GIT Introduction

What is that? Who needs it? How does it work?

DATAVOLUTION - THE SURVIVAL OF THE BITTEST

- UH, OH! Pete
- Separate
- Share!
- Connect!
- Explore!
- Pete et al.
- Citation

UM! It seems like unshared data will easily go extinct.

...explore, connect and share your data, so science can evolve!


Jörg Steinkamp, Moritz Schlarb, Christian Meesters

ZDV Zentrum für Datenverarbeitung
1. What is Git?
2. Who needs Git?
3. How git handles files
4. GitHub.com vs GitLab.com
5. Online Demo
What is Git?

Git is ...

- a Version Control System (VCS)
- a Revision Control System (RCS)
- a Source Code Manager (SCM)
What is Git?

Git is ...

- a Version Control System (VCS)
- a Revision Control System (RCS)
- a Source Code Manager (SCM)

or any meaningful combination of the above words
What is Git?

**Git is ...**

- a **Version Control System (VCS)**
- a Revision Control System (RCS)
- a Source Code Manager (SCM)

or any meaningful combination of the above words

**History**

- 1970s: Source Code Control System (SCCS)
- early 1980s: Revision Control System (RCS)
- 1986: Concurrent Version System (CVS)
- ~2001: Subversion (SVN)
- BitKeeper, Mercurial, Monotone
- 2005: Git

J. Loelinger (2009): Version Control with Git
Distributed
What is Git?

- Distributed
- Scalable
What is Git?

- Distributed
- Scalable
- Fast and efficient
What is Git?

- Distributed
- Scalable
- Fast and efficient
- Integrity and trust
What is Git?

- Distributed
- Scalable
- Fast and efficient
- Integrity and trust
- Immutability
What is Git?

- Distributed
- Scalable
- Fast and efficient
- Integrity and trust
- Immutability
- Branched development
What is Git?

- Distributed
- Scalable
- Fast and efficient
- Integrity and trust
- Immutability
- Branched development
- Free and open
Centralized vs. Distributed

Direct commit & update on server
Centralized vs. Distributed

- Direct commit & update on server
- Push & pull to/from server
- Commit & update to local copy
Performance

Scalability; fast and efficient
- Work individually or in large groups
- Small projects and large projects

Integrity and trust; immutability
Once committed & pushed no manipulation is possible
Branching

What is Git?

Driesen, 2010

Steinkamp, Schlarb, Meesters

GIT Introduction
Development and Operation: DevOps

- concept

plan
Development and Operation: DevOps

- concept
- first draft
Development and Operation: DevOps

- concept
- first draft
- supervisor feedback
Development and Operation: DevOps

- concept
- first draft
- supervisor feedback
- submission
Development and Operation: DevOps

- concept
- first draft
- supervisor feedback
- submission
- in review
- concept
- first draft
- supervisor feedback
- submission
- in review
- feedback
Who needs Git?

Everyone who wants to avoid

HAVE YOU FAILED TO REPRODUCE AN EXPERIMENT?

Many top-rated factors relate to intense competition and time pressure. Most scientists have experienced failure to reproduce results.

How git handles files

- Untracked
- Unmodified
- Modified
- Staged

modified after Chacon & Straub: Pro Git
How git handles files

- **Untracked**
- **Unmodified**
- **Modified**
- **Staged**

Add file(s)

modified after Chacon & Straub: Pro Git
How git handles files

- Untracked
- Unmodified
- Modified
- Staged

Add file(s)

Edit file(s)

modified after Chacon & Straub: Pro Git
How git handles files

- Untracked
- Unmodified
- Modified
- Staged

Add file(s)
Edit file(s)
Stage file(s)

modified after Chacon & Straub: Pro Git
How git handles files

- Untracked
- Unmodified
- Modified
- Staged

Add file(s) → Edit file(s) → Stage file(s)

Remove file(s)

modified after Chacon & Straub: Pro Git
How git handles files

- Untracked
- Unmodified
- Modified
- Staged

Add file(s) → Edit file(s) → Stage file(s) → Commit changes

Remove file(s)

modified after Chacon & Straub: Pro Git
Good filetypes

- all text files
  - .txt
  - .rtf
  - .svg
  - .csv
  - ...

Bad filetypes

- all binary files
  - .doc, .docx
  - .xls, .xlsx
  - .png, .jpg
  - .mp3, .mp4
  - ...
Filetypes

Good filetypes
- all text files
  - .txt
  - .rtf
  - .svg
  - .csv
  - ...

Bad filetypes
- all binary files
  - .doc, .docx
  - .xls, .xlsx
  - .png, .jpg
  - .mp3, .mp4
  - ...

Steinkamp, Schlarb, Meesters
GIT Introduction
How git handles files
Good filetypes

- all text files
  - .txt
  - .rtf
  - .svg
  - .csv
  - ...

Bad filetypes

- all binary files
  - .doc, .docx
  - .xls, .xlsx
  - .png, .jpg
  - .mp3, .mp4
  - ...

... and why???
Filetypes

GIT Introduction

How git handles files 11 / 16

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>XXX The book begins with city council workmen arriving at Arthur</td>
</tr>
<tr>
<td>2</td>
<td>Dent's house. They wish to demolish his house in order to build a XXX bypass.</td>
</tr>
<tr>
<td>3</td>
<td>Arthur's best friend, Ford Prefect, arrives, warning him of the end of the world. Ford is revealed to be an alien who</td>
</tr>
<tr>
<td>4</td>
<td>had come to Earth to research it for the titular Hitchhiker's Guide to the Galaxy, an enormous work providing information about every planet and place in the universe. The two head</td>
</tr>
<tr>
<td>5</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
</tr>
</tbody>
</table>
GitHub.com vs GitLab.com

GitHub
- repos initially public
- coarse access control
- no integrated CI/CD
- no self-hosting
- zenodo integration

GitLab
- repos initially private
- fine access control
- integrated CI/CD
- self-hosting
- no zenodo integration

modified after @Cloudways
GitHub.com vs GitLab.com

GitHub
- repos initially public

GitLab
- repos initially private

modified after @Cloudways
<table>
<thead>
<tr>
<th>GitHub</th>
<th>GitLab</th>
</tr>
</thead>
<tbody>
<tr>
<td>repos initially public</td>
<td>repos initially private</td>
</tr>
<tr>
<td>coarse access control</td>
<td>fine access control</td>
</tr>
</tbody>
</table>

modified after @Cloudways
<table>
<thead>
<tr>
<th>GitHub</th>
<th>GitLab</th>
</tr>
</thead>
<tbody>
<tr>
<td>repos initially public</td>
<td>repos initially private</td>
</tr>
<tr>
<td>coarse access control</td>
<td>fine access control</td>
</tr>
<tr>
<td>no integrated CI/CD</td>
<td>integrated CI/CD</td>
</tr>
</tbody>
</table>

Modified after @Cloudways

©CloudTweaks.com
## GitHub vs GitLab

**GitHub**
- repos initially public
- coarse access control
- no integrated CI/CD
- no self-hosting

**GitLab**
- repos initially private
- fine access control
- integrated CI/CD
- self-hosting

---

Modified after @Cloudways
GitHub vs GitLab.com

<table>
<thead>
<tr>
<th>GitHub</th>
<th>GitLab</th>
</tr>
</thead>
<tbody>
<tr>
<td>repos initially public</td>
<td>repos initially private</td>
</tr>
<tr>
<td>coarse access control</td>
<td>fine access control</td>
</tr>
<tr>
<td>no integrated CI/CD</td>
<td>integrated CI/CD</td>
</tr>
<tr>
<td>no self-hosting</td>
<td>self-hosting</td>
</tr>
<tr>
<td>zenodo integration</td>
<td>no zenodo integration</td>
</tr>
</tbody>
</table>

modified after @Cloudways
Beginner course, introducing
- Gitlab WebUI
- Issues
- Merging

Advanced course, dealing with
- Protected branches
- Code Review in Gitlab
- CI/CD
Introduction to shell programming

- Crash course for beginners
- Mandatory knowledge for HPC.
Scientific Computing – Upcoming Courses

Introduction to shell programming
- Crash course for beginners
- Mandatory knowledge for HPC.

Introduction to High Performance Computing (HPC)
a.k.a. “Mogon-Introduction” – HPC and cluster usage for our local clusters
Scientific Computing – Upcoming Courses

Introduction to shell programming

- Crash course for beginners
- Mandatory knowledge for HPC.

Introduction to High Performance Computing (HPC)
a.k.a. “Mogon-Introduction” – HPC and cluster usage for our local clusters

Various other courses

- Ilias - Registration for HPC
- Ilias - Registration for Linux/Unix
- https://hpc.uni-mainz.de/kurse-und-workshops/
Further readings

- Git-SCM
- git novice intro by swcarpentry
- git - the simple guide
- 'Expert' answers @ StackOverflow
- Oh shit, git! (pardon the profanity)
- How to teach Git
- Git auf deutsch