

# EURO

Source control management with Git at Gitlab, Github and Bitbucket

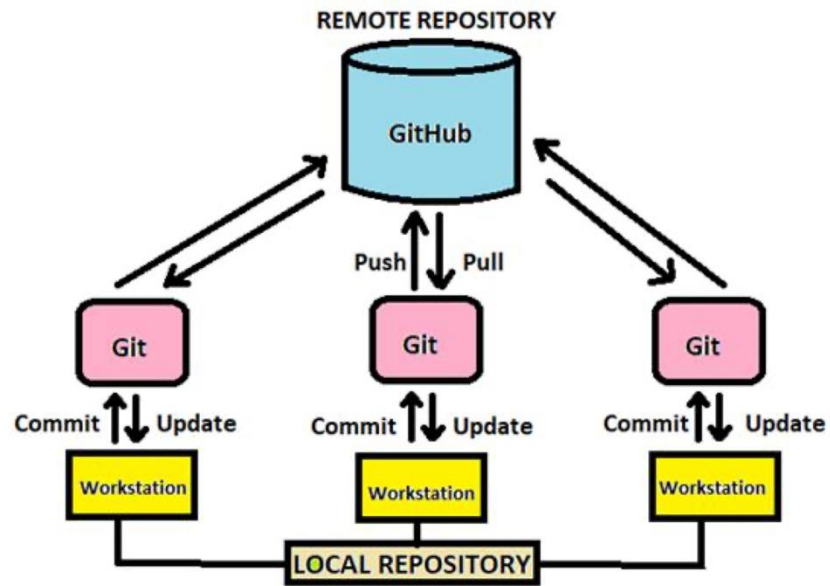
Leon Kos, University of Ljubljana

23 April 2026, Zoom meeting

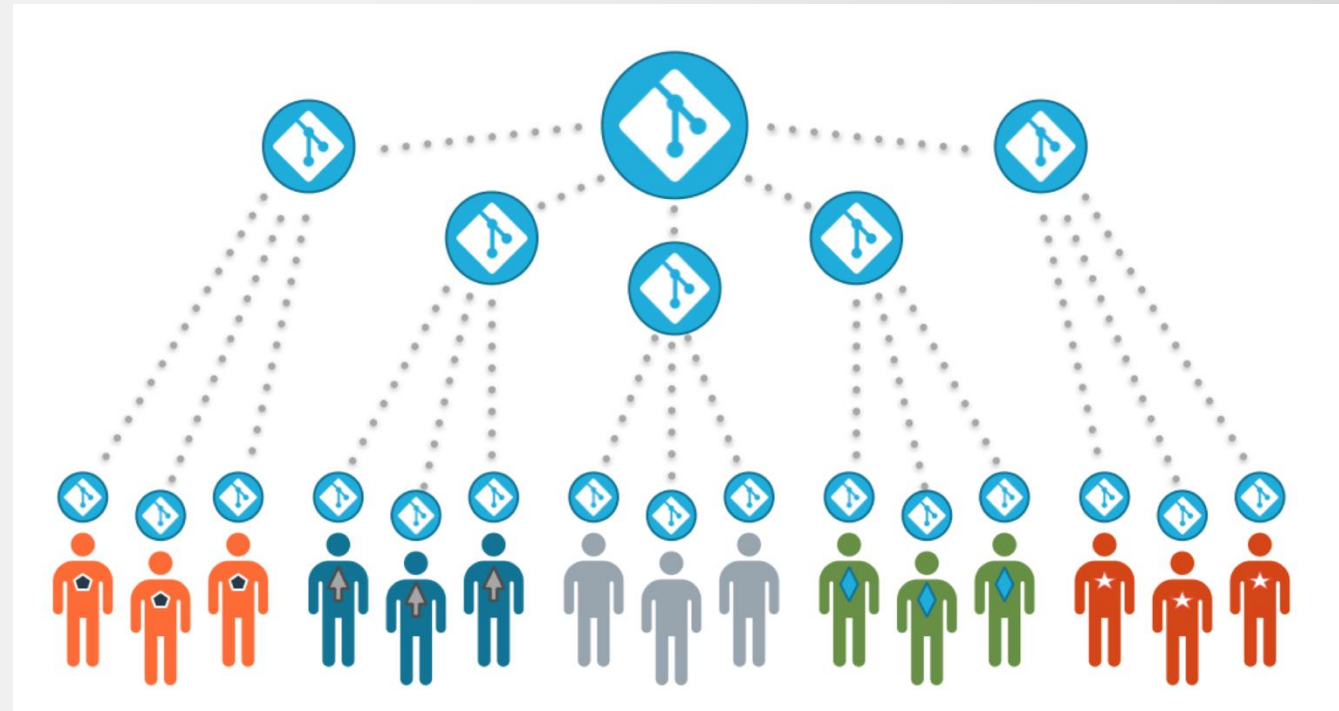


***Source code management (SCM) with Git provides support for versioning with branching and merging in a collaborative work. Git as distributed SCM is usually connected to a central Git server that provides web functionality for source review, pull requests and integration with other services such as continuous integration (CI) and code documentation. In this seminar we will take a look into Git development process and popular servers and integrated services (CI, Read the docs). How to Git version large files in HPC environment along with code and documentation will be discussed from practical and data provenance viewpoint.***

# Distributed SCM



Vir: <https://tutorialslink.com/Articles/What-is-Git/1867>



# Git – daily work from command line



## Setup

1. Add ~/.ssh/id\_rsa.pub to the list of SSH keys on the portal
2. `git clone ssh://git@git.../...git`
3. `git branch -r`
4. `git checkout develop`
5. `git add .gitignore`

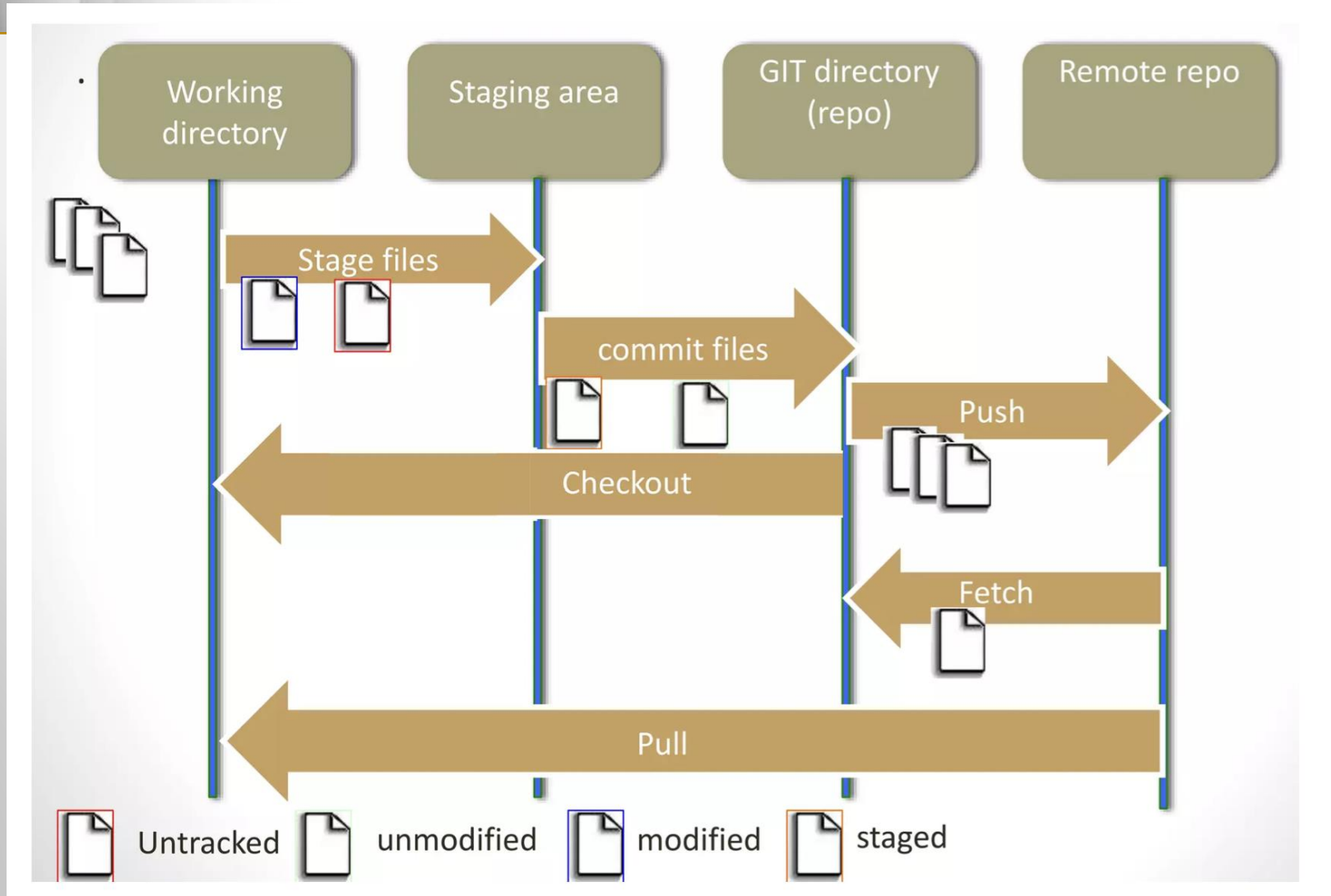
## Develop

1. `git status`
2. `git add file ...`
3. `git pull`
4. `git status`
5. `git commit -m "Short description of changes"`
6. `git push`

# Flow

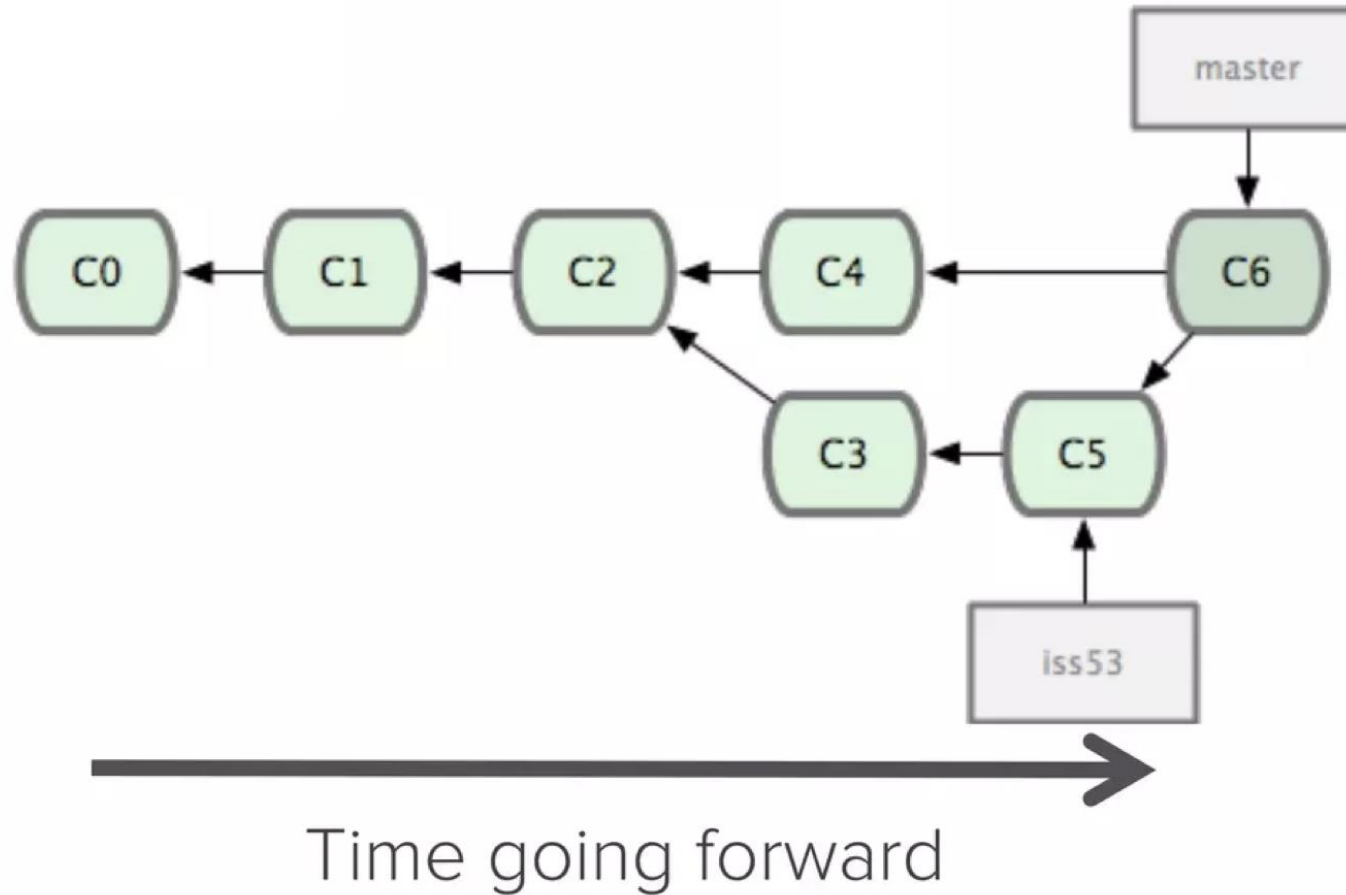


EURO



# Git merge

o z git merge.



# Gitlab



- Good for institutional hosting by deploying Community Edition  
[gitlab.com/gitlab-org/gitlab](https://gitlab.com/gitlab-org/gitlab)
- Not that open anymore (Free trials)  
<https://gitlab.com/explore>
- Integrated Wiki, Snippets, Analytics
- Continuous Integration with Kubernetes
- Demo of projects at  
<https://gitlab.eudat.eu/eudat-prace-2019>

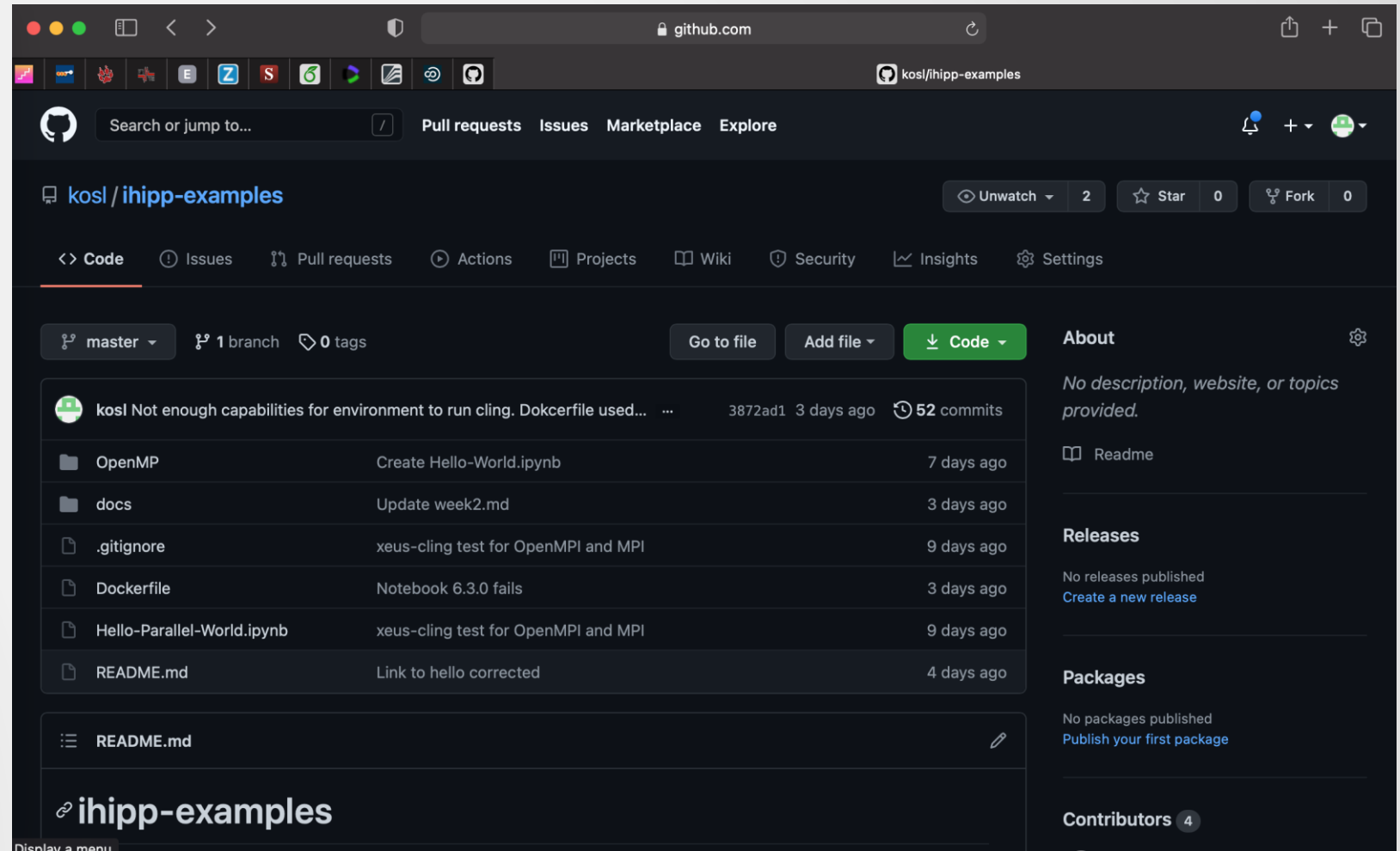
The screenshot shows the GitLab interface for a project named 'computing-architecture'. The left sidebar contains navigation options: Project overview, Details, Activity, Releases, Repository, Issues (0), Merge Requests (0), CI / CD, Operations, Analytics, Wiki, Snippets, and Members. The main content area displays project statistics (4 Commits, 1 Branch, 0 Tags, 143 KB Files, 143 KB Storage) and a file browser for the 'master' branch. A recent commit by Leon Kos is highlighted, showing the addition of 'miniconda.yaml'. Below this, there are buttons for adding various files like README, LICENSE, CHANGELOG, CONTRIBUTING, and setting up CI/CD. A table lists the project's files and their last commit details.

Name	Last commit	Last update
.gitignore	MPI in Python	1 year ago
README.md	Add miniconda yaml	1 year ago
conda-data_analysis-env.yaml	Add miniconda yaml	1 year ago
conda-rtd-env.yaml	Add miniconda yaml	1 year ago
hello-mpi.py	MPI in Python	1 year ago
hello-mpi.sbatch	MPI in Python	1 year ago

# Github

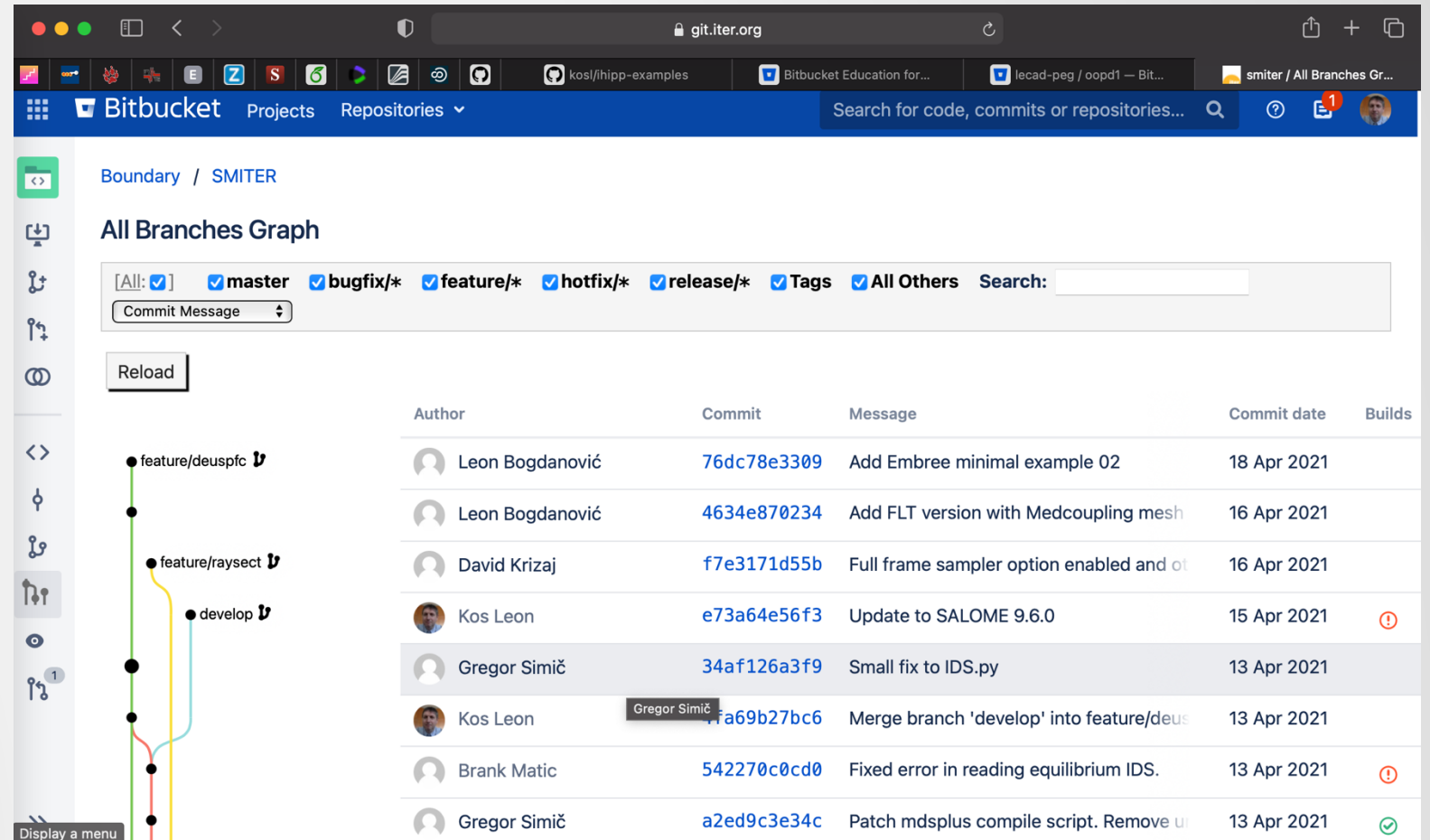


- Open source hosting friendly (e.g. <https://github.com/torvalds/linux>)
- Integration to many other websites (Overleaf, Binder, CircleCI, ...)
- Private repositories, Gists
- Not that great interface
- Many source code rendering and integrated editors
- Github desktop for beginners
- \*.github.io websites
- Actions
- Demo of project at <https://github.com/kosl/ihipp-examples>



# Bitbucket

- Commercial hosting only
- Form 2024 only as cloud offering
- Great Web based integration (Confluence tools, Stash, Jira, Bamboo, Servicedesk, ... )
- Free for academic hosting  
<https://bitbucket.org/product/education> Private repositories
- Limited source code rendering and integrated editors
- Demo of the project at  
<https://bitbucket.org/lecad-peg/oopd1/>
- Demo of Bamboo CI agent



The screenshot shows the Bitbucket web interface for a repository named 'Boundary / SMITER'. The 'All Branches Graph' is visible on the left, showing branches like 'feature/deuspfic', 'feature/raysect', and 'develop'. The main area displays a table of recent commits.

Author	Commit	Message	Commit date	Builds
Leon Bogdanović	76dc78e3309	Add Embree minimal example 02	18 Apr 2021	
Leon Bogdanović	4634e870234	Add FLT version with Medcoupling mesh	16 Apr 2021	
David Krizaj	f7e3171d55b	Full frame sampler option enabled and ot	16 Apr 2021	
Kos Leon	e73a64e56f3	Update to SALOME 9.6.0	15 Apr 2021	🚫
Gregor Simič	34af126a3f9	Small fix to IDS.py	13 Apr 2021	
Kos Leon	fa69b27bc6	Merge branch 'develop' into feature/deus	13 Apr 2021	
Brank Matic	542270c0cd0	Fixed error in reading equilibrium IDS.	13 Apr 2021	🚫
Gregor Simič	a2ed9c3e34c	Patch mdsplus compile script. Remove u	13 Apr 2021	✅

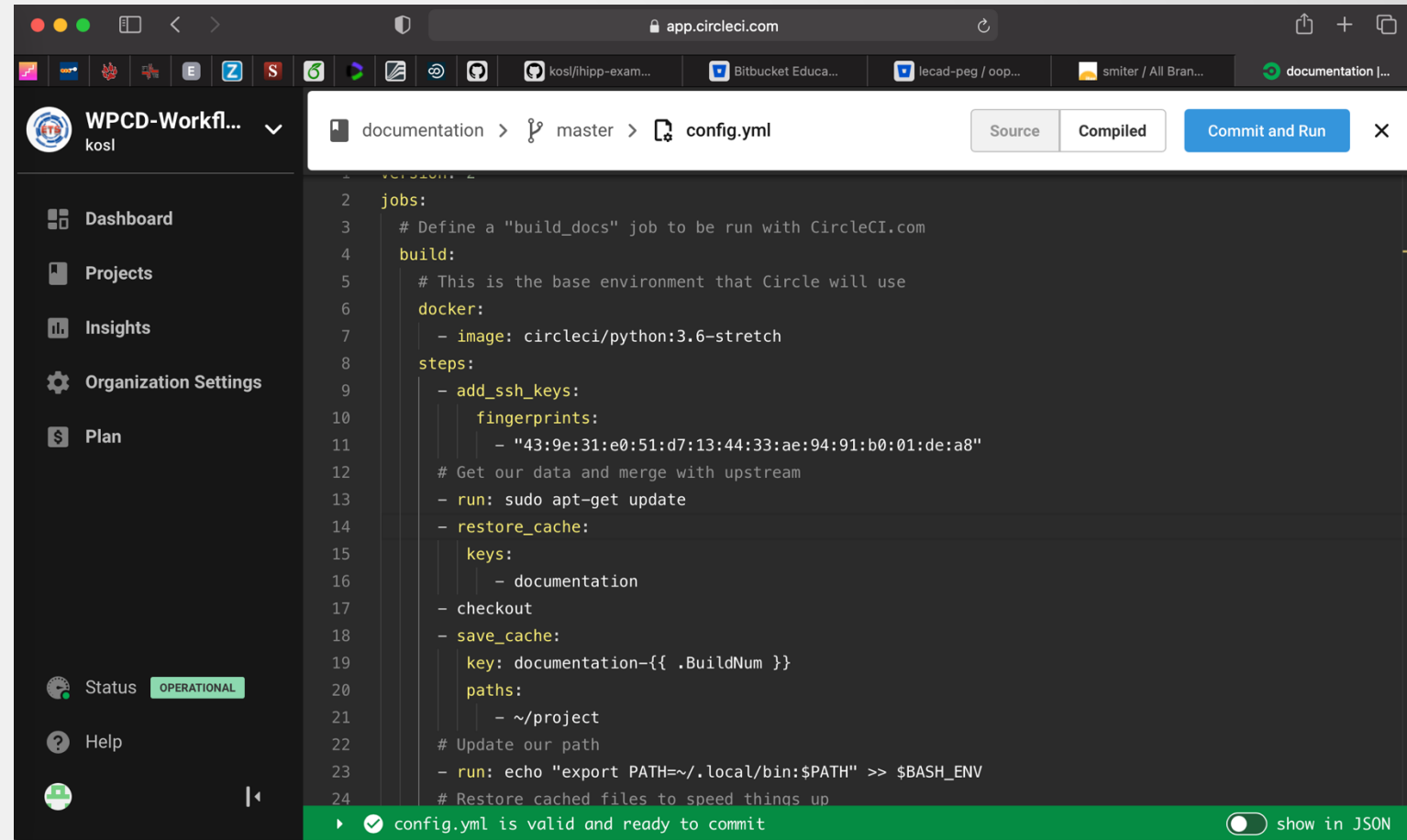
# Continuous Integration (Circle CI)



- Testing and building automatism
  - Testing on different compilers and platforms
  - Build Releases
  - Documentation
- Free minutes can be exhausted unless caches are used for rebuild

Demo of CircleCI

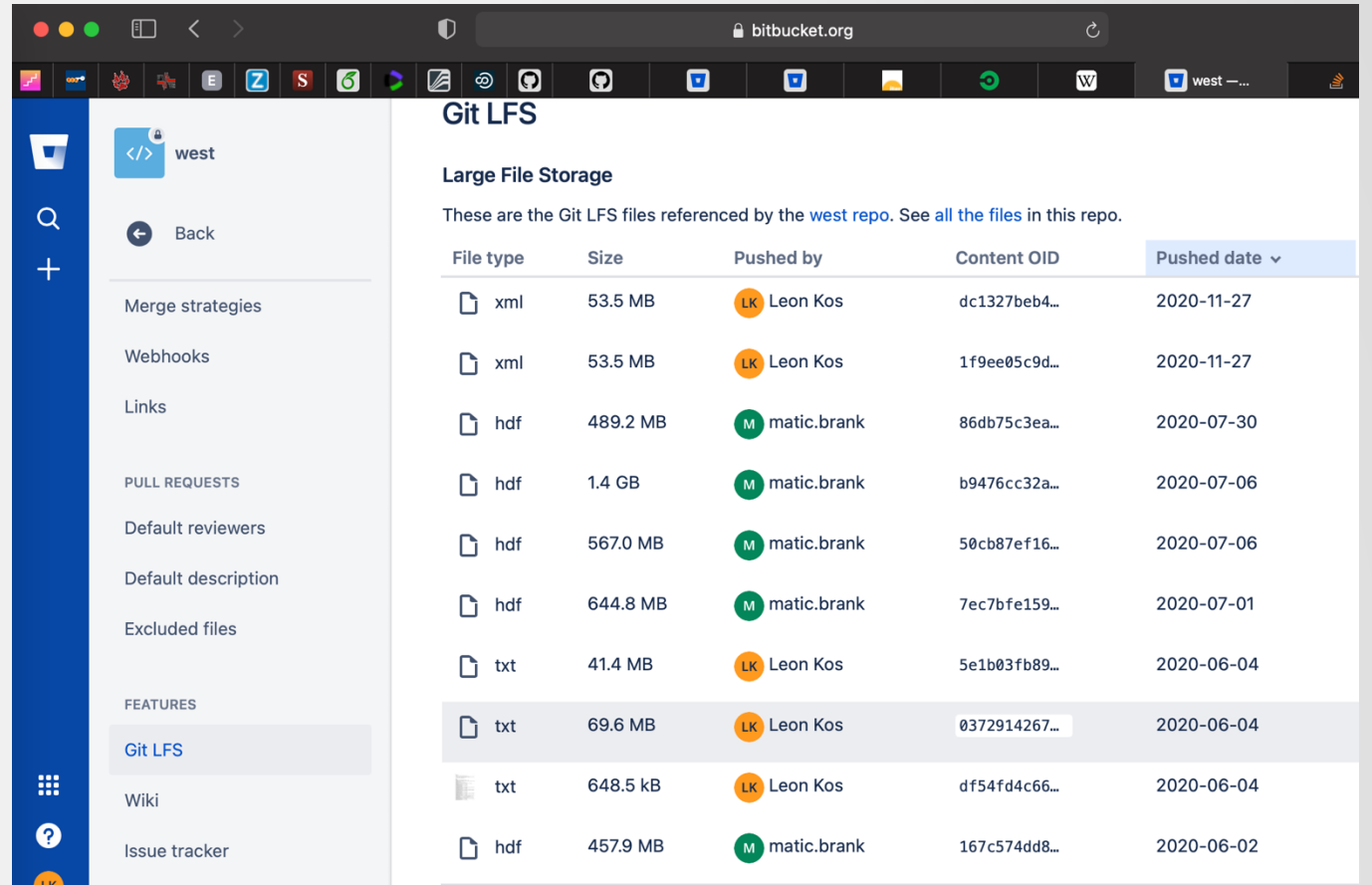
<https://app.circleci.com/projects/github/WPCD-Workflows/documentation/config/>

A screenshot of the CircleCI web interface. The browser address bar shows 'app.circleci.com'. The page title is 'WPCD-Workfl...' and the user is 'kosl'. The left sidebar contains navigation links: Dashboard, Projects, Insights, Organization Settings, and Plan. The main content area shows a configuration file named 'config.yml' for the 'documentation' job on the 'master' branch. The configuration includes a 'build' job with a 'docker' environment (image: circleci/python:3.6-stretch) and several 'steps': 'add\_ssh\_keys' (with fingerprints), 'checkout', 'save\_cache' (key: documentation-{{ .BuildNum }}, paths: ~/project), and 'run' (echoing the PATH environment variable). A green status bar at the bottom indicates 'config.yml is valid and ready to commit'.

```
1 version: 2.1
2 jobs:
3   # Define a "build_docs" job to be run with CircleCI.com
4   build:
5     # This is the base environment that Circle will use
6     docker:
7       - image: circleci/python:3.6-stretch
8     steps:
9       - add_ssh_keys:
10         fingerprints:
11           - "43:9e:31:e0:51:d7:13:44:33:ae:94:91:b0:01:de:a8"
12       # Get our data and merge with upstream
13       - run: sudo apt-get update
14       - restore_cache:
15         keys:
16           - documentation
17       - checkout
18       - save_cache:
19         key: documentation-{{ .BuildNum }}
20         paths:
21           - ~/project
22       # Update our path
23       - run: echo "export PATH=~/.local/bin:$PATH" >> $BASH_ENV
24       # Restore cached files to speed things up
```

# Git Large File Support (LFS)

- Install from <https://git-lfs.github.com>
  - git lfs install
  - git lfs track "\*.stp"
  - git add .gitattributes
  - git lfs pull
- Limited storage size (5GB) on public repositories. Use [git filter-repo](#) to remove the objects from the repository.
- No site-wide cache!
  - Increased local storage
  - bandwidth

A screenshot of the Bitbucket web interface showing the 'Git LFS' section for a repository named 'west'. The interface displays a table of large file storage entries with columns for file type, size, pushed by, content OID, and pushed date. The table lists several files, including XML, HDF, and TXT files, with sizes ranging from 41.4 MB to 1.4 GB. The user 'Leon Kos' is shown as the pusher for several files, while 'matic.branks' is shown for others. The interface also includes a sidebar with navigation options like 'Merge strategies', 'Webhooks', and 'Links', and a top navigation bar with the Bitbucket logo and repository name.

File type	Size	Pushed by	Content OID	Pushed date
xml	53.5 MB	LK Leon Kos	dc1327beb4...	2020-11-27
xml	53.5 MB	LK Leon Kos	1f9ee05c9d...	2020-11-27
hdf	489.2 MB	M matic.branks	86db75c3ea...	2020-07-30
hdf	1.4 GB	M matic.branks	b9476cc32a...	2020-07-06
hdf	567.0 MB	M matic.branks	50cb87ef16...	2020-07-06
hdf	644.8 MB	M matic.branks	7ec7bfe159...	2020-07-01
txt	41.4 MB	LK Leon Kos	5e1b03fb89...	2020-06-04
txt	69.6 MB	LK Leon Kos	0372914267...	2020-06-04
txt	648.5 kB	LK Leon Kos	df54fd4c66...	2020-06-04
hdf	457.9 MB	M matic.branks	167c574dd8...	2020-06-02

# „Externaldata“ Git LFS support



- External repository to push assets
  - http for public hashes
  - ssh for private repos
- Site-wide cache possible
  - One copy only with symlinks
- Github example <https://github.com/WPCD-Workflows/documentation/blob/master/Makefile>
- GForge SSH Kerberos example <https://gforge-next.eufus.eu/#/project/publicdb/scm/file/Makefile>
- SMITER demo

```
258 # If the asset already exist the we don't want to change its timestamp as
259 # this may impact other's (re)builds. Therefore MD5 and timestamps needs
260 # to be backdated in this case.
261 % : %.md5-stamp
262     if test -f $(ASSETDIR)/$(shell cat $^); \
263     then touch -r $(ASSETDIR)/$(shell cat $^) -d "-1 minute" $^; \
264         touch -r $^ -d "-1 minute" $(patsubst %.md5-stamp, %.md5, $^); \
265     else make $(ASSETDIR)/$(shell cat $^); fi
266     $(ASSET_LINK) $(ASSETDIR)/$(shell cat $^) $@
267
268 %.md5-stamp : %.md5
269     cat $^ > $@
270
271 % : %.zip
272     unzip -q -d $@ $^
273
274 # Downloads asset and makes it available to the group for linking to it
275 $(ASSETDIR)/% :
276     if ! test -d $(ASSETDIR); \
277     then mkdir -m 3775 -p $(ASSETDIR); \
278     if test "x$(ASSETDIR_GROUP)" != "x"; \
279     then chgrp $(ASSETDIR_GROUP) $(ASSETDIR); \
280     fi; \
281     fi
282     curl --location --fail --insecure \
283         --output $(ASSETDIR)/$(@F) $(EXTERNALDATA)/$(@F)
284     if test $$$(md5sum $(ASSETDIR)/$(@F)|cut -c 1-32) = $(@F) ; then \
285     chmod g+rw $(ASSETDIR)/$(@F); \
286     else \
287     rm -f $(ASSETDIR)/$(@F); \
288     echo "Error: Checksum of $(ASSETDIR)/$(@F) incorrect! Removing it."; \
289     exit 1; \
a menu     fi
```

# Restructured text with Sphinx

- Repository should version
  - Source Code
  - Large data
  - Documentation
- Data provenance
- Containers?
- Binder?



The image displays the SMITER web interface and its graphical user interface (GUI) for ITER. The top part shows the static.iter.org website with a search bar and a table of contents. The bottom part shows a multi-panel GUI with various plots and 3D models of the ITER tokamak components.

**1. Introduction**

1.1. Power Deposition Model

1.2. Workflow

1.3. Description of Constituent Codes

1.4. Graphical User Interface

2. Tutorials

3. Thermal modelling

4. HOWTOs

5. SMITER module description

6. Additional resources

Fig. 1.1 Illustration of the SMITER Graphical User Interface for ITER. The code allows several run-cases in one study (a) to be run (f) in parallel on a compute cluster. ParaView window (b) shows resulting target top panels (blue), selected characteristic field lines (red), omp disk (yellow), a complete blanket sector CAD model from the Geometry module and a shadow mesh from the Meshing module (grey) augmented a for overall evaluation of run-case setup. The magnetic equilibrium with LCFS and Limiter/Wall geometry (c) and other details can be further analyzed with built-in 2D and 3D plots such as flux function detail (d). Triangular meshes (j) are the main run-case geometry setup (e) that are directly imported or meshed from CAD models (i) deformed to retain only the required PFC surfaces for meshing with different algorithms and hypotheses. Resulting heat fluxes on the panels (g) can be further processed to get ITER first wall panel (FWP) (h) using FEM thermal models built into the SALOME environment of normal heat flux (NHF) or enhanced heat flux (EHF) cooling sub-structures.

The SMITER fieldline tracing code together with its graphical user interface (GUI) provides a



SLING EURO

# Thanks!



**EuroHPC**  
Joint Undertaking

This project has received funding from the European High-Performance Computing Joint Undertaking (JU) under grant agreement No 951732. The JU receives support from the European Union's Horizon 2020 research and innovation programme and Germany, Bulgaria, Austria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, Greece, Hungary, Ireland, Italy, Lithuania, Latvia, Poland, Portugal, Romania, Slovenia, Spain, Sweden, United Kingdom, France, Netherlands, Belgium, Luxembourg, Slovakia, Norway, Switzerland, Turkey, Republic of North Macedonia, Iceland, Montenegro