Software management and HPC computing

marco milanesio, paola goatin, regis duvigneau

11-12 / 3 / 2024

part II: Collaborating

UNIVERSITÉ CÔTE D'AZUR

UNIVERSITÉ CÔTE D'AZUR
MAISON DE LA MODÉLISATION, DE LA SIMULATION ET DES INTERACTIONS

Inria

Agenda

- 11 - 3 - 2024
  - Scientific software
  - Design and development
  - Data structures
  - **Collaborating**
  - Open science

- 12 - 3 - 2024
  - Parallelisation and HPC
  - Documentation

Versioning

Version control

Version control is a system that records changes to a file or set of files over time so that you can recall specific versions later.

- Local (you should do it (and call it backup))
- Central (you don't have to)
- Fully distributed (you should really do it)

Version control

Version control is a system that records changes to a file or set of files over time so that you can recall specific versions later.

- Local (you should do it (and call it backup))
- Central (you don't have to)
- Fully distributed (you should really do it)

- SVN (central)
- TFS (central)
- Mercurial (distributed)
- Git (distributed)

Version control

Version control is a system that records changes to a file or set of files over time so that you can recall specific versions later.

- Local (you should do it (and call it backup))
- Central (you don't have to)
- Fully distributed (you should really do it)

- SVN (central)
- TFS (central)
- Mercurial (distributed)
- Git (distributed)

Git

- Speed
- Simple design
- Strong support for non-linear development (thousands of parallel branches)
- Fully distributed
- Able to handle large projects like the Linux kernel efficiently (speed and data size)

- GitLab: repository management software
- GitHub: platform to upload copies of the repositories

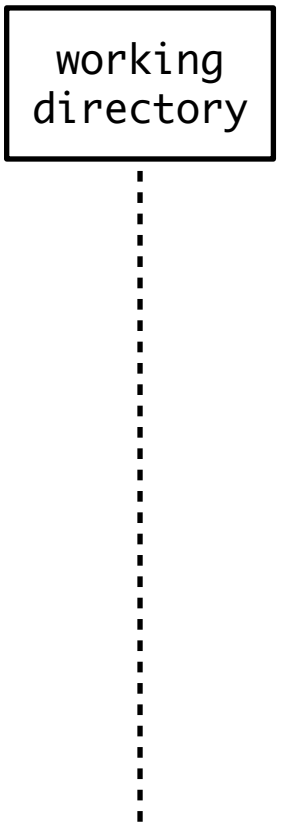
Git

- Use it every time there's development
- Just type: `git init`
- You're versioning, on branch `main` (`master`)
- https://git-scm.com/book/en/v2

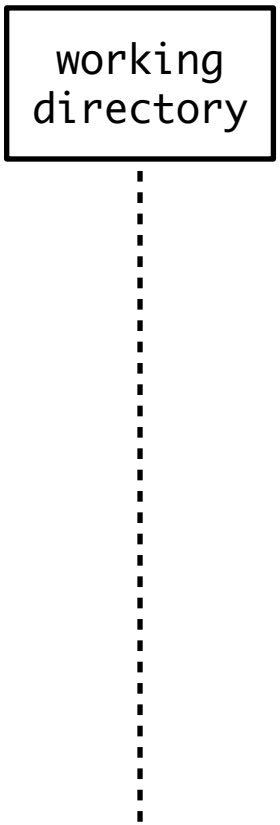
Hands on

Workflow

Workflow

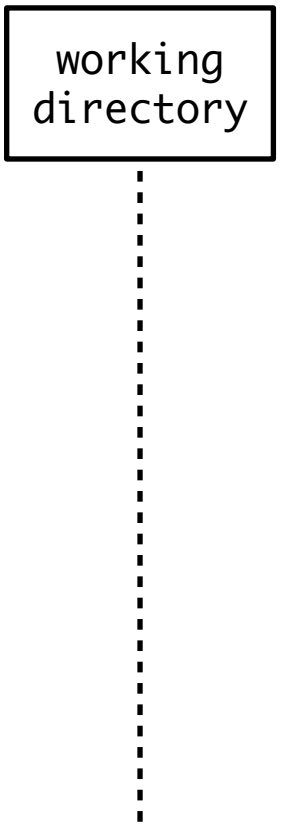


Workflow



file.txt

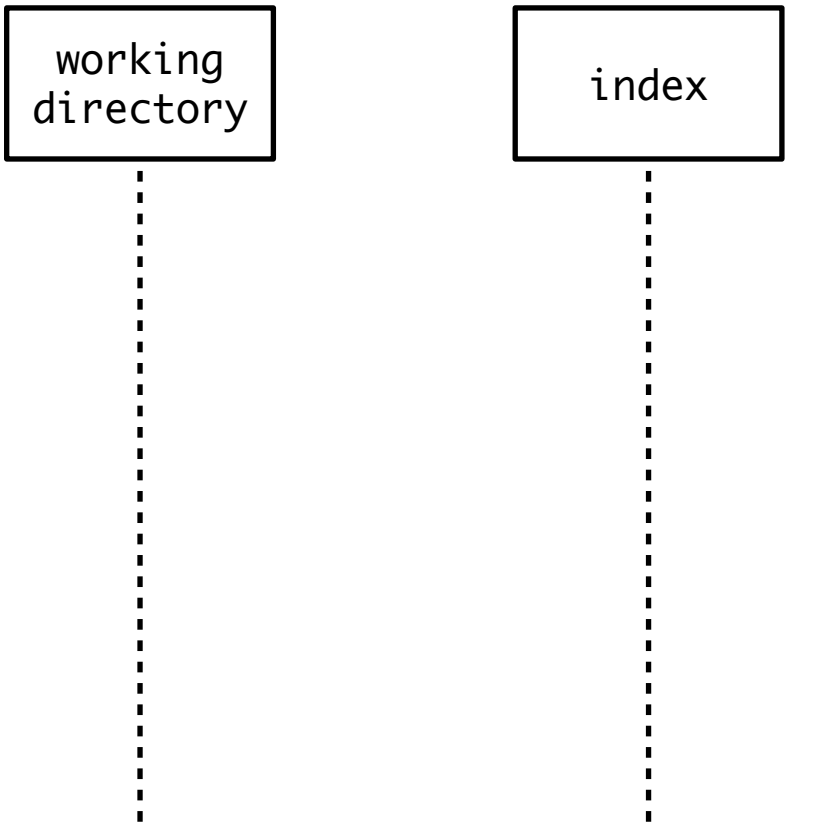
Workflow



git init

file.txt

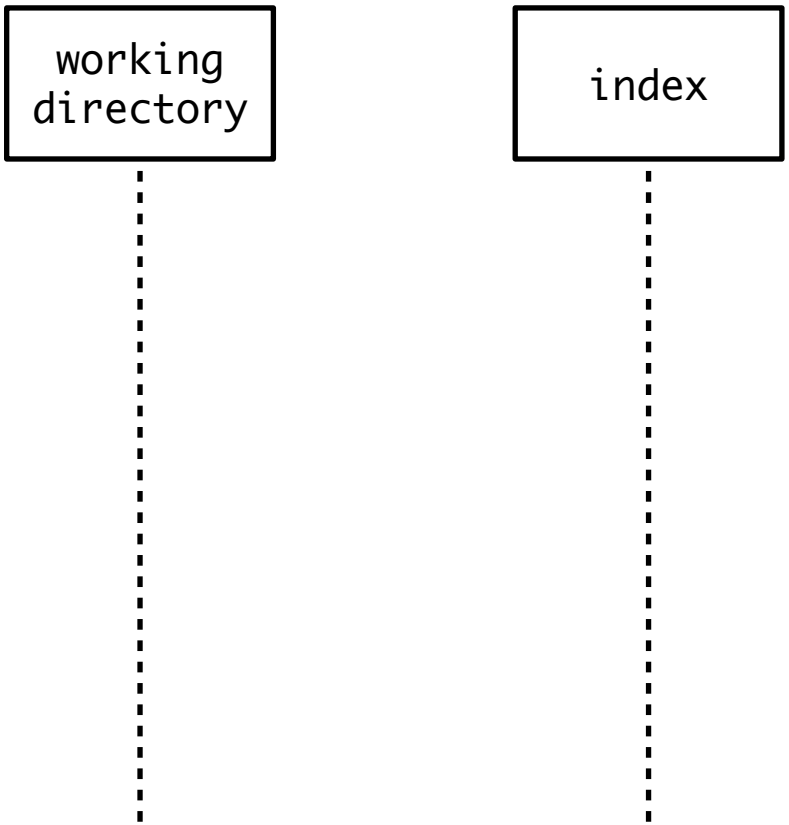
Workflow



`git init`

`file.txt`

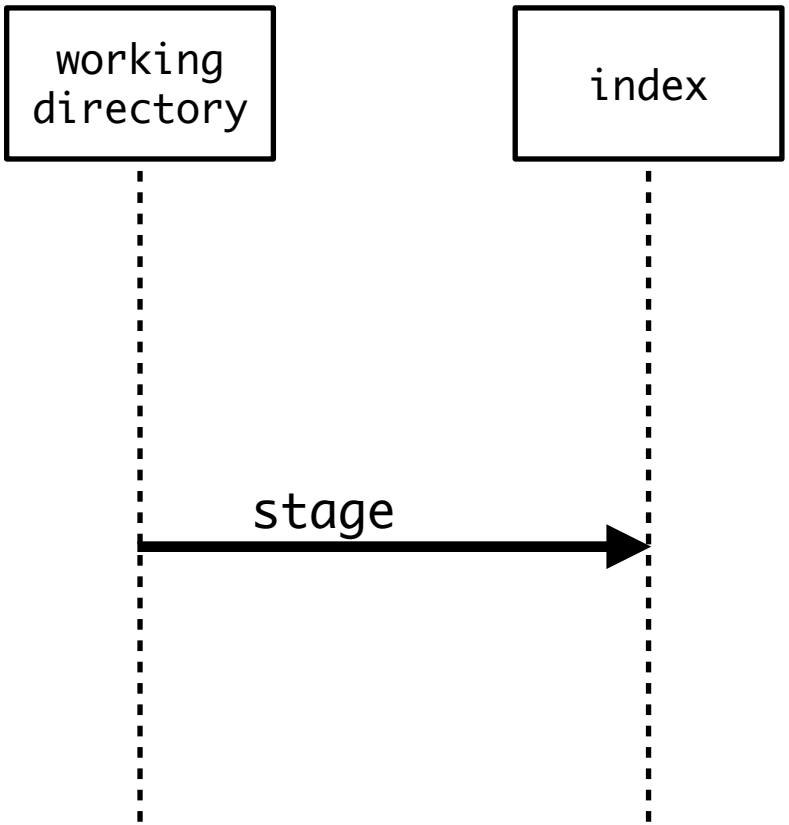
Workflow




```
git init
git add file.txt
```

file.txt

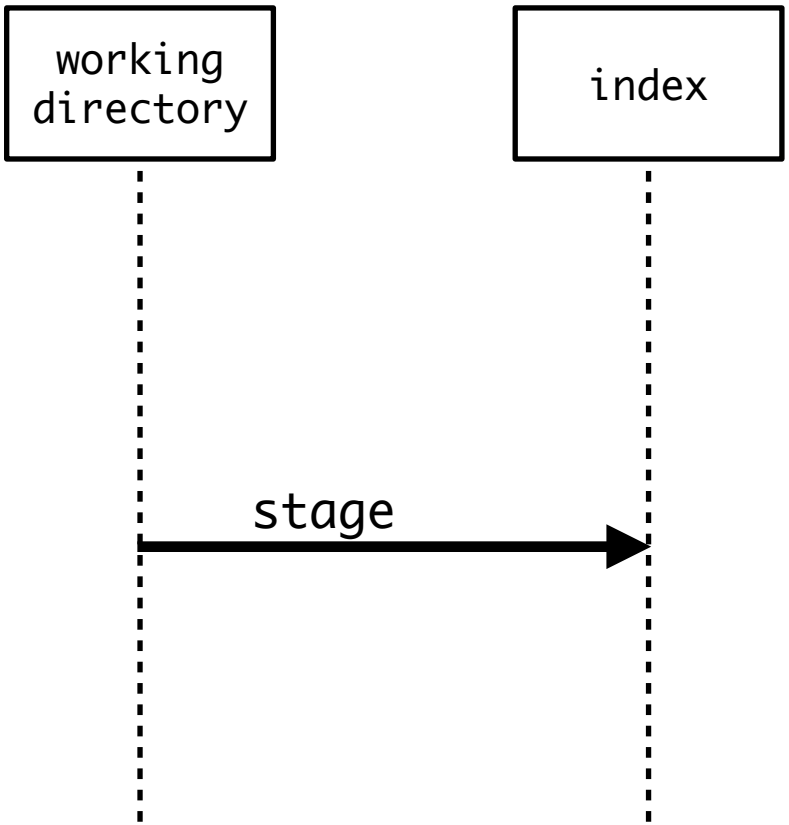
Workflow




```
git init
git add file.txt
```

file.txt file.txt

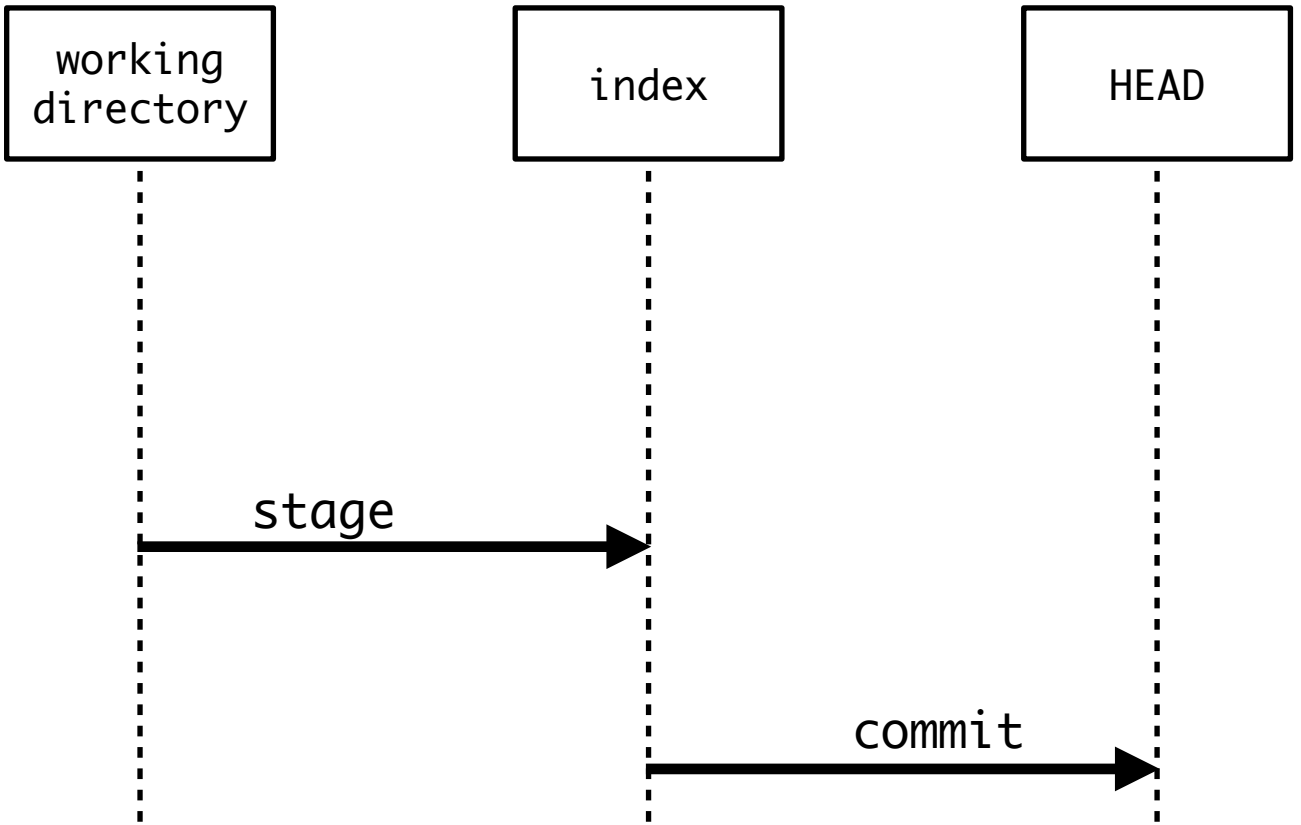
Workflow




```
git init
git add file.txt
git commit -m "1st"
```

file.txt file.txt

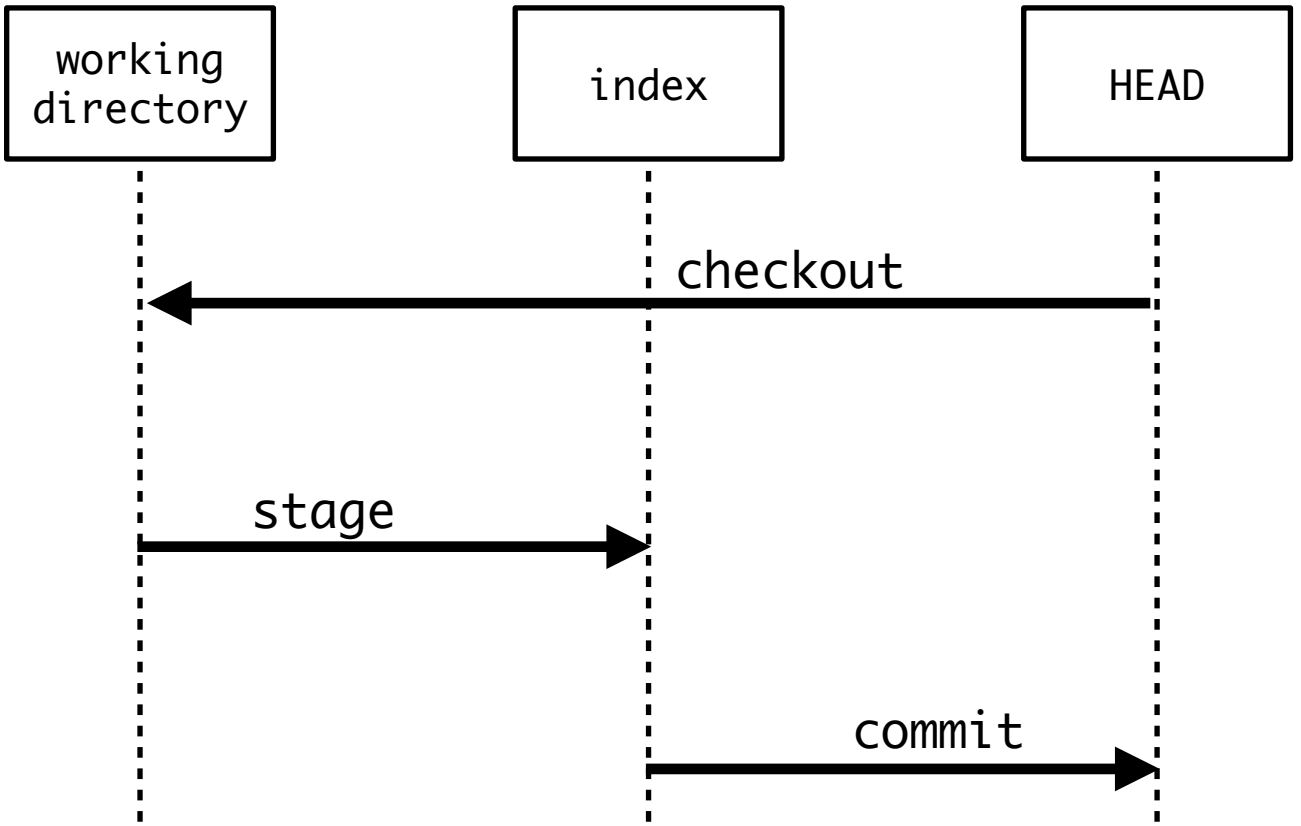
Workflow




```
git init
git add file.txt
git commit -m "1st"
```

file.txt file.txt file.txt

Workflow




```
git init
git add file.txt
git commit -m "1st"
```

file.txt file.txt file.txt

Branching

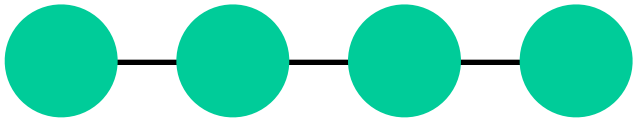
- Explore new ideas
- Bug fix
- Add new features

Branching

- Explore new ideas
- Bug fix
- Add new features

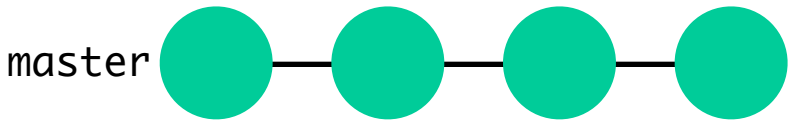
Branching

- Explore new ideas
- Bug fix
- Add new features



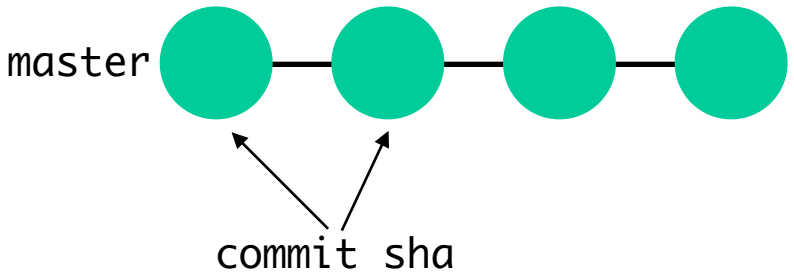
Branching

- Explore new ideas
- Bug fix
- Add new features



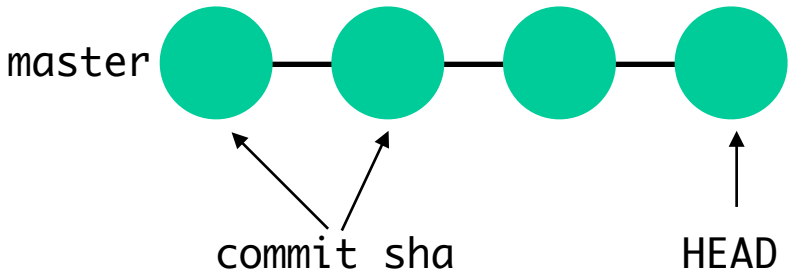
Branching

- Explore new ideas
- Bug fix
- Add new features



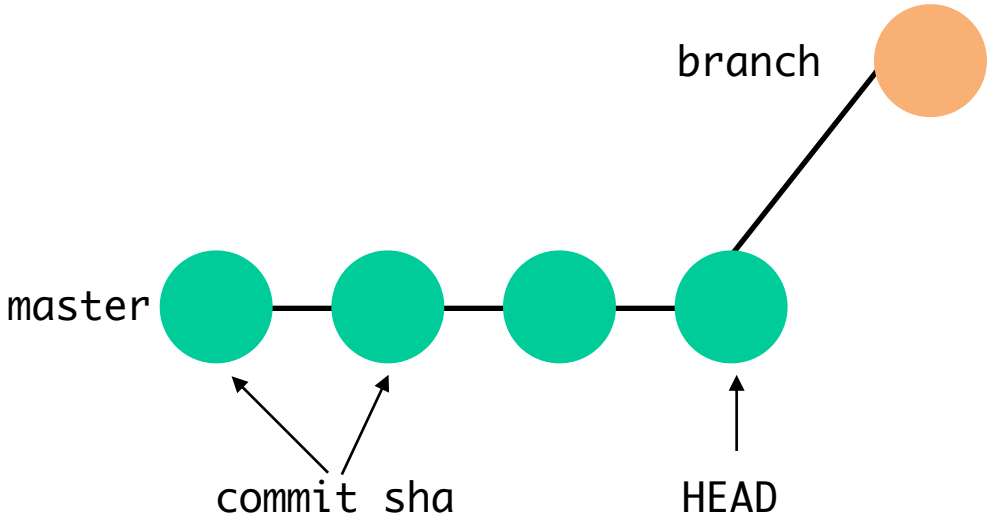
Branching

- Explore new ideas
- Bug fix
- Add new features



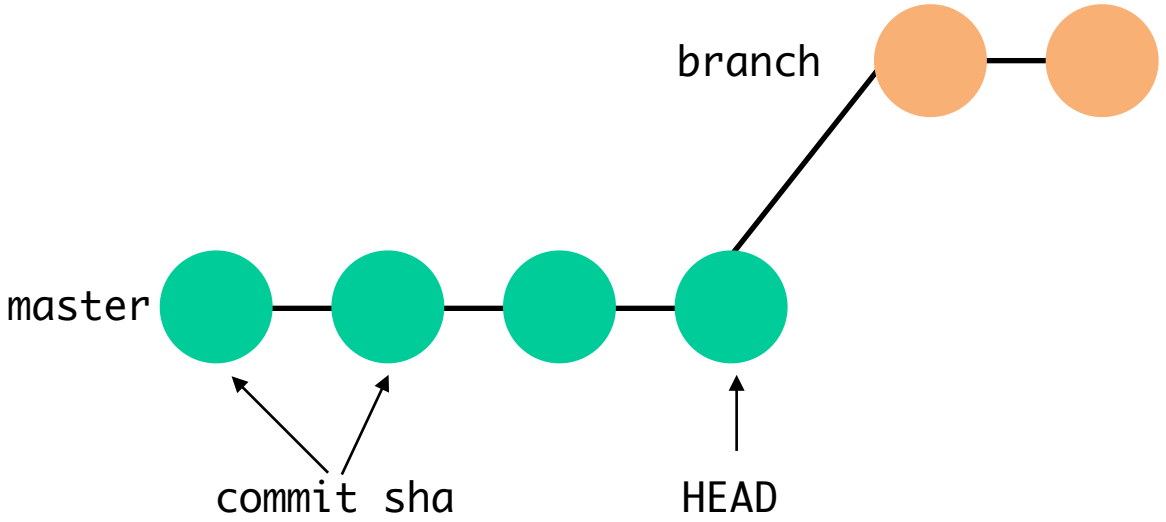
Branching

- Explore new ideas
- Bug fix
- Add new features



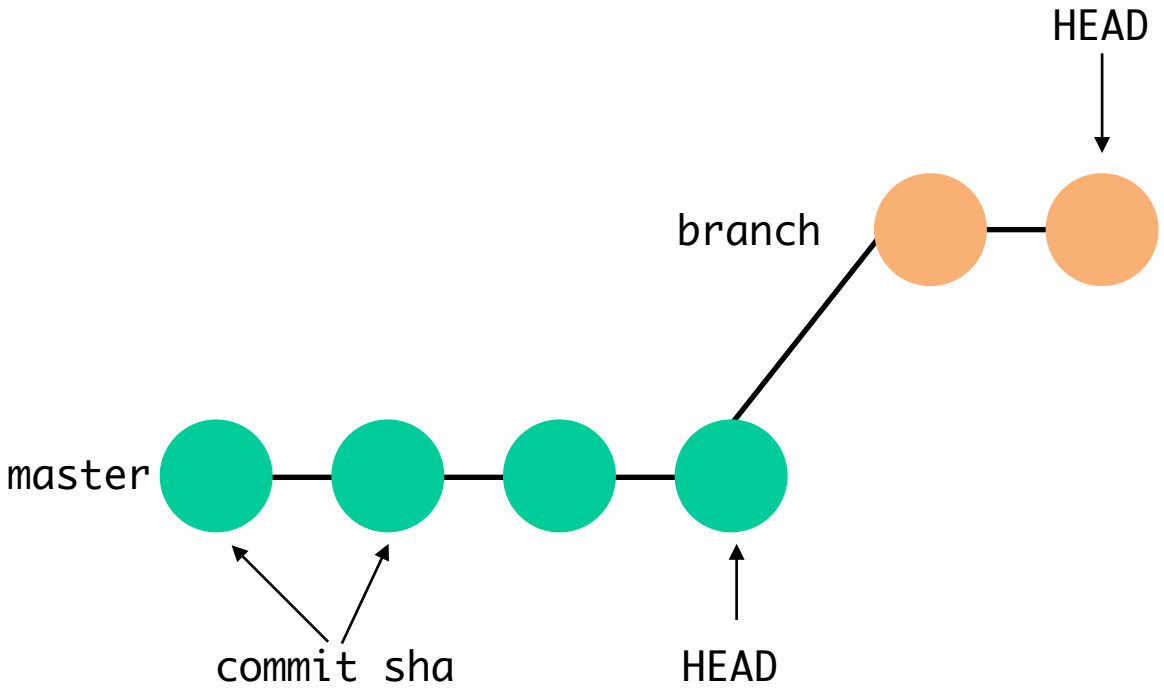
Branching

- Explore new ideas
- Bug fix
- Add new features



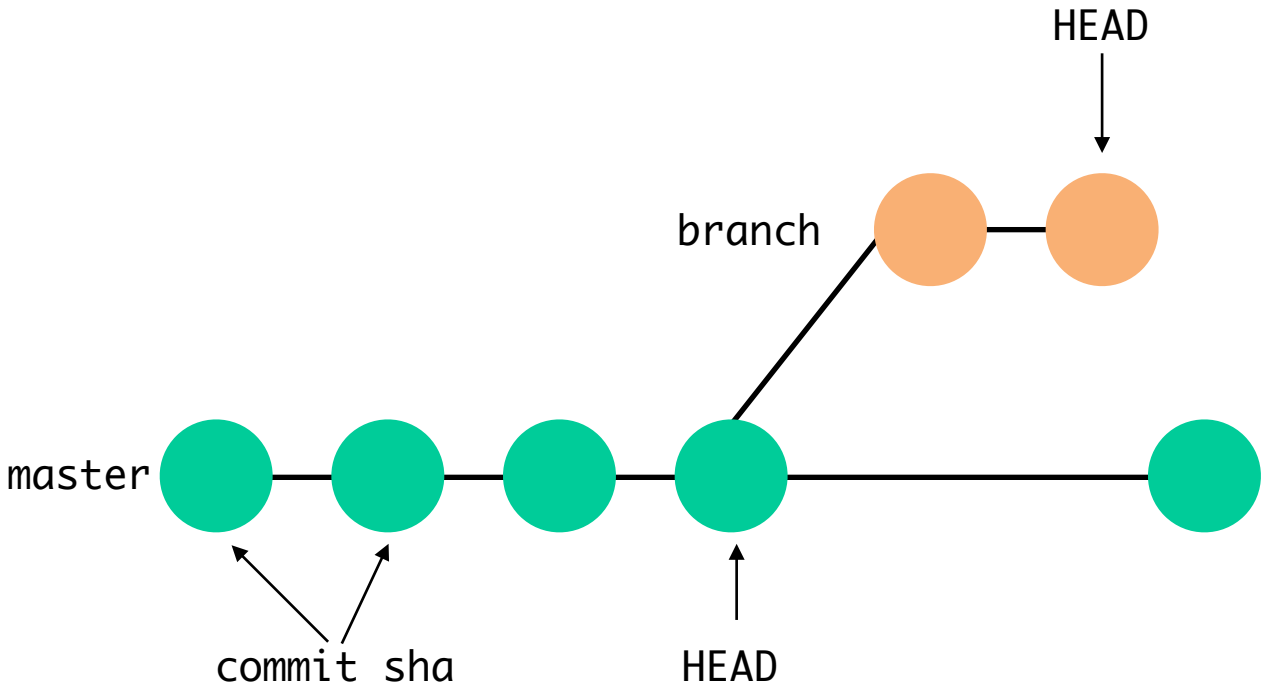
Branching

- Explore new ideas
- Bug fix
- Add new features



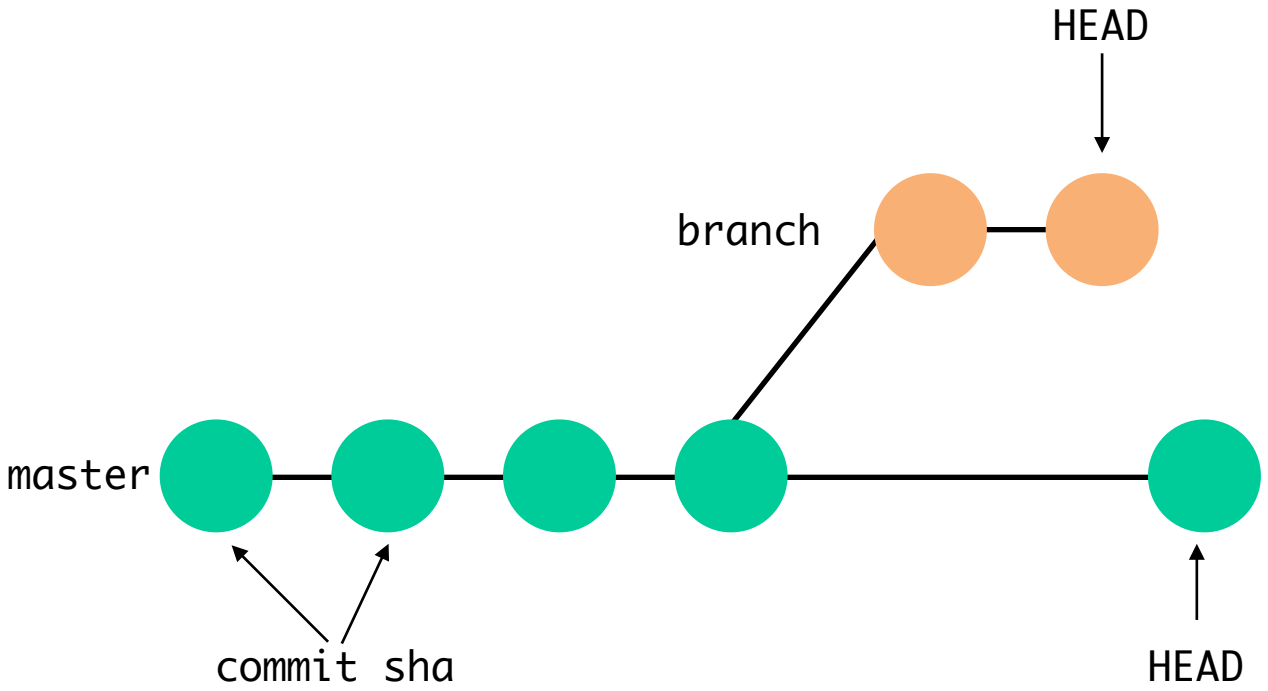
Branching

- Explore new ideas
- Bug fix
- Add new features



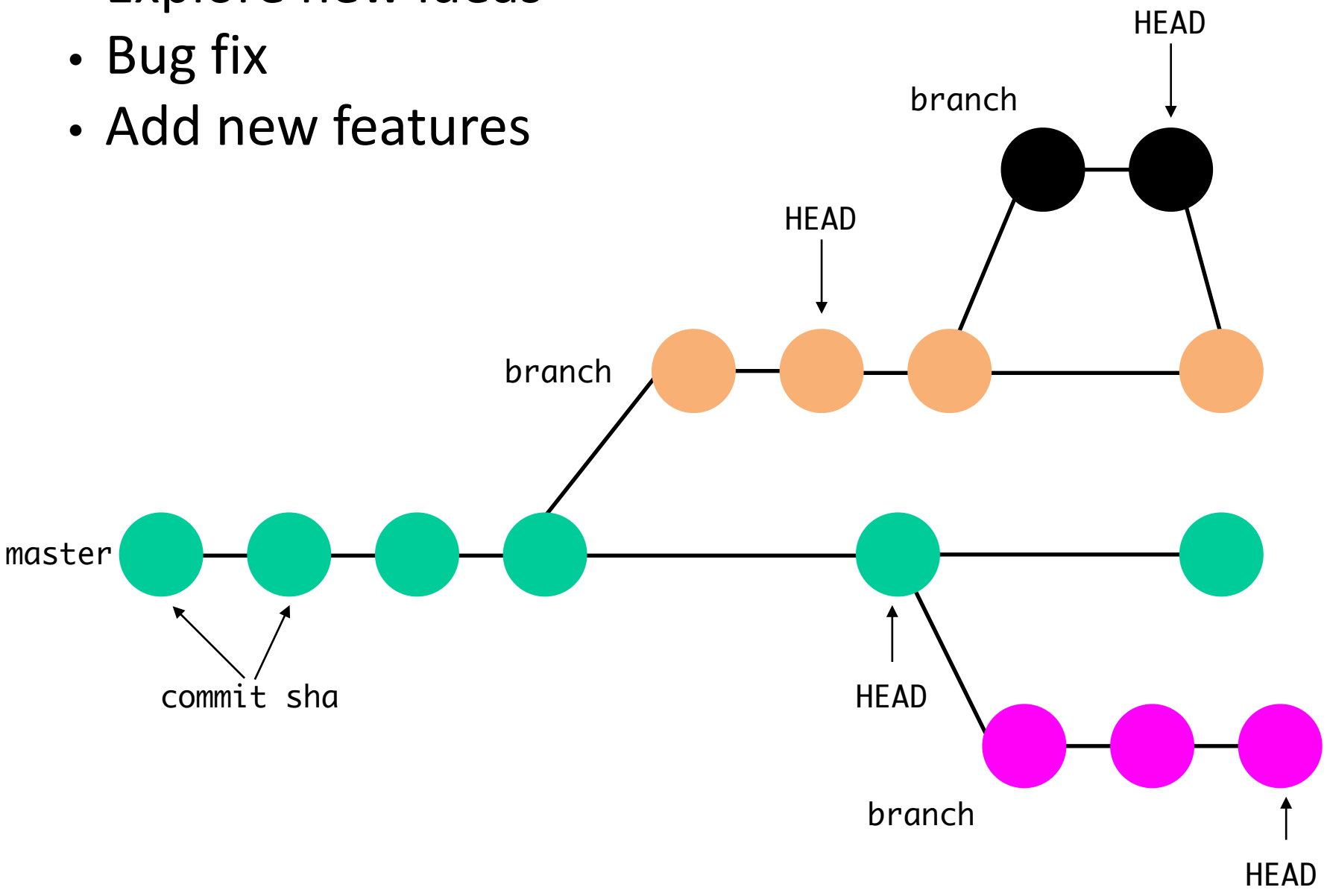
Branching

- Explore new ideas
- Bug fix
- Add new features



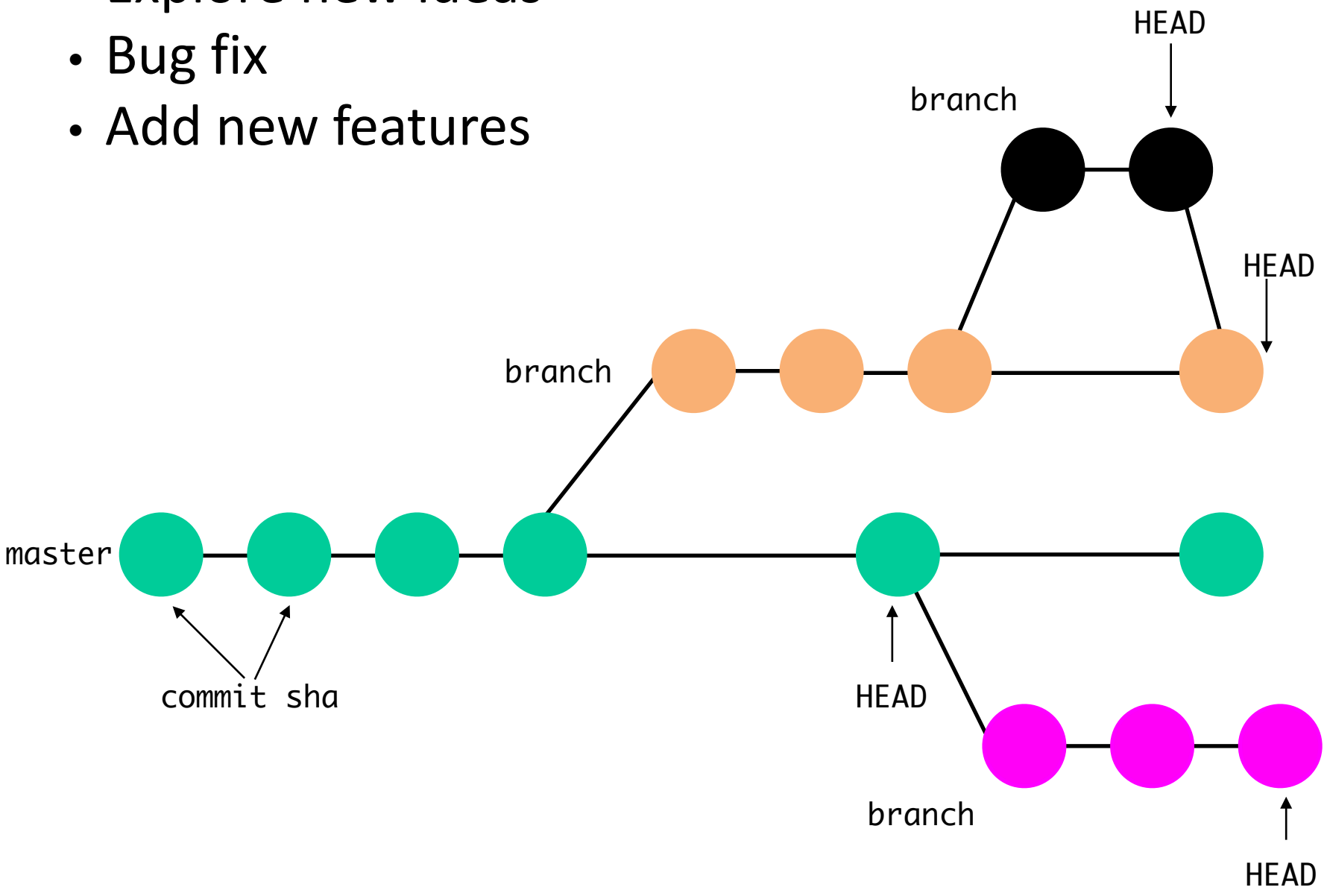
Branching

- Explore new ideas
- Bug fix
- Add new features



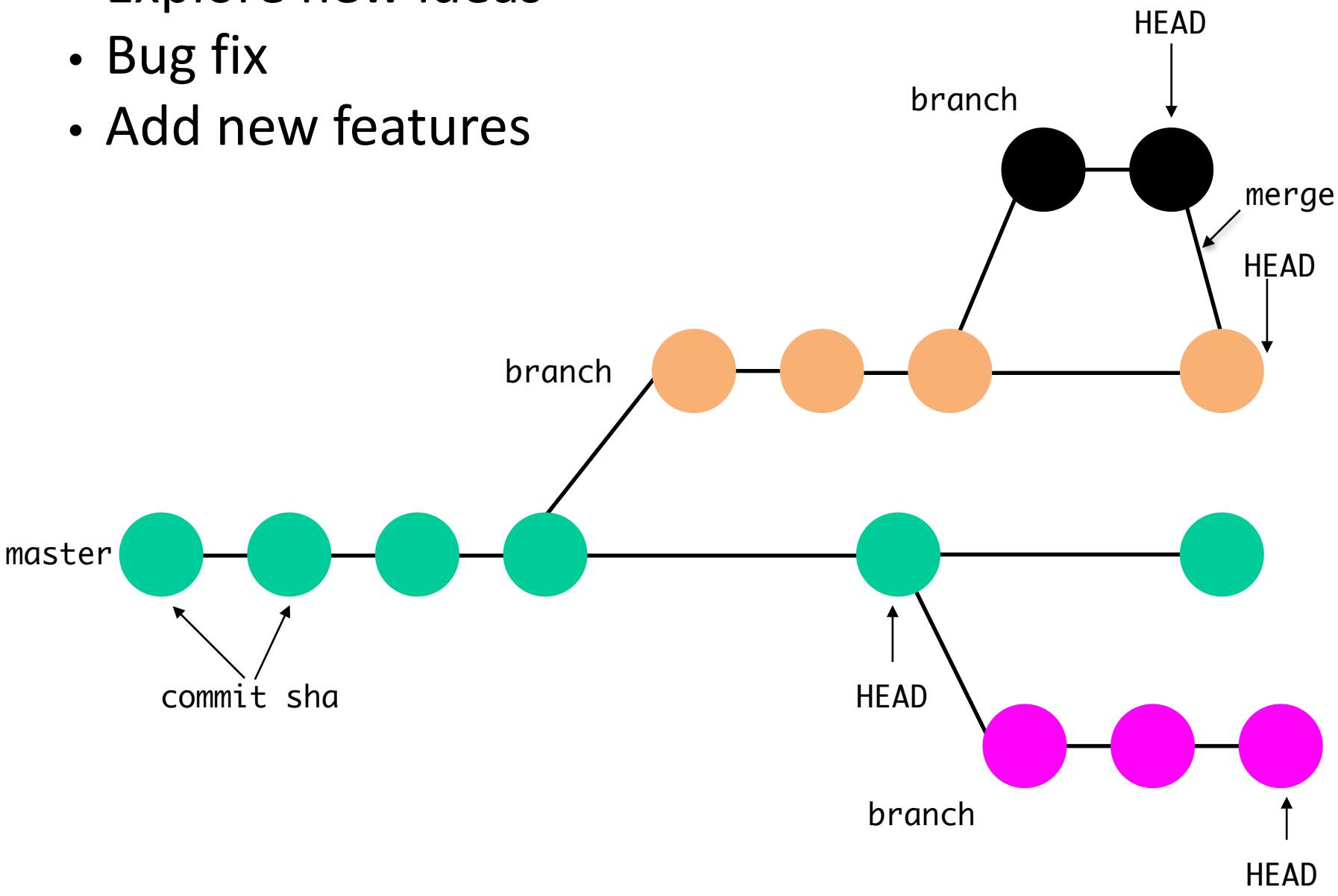
Branching

- Explore new ideas
- Bug fix
- Add new features



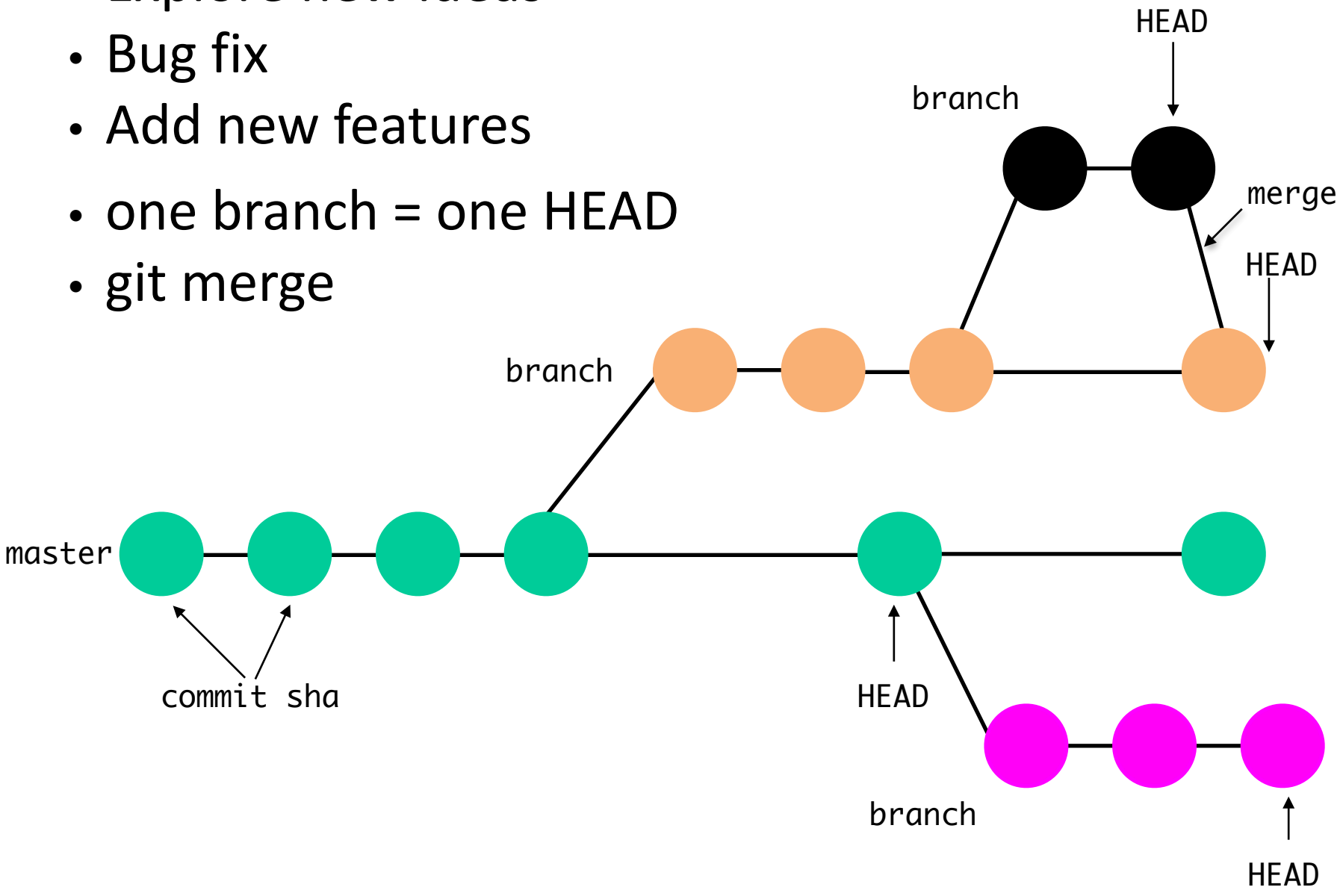
Branching

- Explore new ideas
- Bug fix
- Add new features



Branching

- Explore new ideas
- Bug fix
- Add new features
- one branch = one HEAD
- git merge

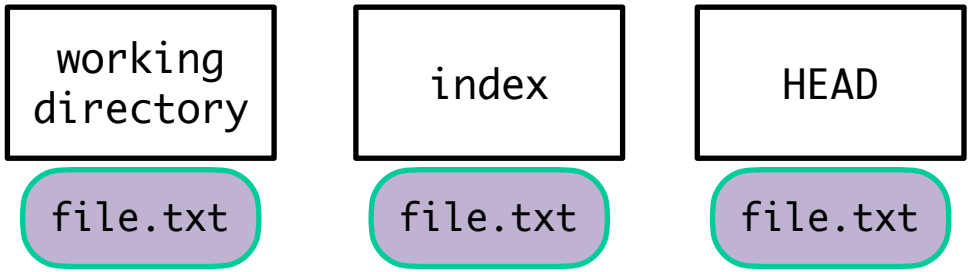


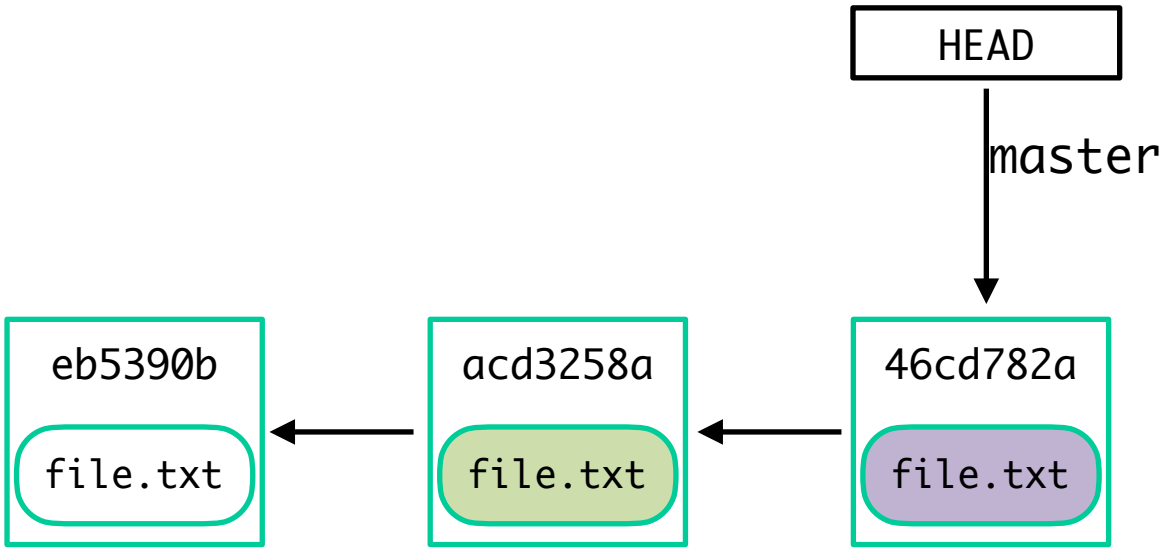
Git merge

- Update another branch
- Easy: if target branch hasn’t "moved"
- Hard: if target branch has "moved"
- merge conflicts

Git reset

- Most feared command
- Really not scary

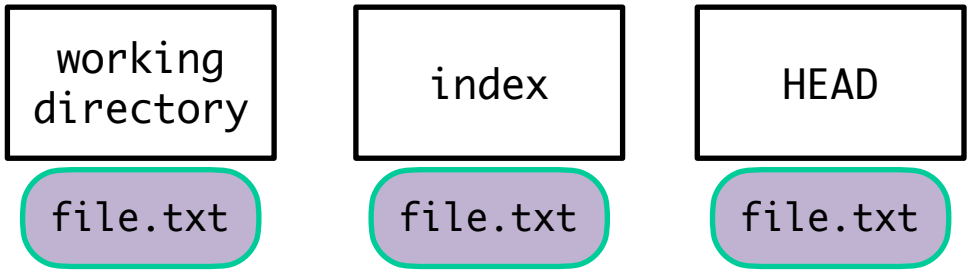


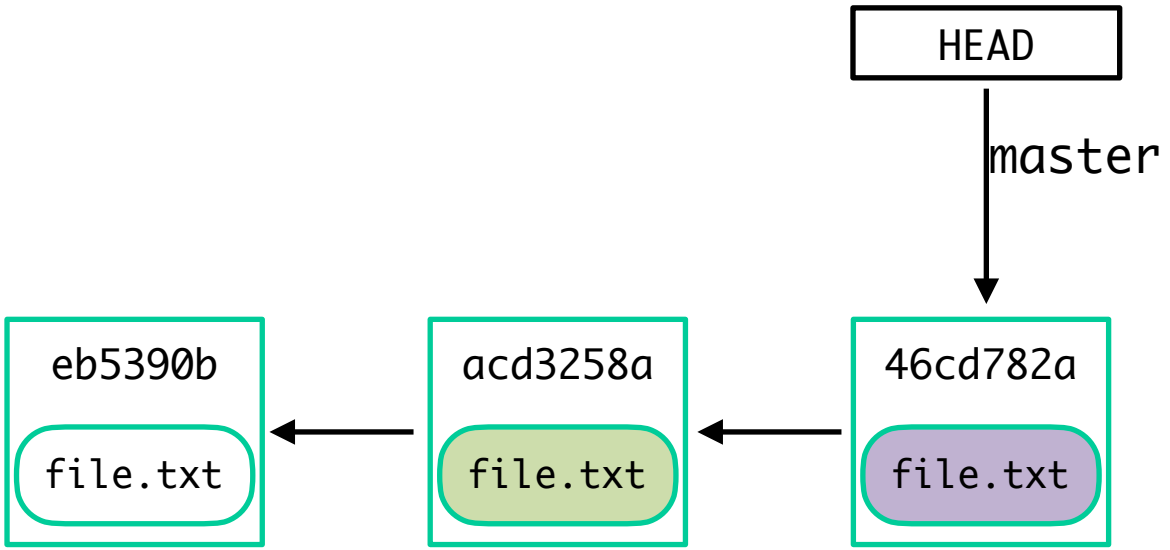


Git reset

- Most feared command
- Really not scary

```
git reset acd3258a
```

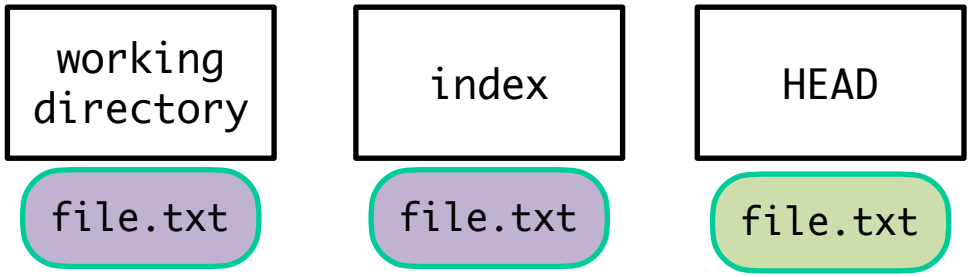


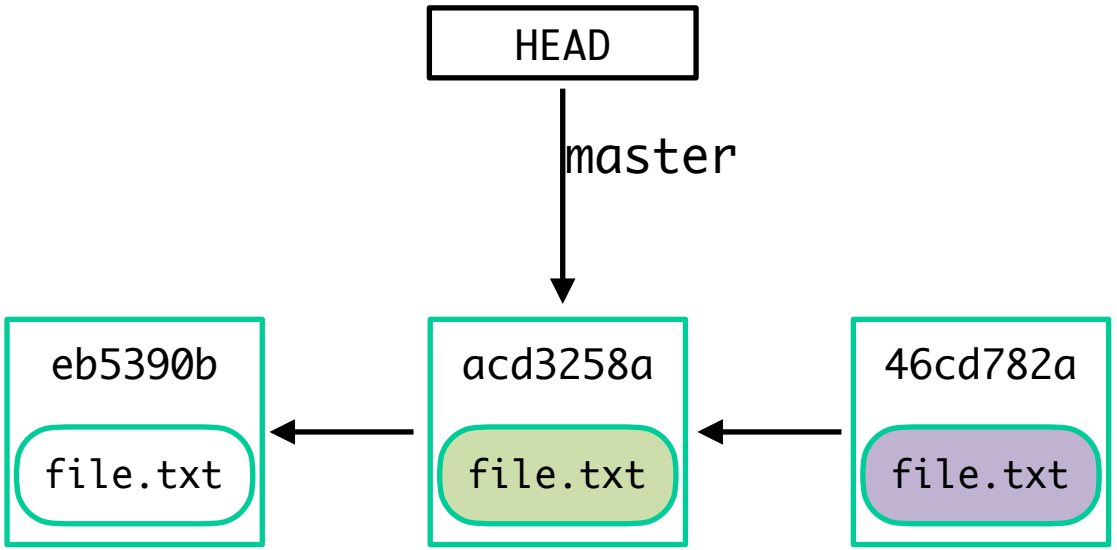


Git reset

- Most feared command
- Really not scary

```
git reset acd3258a
```

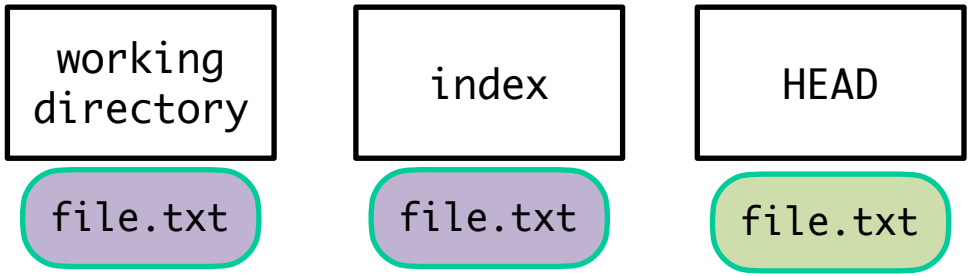


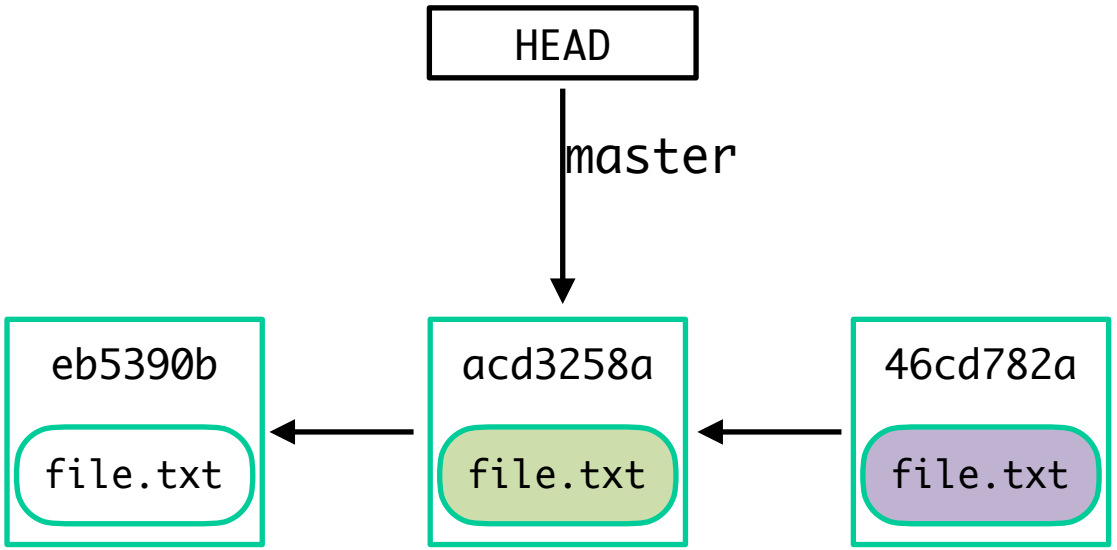


Git reset

- Most feared command
- Really not scary

```
git reset --soft HEAD~
```

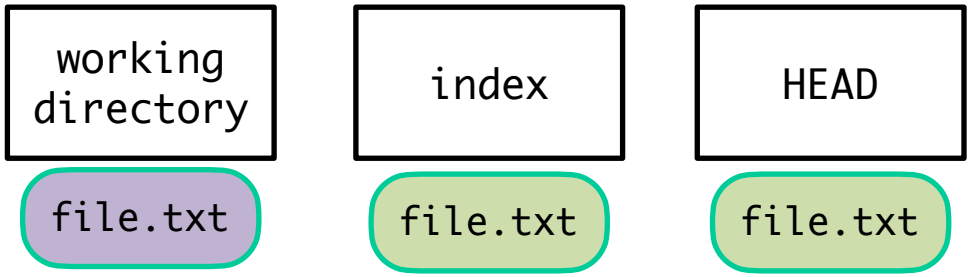


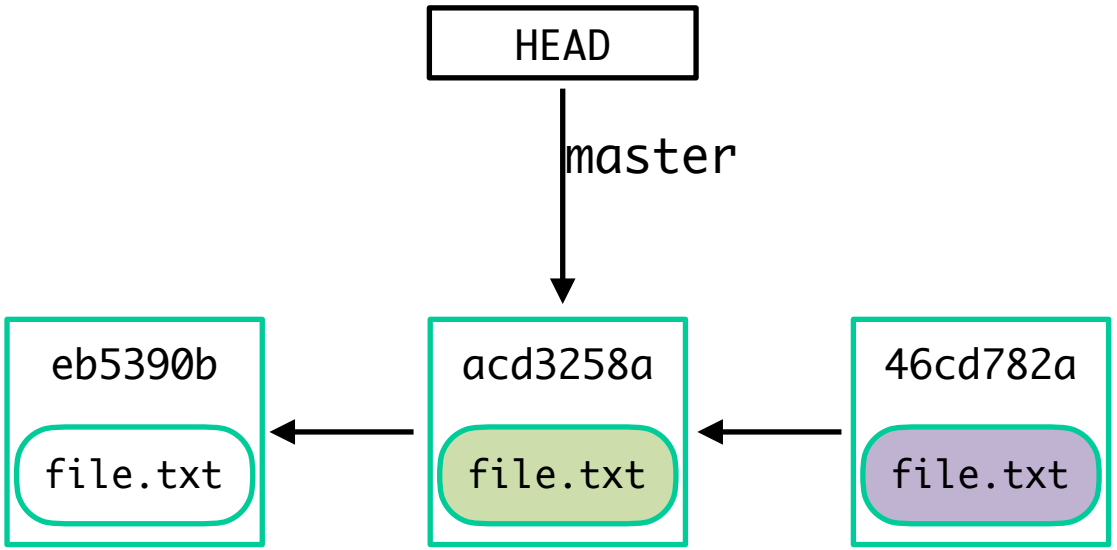


Git reset

- Most feared command
- Really not scary

```
git reset --soft HEAD~
git reset [--mixed] HEAD~
```

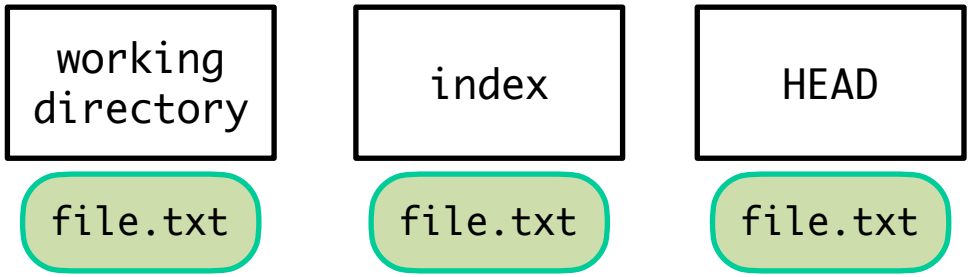


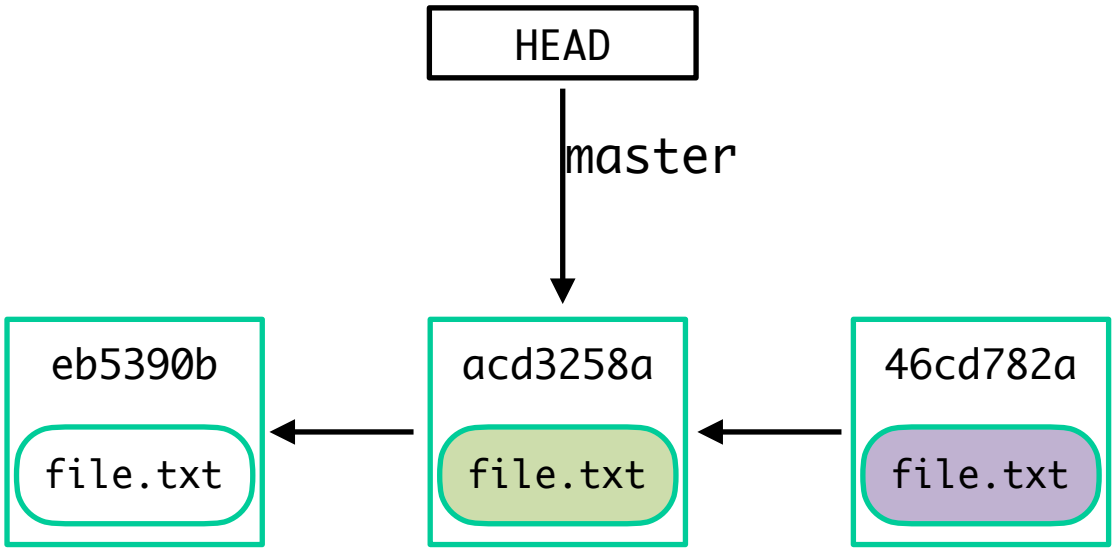


Git reset

- Most feared command
- Really not scary

```
git reset --soft HEAD~
git reset [--mixed] HEAD~
git reset --hard HEAD~
```

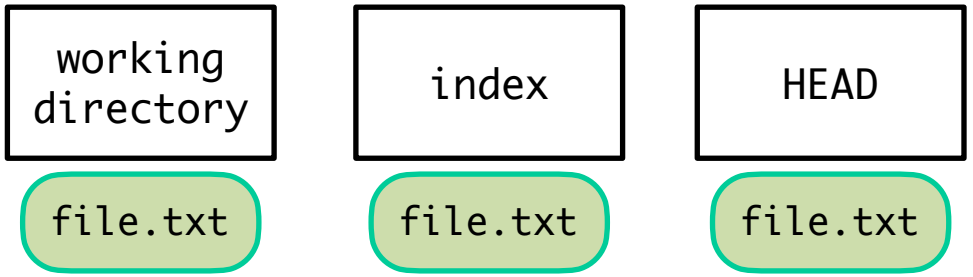


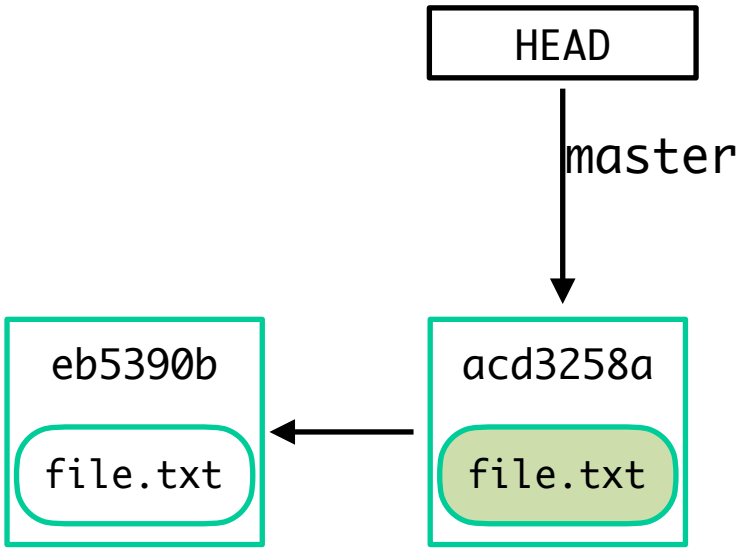


Git reset

- Most feared command
- Really not scary

```
git reset --soft HEAD~
git reset [--mixed] HEAD~
git reset --hard HEAD~
```





GitHub

- github.com
- "push" HEAD to the cloud
- That’s it.

- No, there’s more
- Team collaboration
- github actions
- CI/CD

Testing

Testing

- The fundamental skill
- Not only for being nice to others
- It’s for you!

- Testing rocks
- Debugging sucks
- There is no compiler to help you out there
- Make sure your code does what you want

Testing

- assertions / contracts
  - to validate program invariants
- unittest Module
  - builtin
  - used in standard library
  - can grow quite dramatically
- pytest an alternative

Test Driven Development

- Write a "single" unit test for a new feature
- Run the test(s): fail
- Write "just enough" code just to make it pass
- Run the test(s): success
- Refactor your code
- Run the test(s): success
- Repeat (new feature, new tests)

Hands on

Unit test vs Integration test

- *Misleading*
- Can be done in the same class (unittest.TestCase)
  - Not a good idea, though
- Different scopes:
  - single module
  - between modules
- White Box vs Black Box

CI / CD

- Continuous Integration / Continuous Delivery
- Github actions
- Integration testing
- Automatic test execution
  - before pull req / merge
  - after each commit / push
  - fully customisable

Agenda

- 11 - 3 - 2024
  - Scientific software
  - Design and development
  - Data structures
  - Collaborating
  - **Open science**

- 12 - 3 - 2024
  - Parallelisation and HPC
  - Documentation

Open Science

Open source

- Community contribution
- Code quality
- Collaboration
- Source code availability
- Licenses

Licences

- Legal instruments governing the distribution
- Two common categories
  - Proprietary software
  - Free and Open Source
- Many types:
  - Public domain
  - FOSS
  - Copyleft
  - Freeware / Shareware
  - Proprietary
  - Secret
- It’s a matter of rights

Rights

- Copyright retainment
- Right to perform
- Right to display
- Right to copy
- Right to modify
- Right to distribute
- Right to sublicense
- Right to sell

Choose a license

- **University policies or requirements**
- **Funding source**
- **Collaboration**
- **Commercialization potential**

- **MIT**
- **Apache**
- **MPL**
- **GPL**

MIT

Copyright <YEAR> <COPYRIGHT HOLDER>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT

Copyright <YEAR> <COPYRIGHT HOLDER>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

GPL v3

Copyright (C) <year> <name of author>

    This program is free software: you can redistribute it and/or modify
    it under the terms of the GNU General Public License as published by
    the Free Software Foundation, either version 3 of the License, or
    (at your option) any later version.

    This program is distributed in the hope that it will be useful,
    but WITHOUT ANY WARRANTY; without even the implied warranty of
    MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
    GNU General Public License for more details.

    You should have received a copy of the GNU General Public License
    along with this program. If not, see <https://www.gnu.org/licenses/>.

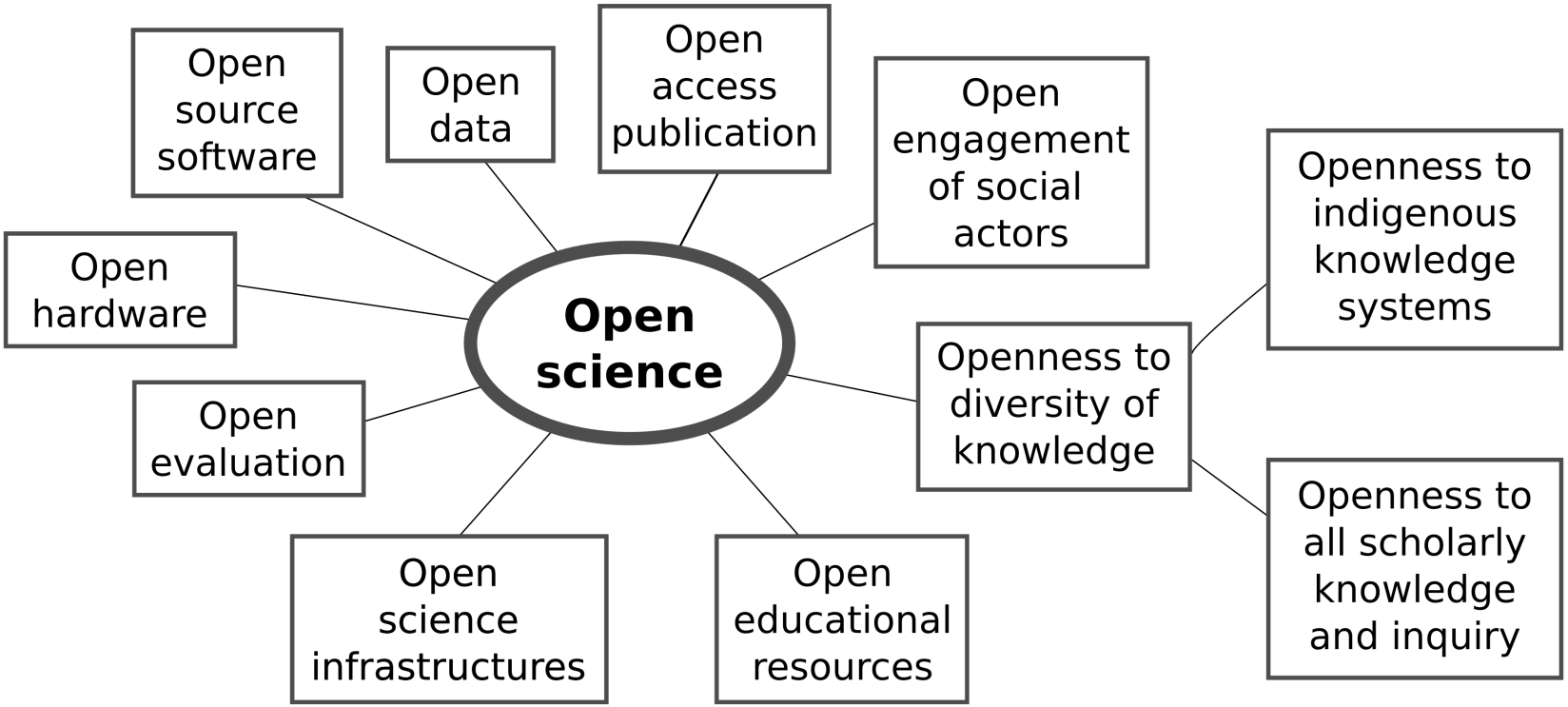
GPL v3

```
Copyright (C) <year>  <name of author>

    This program is free software: you can redistribute it and/or modify
    it under the terms of the GNU General Public License as published by
    the Free Software Foundation, either version 3 of the License, or
    (at your option) any later version.

    This program is distributed in the hope that it will be useful,
    but WITHOUT ANY WARRANTY; without even the implied warranty of
    MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
    GNU General Public License for more details.

    You should have received a copy of the GNU General Public License
    along with this program.  If not, see <https://www.gnu.org/licenses/>.
```

Enforcing

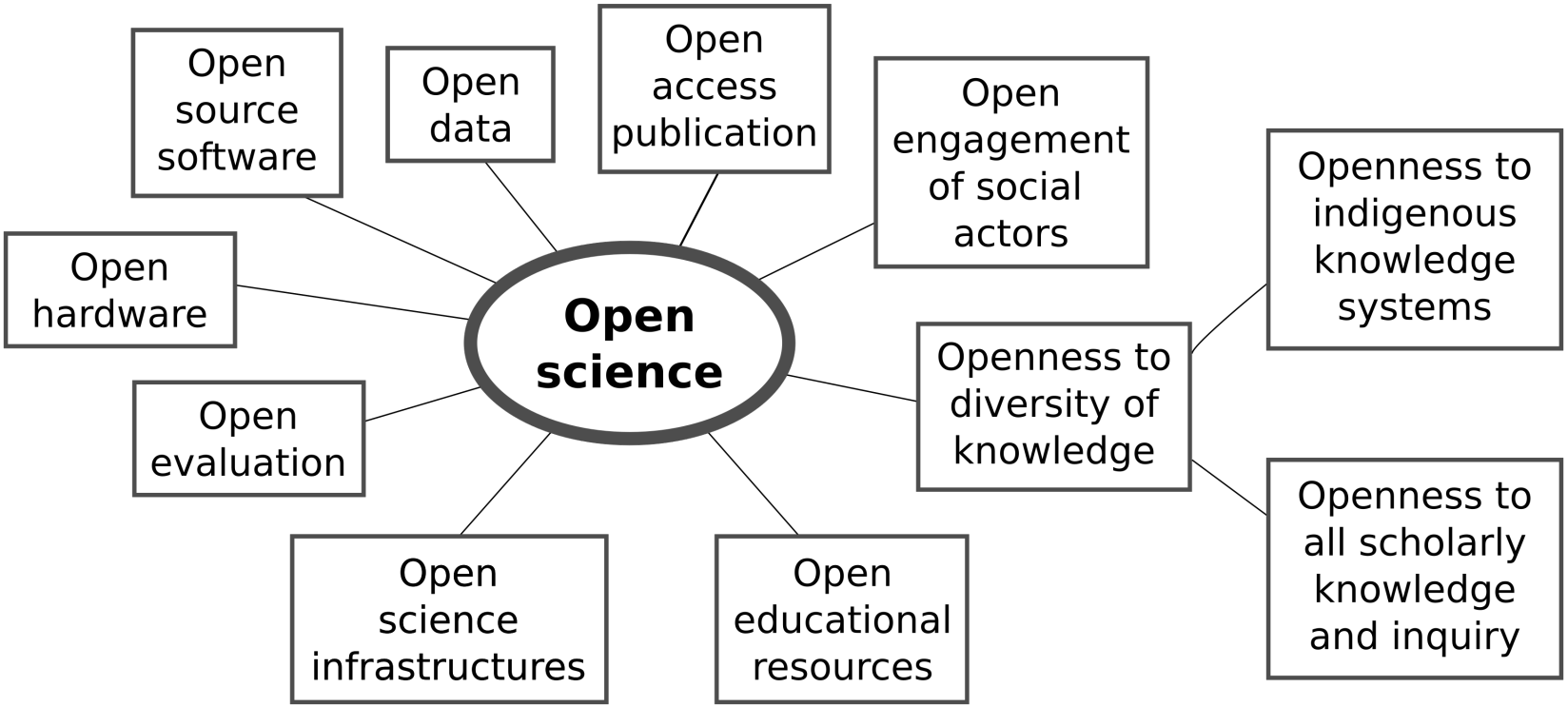
- Self governance and community enforcement
  - OSS communities
  - License compliance tools
- Legal system and Copyright Law
  - in case of blatant violation
  - CC is a nice middle ground here
- Public advocacy and Education
  - OSS organisations and foundations (e.g., GNU)
  - Universities
- It’s a collective effort: go out there and participate!

Open Science



By RobbielanMorrison - Own work, CC BY 4.0, https://commons.wikimedia.org/w/index.php?curid=100144897

Open Science



By RobbielanMorrison - Own work, CC BY 4.0, https://commons.wikimedia.org/w/index.php?curid=100144897

Inria

Open science/Open Access

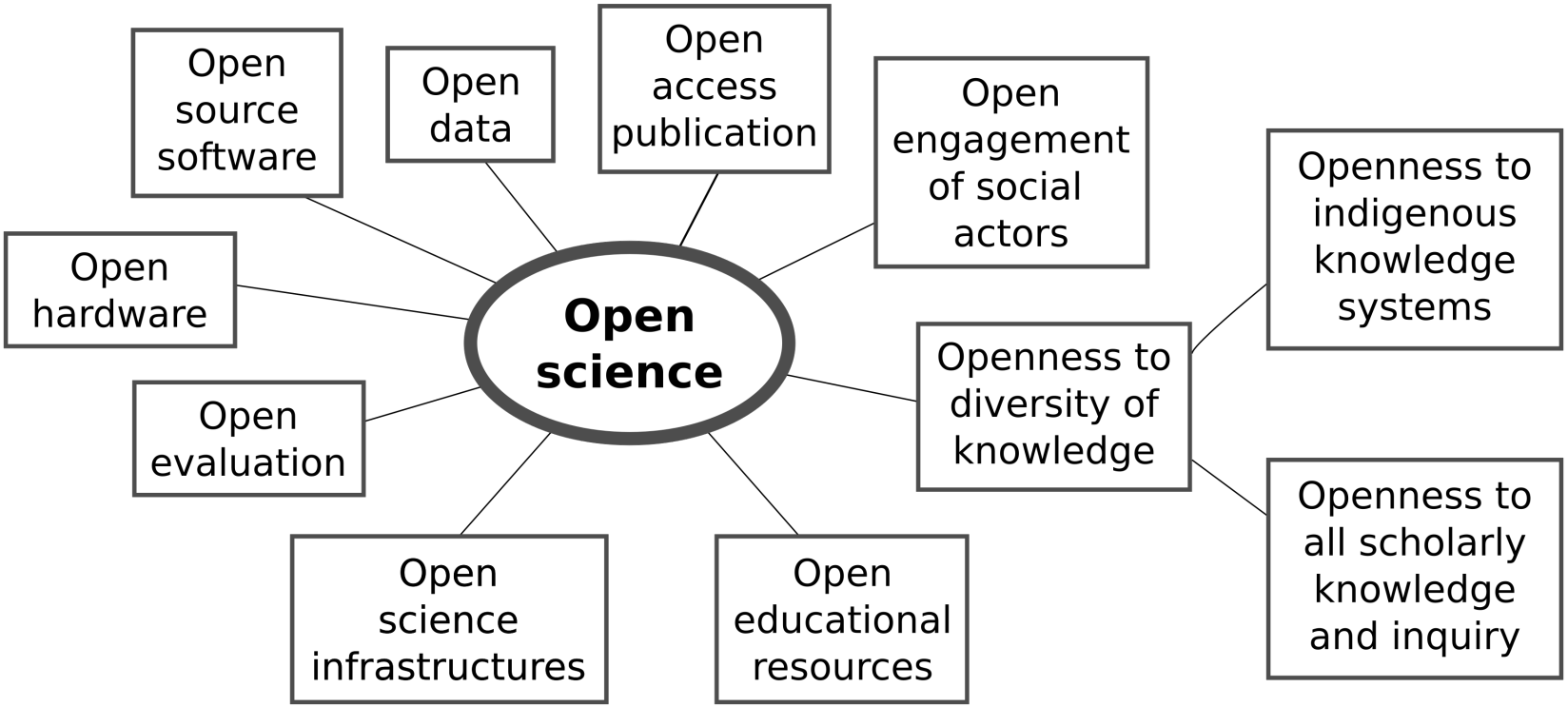
What role does Inria play when it comes to Open Science?

Date: 10 Feb. 2022

Home > News and events > What role does Inria play when it comes to Open Science?

https://www.inria.fr/en/open-science-inria-role

Open Science



By RobbielanMorrison - Own work, CC BY 4.0, https://commons.wikimedia.org/w/index.php?curid=100144897

Inria

Open science/Open Access

What role does Inria play w

Date: 10 Feb. 2022

Home > News and events > What role does Inria play when it comes to Open Science?

https://www.inria.fr/en/open-science-inria-role

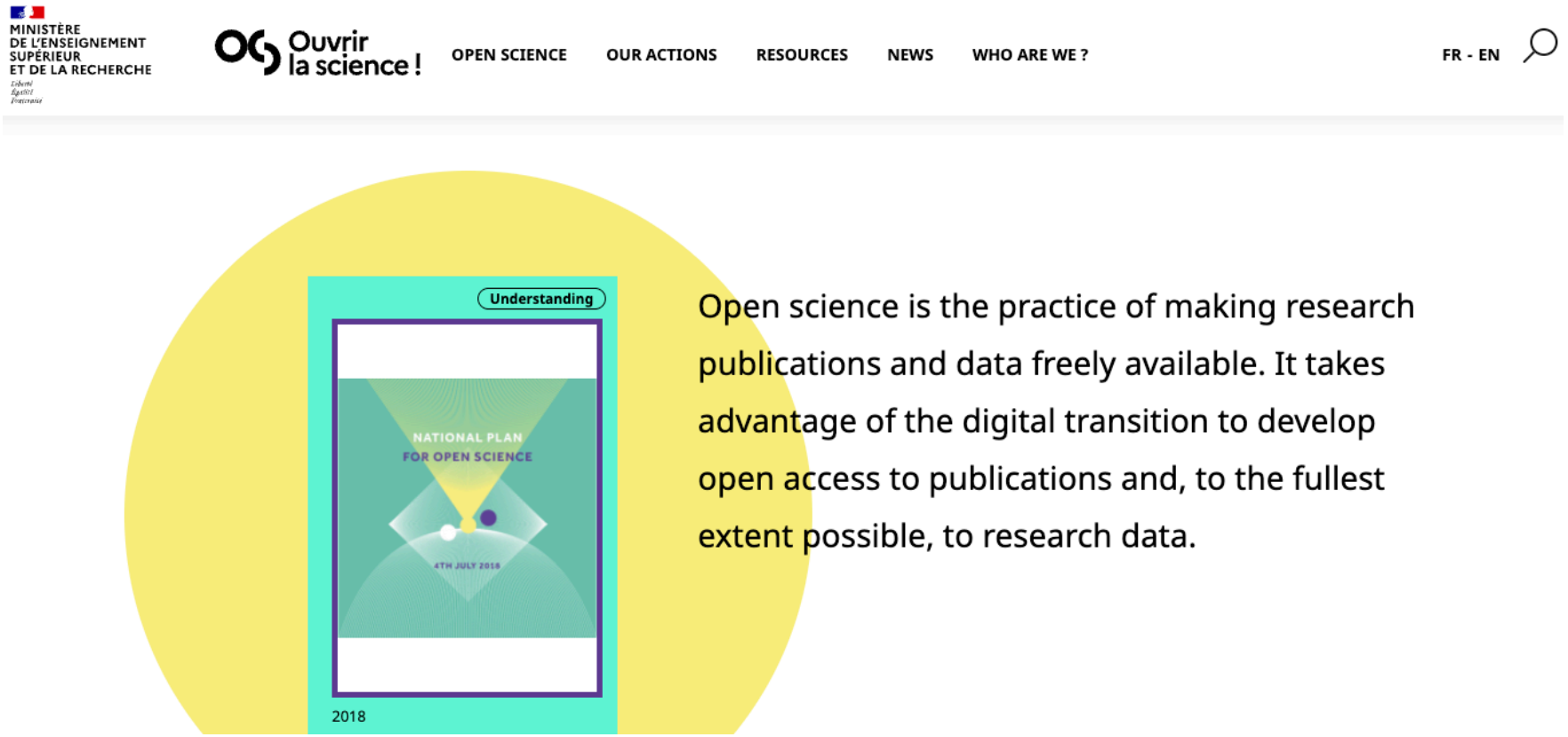
SCIENCE OUVERTE

La science ouverte est la diffusion sans entrave des résultats, des méthodes et des produits de la recherche scientifique. Cette démarche globale s'oppose à la privatisation du savoir scientifique et aux limitations induites à sa diffusion pour restaurer son rôle sociétal, démocratique et rétablir les conditions de son fonctionnement comme science efficace et de qualité.

Université Côte d'Azur soutient fermement cette dynamique en proposant un panel de services destinés à favoriser l'ouverture des publications et des données produites par ses chercheurs.

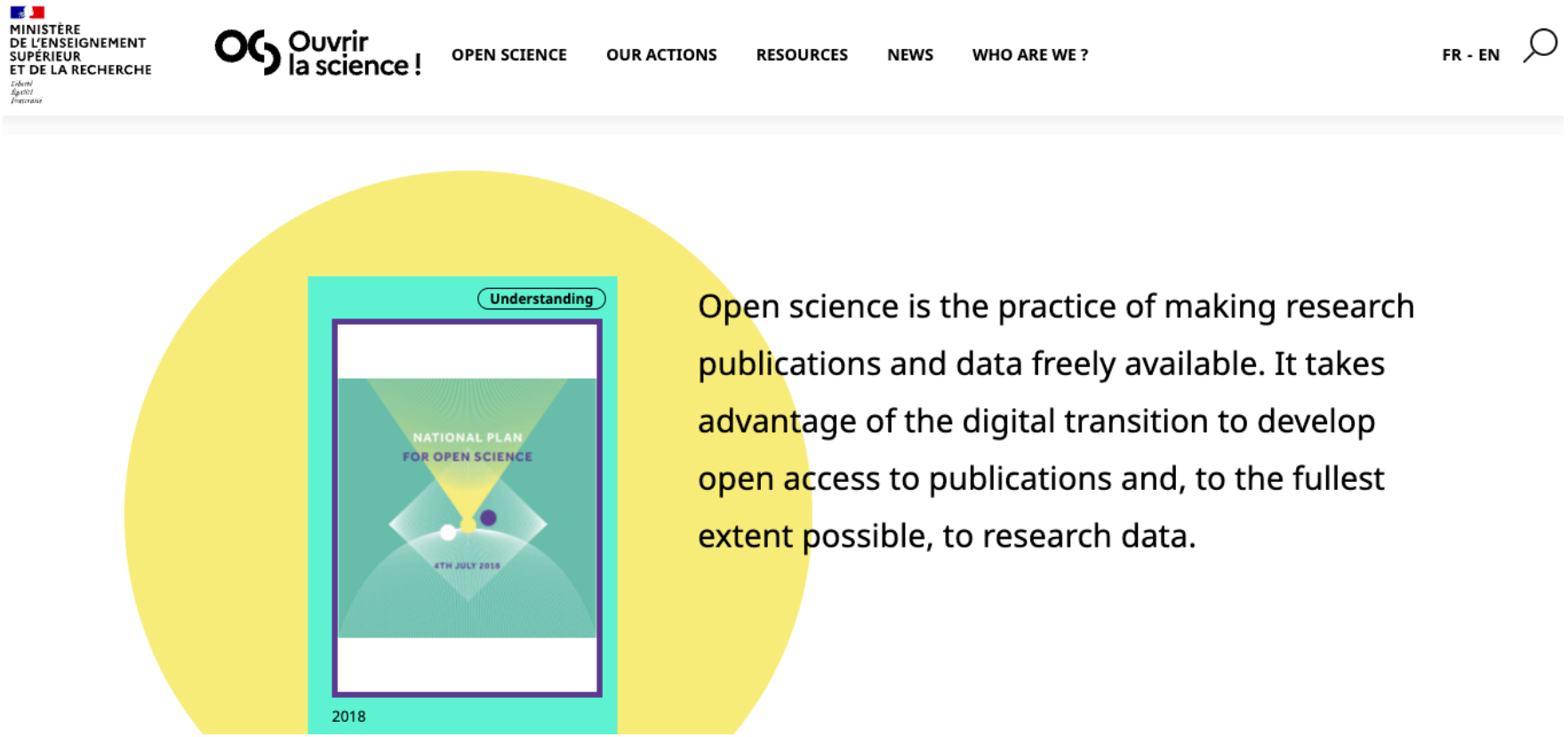
https://univ-cotedazur.fr/recherche-innovation/science-ouverte

Open Science



https://www.ouvrirlascience.fr/national-plan-for-open-science-4th-july-2018/

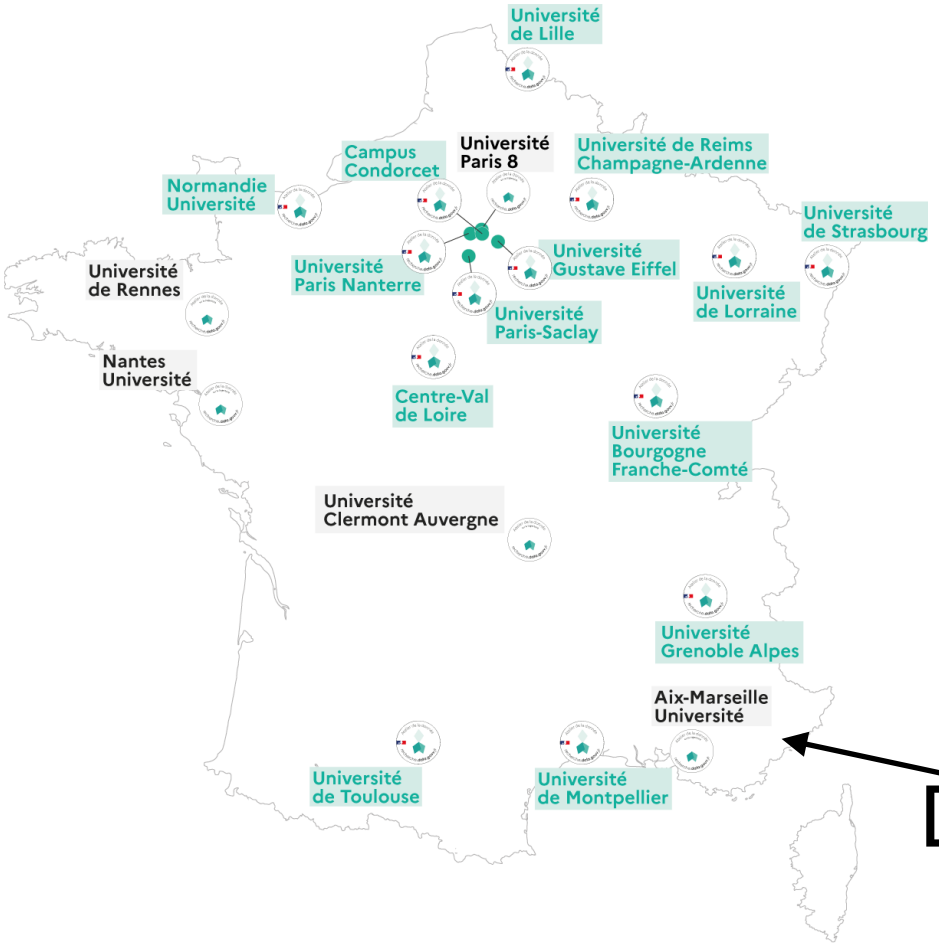
Open Science



https://www.ouvrirlascience.fr/national-plan-for-open-science-4th-july-2018/

research.data.gouv.fr

research.data.gouv.fr



EOSC

What the European Open Science Cloud is

The ambition of the European Open Science Cloud (EOSC) is to provide European researchers, innovators, companies and citizens with a federated and open multi-disciplinary environment where they can publish, find and reuse data, tools and services for research, innovation and educational purposes.

This environment will operate under well-defined conditions to ensure trust and safeguard the public interest.

The EOSC enables a step change across scientific communities and research infrastructures towards

- seamless access
- FAIR (Findability, Accessibility, Interoperability and Reusability) management
- reliable reuse of research data and all other digital objects produced along the research life cycle (e.g. methods, software and publications)

The European Open Science Cloud (EOSC) ultimately aims to develop a 'Web of FAIR Data and services' for science in Europe upon which a wide range of value-added services can be built. These range from visualisation and analytics to long-term information preservation or the monitoring of the uptake of open science practices.

The EOSC is recognised by the Council of the European Union among the 20 actions of the policy agenda 2022-2024 of the European Research Area (ERA) with the specific objective to deepen open science practices in Europe. It is also recognised as the "science, research and innovation data space" which will be fully articulated with the other sectoral data spaces defined in the European strategy for data.

Full deployment of the EOSC will lead to higher research productivity, new insights and innovations, as well as improved reproducibility and trust in science.

bit.ly/eosc

EOSC

What the European Open Science Cloud is

The ambition of the European Open Science Cloud (EOSC) is to provide European researchers, innovators, companies and citizens with a federated and open multi-disciplinary environment where they can publish, find and reuse data, tools and services for research, innovation and educational purposes.

This environment will operate under well-defined conditions to ensure trust and safeguard the public interest.

The EOSC enables a step change across scientific communities and research infrastructures towards

- seamless access
- FAIR (Findability, Accessibility, Interoperability and Reusability) management
- reliable reuse of research data and all other digital objects produced along the research life cycle (e.g. methods, software and publications)

The European Open Science Cloud (EOSC) ultimately aims to develop a 'Web of FAIR Data and services' for science in Europe upon which a wide range of value-added services can be built. These range from visualisation and analytics to long-term information preservation or the monitoring of the uptake of open science practices.

The EOSC is recognised by the Council of the European Union among the 20 actions of the policy agenda 2022-2024 of the European Research Area (ERA) with the specific objective to deepen open science practices in Europe. It is also recognised as the "science, research and innovation data space" which will be fully articulated with the other sectoral data spaces defined in the European strategy for data.

Full deployment of the EOSC will lead to higher research productivity, new insights and innovations, as well as improved reproducibility and trust in science.

bit.ly/eosc

Problems

- Ultimately: data sharing
- "Easy"
  - experiments and results
    - publications
  - HAL
  - journals
  - arxiv
- "Hard"
  - software
- "Harder"
  - data

Problems: Hard

- Publish software
- License
- Tests
- Installation / configuration
- [multi]platform

Problems: Harder

- Data and databases
- License
- Management
  - storage
  - access
- Documentation
- Format

Your role

- Try to publish your data
  - Ask your institution
  - Ask your supervisor
  - Find a data center for your data domain
    - Human and social science
    - Astrophysics
    - Chemistry
    - Environment
    - …
- And ask Datazur for help in publishing a data paper
  - and the data too