days ago

Add more commits by pushing to the **feature/add_hostname** branch on **eth-cscs/UserLabDay**.

All checks have passed
1 successful check
Hide all checks

CSCS CI Details

This branch has no conflicts with the base branch
Merging can be performed automatically.

Reviewers
No reviews

Assignees
teojgo

Labels
None yet

Projects
None yet

Milestone
No milestone

Notifications
Unsubscribe

Retriggering a build

Add hostname command #4

Edit

Open **teojgo** wants to merge 1 commit into `master` from `feature/add_hostname`

Conversation 0 | Commits 1 | Checks 0 | Files changed 1 | +1 −0

teojgo commented 6 days ago — Member

No description provided.

`Add hostname command` 22ffbd5

teojgo self-assigned this 6 days ago

teojgo commented 16 seconds ago — Member

retest this please

Add more commits by pushing to the **`feature/add_hostname`** branch on **eth-cscs/UserLabDay**.

Some checks haven't completed yet — Hide all checks

1 pending check

CSCS CI *Pending — Build started for merge commit.* Details

This branch has no conflicts with the base branch

Merging can be performed automatically.

Merge pull request or view command line instructions.

Reviewers

No reviews

Assignees

teojgo

Labels

None yet

Projects

None yet

Milestone

No milestone

Notifications

Unsubscribe

You're receiving notifications because you were assigned.

1 participant

PR from non-whitelisted user

rsarm commented 5 minutes ago — Member

No description provided.

`Test commit` — b420dd3

jenkins-cscs commented 4 minutes ago — Collaborator

Can one of the admins verify this patch?

teojgo commented 37 seconds ago — Member

test this please

Add more commits by pushing to the **`test-branch`** branch on **eth-cscs/UserLabDay**.

Some checks haven't completed yet — Hide all checks

1 pending check

CSCS CI *Pending — Build triggered for merge commit.*

This branch has no conflicts with the base branch

Merging can be performed automatically.

Merge pull request or view command line instructions.

Write Preview

Reviewers

No reviews

Assignees

No one—assign yourself

Labels

None yet

Projects

None yet

Milestone

No milestone

Notifications

Unsubscribe

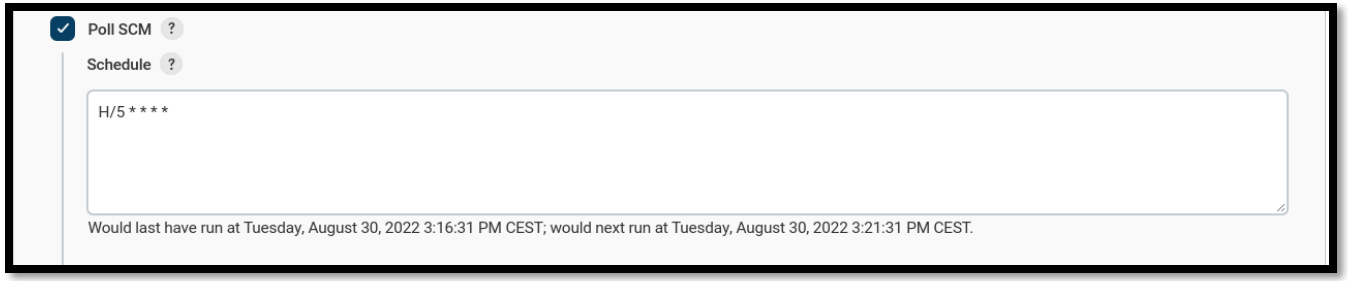
You're receiving notifications because you commented.

3 participants

Lock conversation

Polling a remote repository

- An alternative to ghprb since the CSCS Jenkins is polling the remote repository for changes
- The option can be enabled by selecting **Poll SCM** under Build Triggers:



Conclusions

Conclusions

- The Jenkins CSCS CI service can be used to test software on the actual HPC systems
- Access to the service is requested by a PI of a project
- Several best practices must be followed to effectively incorporate the CI on your project
- Ghprb and Polling are two of the proposed ways to trigger pipelines based on PRs/MRs

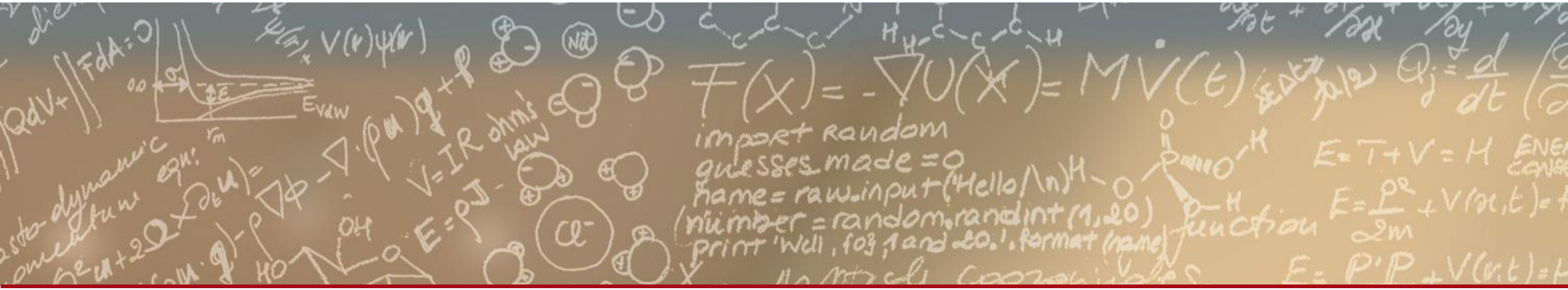
Additional Resources

Online Resources:

- ❑ CSCS CI documentation
- ❑ Jenkins User Documentation
- ❑ Jenkins Handbook

Books:

- ❑ Jenkins 2 Up and Running
- ▪ CI/CD Pipeline Using Jenkins Unleashed
- ▪ Jenkins Administrator's Guide



Thank you for your attention.