Azrieli
College of Engineering
Jerusalem

עזריאלי
מכללה אקדמית להנדסה
ירושלים

הנדסת תוכנה

6. בקרת גרסאות
git / github

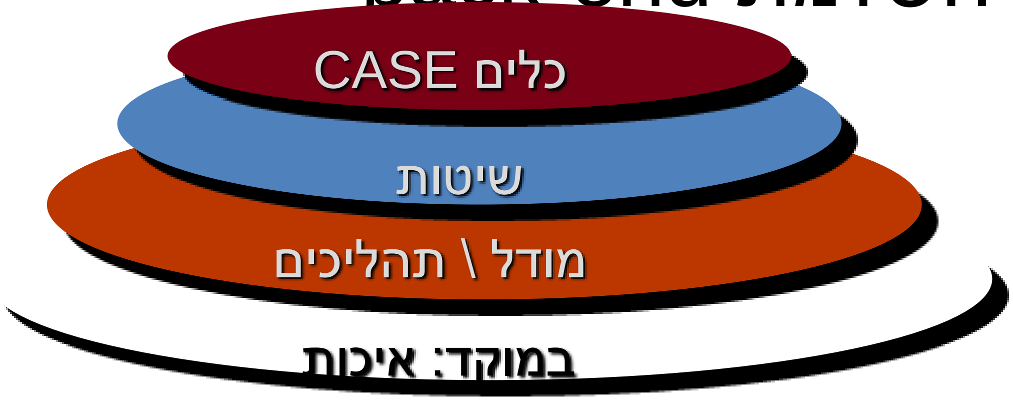
Pragmatic Programmer Tip :

Always Use Source Code Control

Source code control is a time machine for your work—you can go back.

מה היום?

- בקרת גרסאות קוד – Version Control
- Git
- Github
- סקר ZFR
- פרויקט: תחילת סבבים
- תרגיל – סקר ZFR + השלמת back-end



בקרת גרסאות קוד - מקורות

- Sink, Version Control by Example
- Google Tech Talk: Linus Torvalds on git
- Spolsky, Hg Init: a Mercurial tutorial http://hginit.com
- Sink, Source Control HOWTO http://www.ericsink.com/scm/source_control.html
- Intro to Distributed Version Control http://betterexplained.com/articles/intro-to-distributed-version-control-illustrated/

VCS Links

- Eric Raymond on vcs
- The 10 commandments of good source control management, blog 2011
- FogBugz and Kiln video, 2011
  - DVCS University Slides (*)
- Spolsky, Hg Init: a Mercurial tutorial
- Azad, A Visual Guide to Version Control
  - Intro to Distributed Version Control (Illustrated)
- Eric Sink, Source Control HOWTO

Git Links

- **http://git-scm.com/** (getting started)
- Set Up Git (Win), ssh issues http://help.github.com/win-set-up-git/
- Info: http://git-scm.com, gitref.org, progit.org (book, basic),
- Windows client list, Egit (Eclipse), VS, power-shell, gihtub for windows (slds)
- O'Reilly Webcast: Git in One Hour
- git internals video

More Git / Github Links

- Git branching with git-flow, Heb. Video, 2013
- Videos: Git For Ages 4 And Up,
  Git Going(oredev'12), Think like a Git
- learn.github.com/, try.github.com/
  Gitflow
- saastv: Using Branches with Git
- no branches at flickr
- Insider Guide to GitHub (video)
- Articles: article (including .gitignore), post,
  difficulties,

איפה אנחנו בפרויקט (בקורס)?

- למה?
  בעיה (פלט: הצעת פרויקט\חזון\SOW)
- מי?
  צוות (Inception, אתחול\תכנון פרויקט)
- מה?
  דרישות (SRS)
- איך?
  תיכון (ארכיטקטורה) (SDS)
- מתי?
  תכנון וניהול – (ZFR)
- בניה
  (סבבי פיתוח)

Github

- ויקי
- ~~הורדות~~
- חברתי \ גראפים
- ✓ ניהול משימות
- ➢ בקרת גרסאות
- תרומות
- סקרי קוד (בהמשך)
- קישור לאתרי רזומה
- חינם לקוד פתוח

Have projects on GitHub?
Import them easily to your profile

Version Control – בקרת גרסאות

- איך (האם?) אתם שומרים את תוצרי העבודה שלכם?
- האם אפשר לשפר?
- האם יש הבדל בין מפתח בודד לחברה גדולה?
- שמות שונים:
  - בקרת תצורה
  - Revision Control
  - Software Configuration Management
  - Source-Code/Version Control System

Joel Test (~2000 / stackoverflow)

1. Do you use source control?

וגם היום...

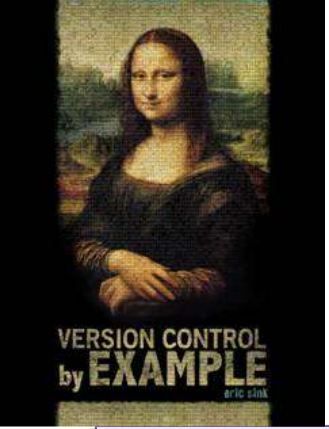
"You’ve just spent twenty minutes doing a presentation for your teammates on adopting source control. Yeah, they don’t do source control at all. Yep, not at all—it’s as if the last 20 years of computer science never happened. But better late than never, and frankly any source control is better than none, because disaster is one errant delete away."

- “Driving Technical Change: Why People on Your Team Don't Act on Good Ideas, and How to Convince Them They Should”, chap. The Cynic

בקרת גרסאות – בשביל מה? יעדים

- שיתוף מספר מפתחים (מרוחקים) בו זמנית
- לדאוג שלא יהיו סתירות בין המפתחים
- איסוף כל הגרסאות ומעקב אחרי שינויים
  - מאפשר מחיקת קוד
- ניהול מספר גרסאות במקביל
- גיבוי
- מאגר מעודכן של תוצרי הפרויקט
  - במיוחד עם Continuous Integration\daily build

בפרויקט תדרשו להדגים את בקרת התצורה שלכם



היסטוריה (השוואה)

Generation	Networking	Operations	Concurrency	Examples
1	None	One file at a time	Locks	RCS, SCCS
2	Centralized	Multi-file	Merge before commit	CVS, SourceSafe, Subversion, Team Foundation Server, IBM Rational ClearCase
3	Distributed	Changesets	Commit before merge	Bazaar, Git, Mercurial

משל גיט

- Tom Preston-Werner http://tom.preston-werner.com/2009/05/19/the-git-parable.html
- Herland, http://www.infoq.com/presentations/git-details, slides: https://github.com/jherland/git_parable

The Git Parable

Johan Herland

johan@herland.net

The Git Parable

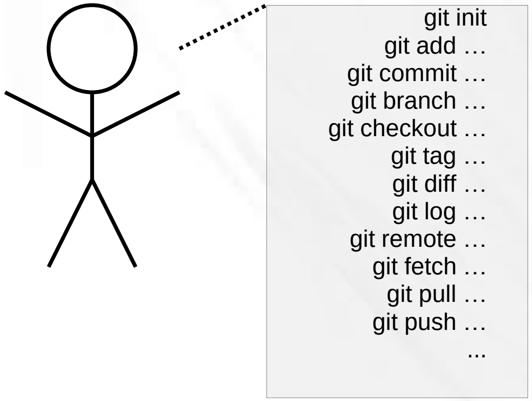
- Shamelessly stolen from Tom Preston-Werner
  http://tom.preston-werner.com/2009/05/19/the-git-parable.html
- I'm lazy...
- Also: Best introduction to Git I've found so far

The Git Parable

- Git - simple & powerful

The Git Parable

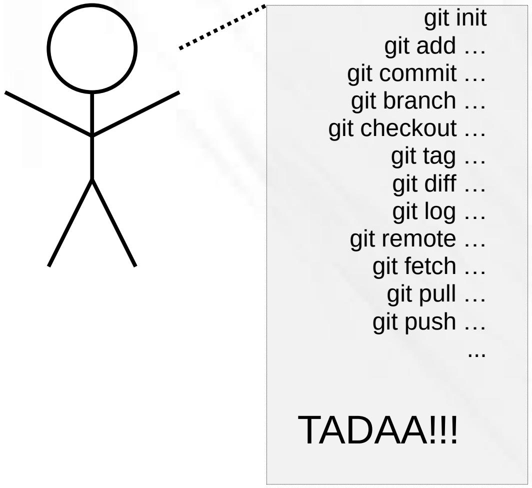
- Git - simple & powerful



```
git init
git add …
git commit …
git branch …
git checkout …
git tag …
git diff …
git log …
git remote …
git fetch …
git pull …
git push …
...
```

The Git Parable

- Git - simple & powerful

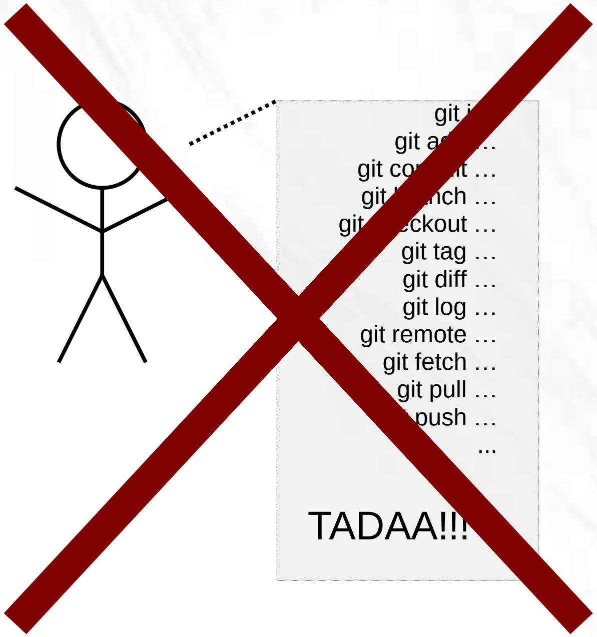


```
git init
git add …
git commit …
git branch …
git checkout …
git tag …
git diff …
git log …
git remote …
git fetch …
git pull …
git push …
...
```

TADAA!!!

The Git Parable

- Git - simple & powerful

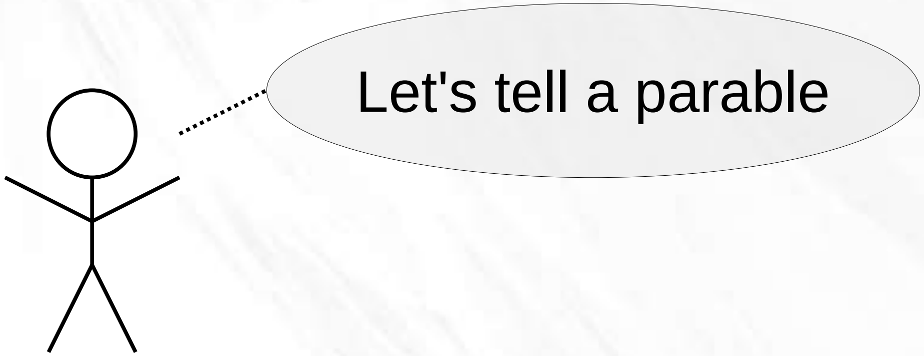


git init …
git add …
git commit …
git branch …
git checkout …
git tag …
git diff …
git log …
git remote …
git fetch …
git pull …
git push …
...

TADAA!!!

The Git Parable

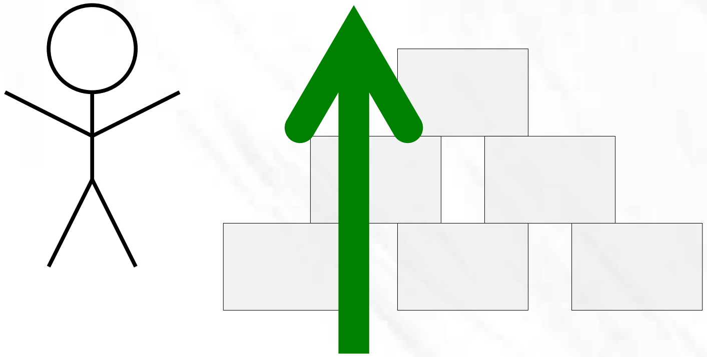
- Git - simple & powerful



Let's tell a parable

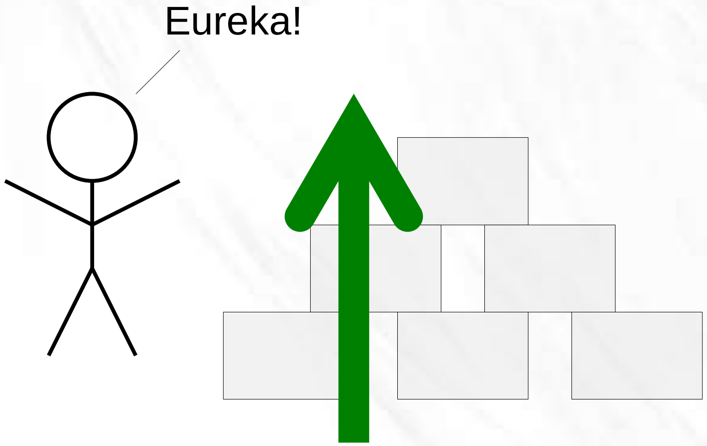
The Git Parable

- Git - simple & powerful



The Git Parable

- Git - simple & powerful



The Parable

- A simple computer
  - A text editor
  - A few filesystem commands

The Parable

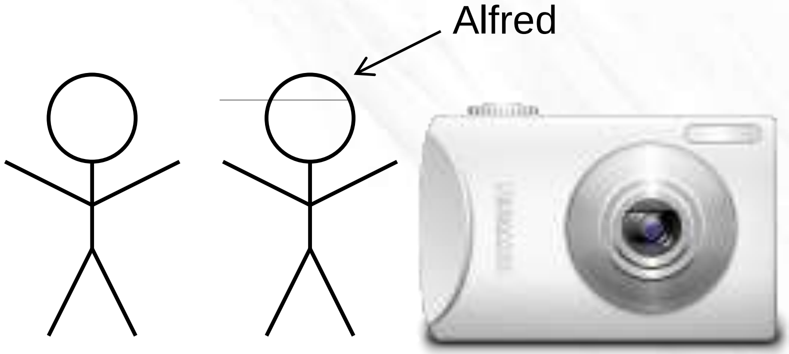
- Write a large software program

The Parable

- Write a large software program
- Invent some method to keep track of versions
  - retrieve code that you changed/deleted

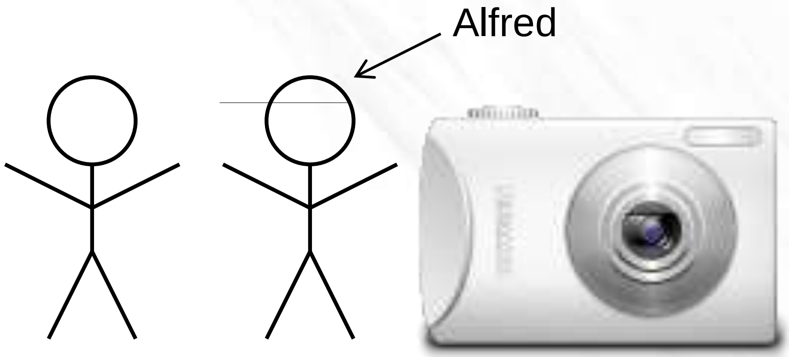
Snapshots

- Alfred, the photographer



Snapshots

- Alfred, the photographer



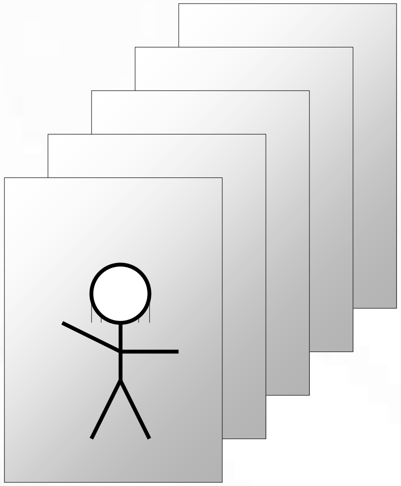
Snapshots

- Alfred, the photographer
- Hazel and her daughter

Hazel
daughter

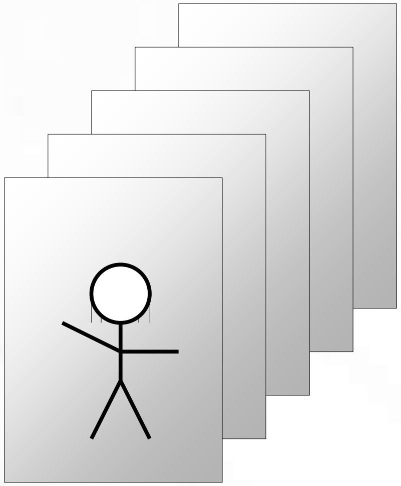
Snapshots

- Alfred, the photographer
- Hazel and her daughter
    - Remember what the daughter was like at each different stage



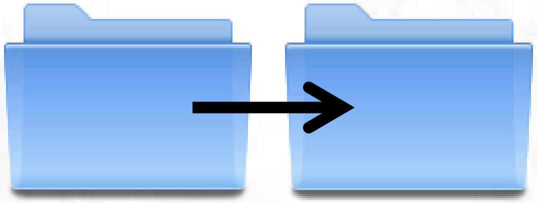
Snapshots

- Alfred, the photographer
- Hazel and her daughter
    - Remember what the daughter was like at each different stage



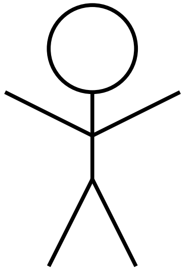
Snapshots

Snapshots

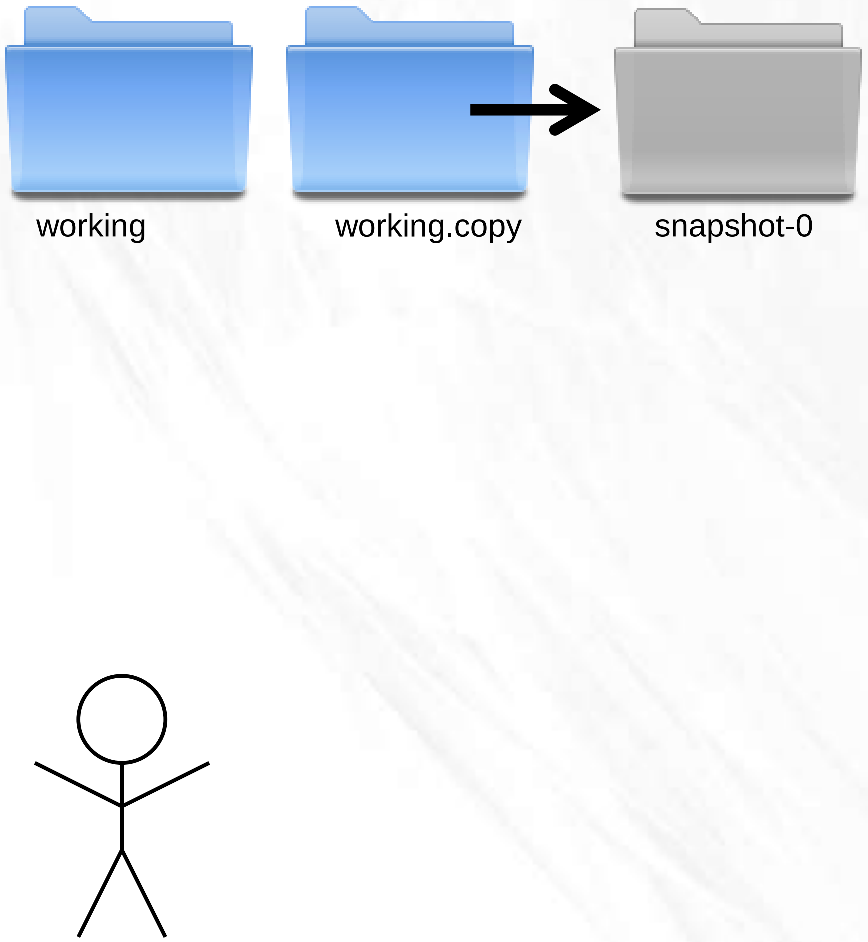


working

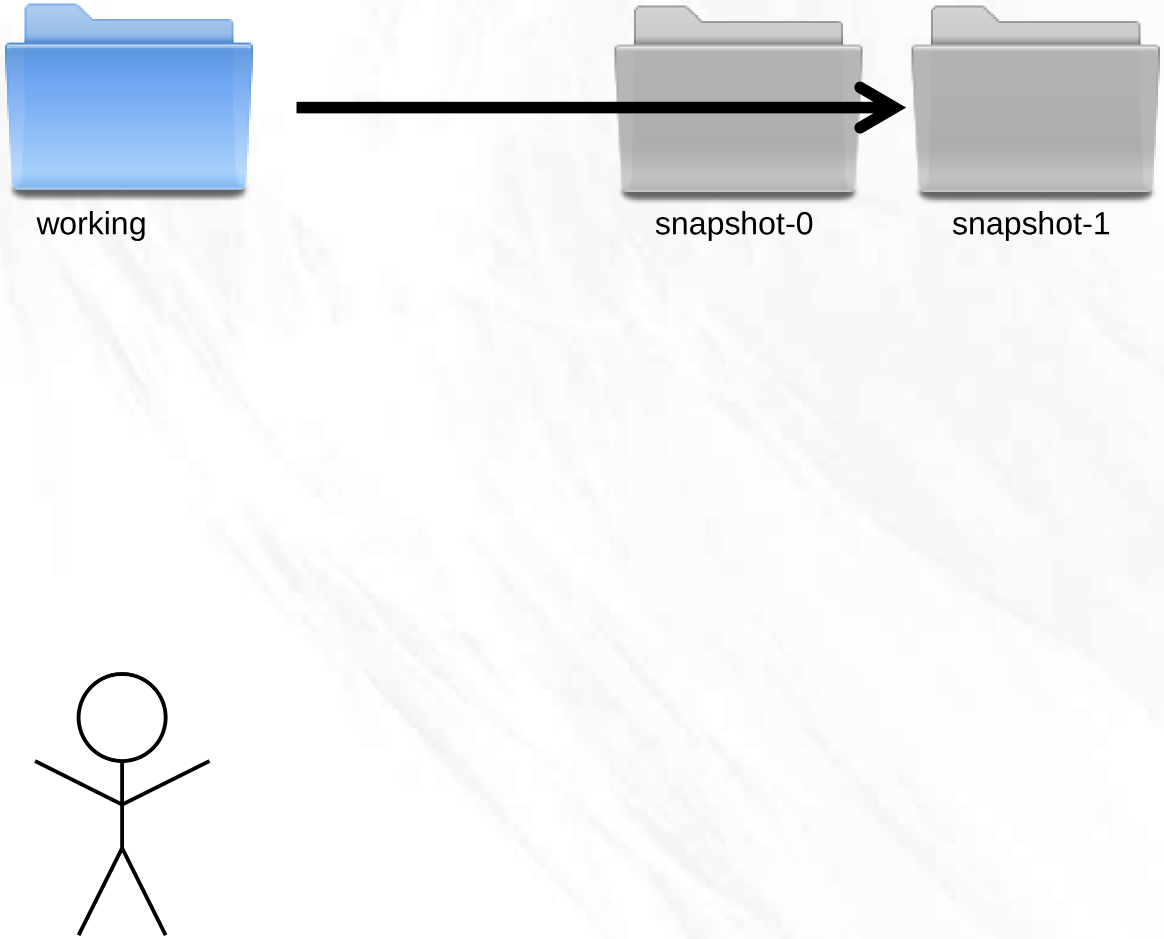
working.copy



Snapshots



Snapshots

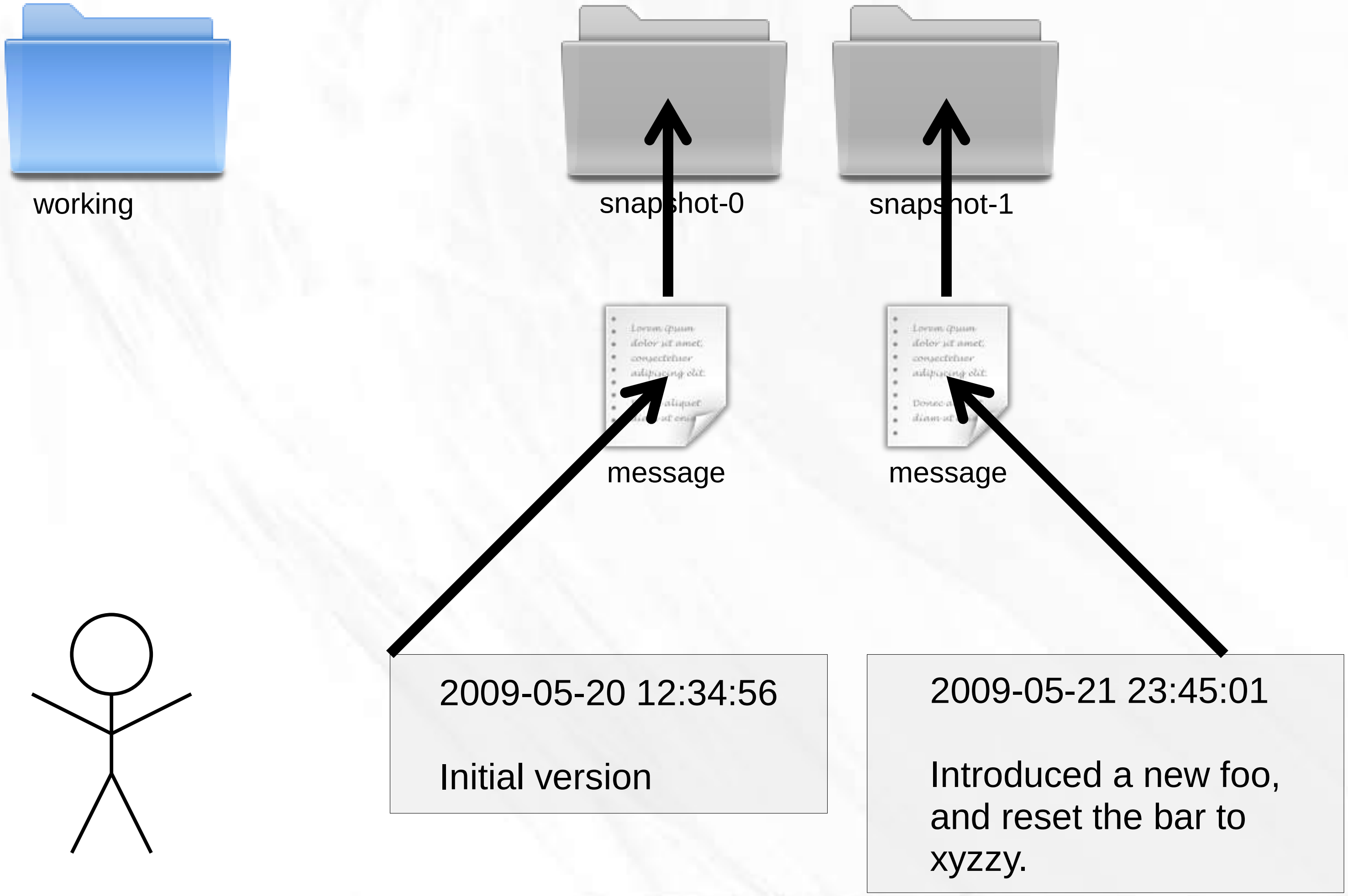


working

snapshot-0

snapshot-1

Snapshots



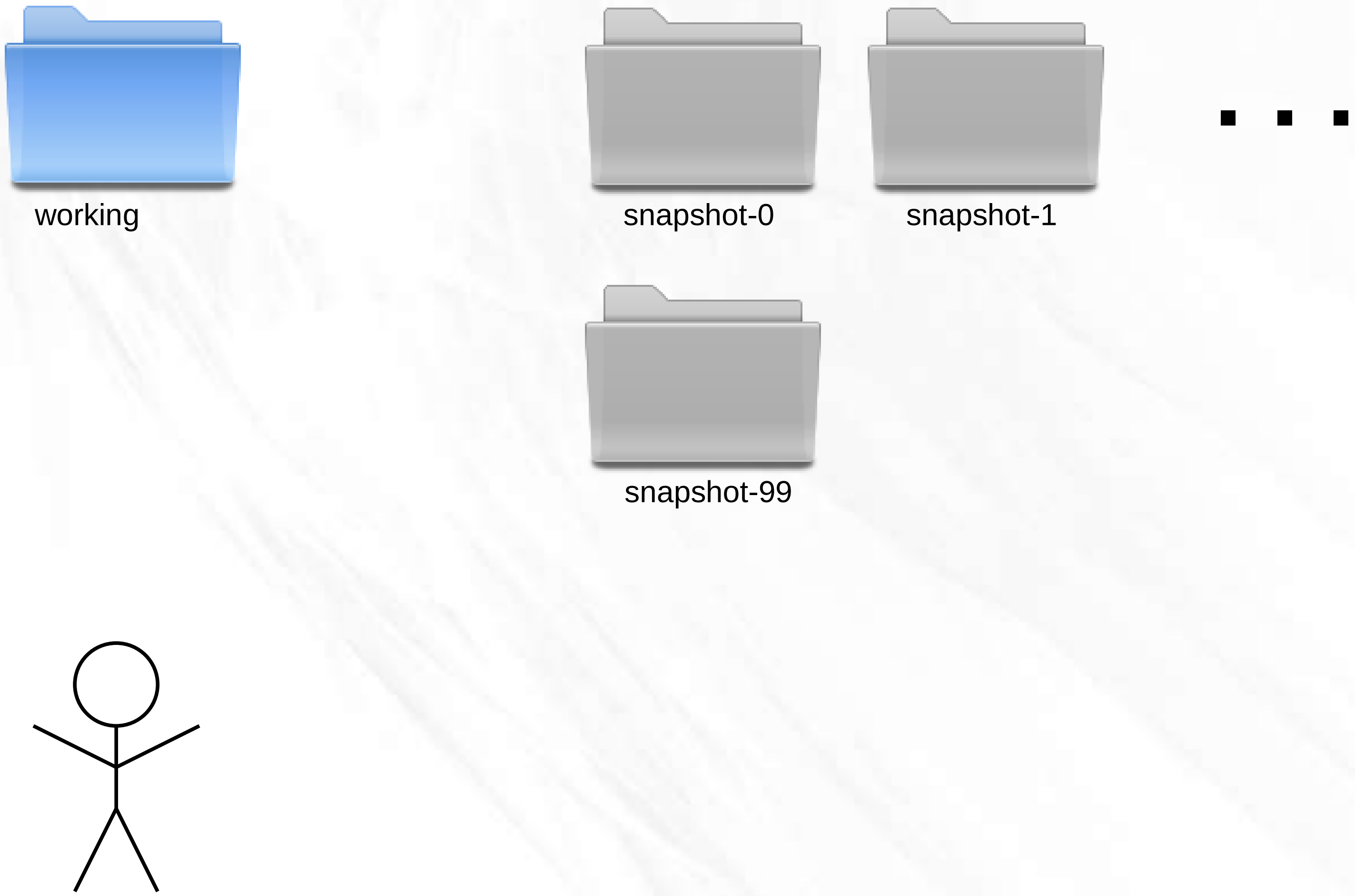
2009-05-20 12:34:56

Initial version

2009-05-21 23:45:01

Introduced a new foo, and reset the bar to xyzzy.

Branches



working

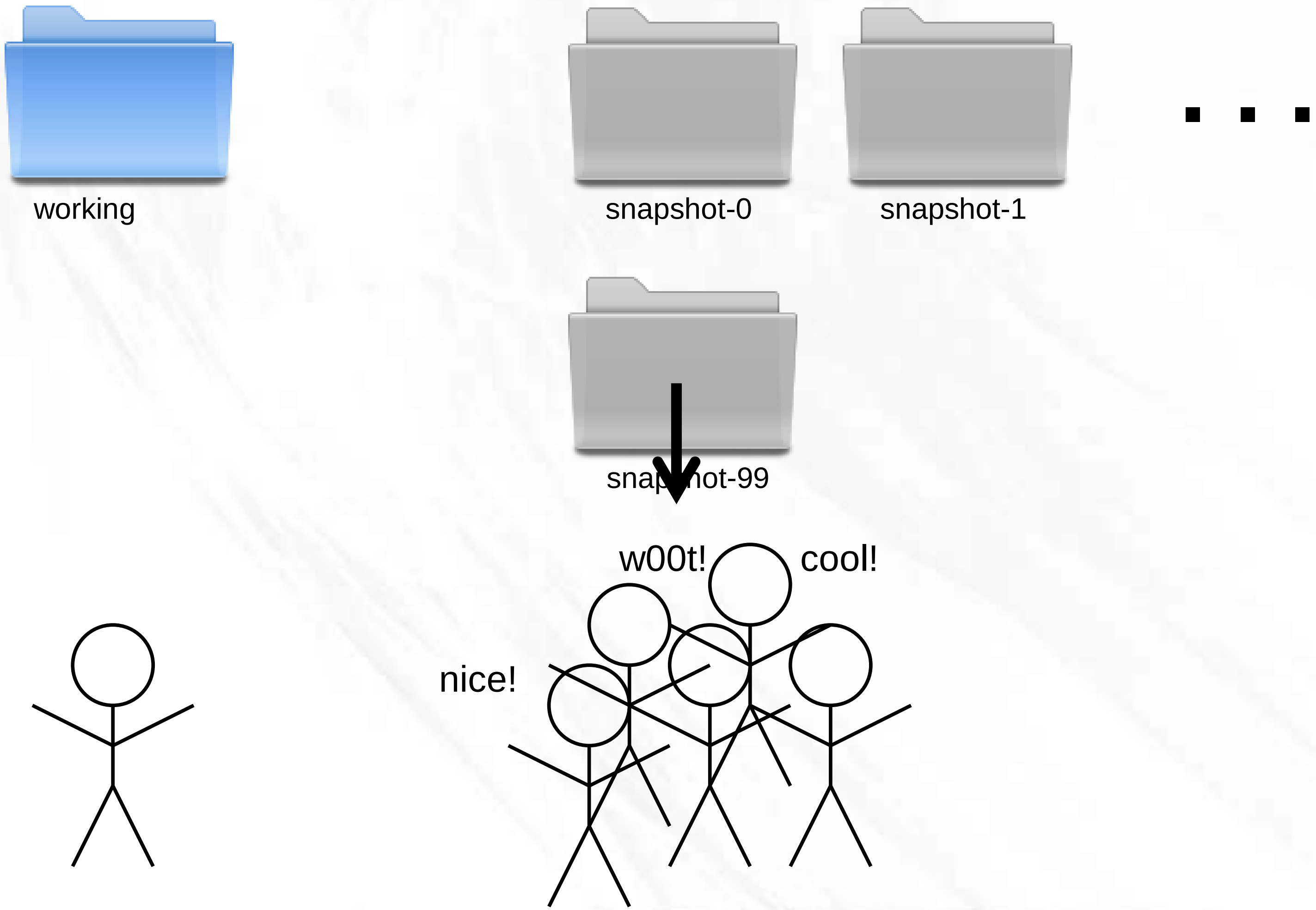
snapshot-0

snapshot-1

. . .

snapshot-99

Branches



working

snapshot-0

snapshot-1

...

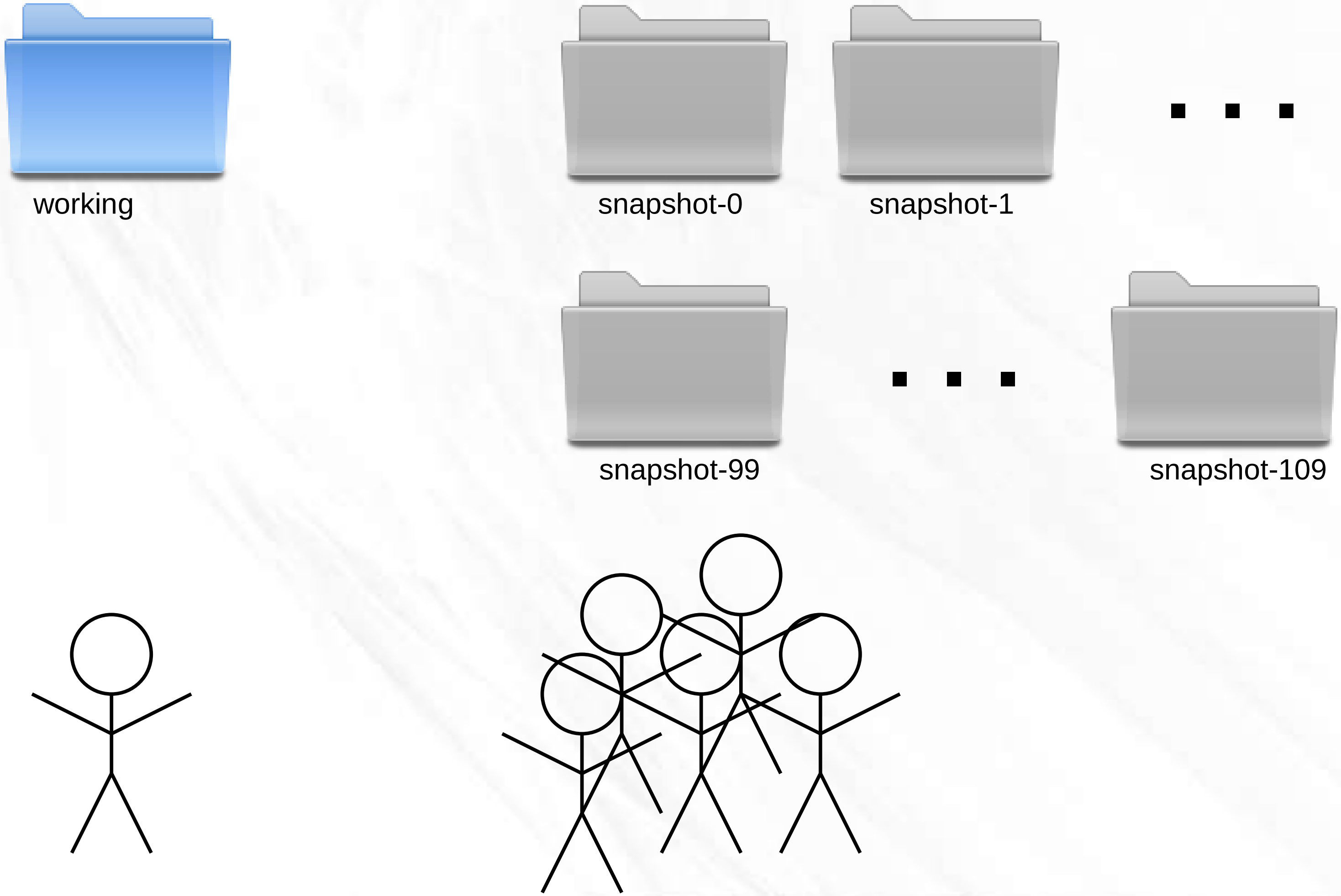
snapshot-99

w00t!

cool!

nice!

Branches



working

snapshot-0

snapshot-1

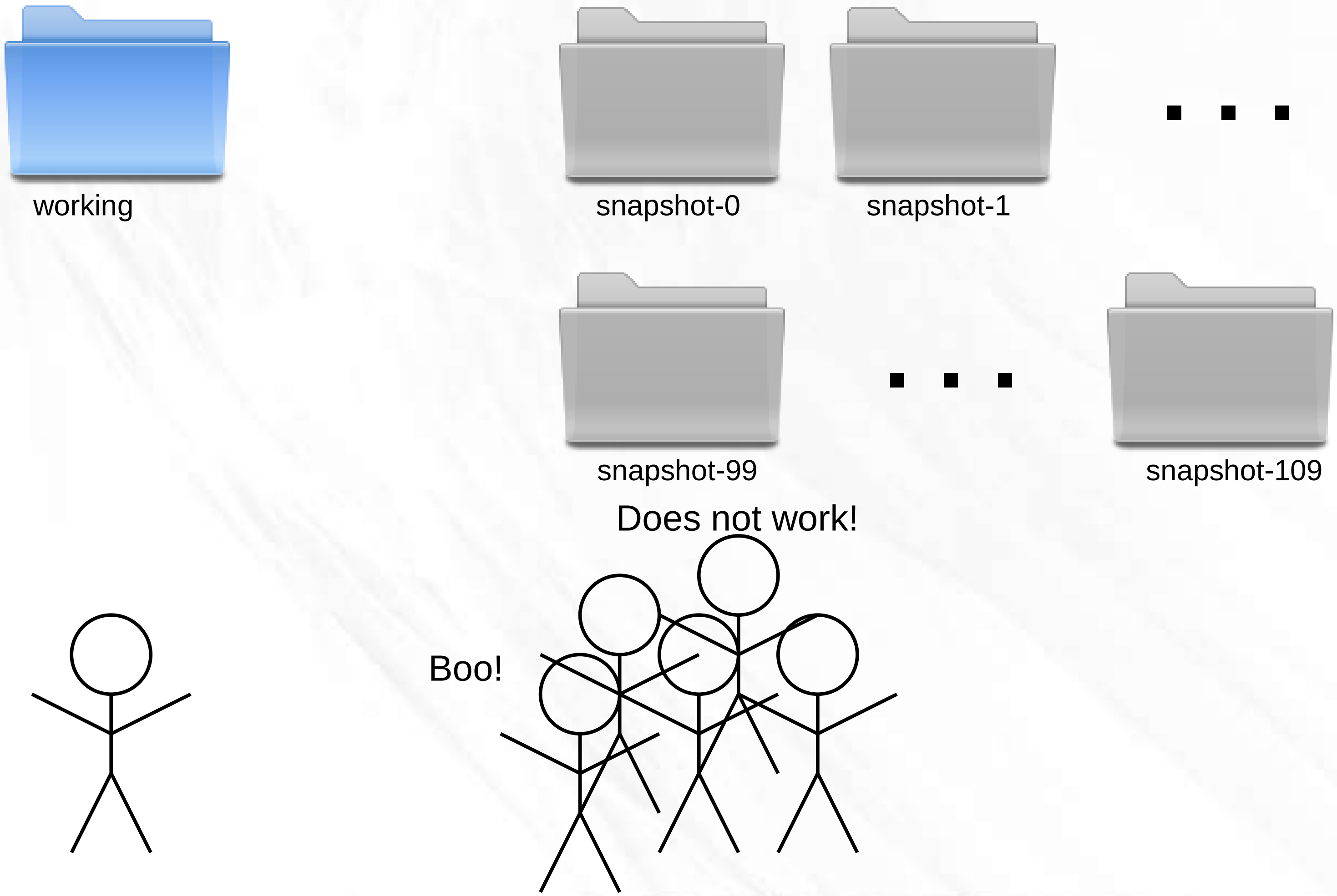
. . .

snapshot-99

. . .

snapshot-109

Branches



working

snapshot-0

snapshot-1

...

snapshot-99

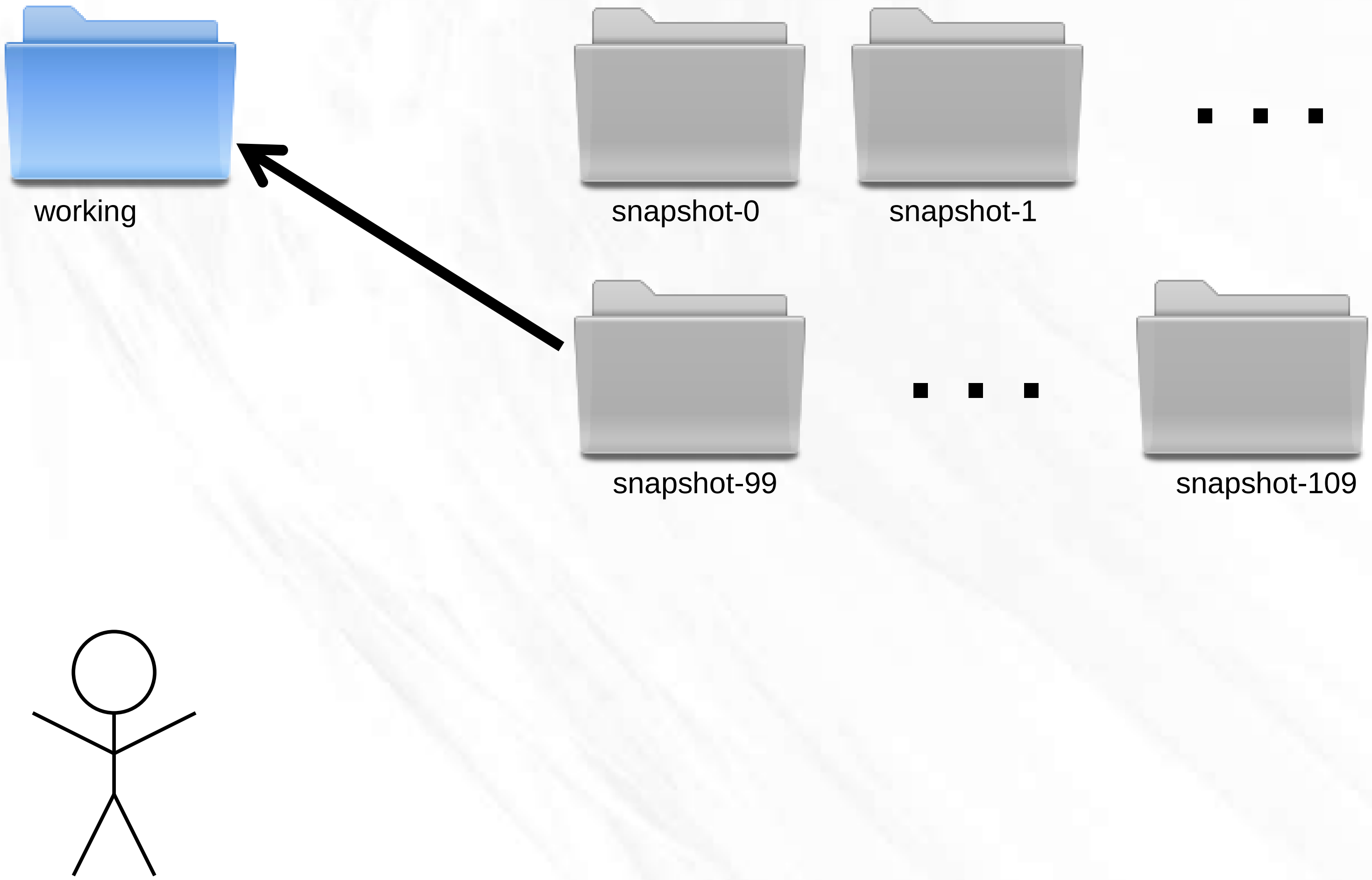
...

snapshot-109

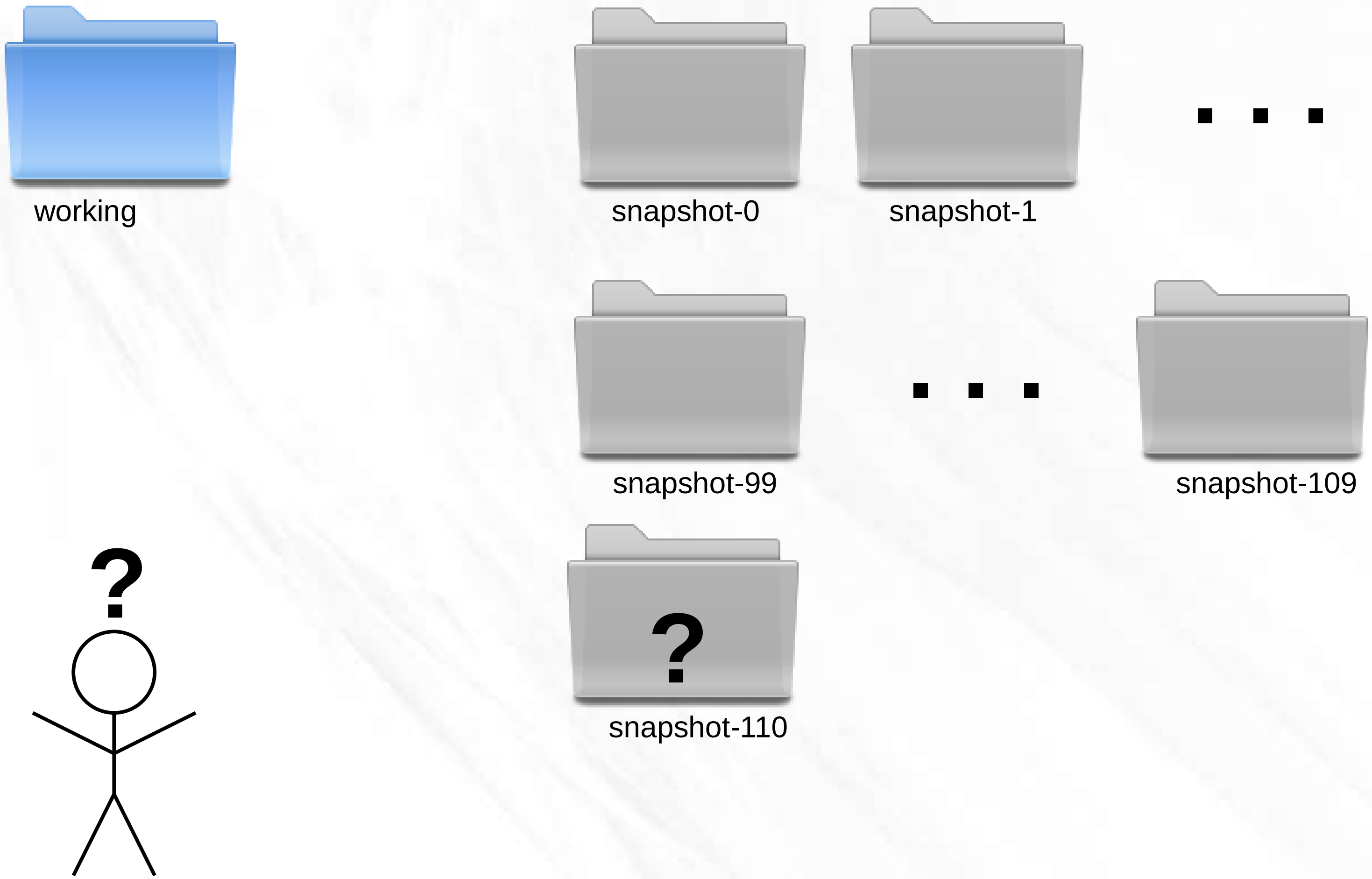
Does not work!

Boo!

Branches



Branches



working

snapshot-0

snapshot-1

. . .

snapshot-99

. . .

snapshot-109

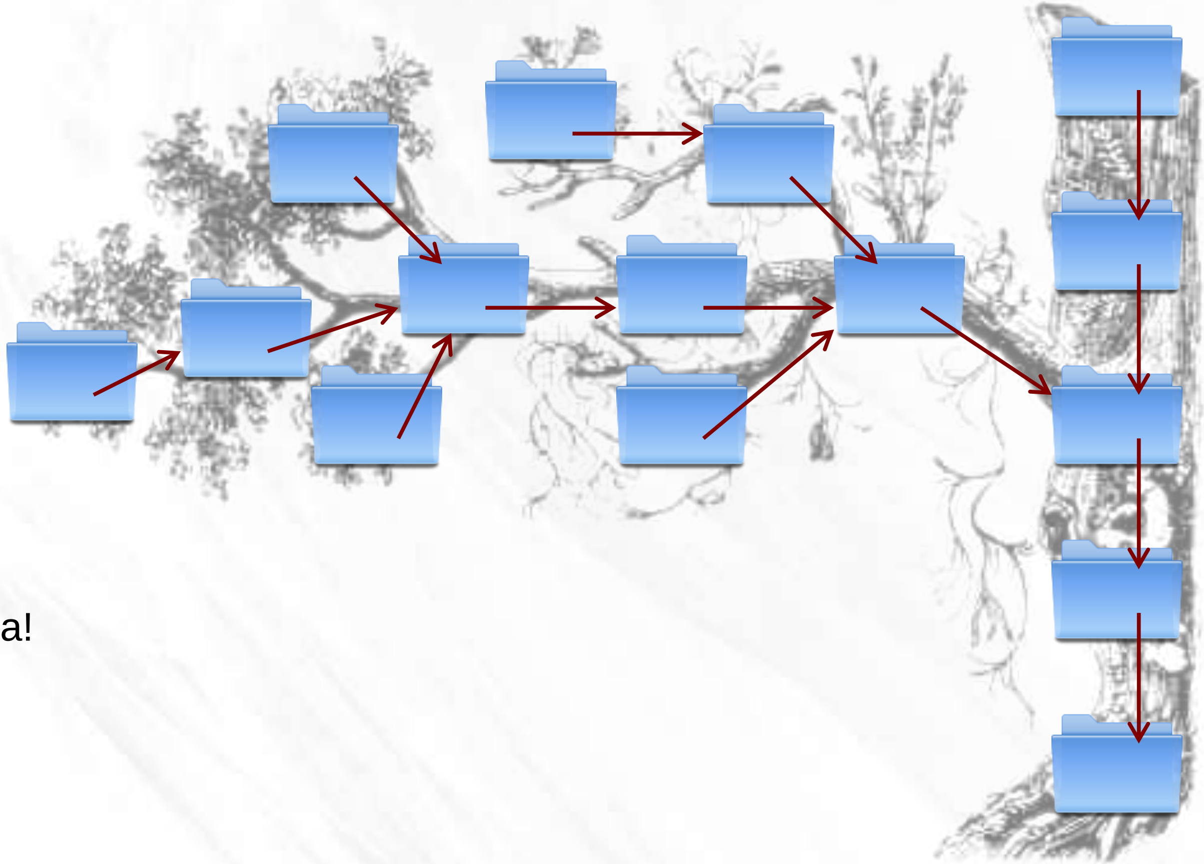
?

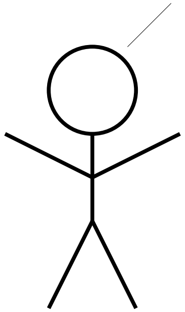
?

snapshot-110

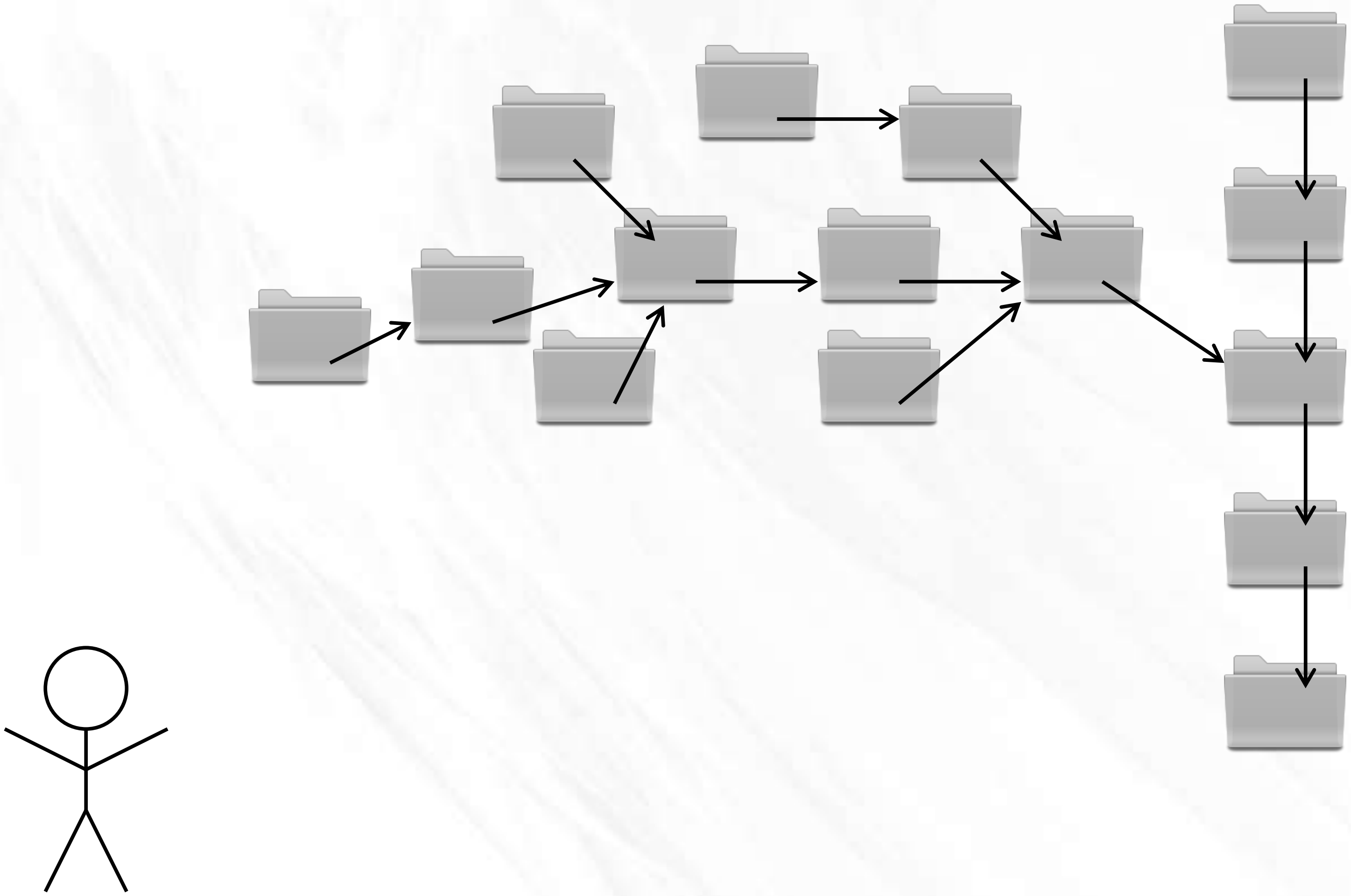
Branches

Branches

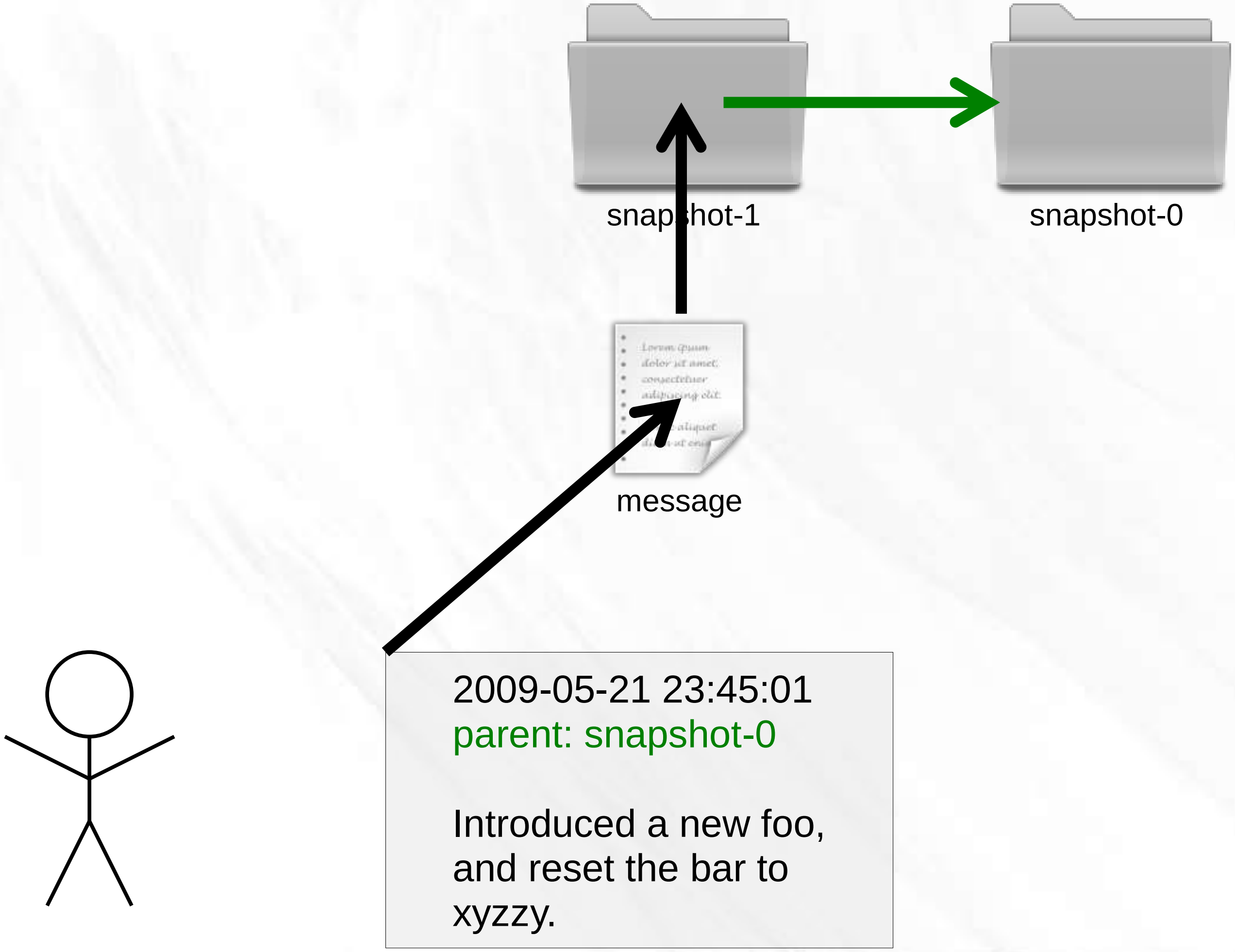




Branches



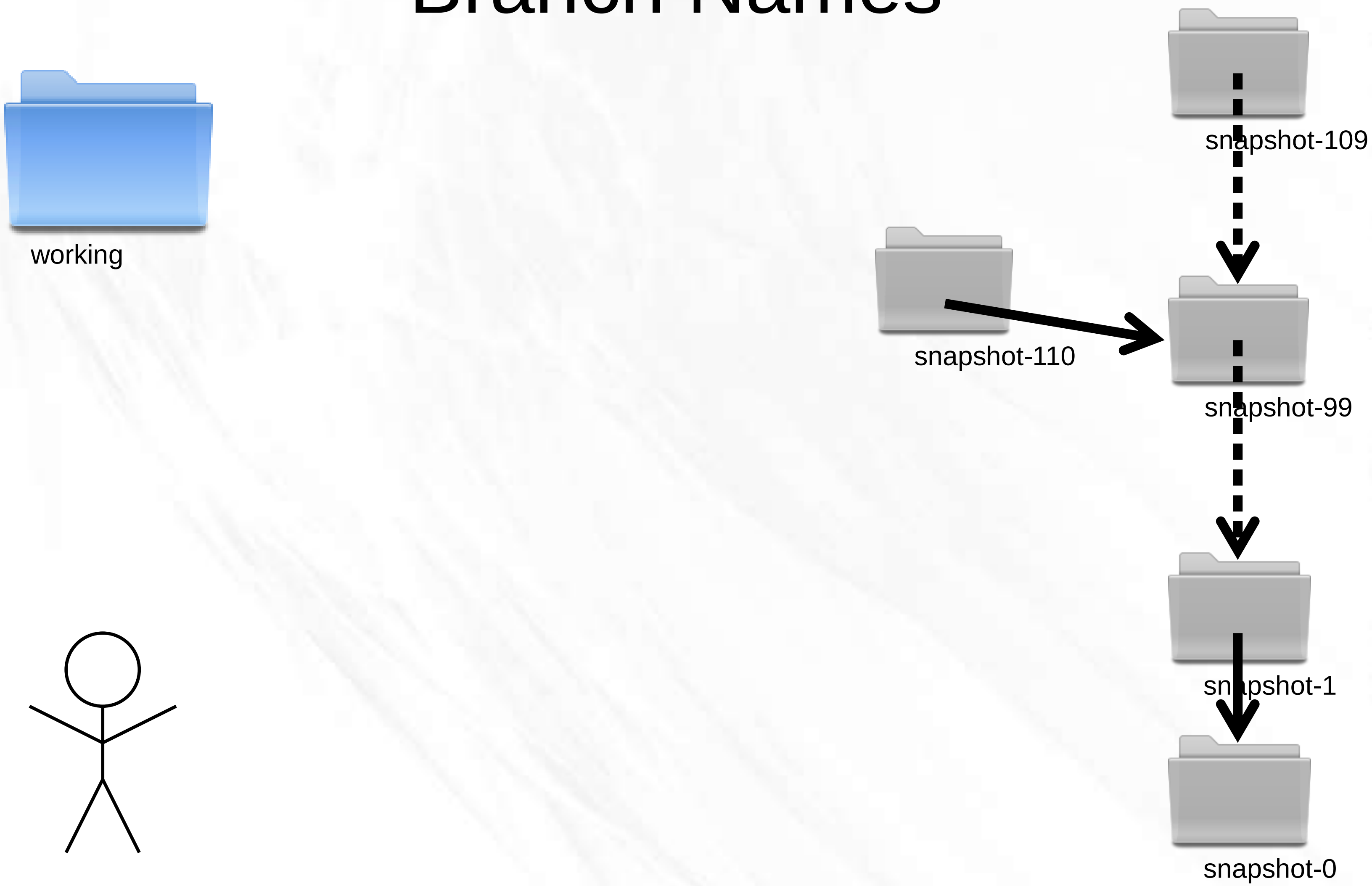
Branches



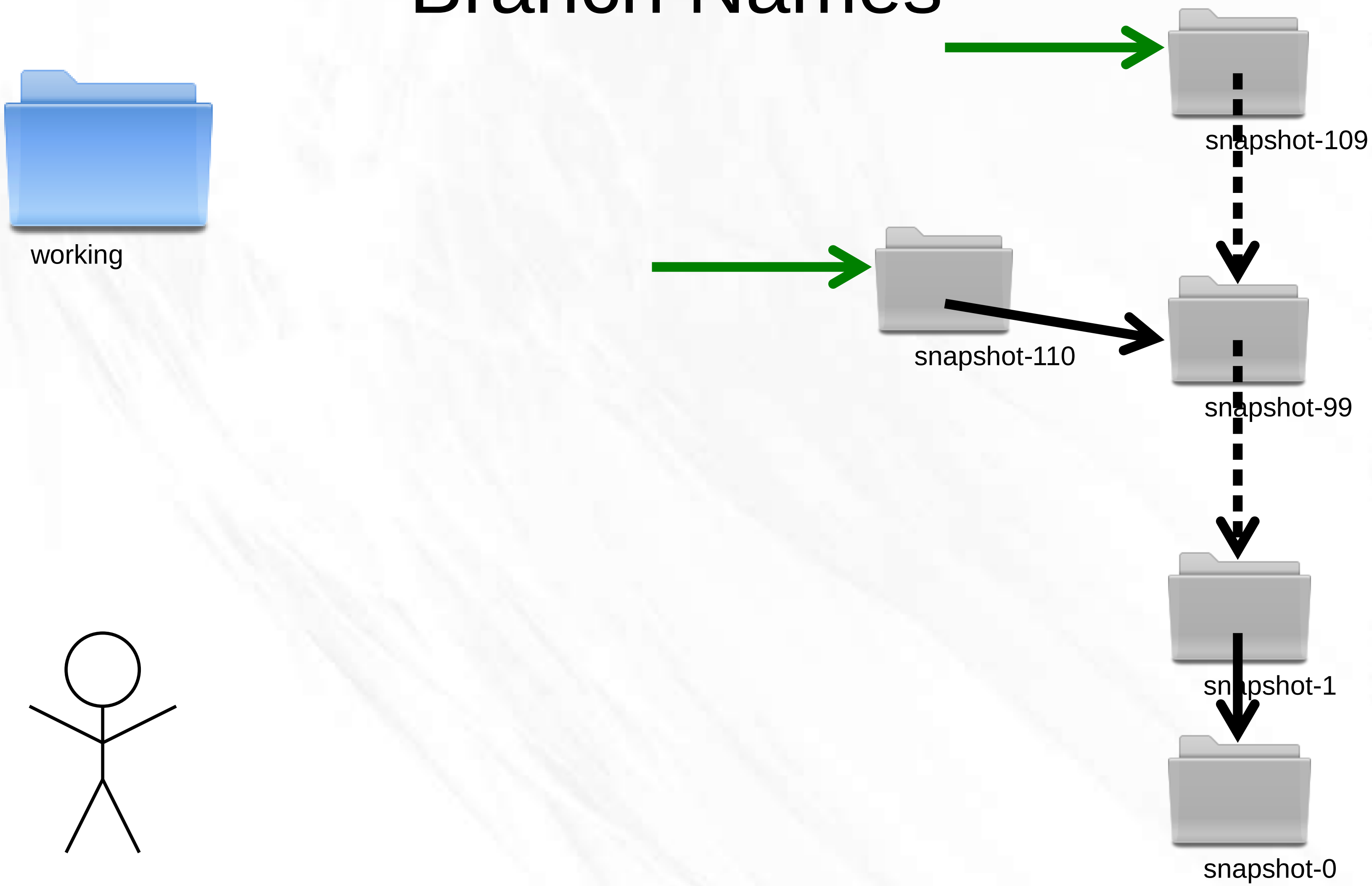
2009-05-21 23:45:01
parent: snapshot-0

Introduced a new foo, and reset the bar to xyzzy.

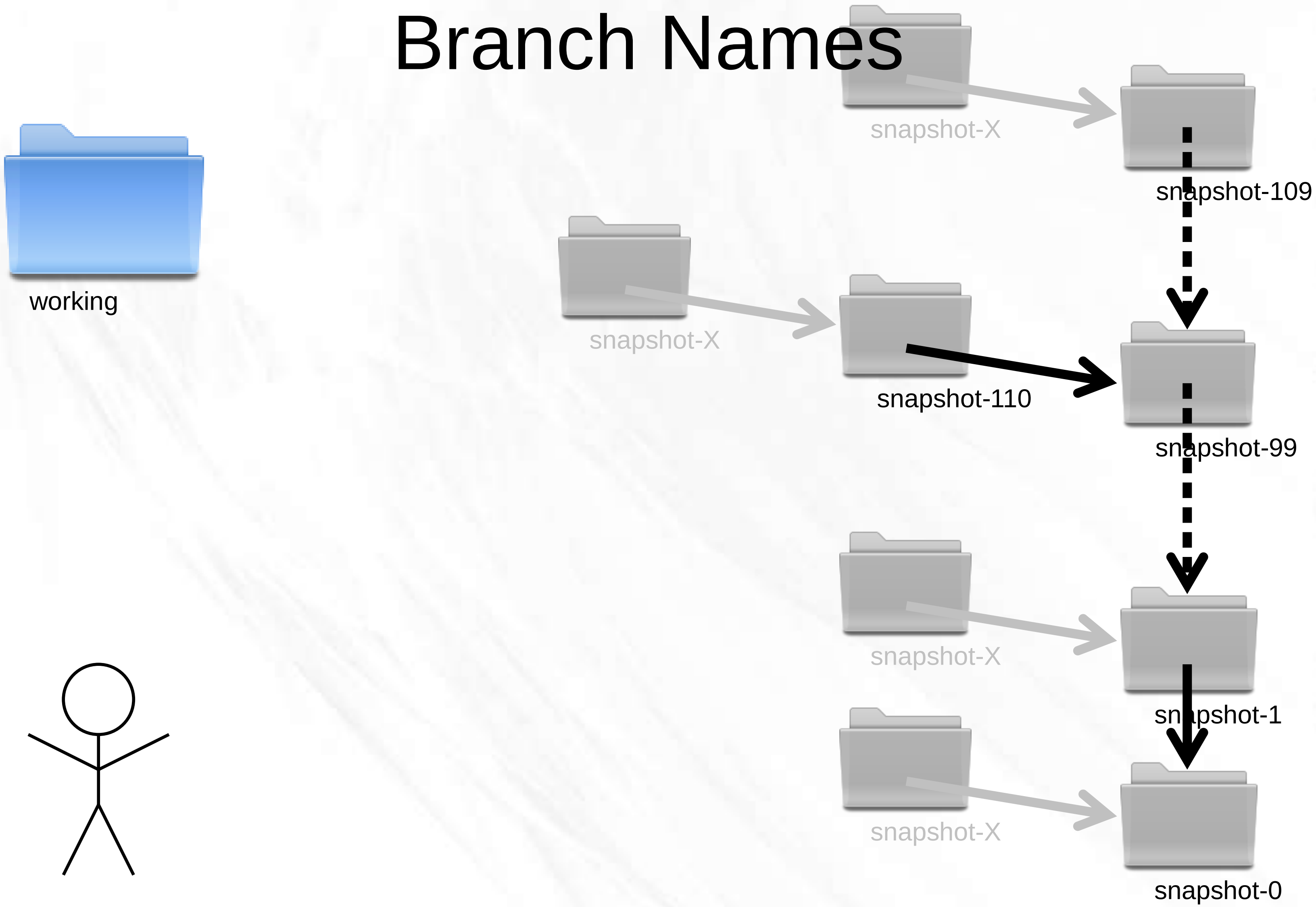
Branch Names



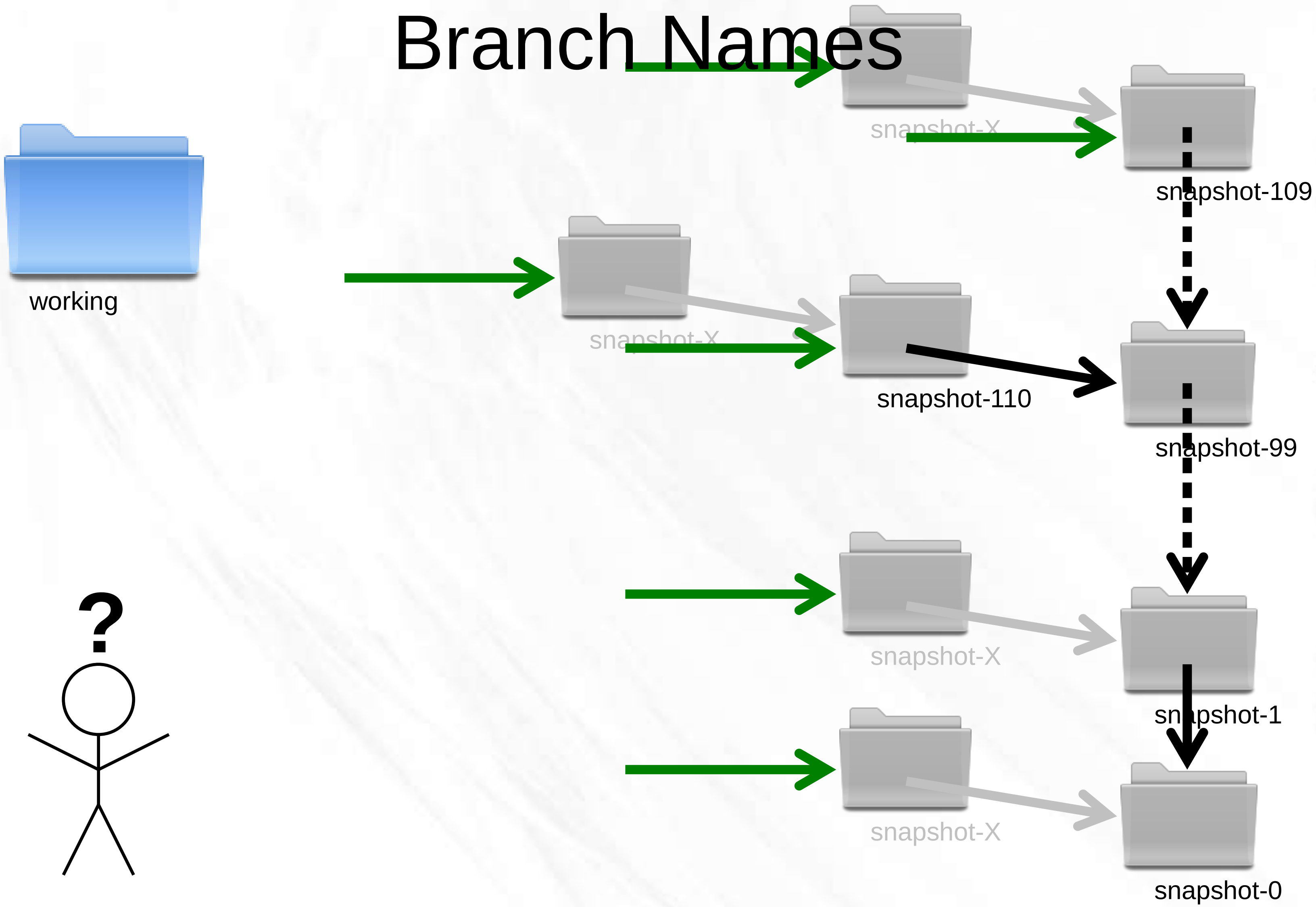
Branch Names



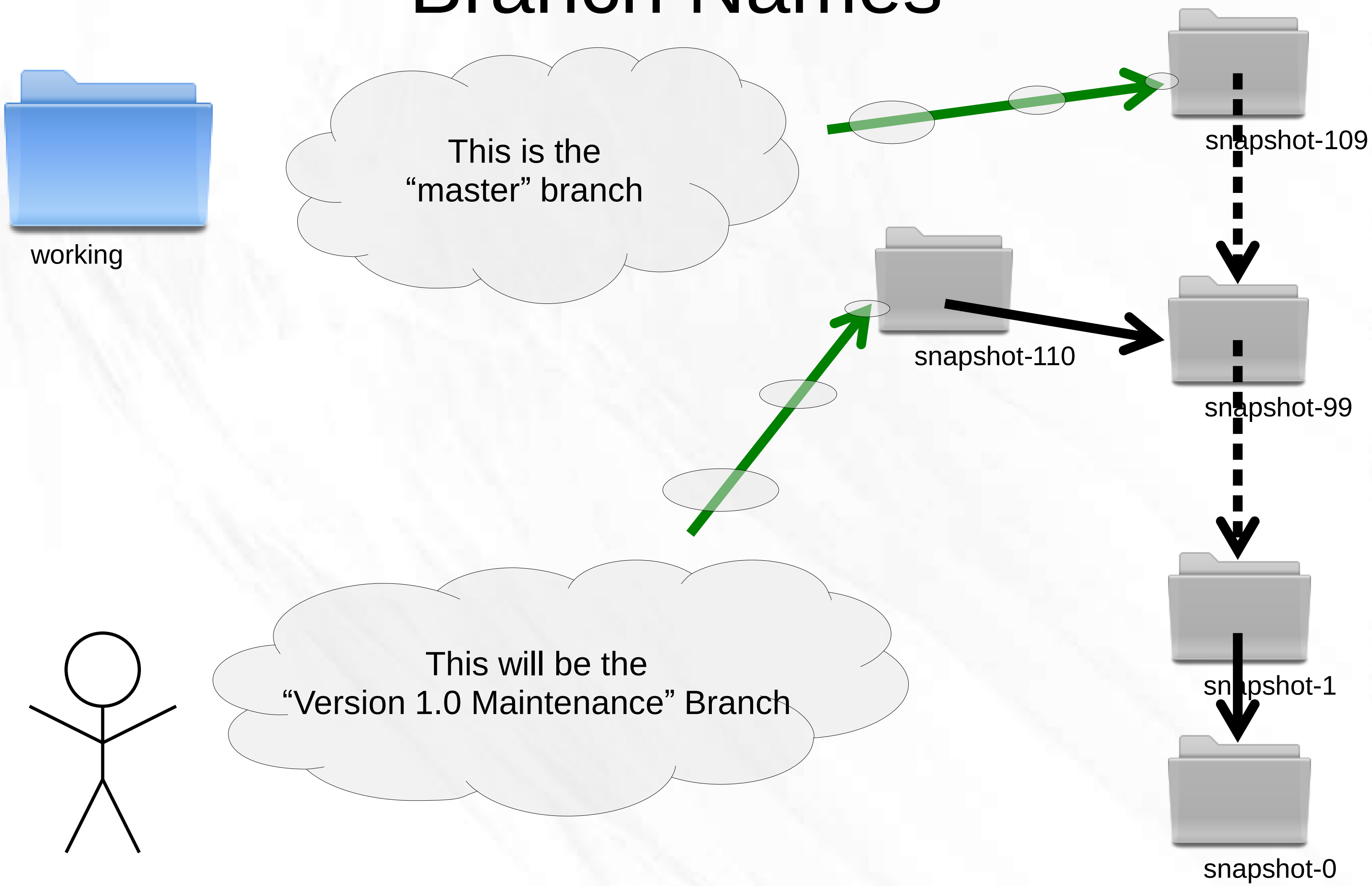
Branch Names



Branch Names



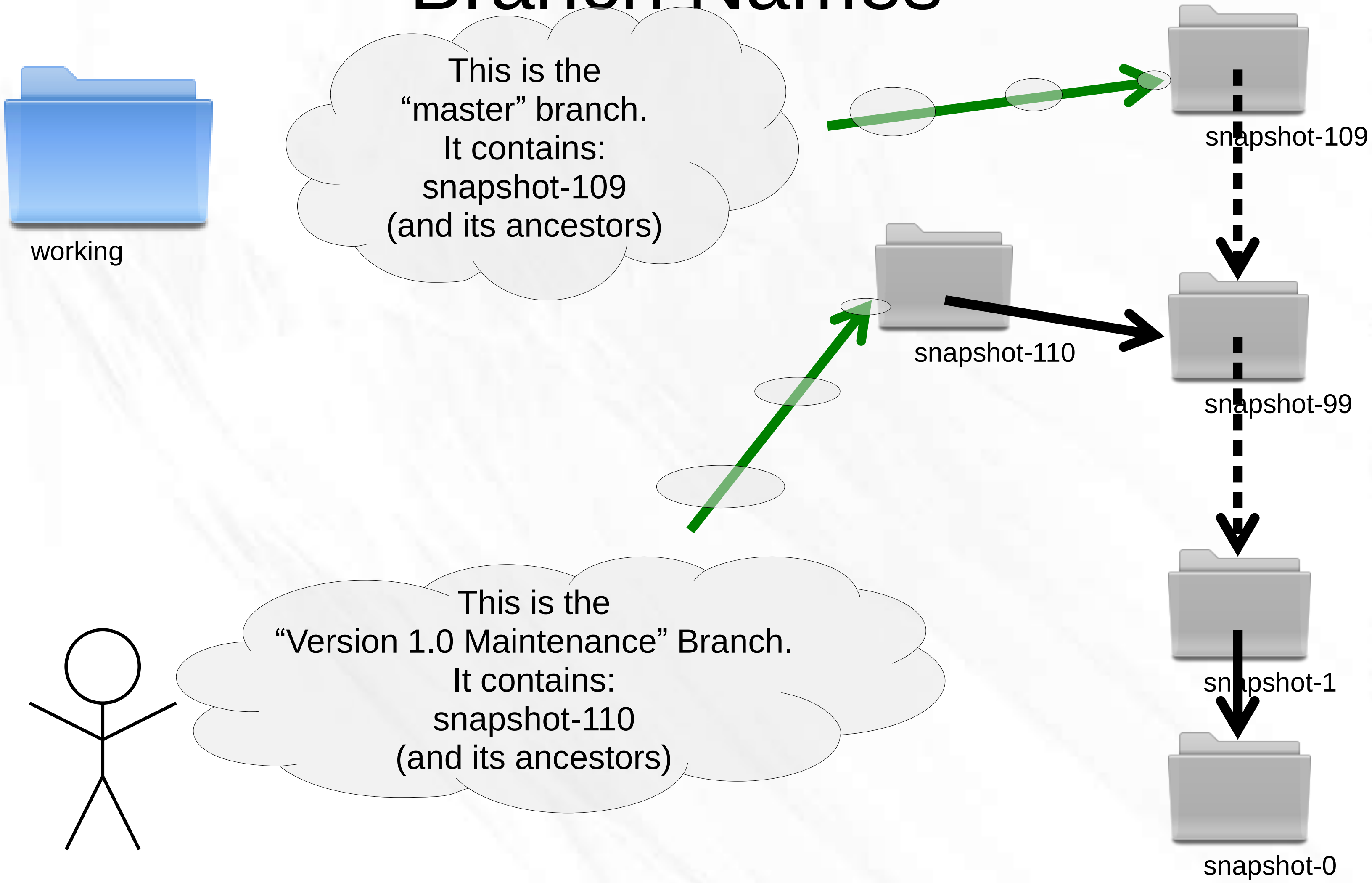
Branch Names



This is the
“master” branch

This will be the
“Version 1.0 Maintenance” Branch

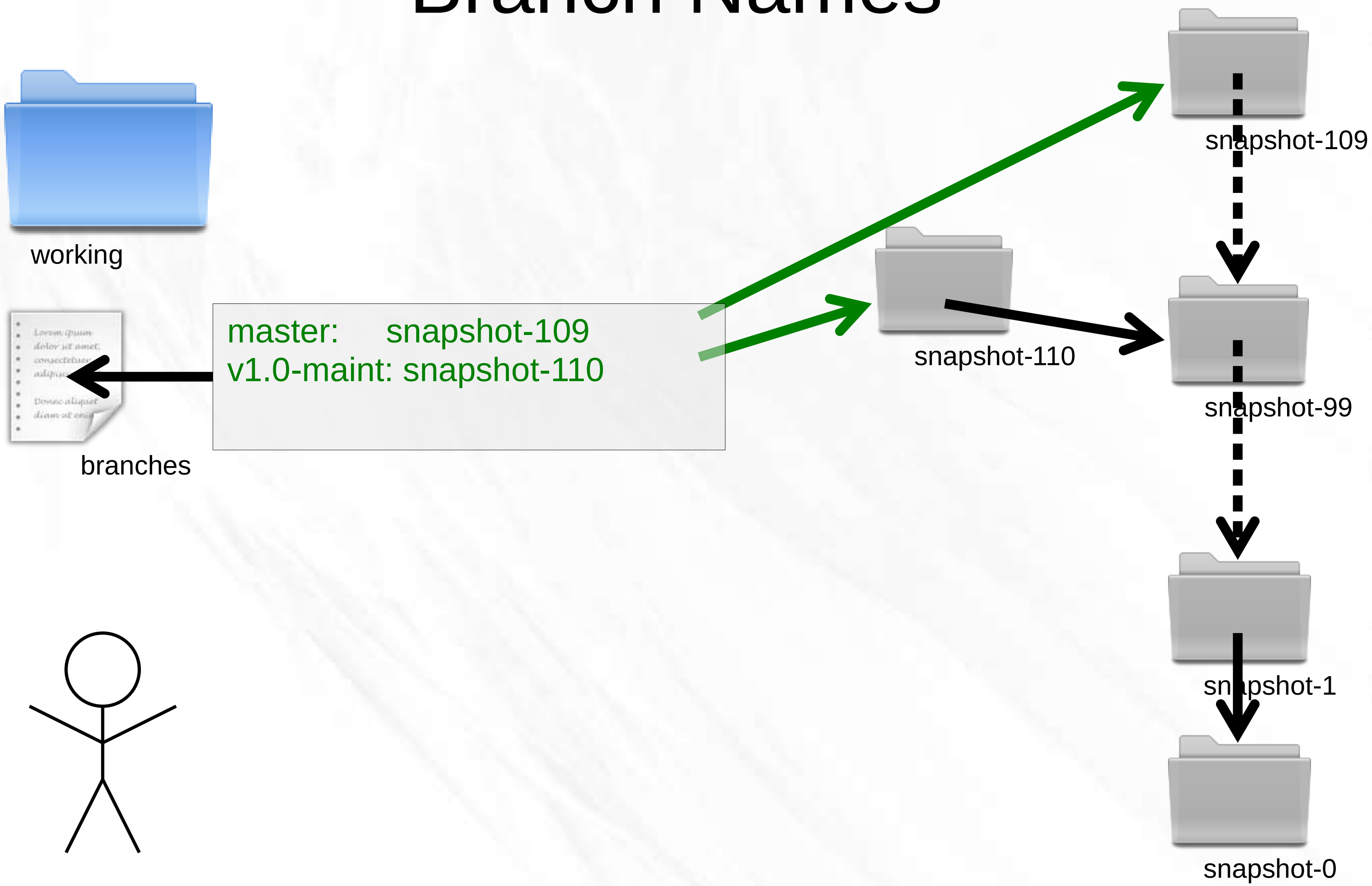
Branch Names



This is the
“master” branch.
It contains:
snapshot-109
(and its ancestors)

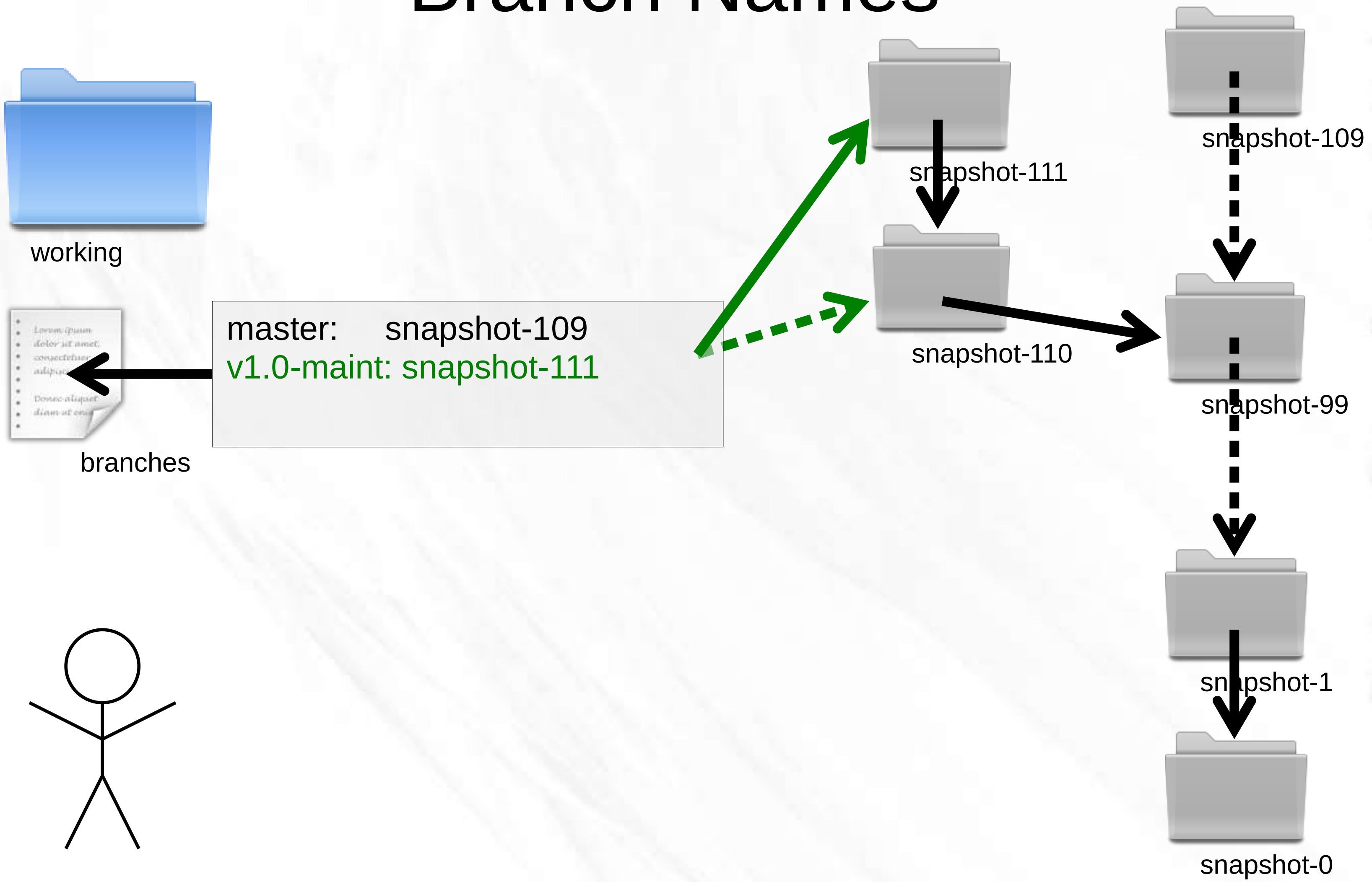
This is the
“Version 1.0 Maintenance” Branch.
It contains:
snapshot-110
(and its ancestors)

Branch Names



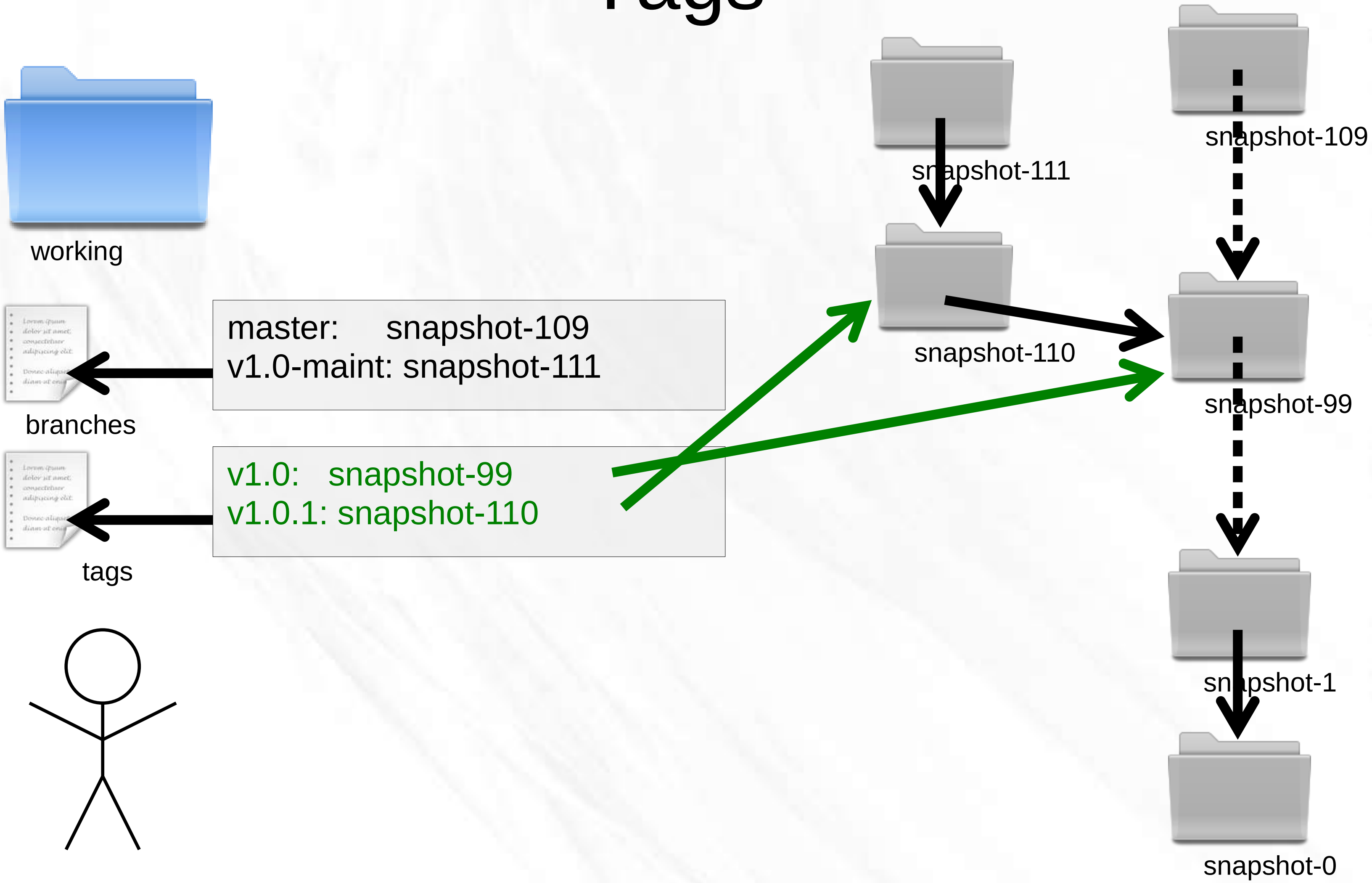
master: snapshot-109
v1.0-maint: snapshot-110

Branch Names



master: snapshot-109
v1.0-maint: snapshot-111

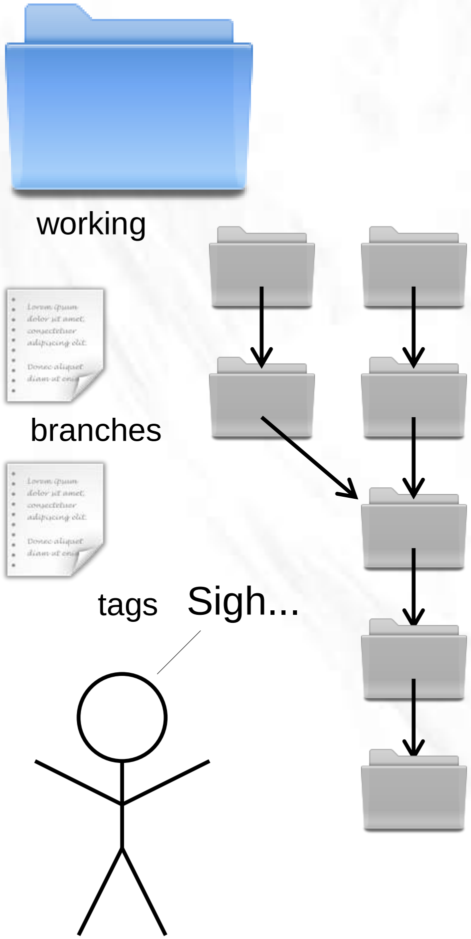
Tags



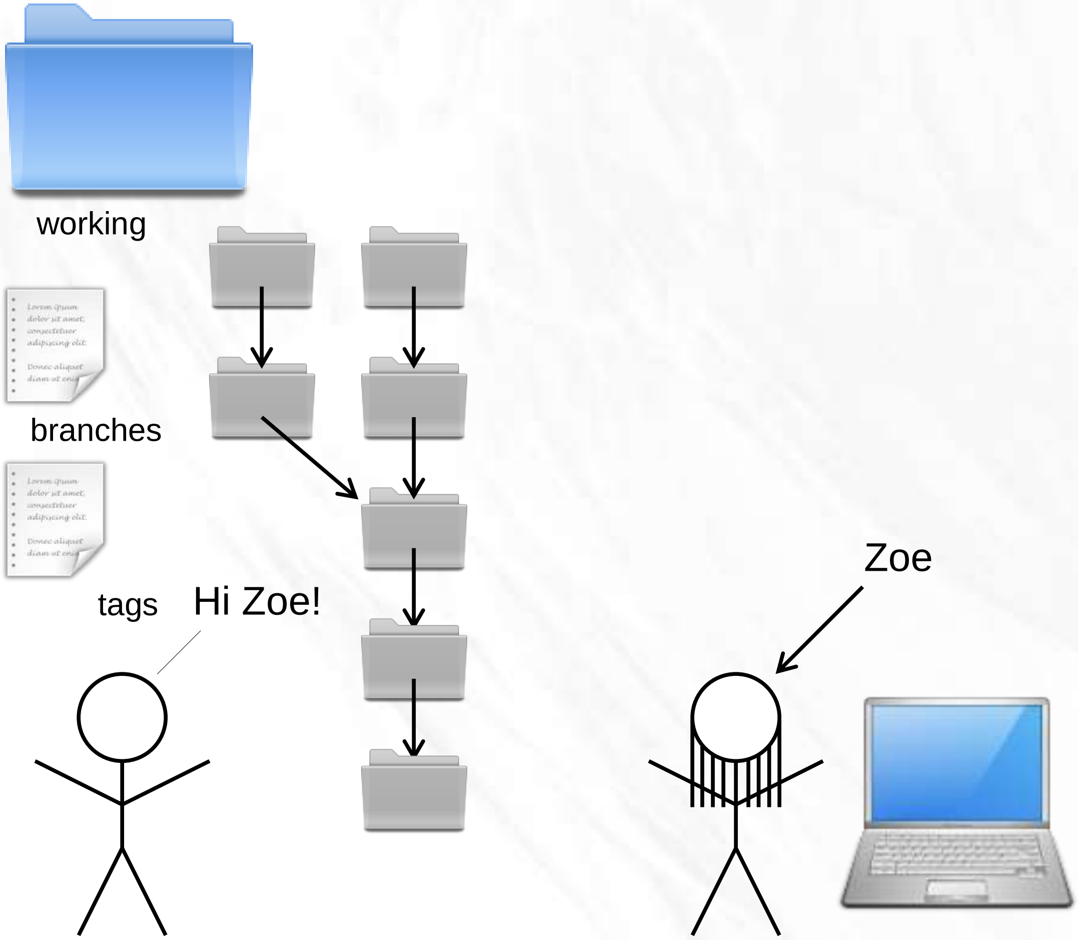
master: snapshot-109
v1.0-maint: snapshot-111

v1.0:   snapshot-99
v1.0.1: snapshot-110

Distributed



Distributed



working

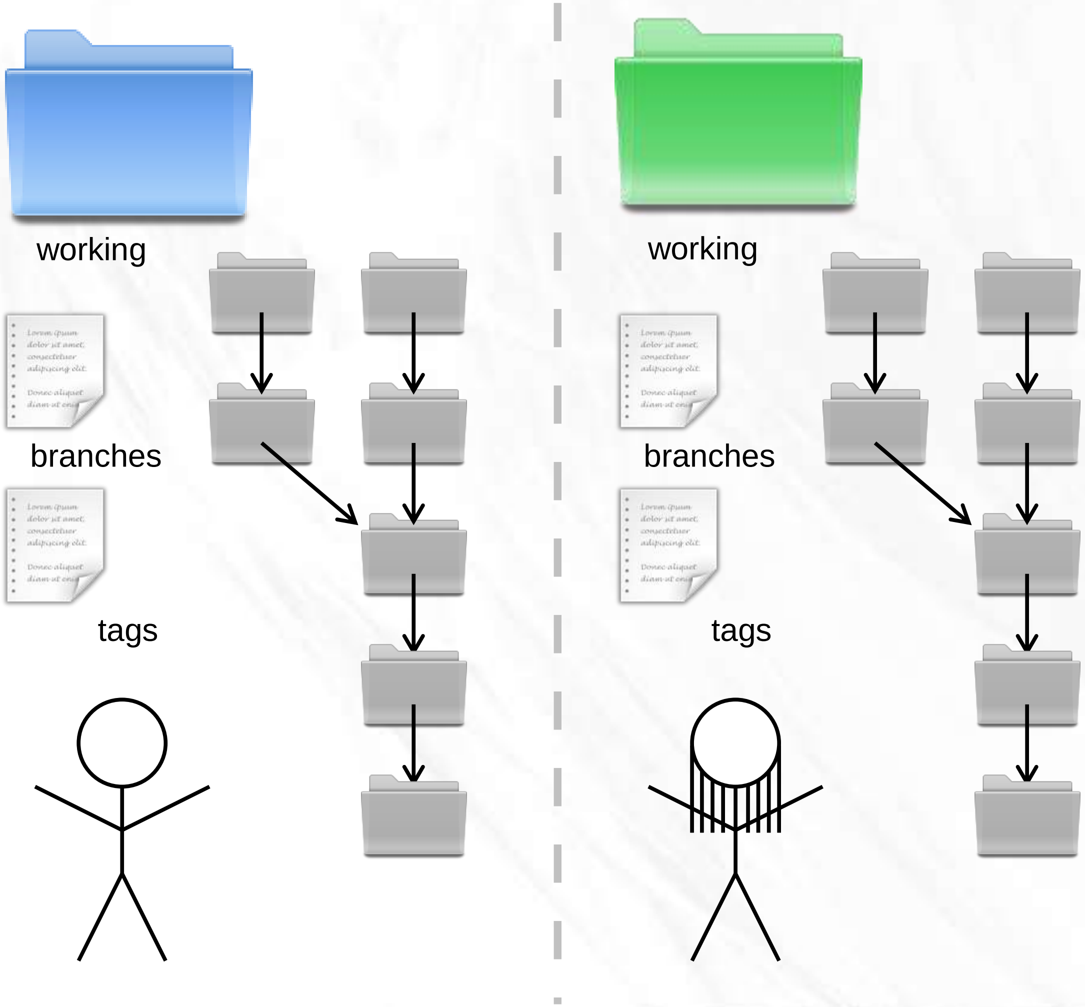
branches

tags

Hi Zoe!

Zoe

Distributed



working

branches

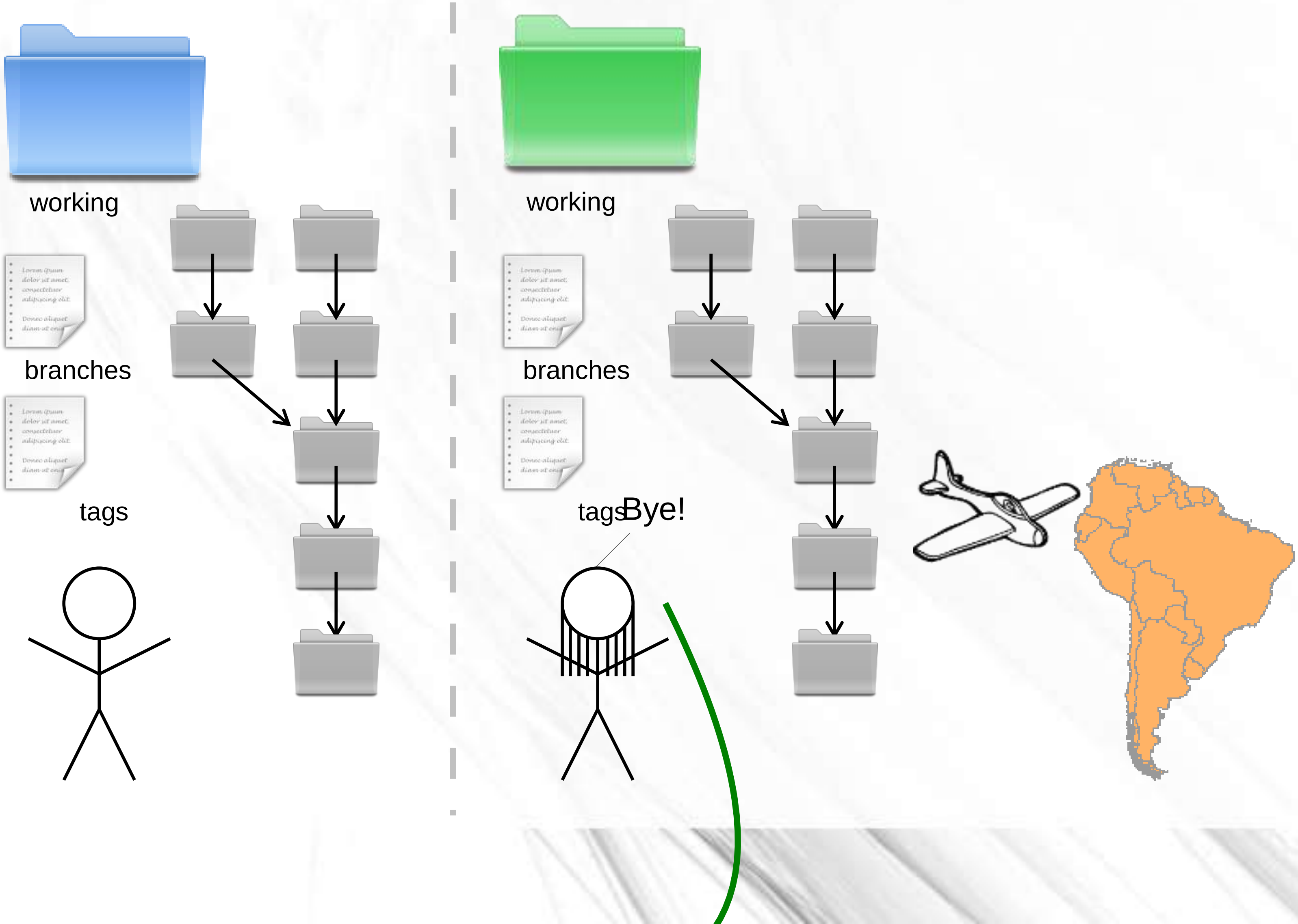
tags

working

branches

tags

Distributed



working

branches

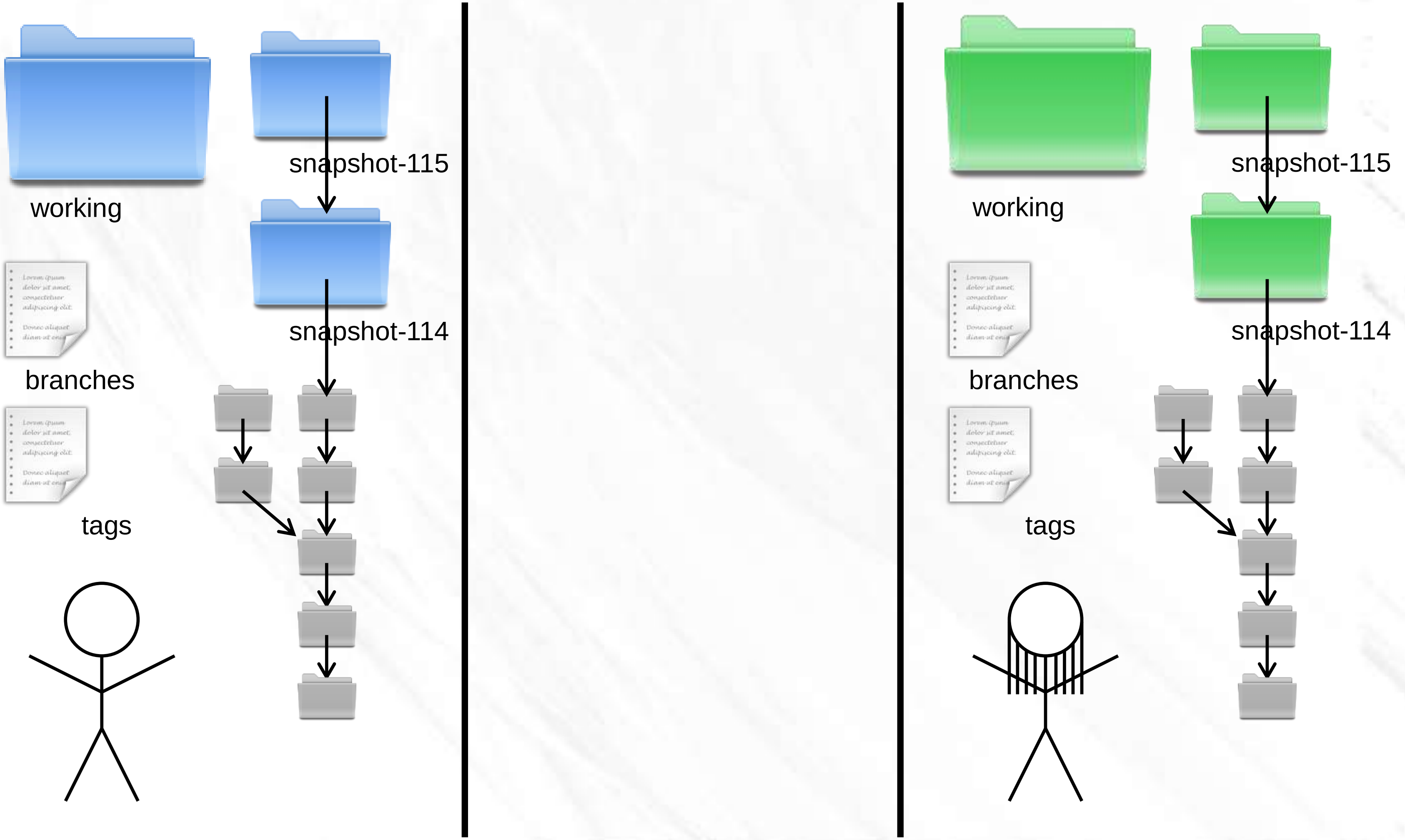
tags

working

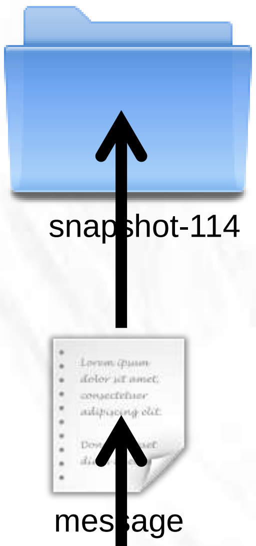
branches

tags

Distributed

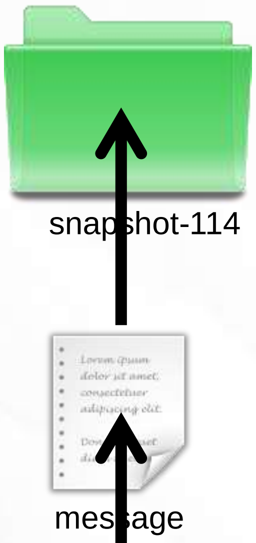


Distributed



2009-05-22 12:12:12
parent: snapshot-113
author: Me <me@me.me>

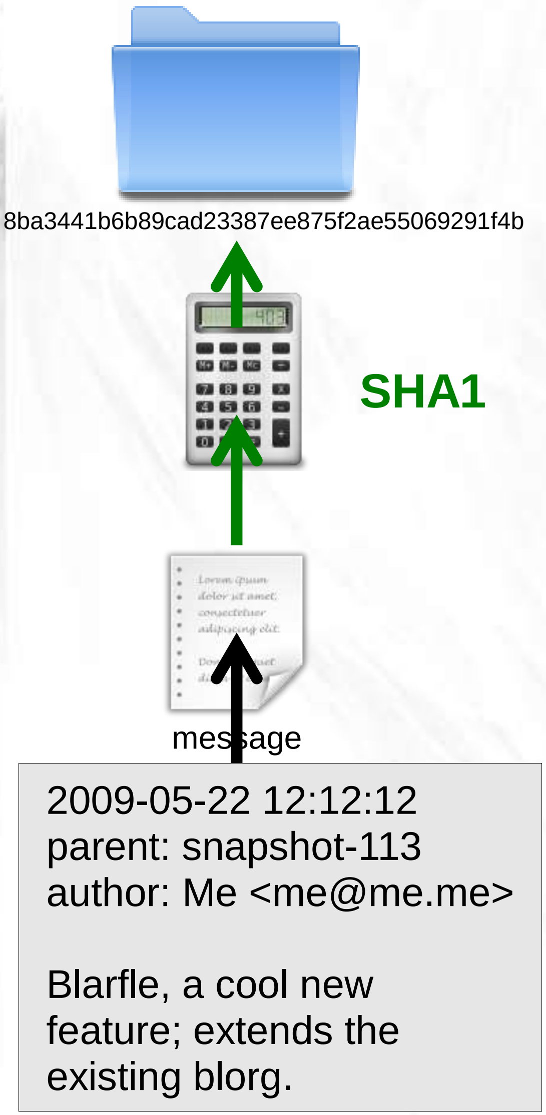
Blarfle, a cool new feature; extends the existing blorg.



2009-05-21 23:45:01
parent: snapshot-113
author: Zoe <zoe@z.oe>

Introduced a new foo, and reset the bar to xyzzy.

Distributed



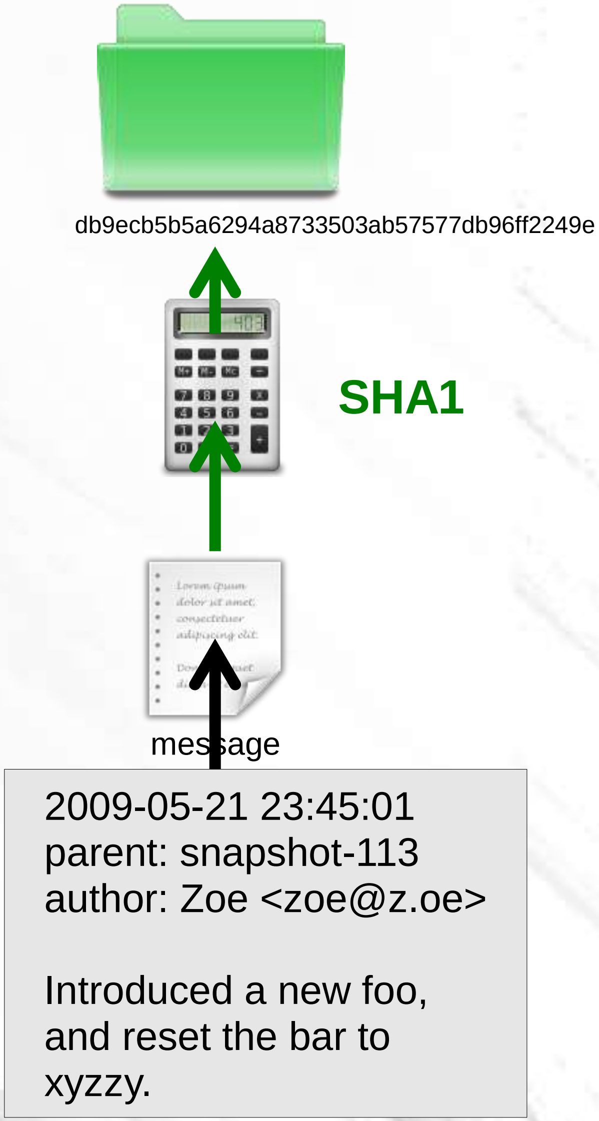
8ba3441b6b89cad23387ee875f2ae55069291f4b

SHA1

message

2009-05-22 12:12:12
parent: snapshot-113
author: Me <me@me.me>

Blarfle, a cool new feature; extends the existing blorg.



db9ecb5b5a6294a8733503ab57577db96ff2249e

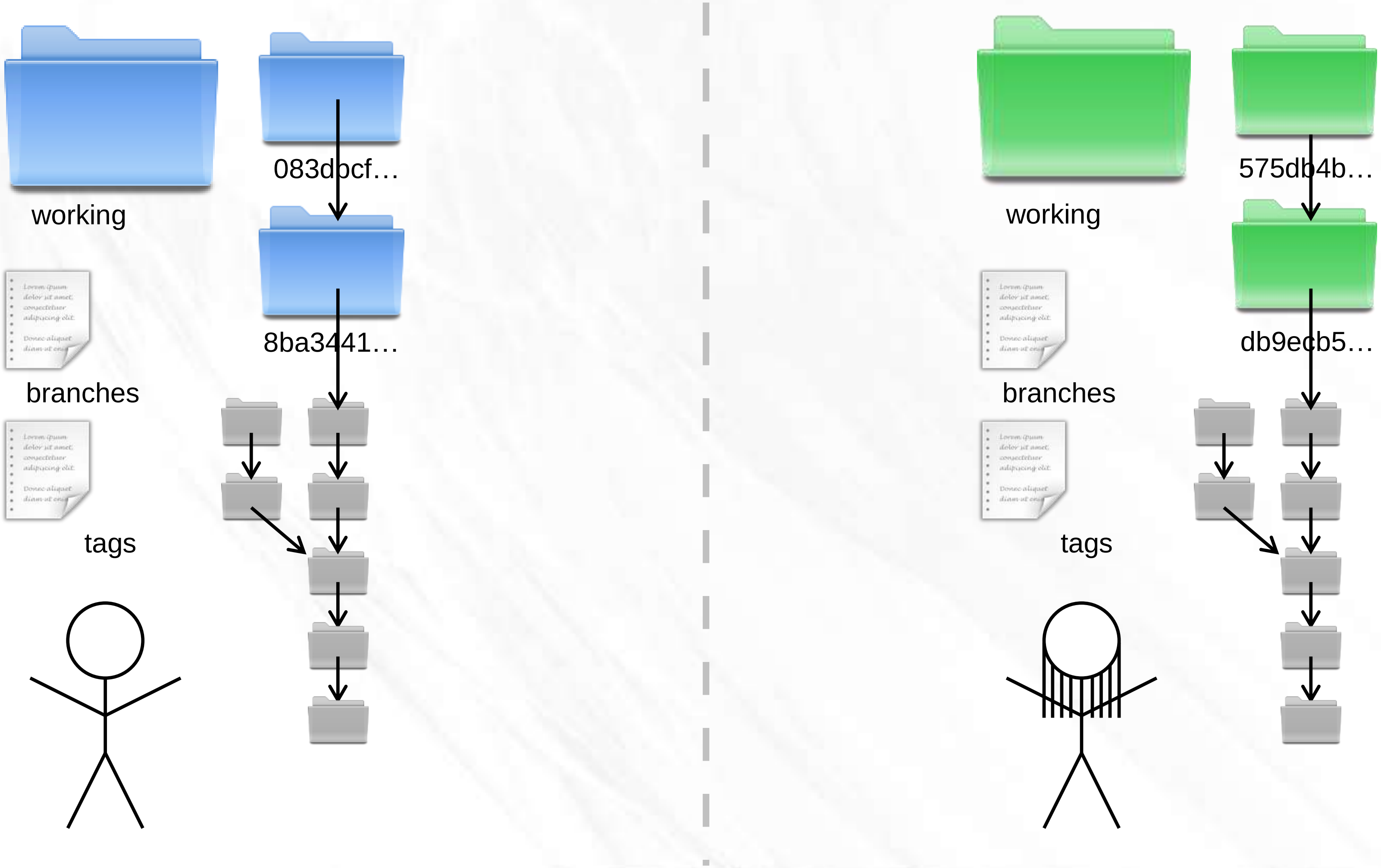
SHA1

message

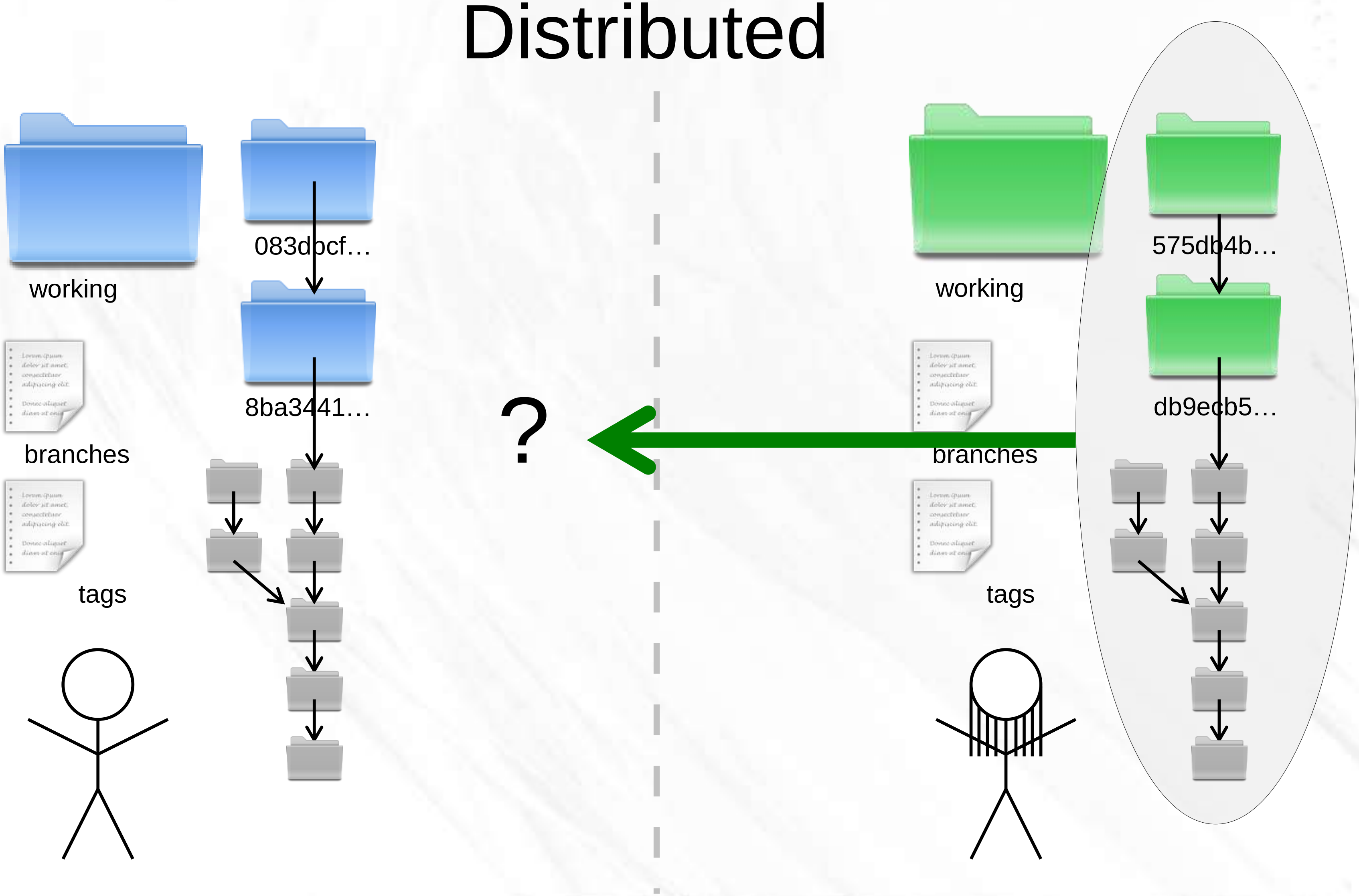
2009-05-21 23:45:01
parent: snapshot-113
author: Zoe <zoe@z.oe>

Introduced a new foo, and reset the bar to xyzzy.

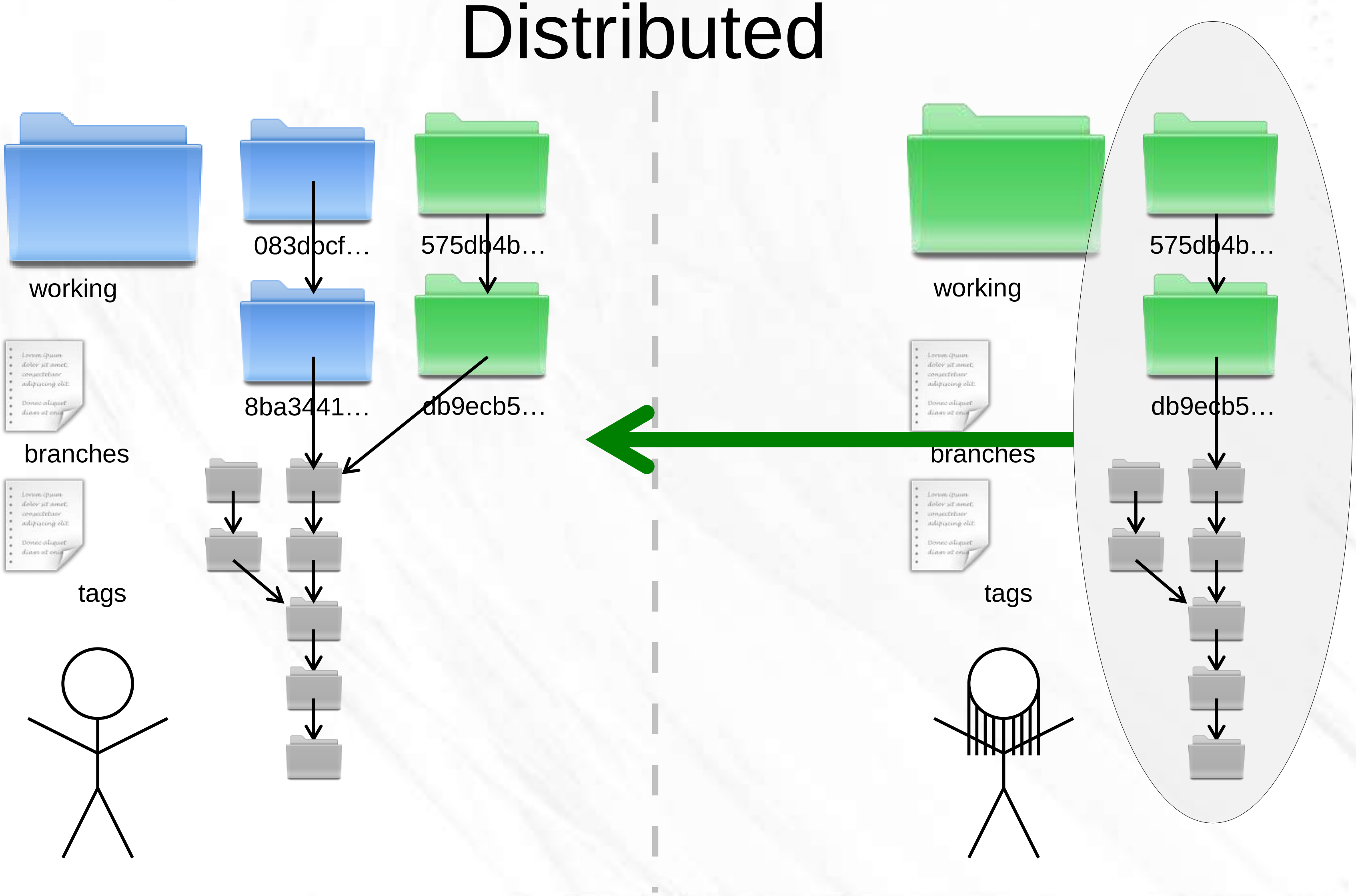
Distributed



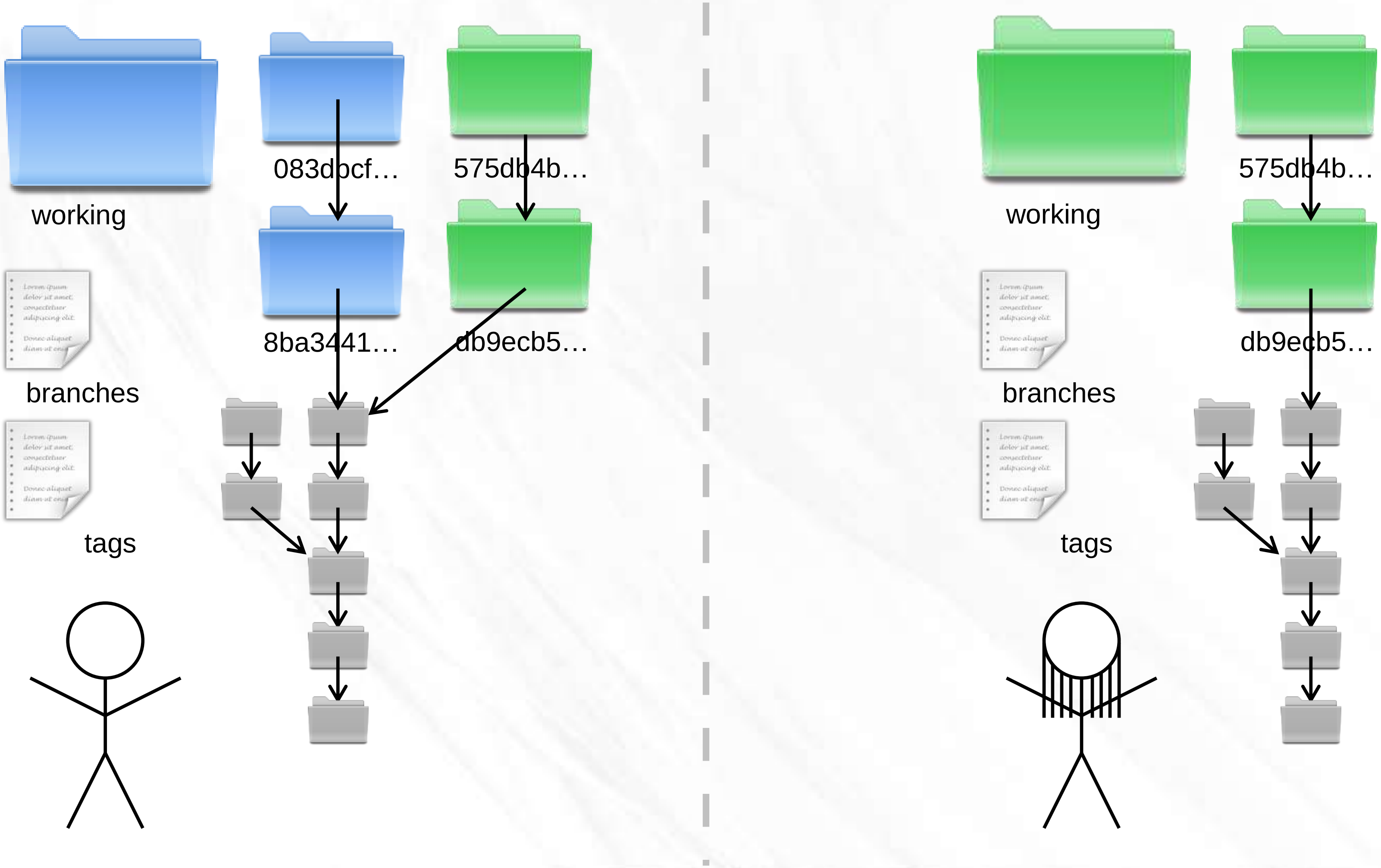
Distributed



Distributed



Distributed



working

083d0cf…

575db4b…

8ba3441…

db9ecb5…

branches

tags

working

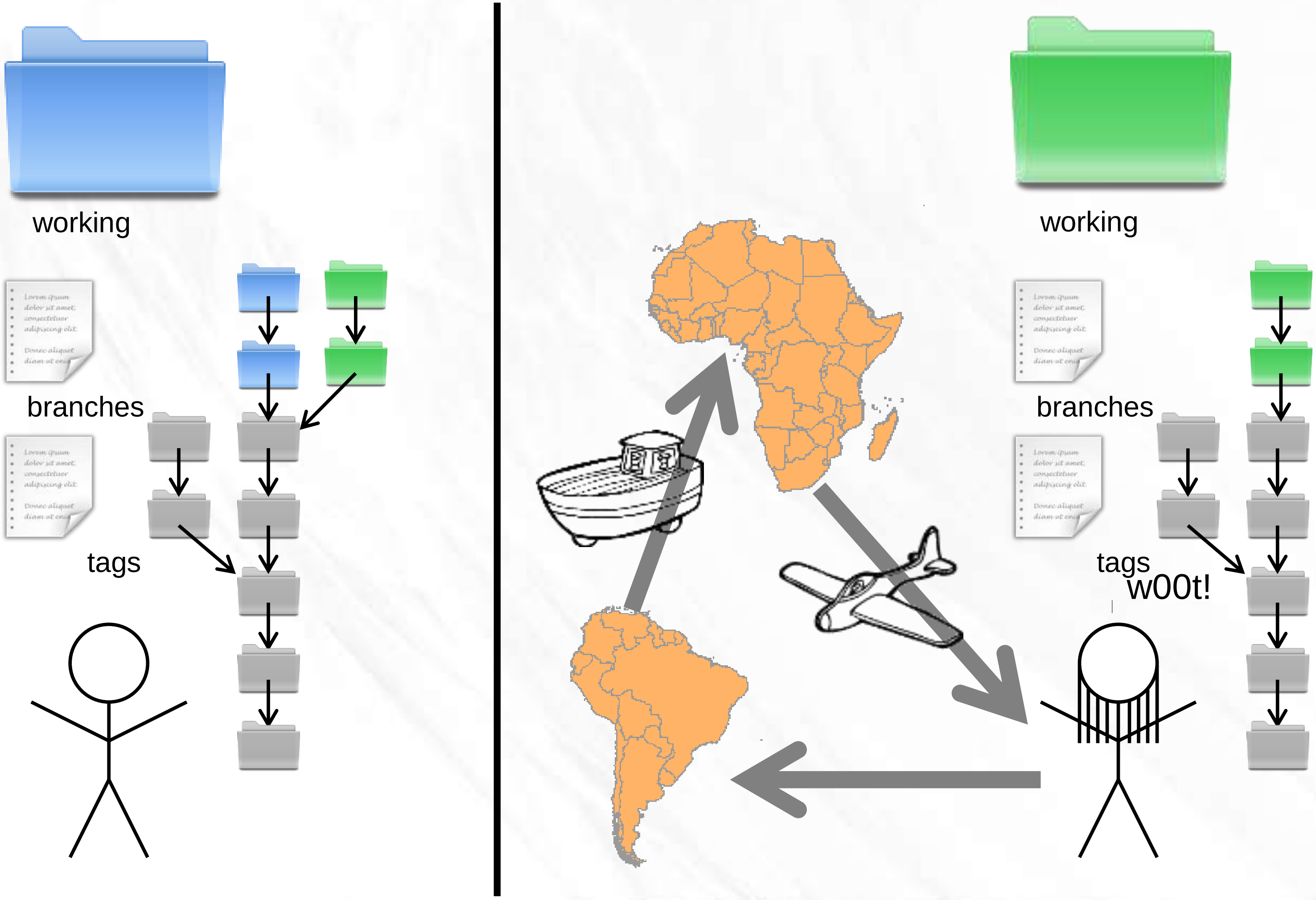
575db4b…

db9ecb5…

branches

tags

Offline



working

branches

tags

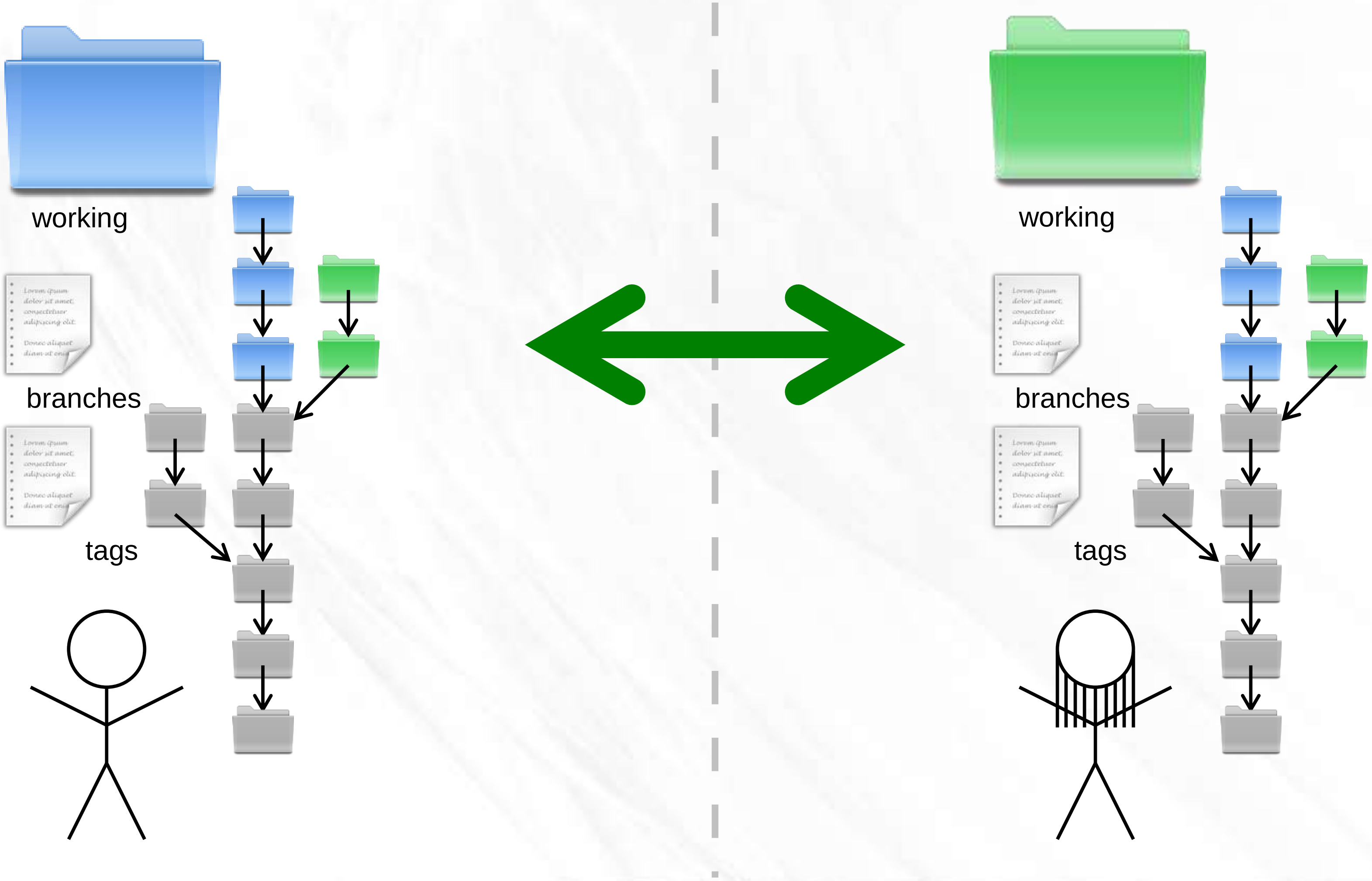
working

branches

tags

w00t!

Offline



working

branches

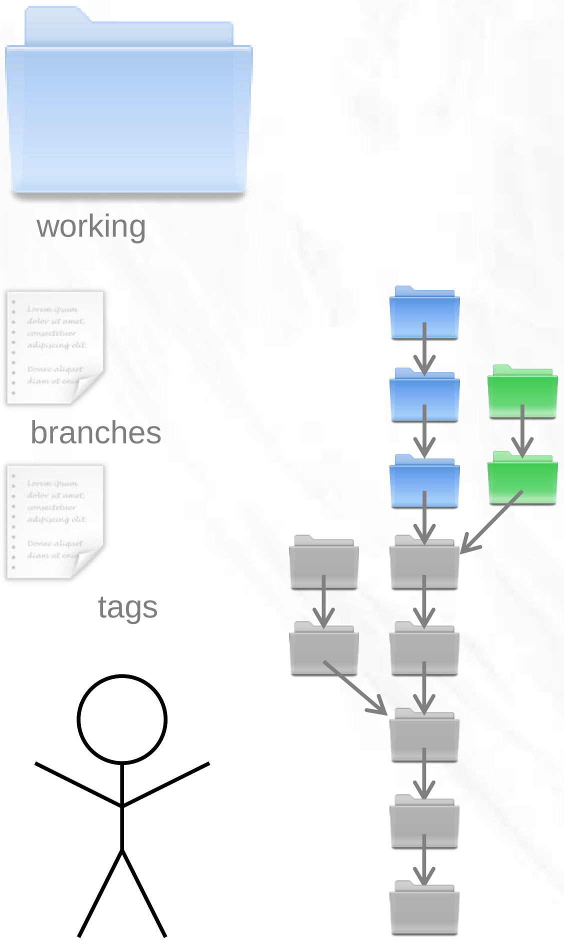
tags

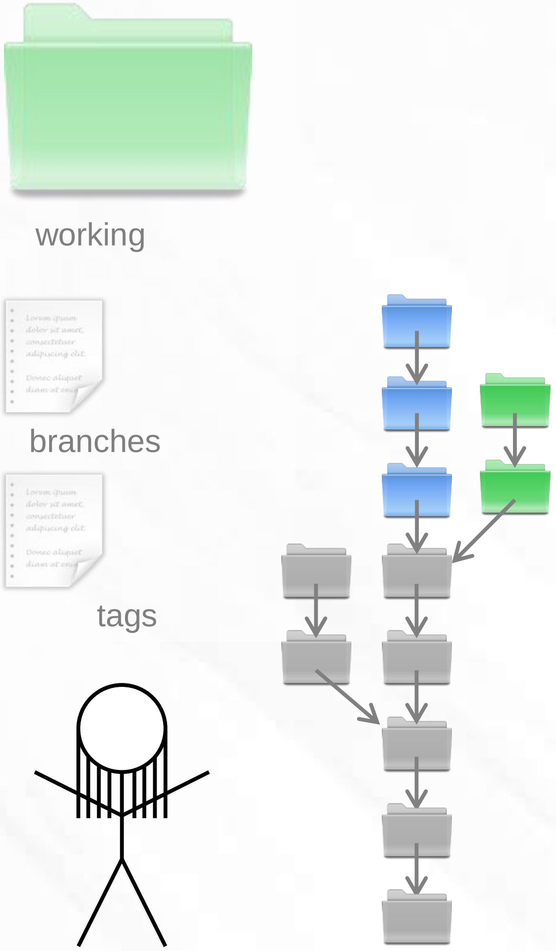
working

branches

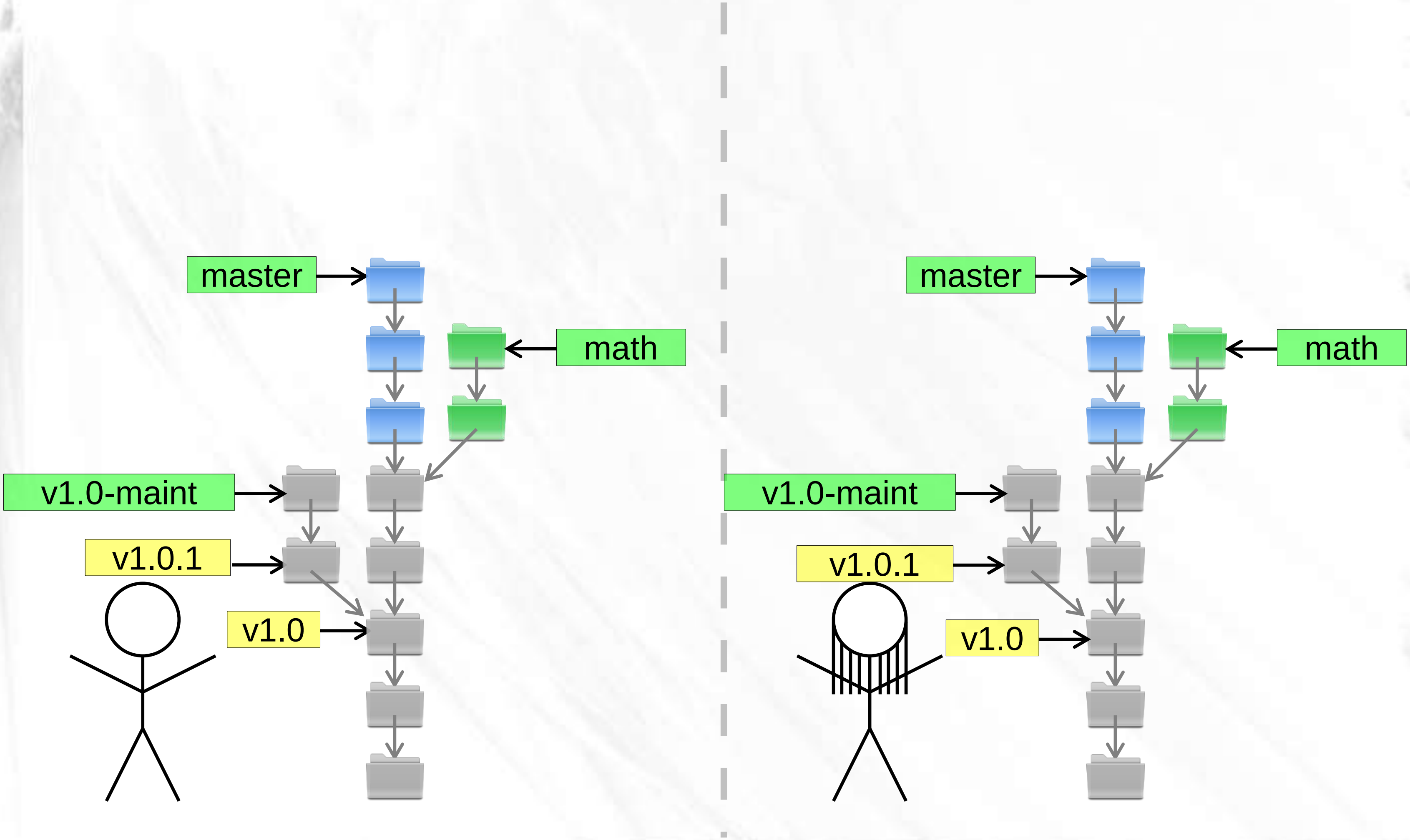
tags

(simpler drawings)





(simpler drawings)



master

math

v1.0-maint

v1.0.1

v1.0

master

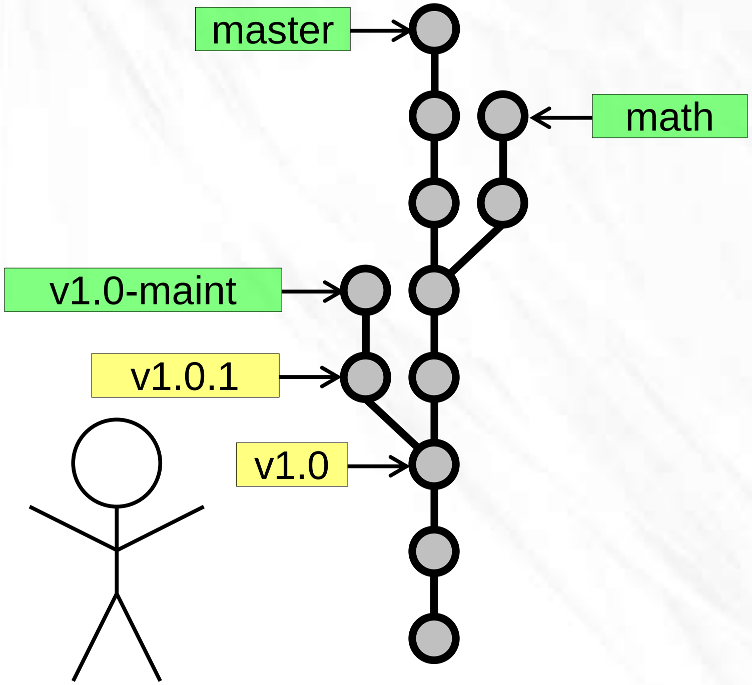
math

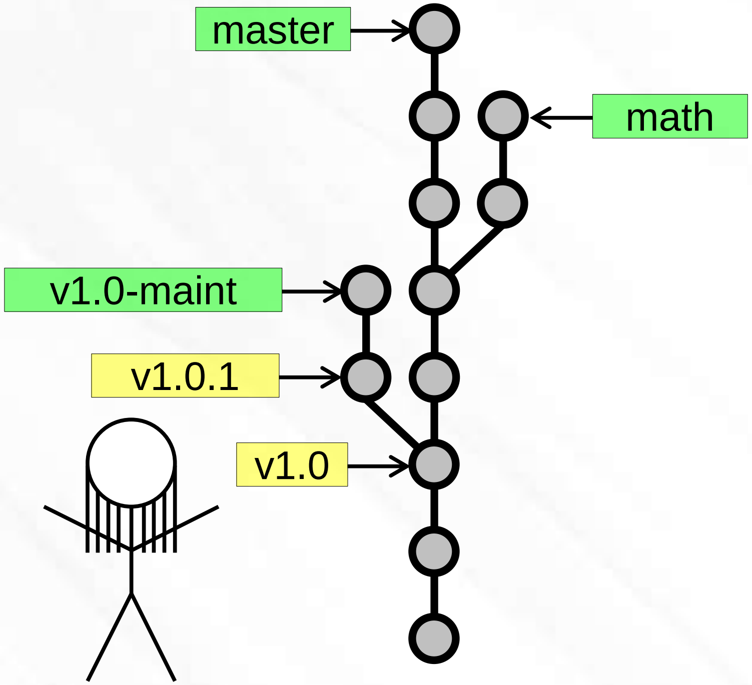
v1.0-maint

v1.0.1

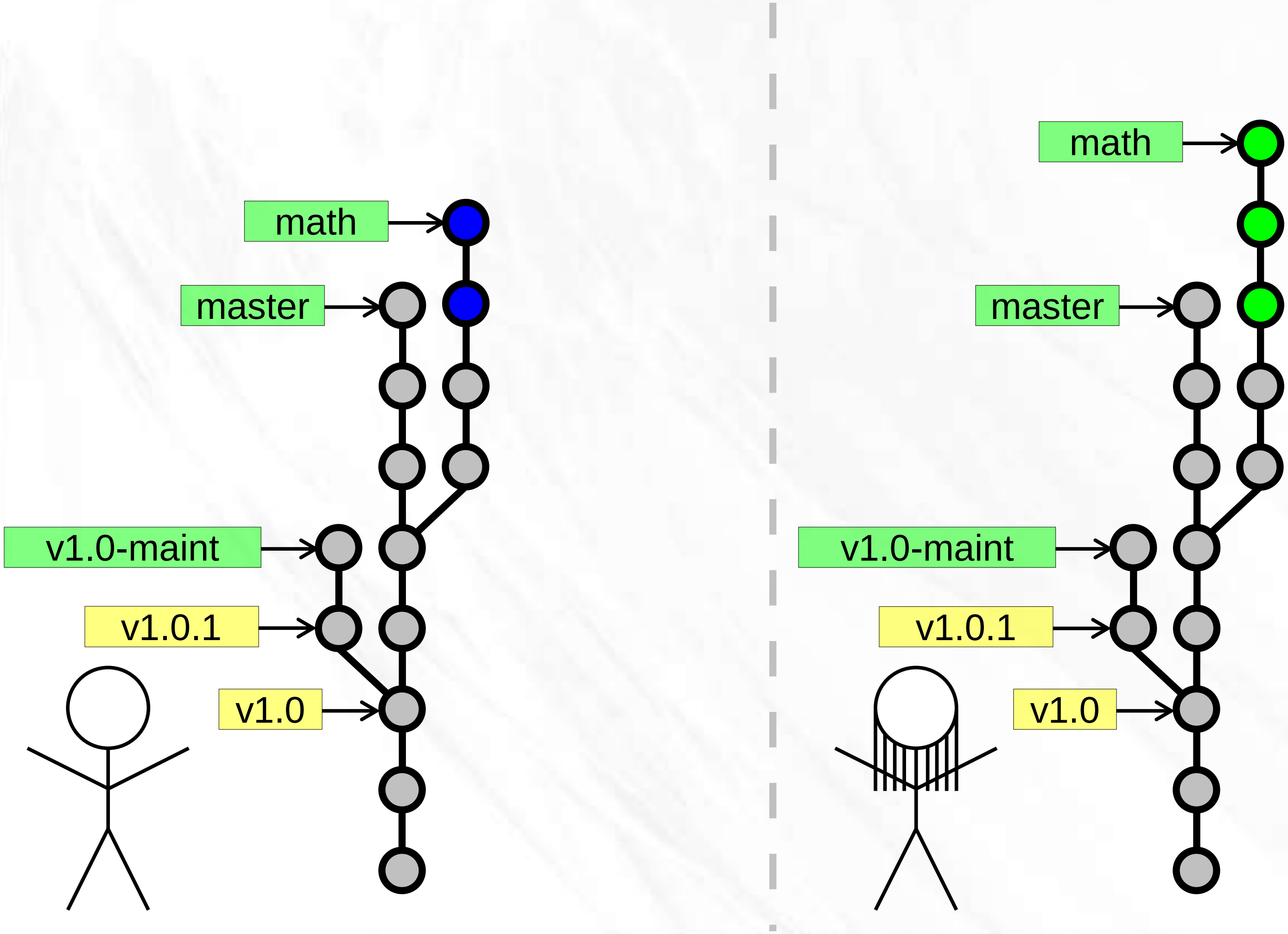
v1.0

(simpler drawings)

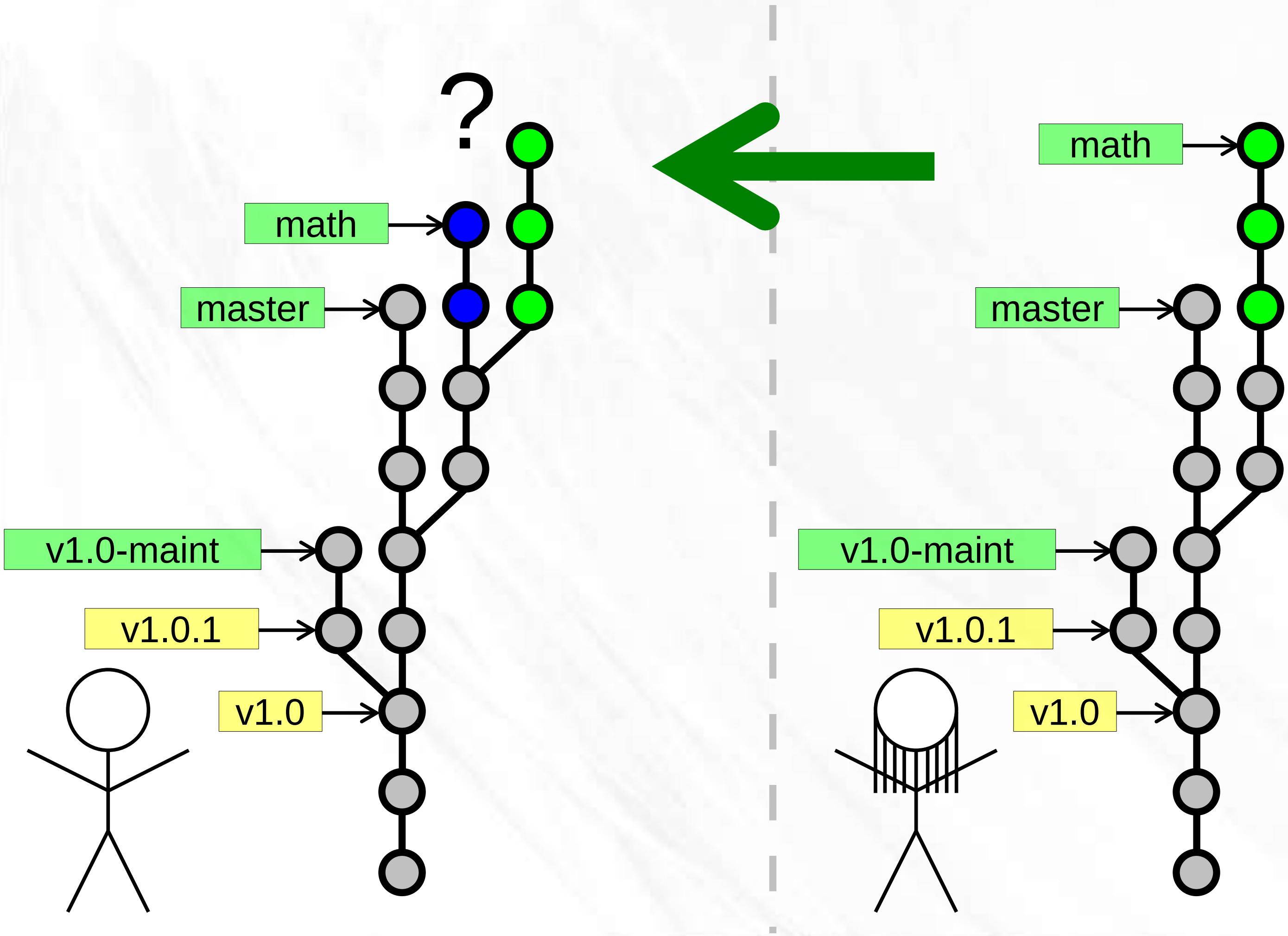




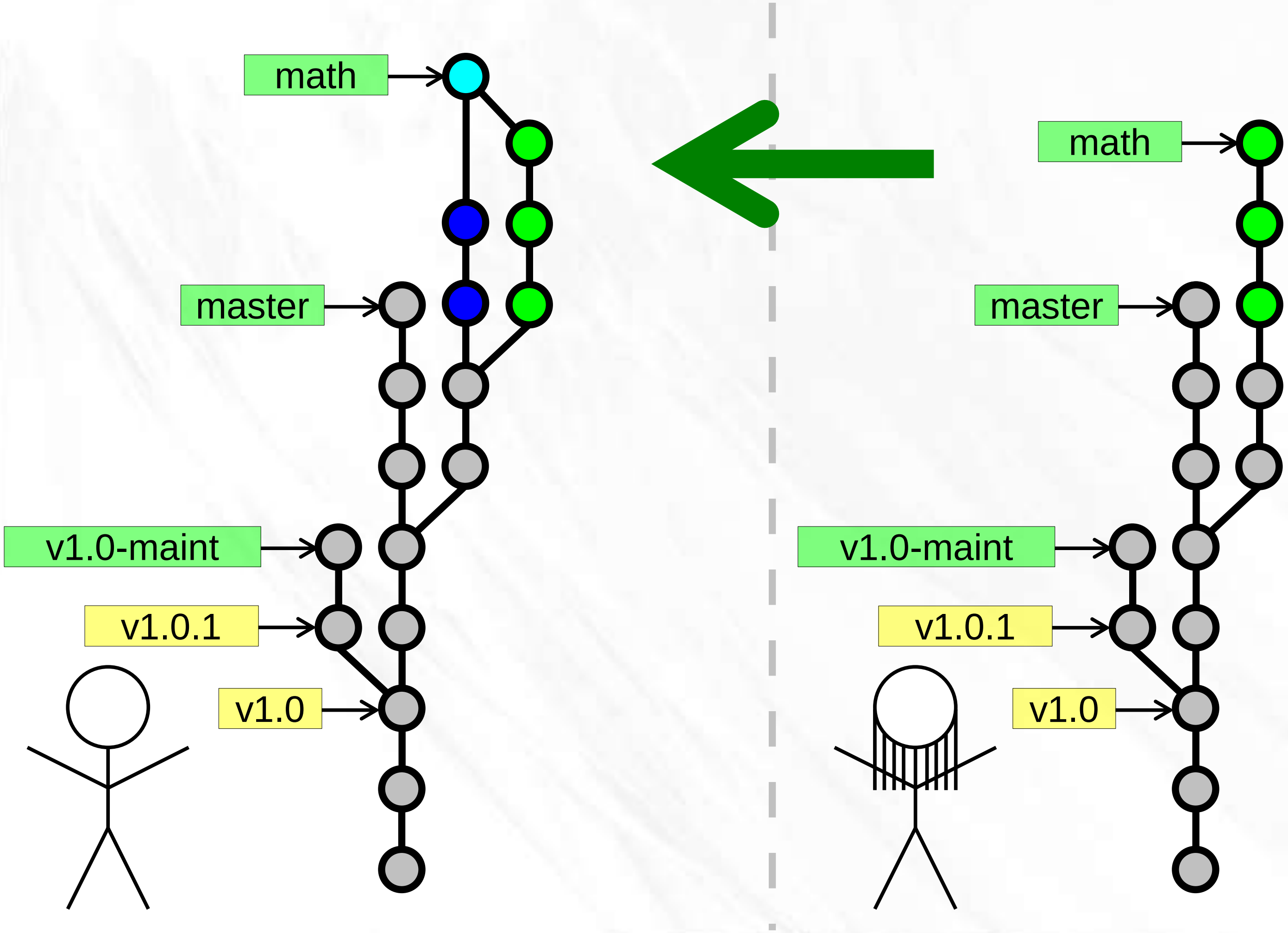
Merges


math
master
v1.0-maint
v1.0.1
v1.0
math
master
v1.0-maint
v1.0.1
v1.0

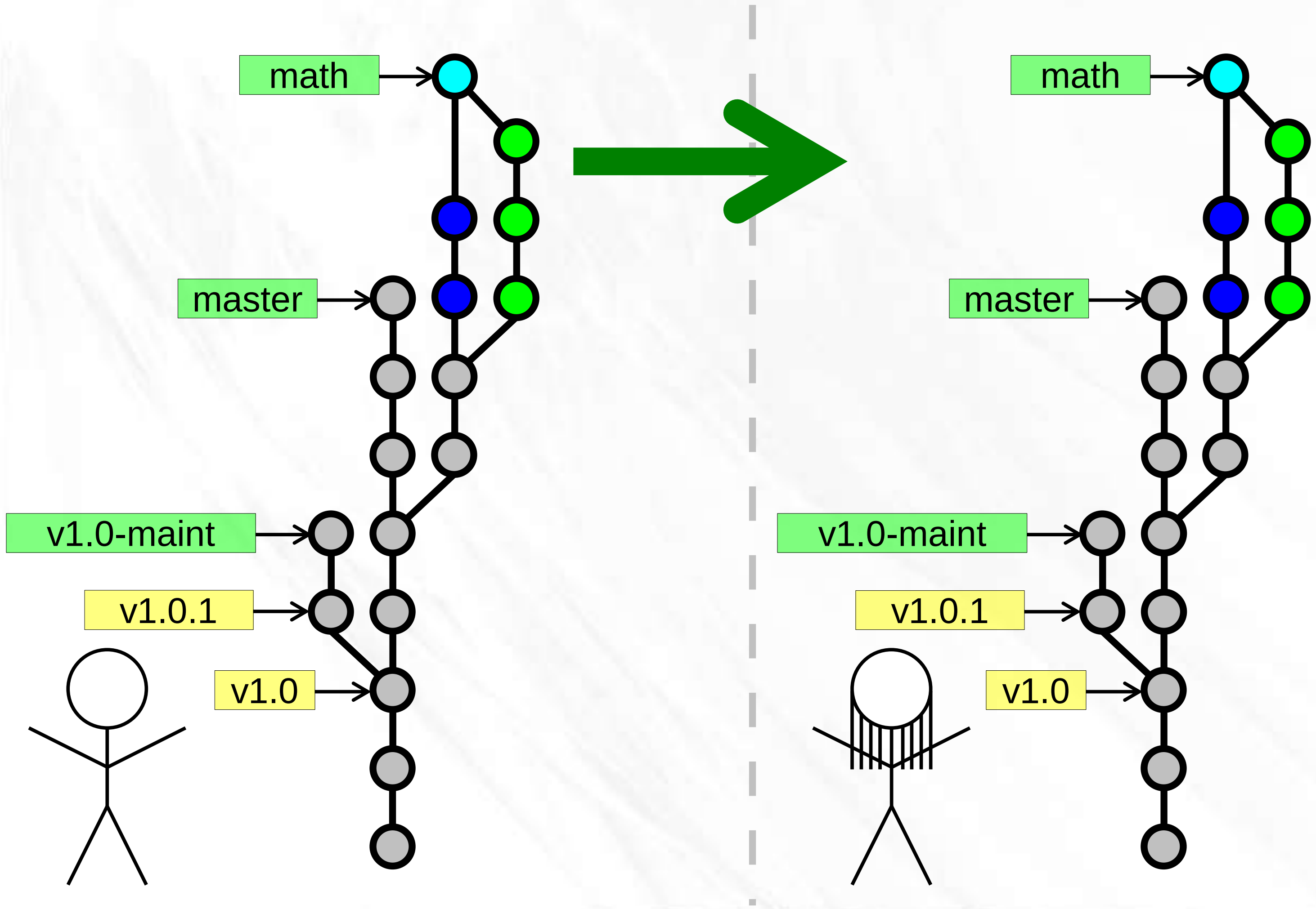
Merges


?
math
master
v1.0-maint
v1.0.1
v1.0
math
master
v1.0-maint
v1.0.1
v1.0

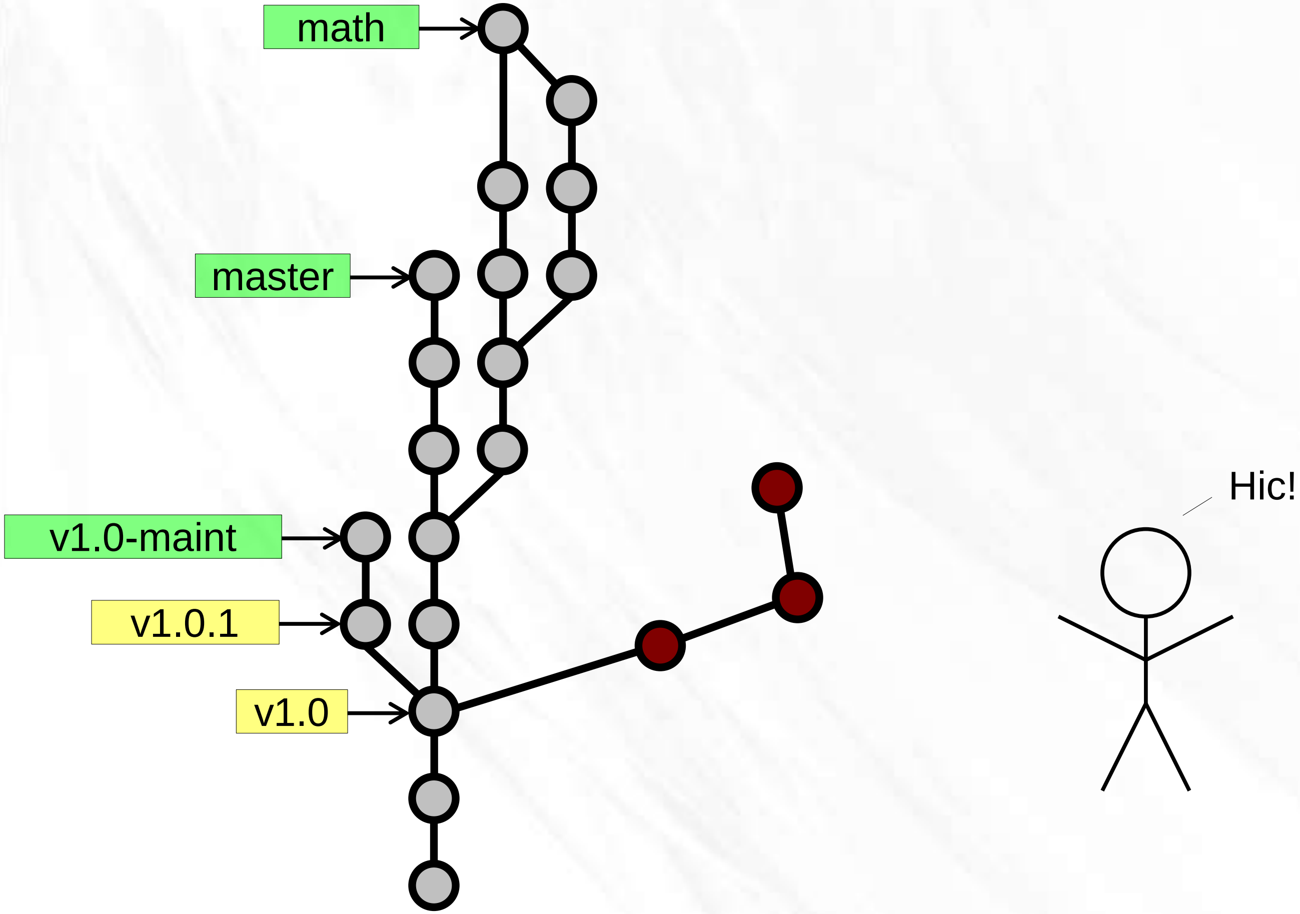
Merges


math
master
v1.0-maint
v1.0.1
v1.0
math
master
v1.0-maint
v1.0.1
v1.0

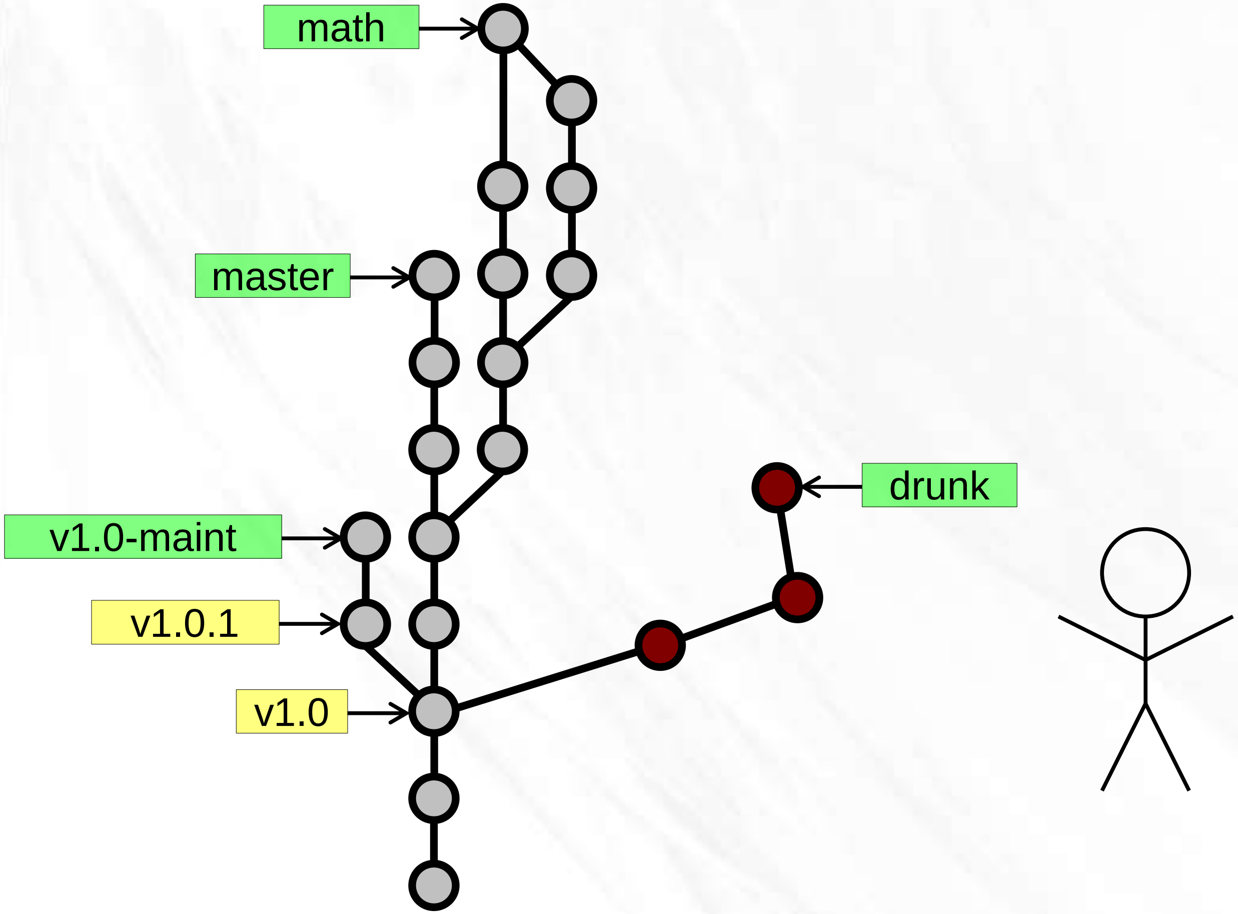
Merges


math
master
v1.0-maint
v1.0.1
v1.0
math
master
v1.0-maint
v1.0.1
v1.0

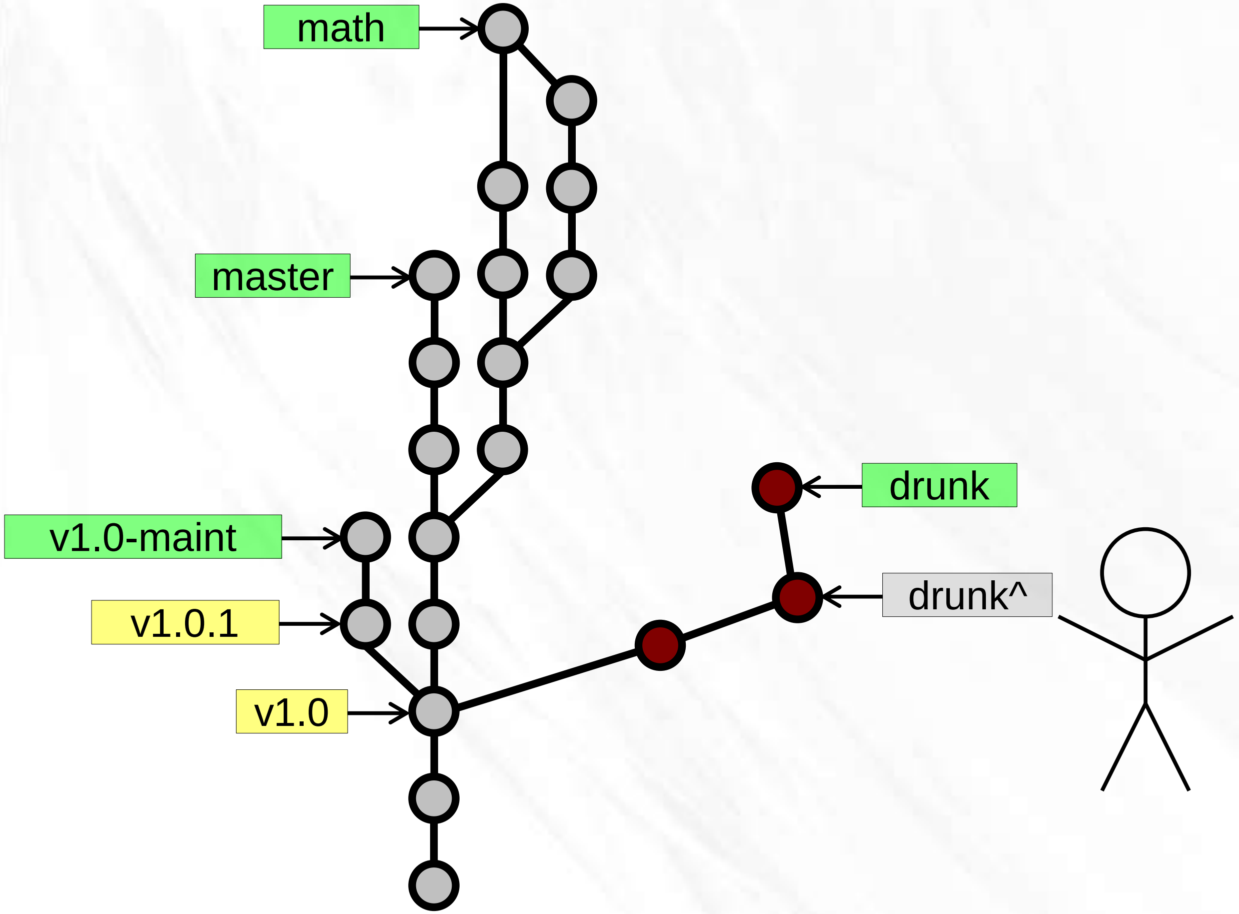
Rewriting History



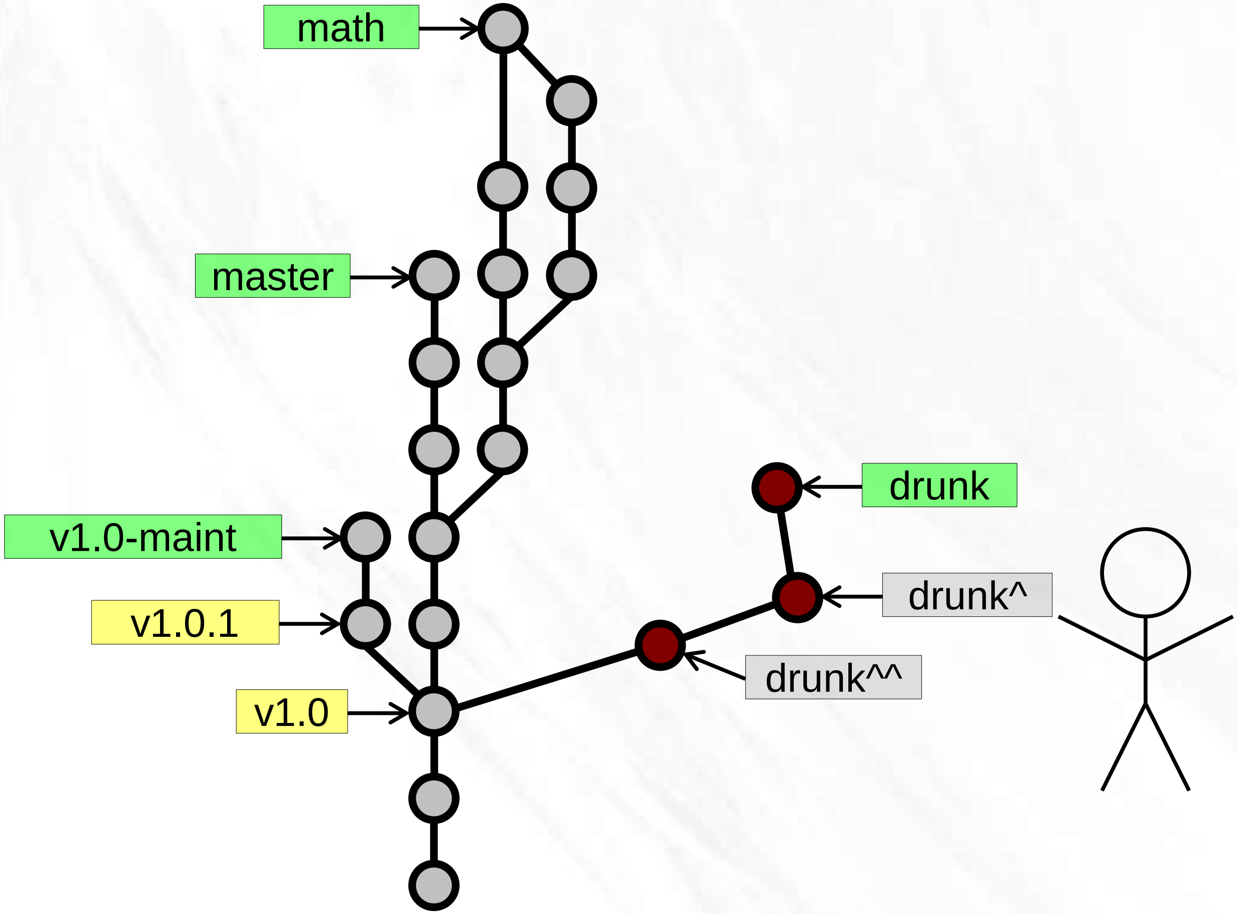
Rewriting History



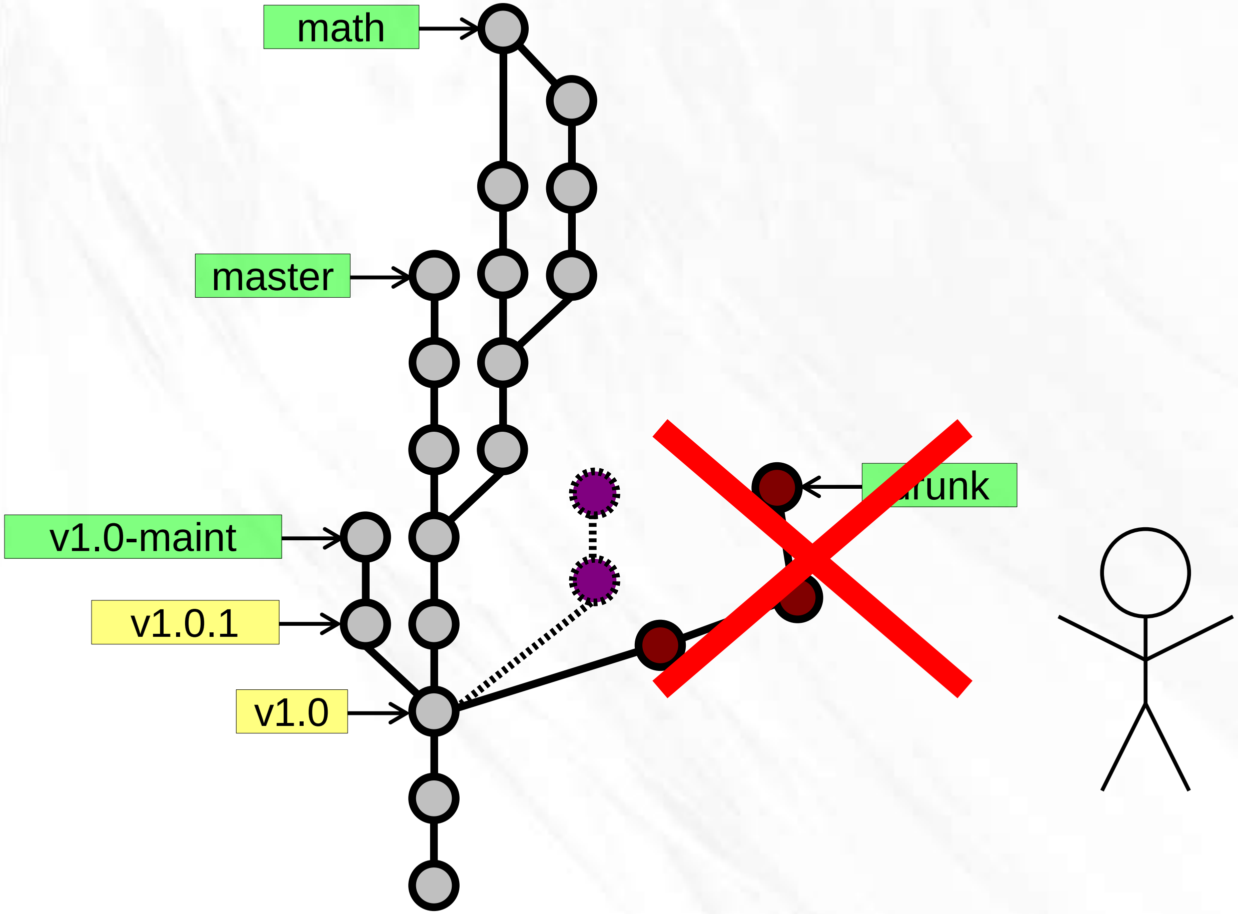
Rewriting History



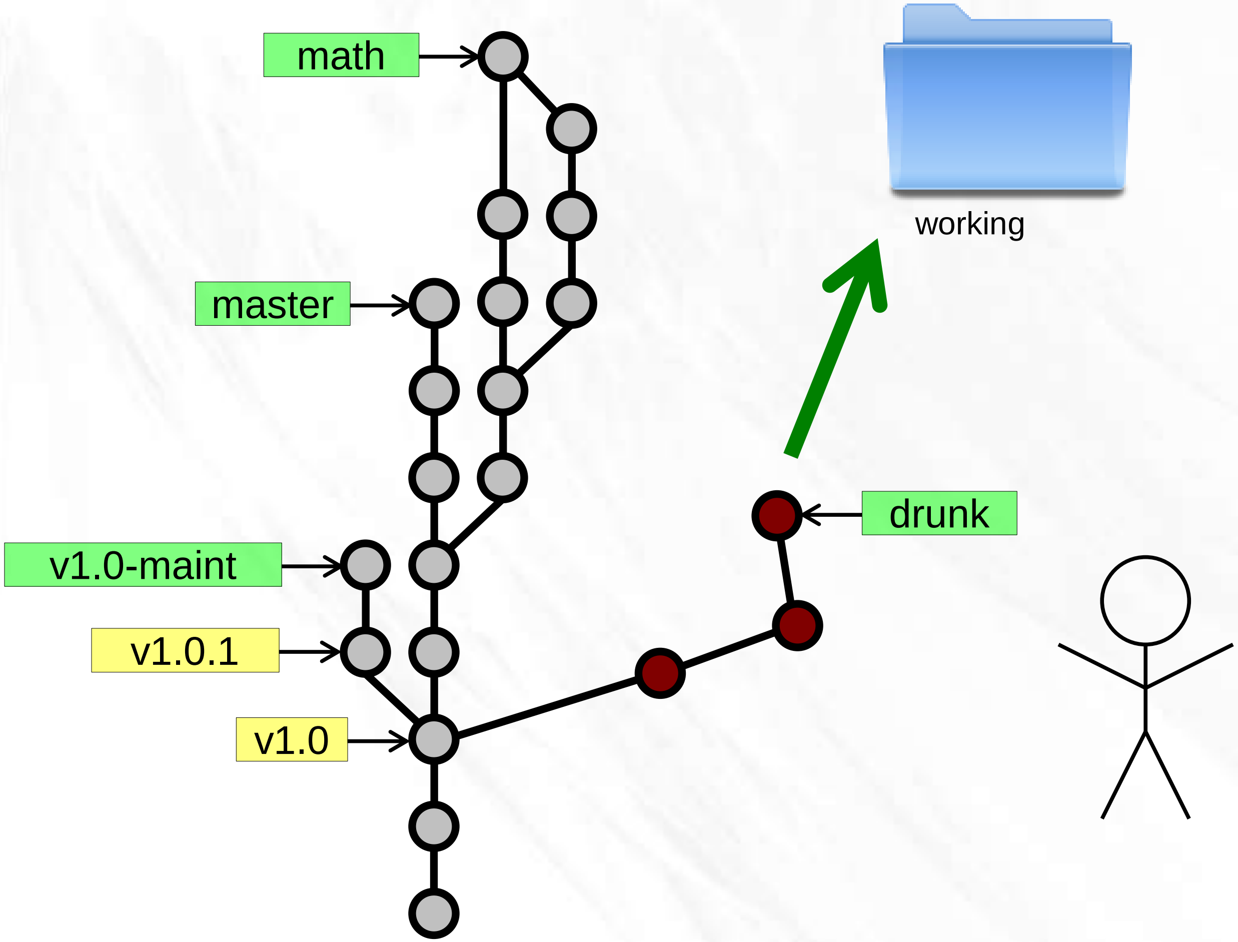
Rewriting History



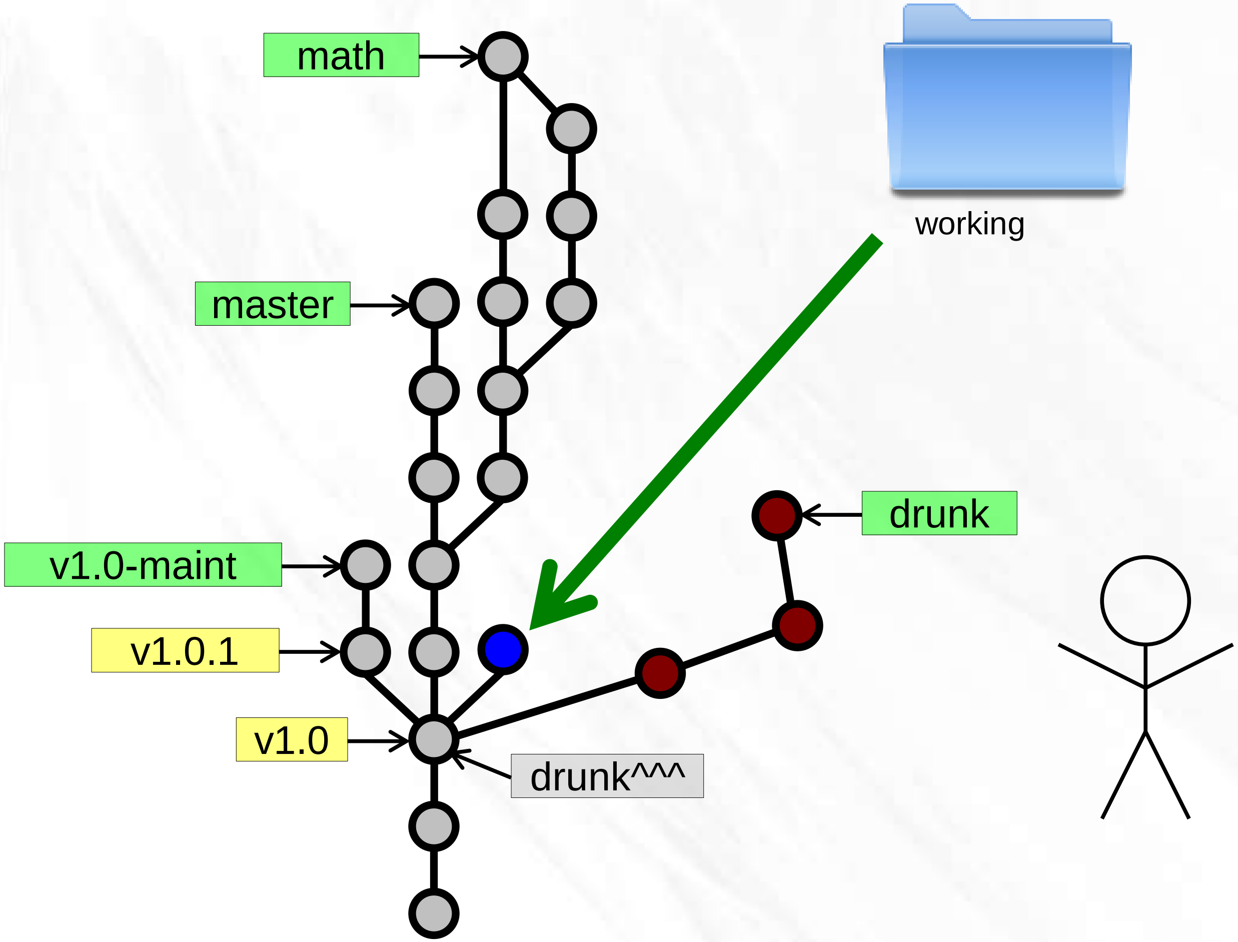
Rewriting History



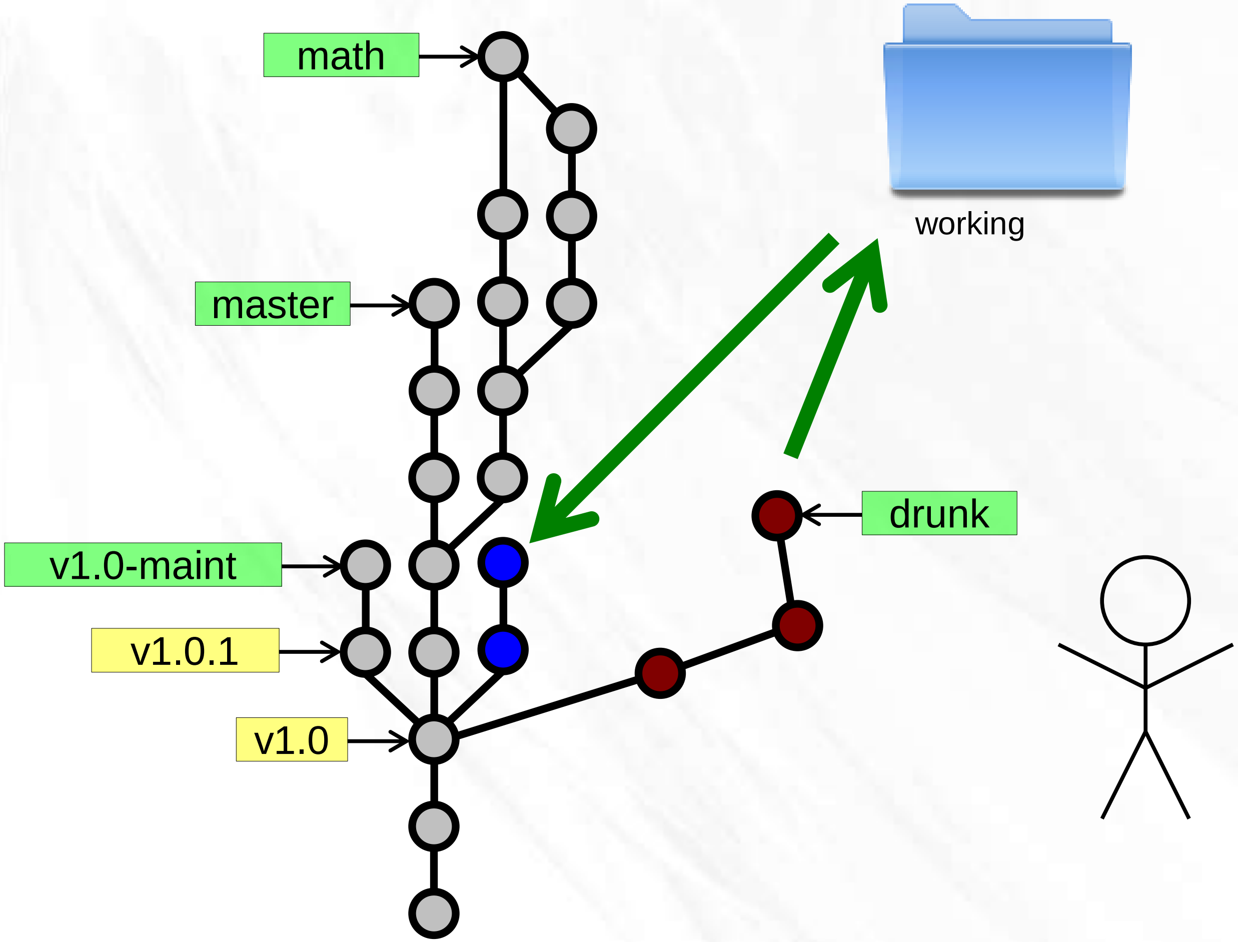
Rewriting History



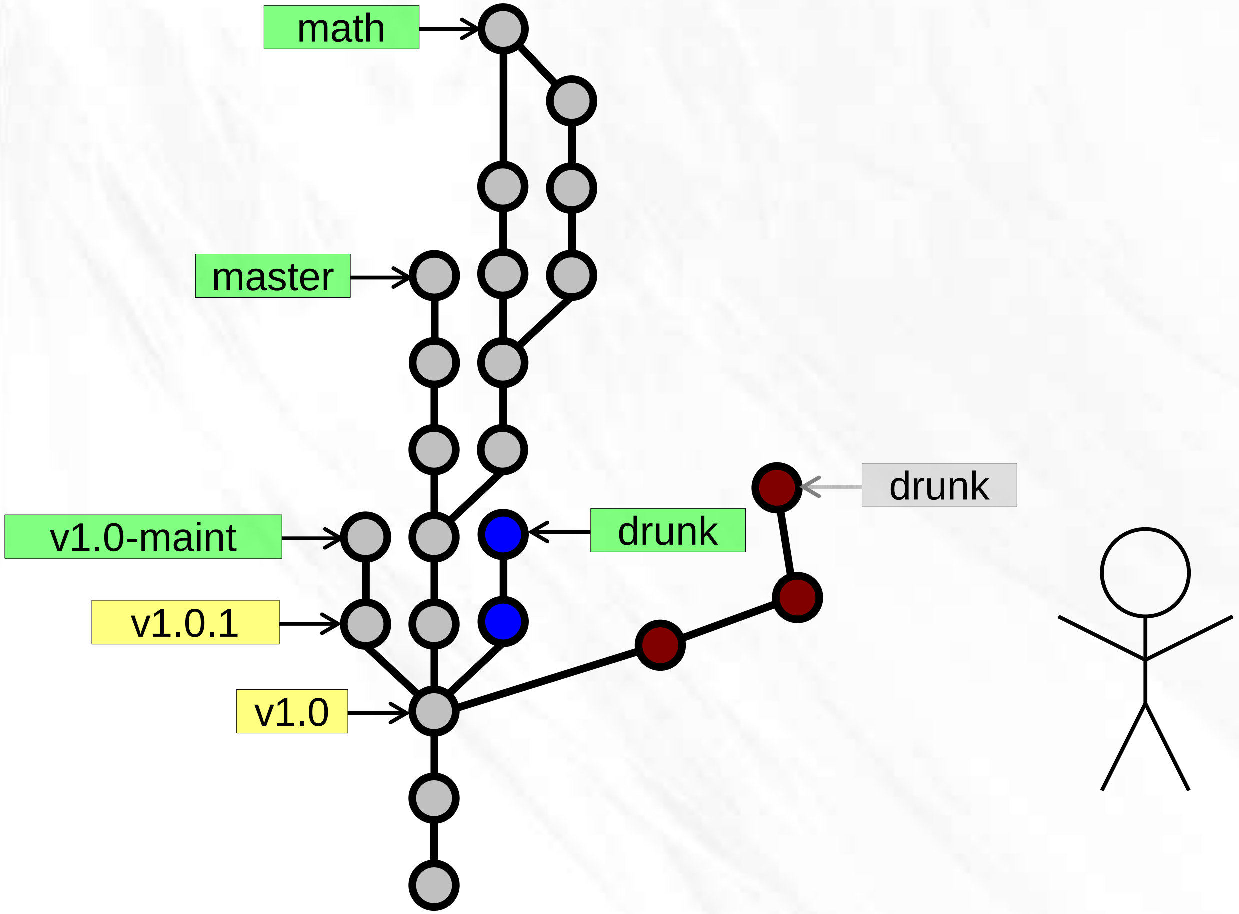
Rewriting History



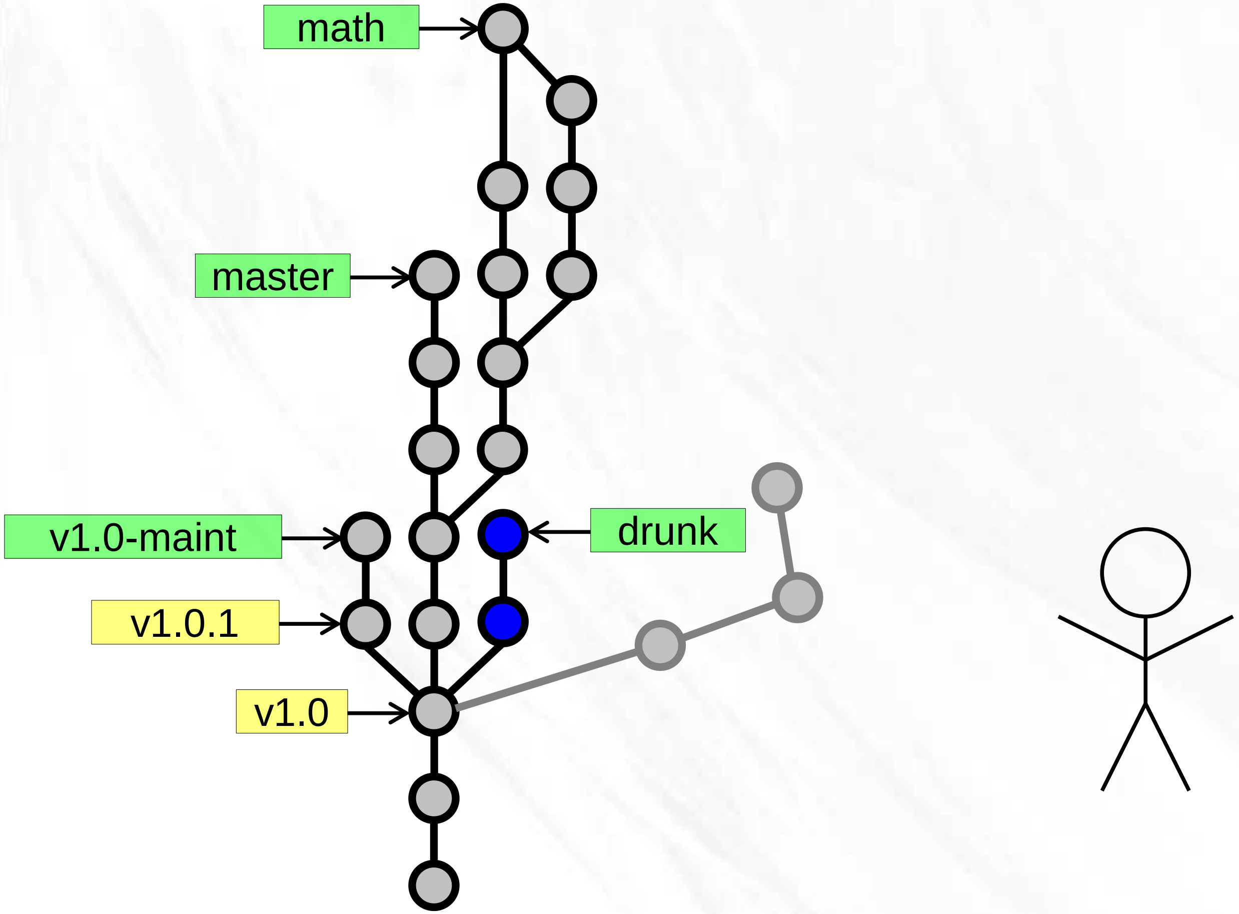
Rewriting History



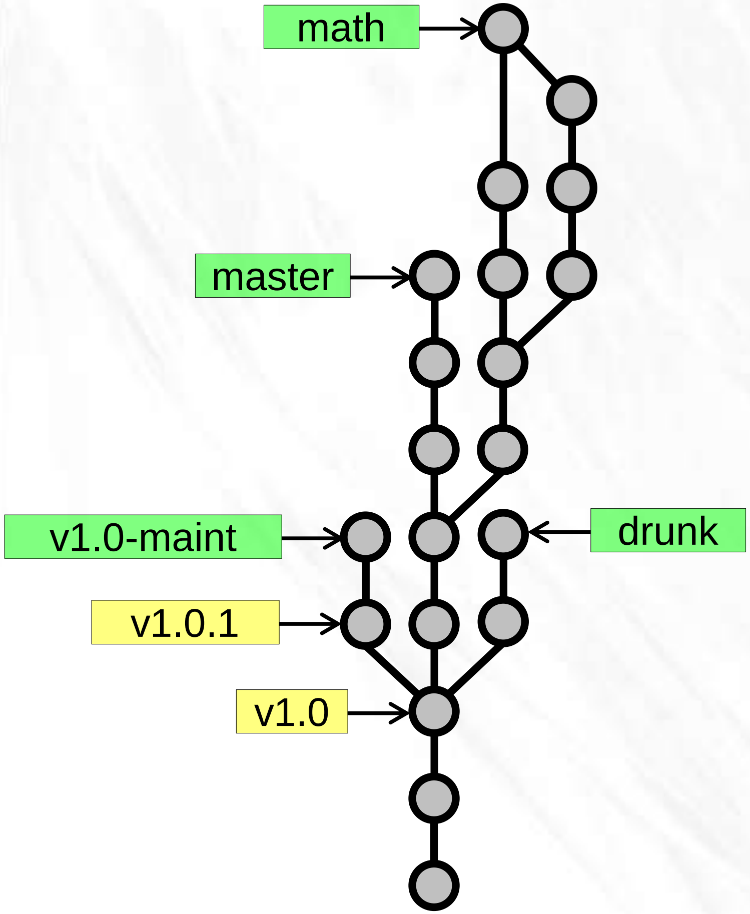
Rewriting History

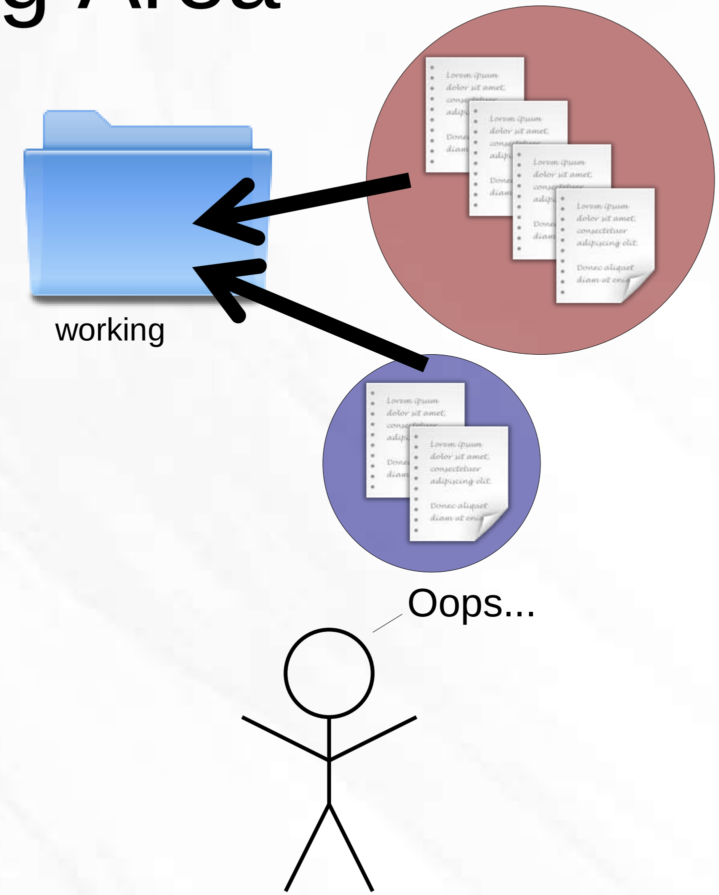


Rewriting History

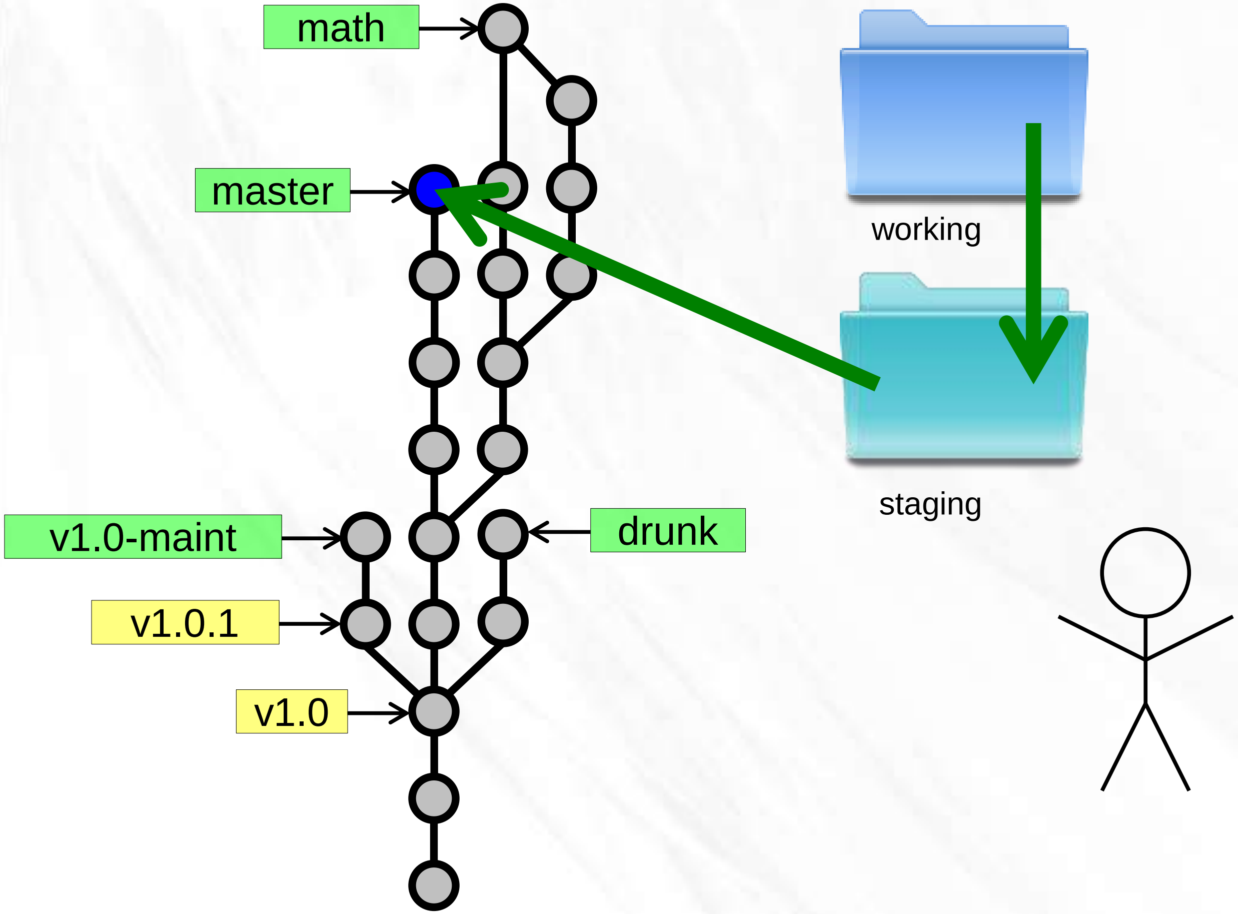


Staging Area

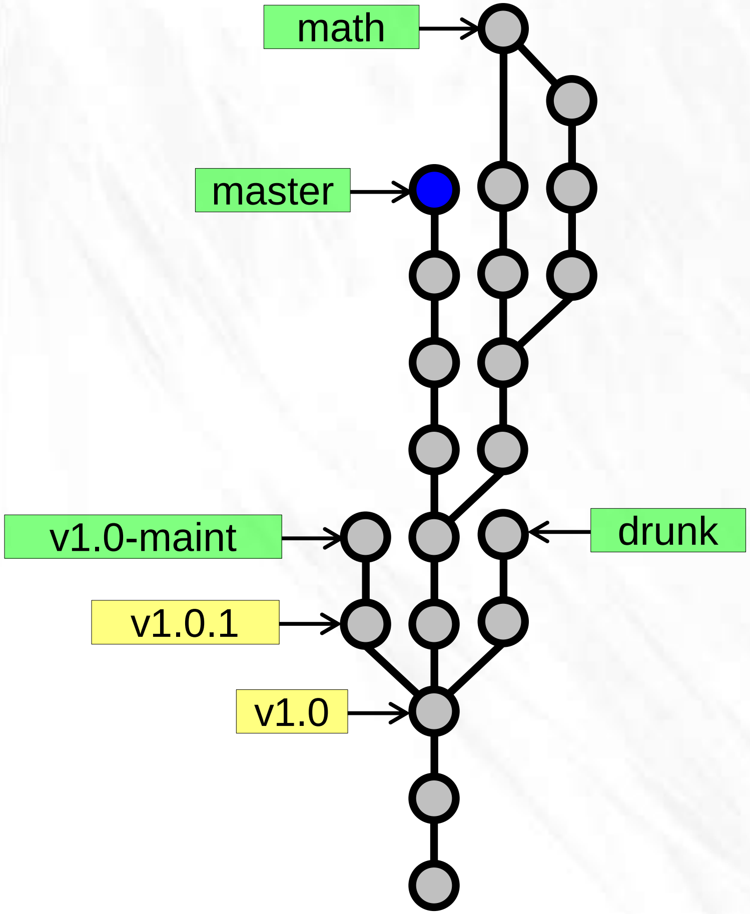


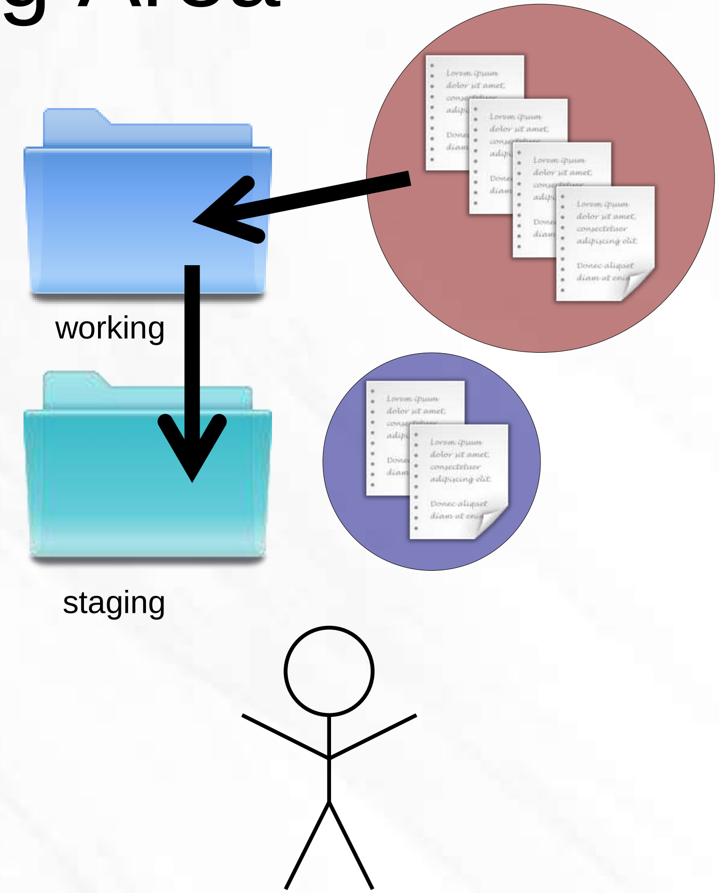


Staging Area

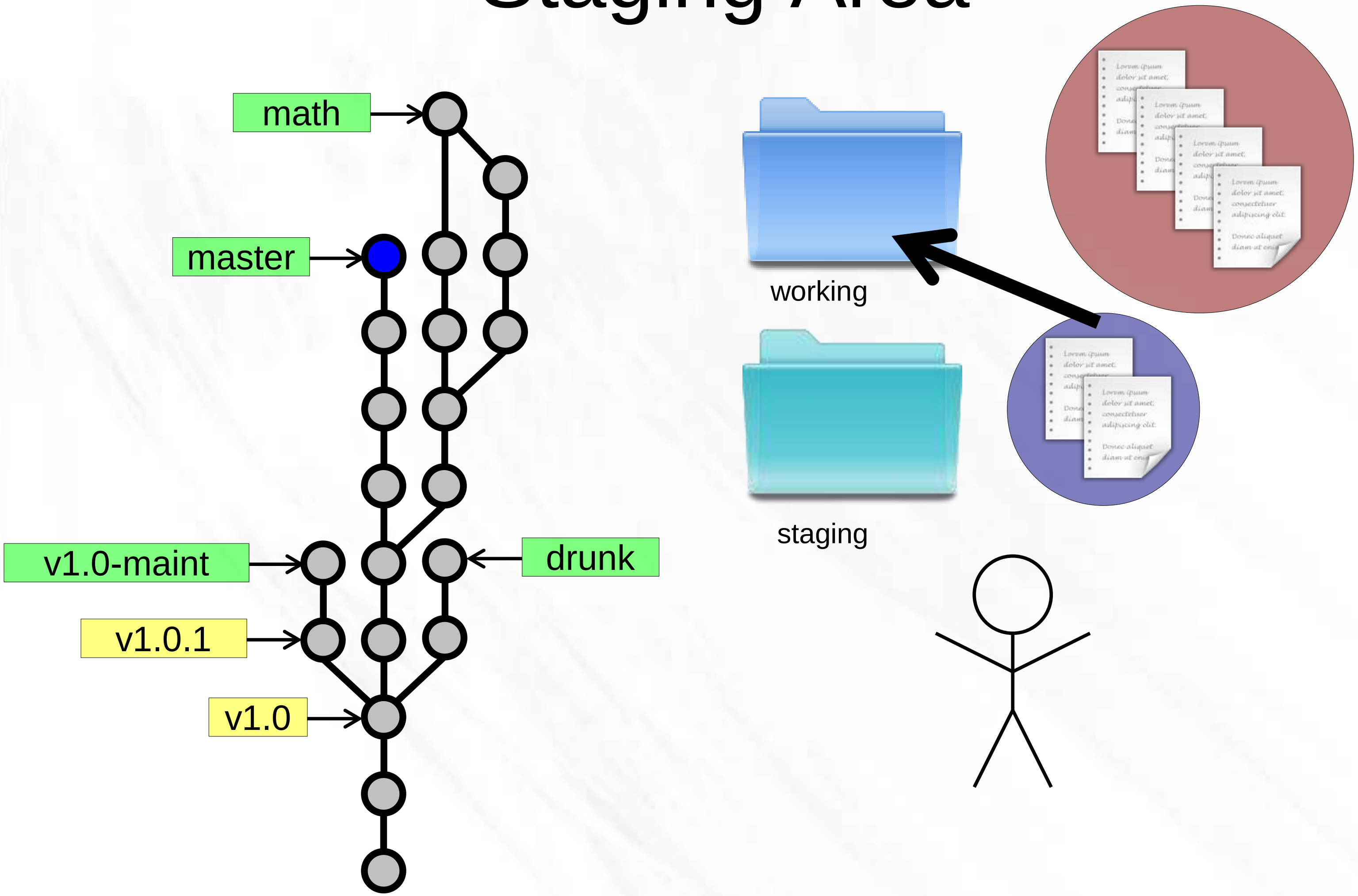


Staging Area

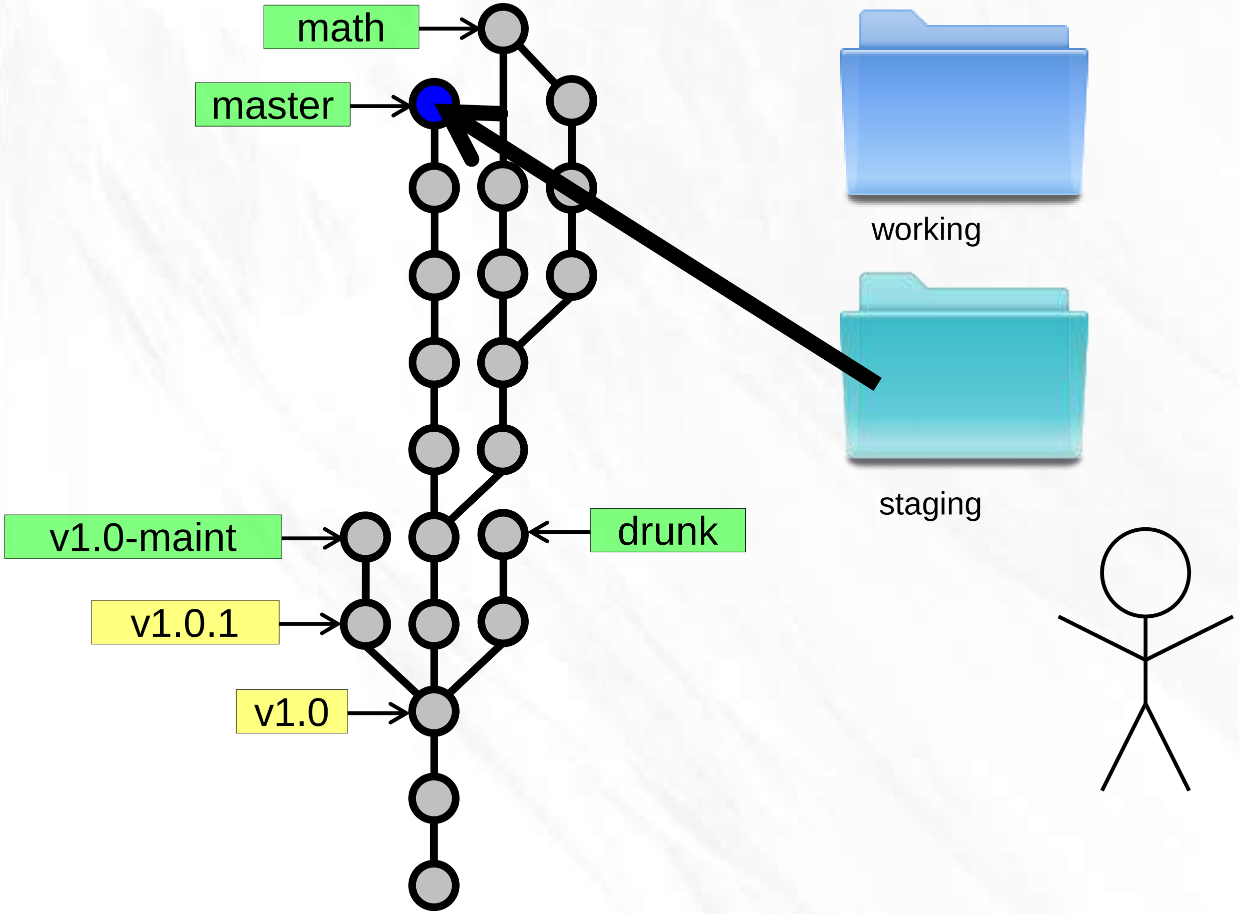




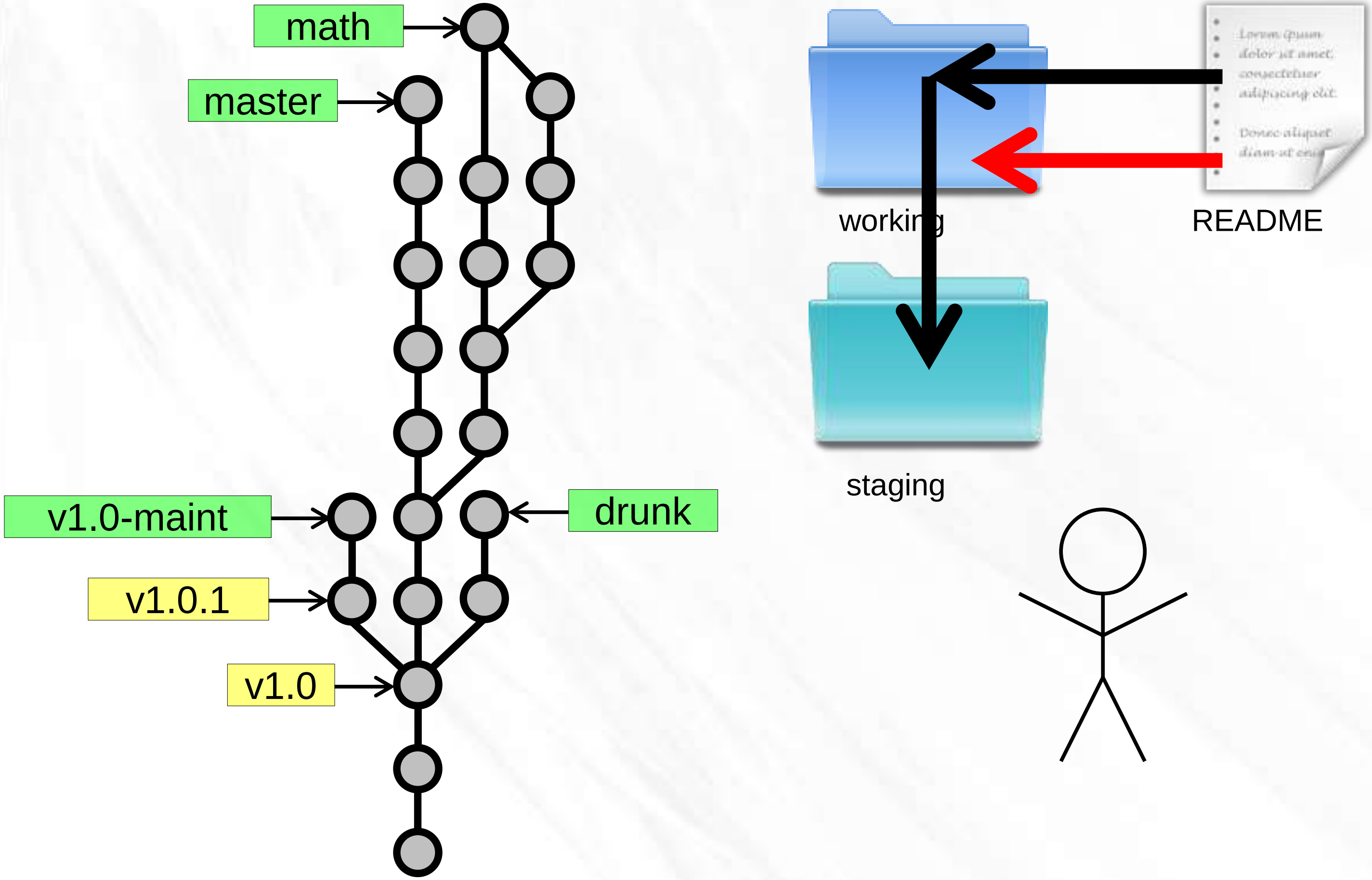
Staging Area



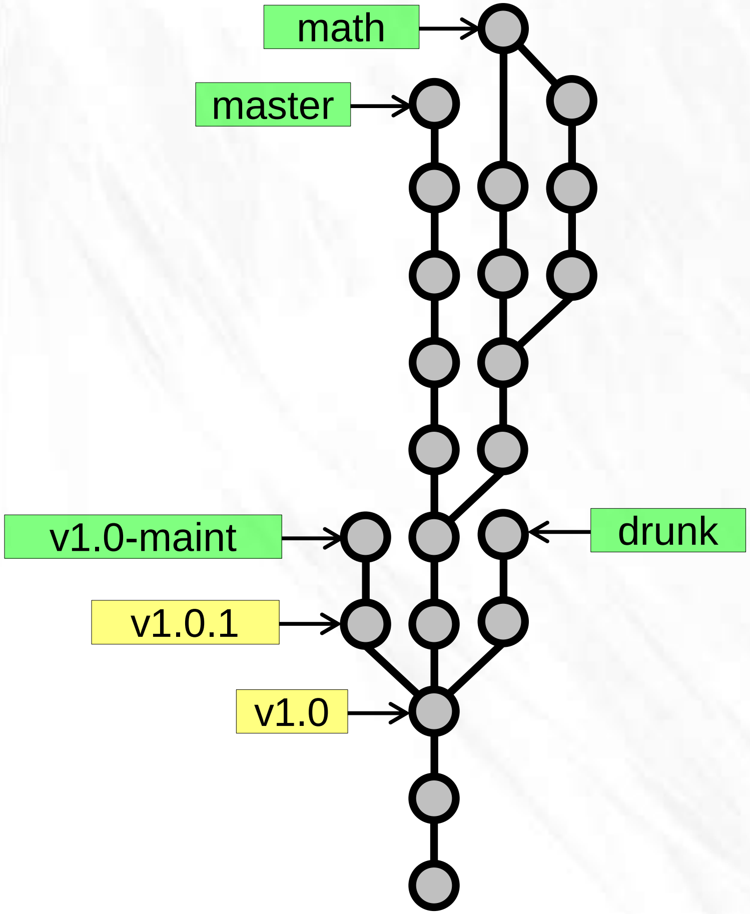
Staging Area

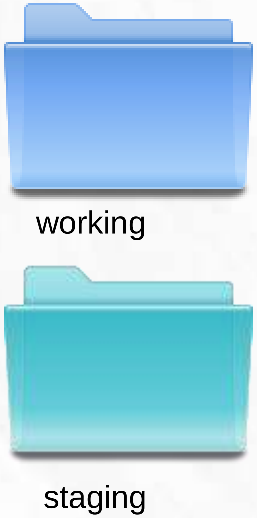


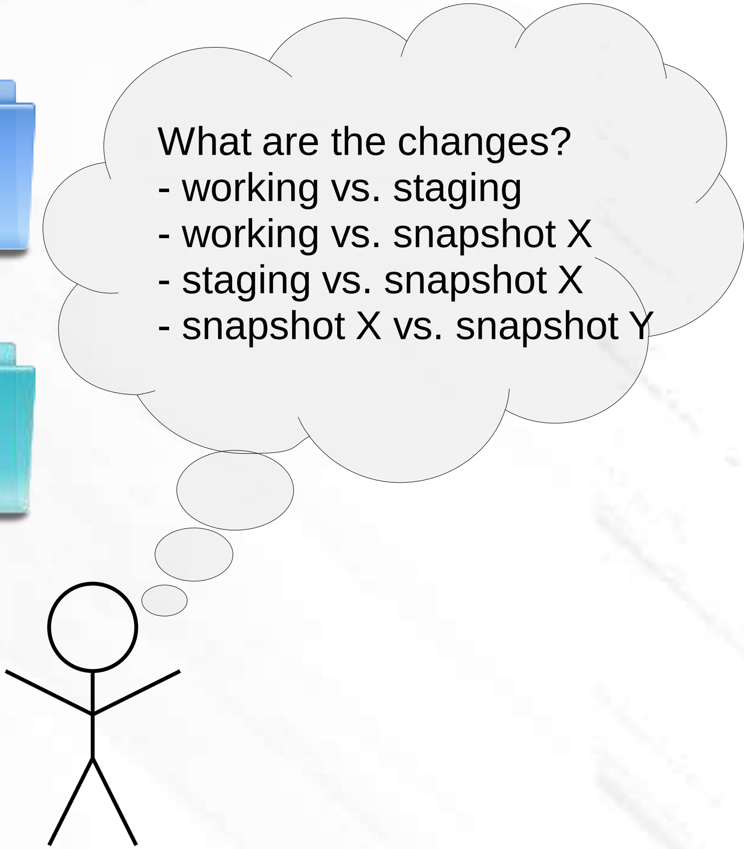
Staging Area



Diffs

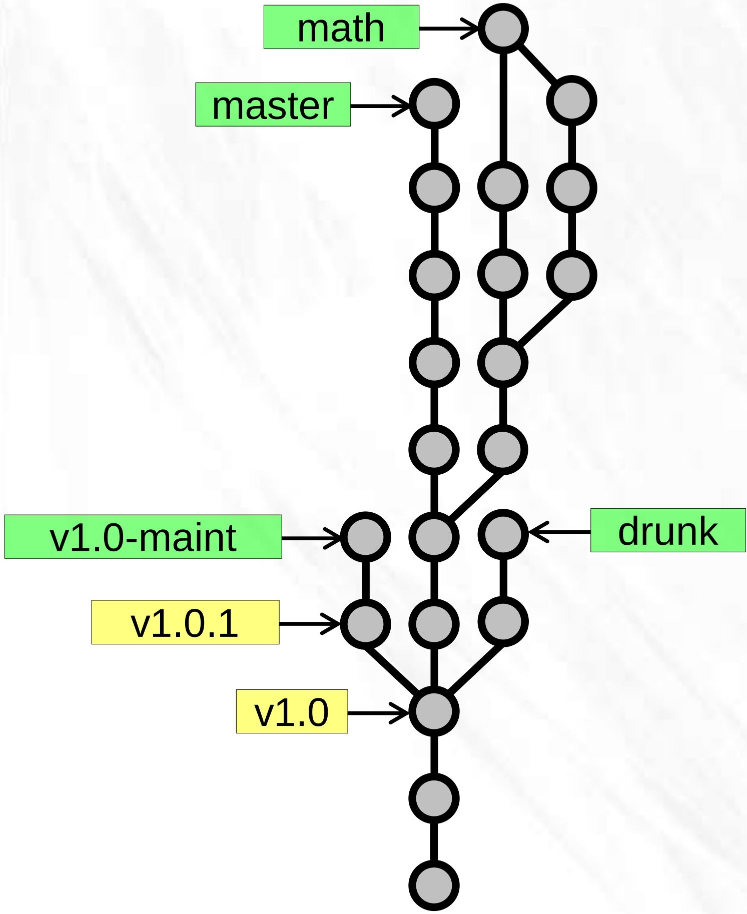


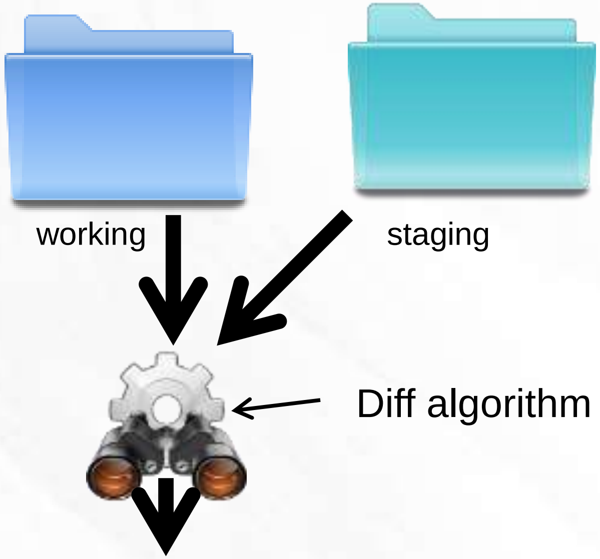




What are the changes?
- working vs. staging
- working vs. snapshot X
- staging vs. snapshot X
- snapshot X vs. snapshot Y

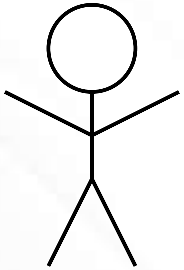
Diffs



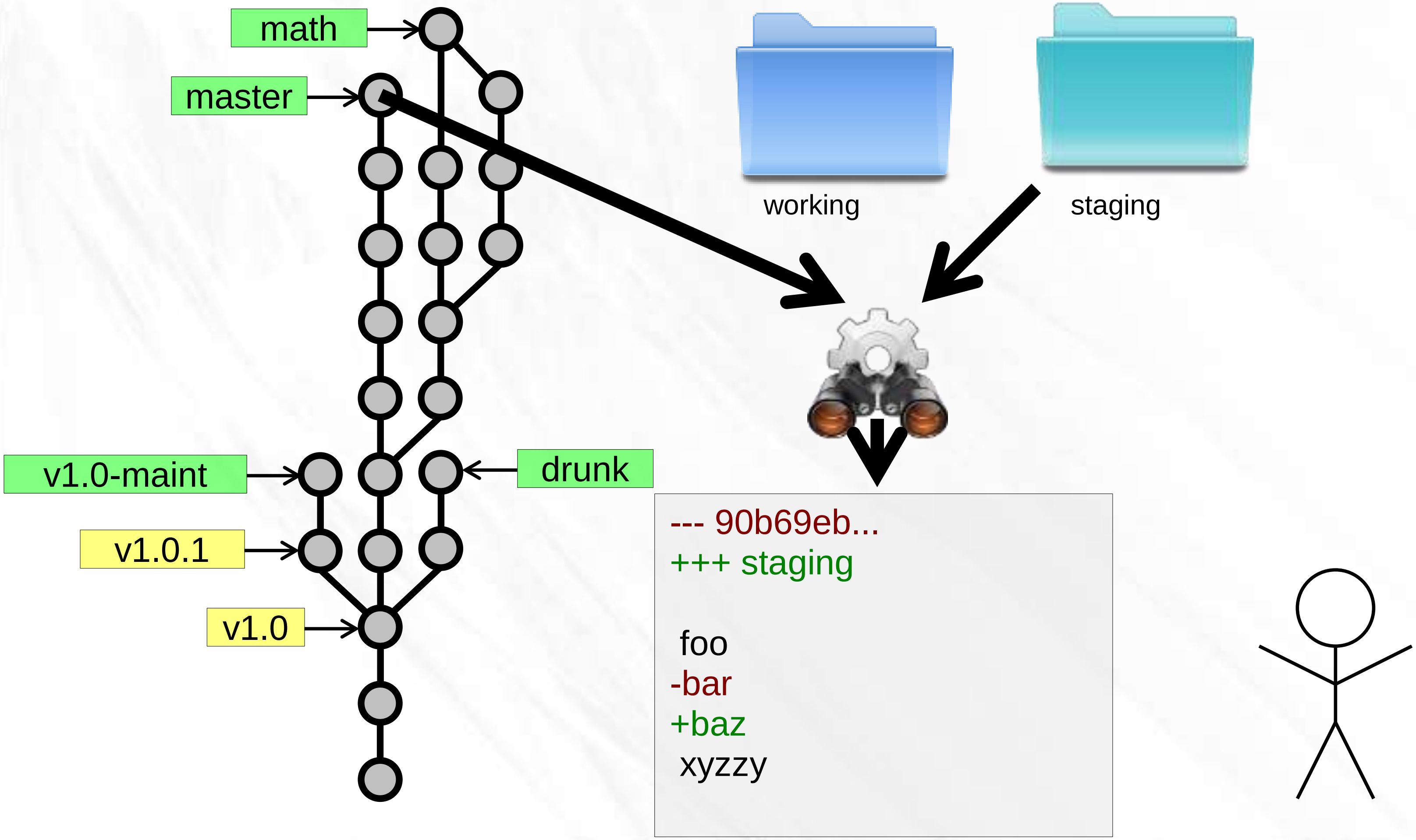


```
--- staging
+++ working

 foo
-baz
+bazzle
 xyzzy
```



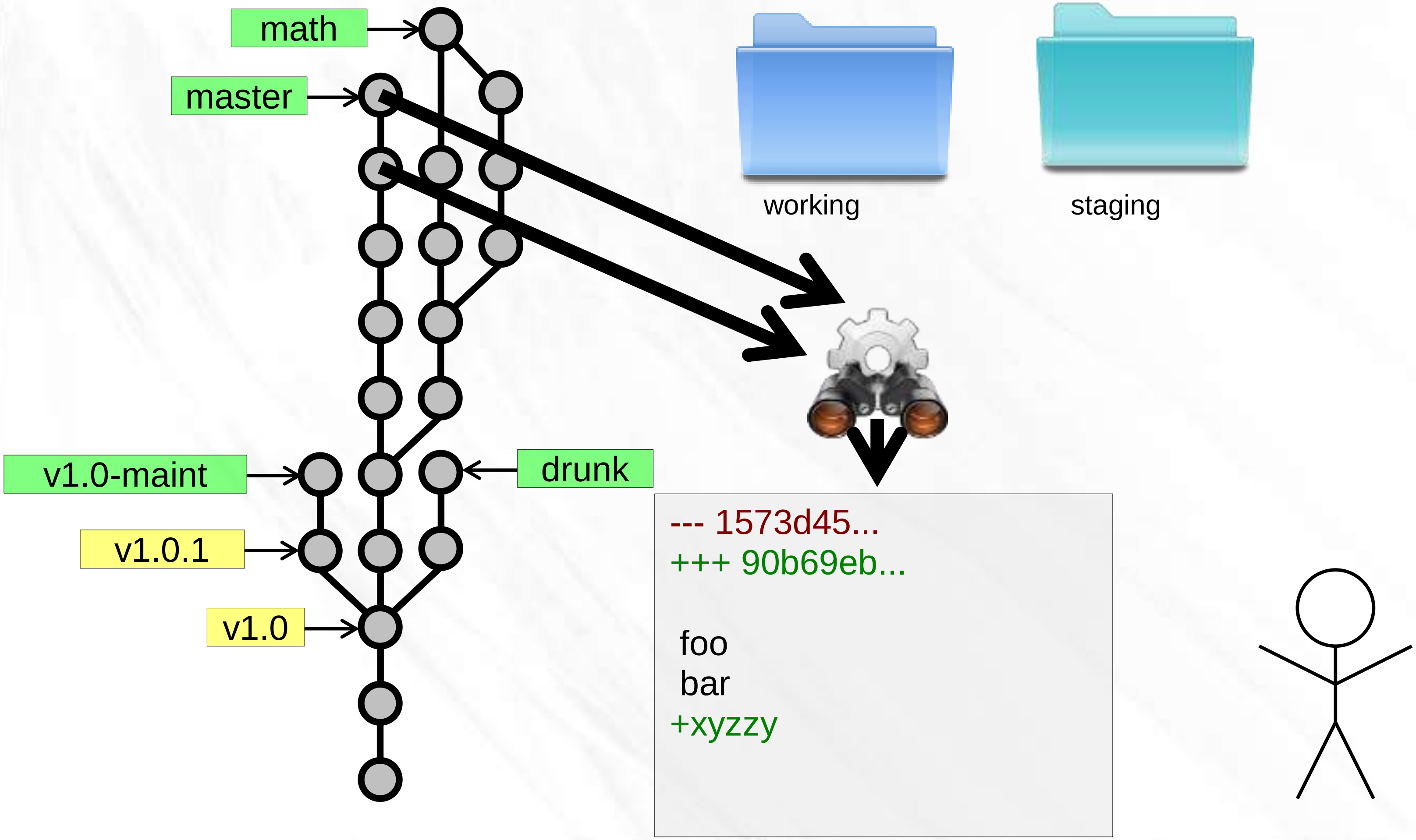
Diffs



```
--- 90b69eb...
+++ staging

 foo
-bar
+baz
 xyzzy
```

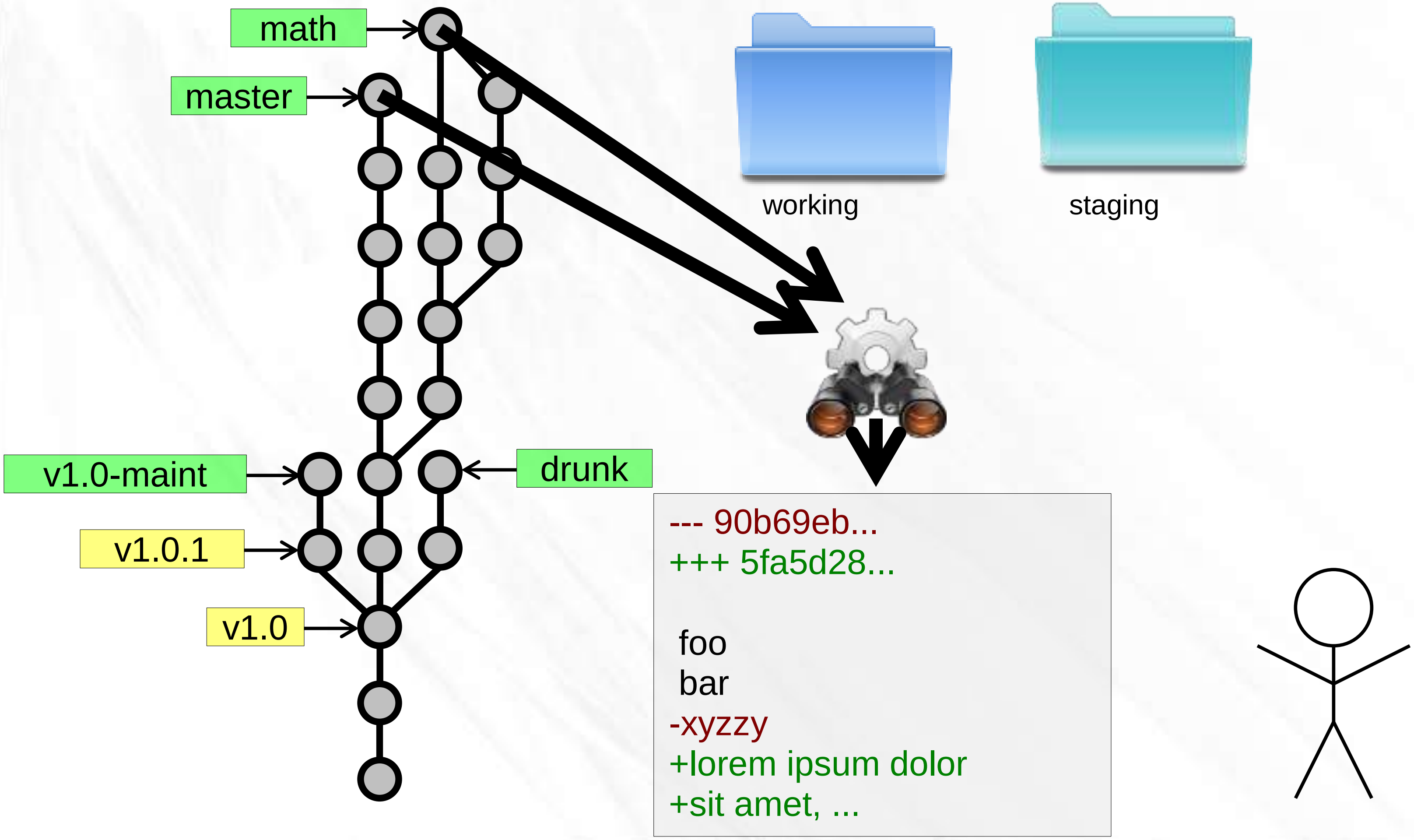
Diffs



```
--- 1573d45...
+++ 90b69eb...

 foo
 bar
+xyzzy
```

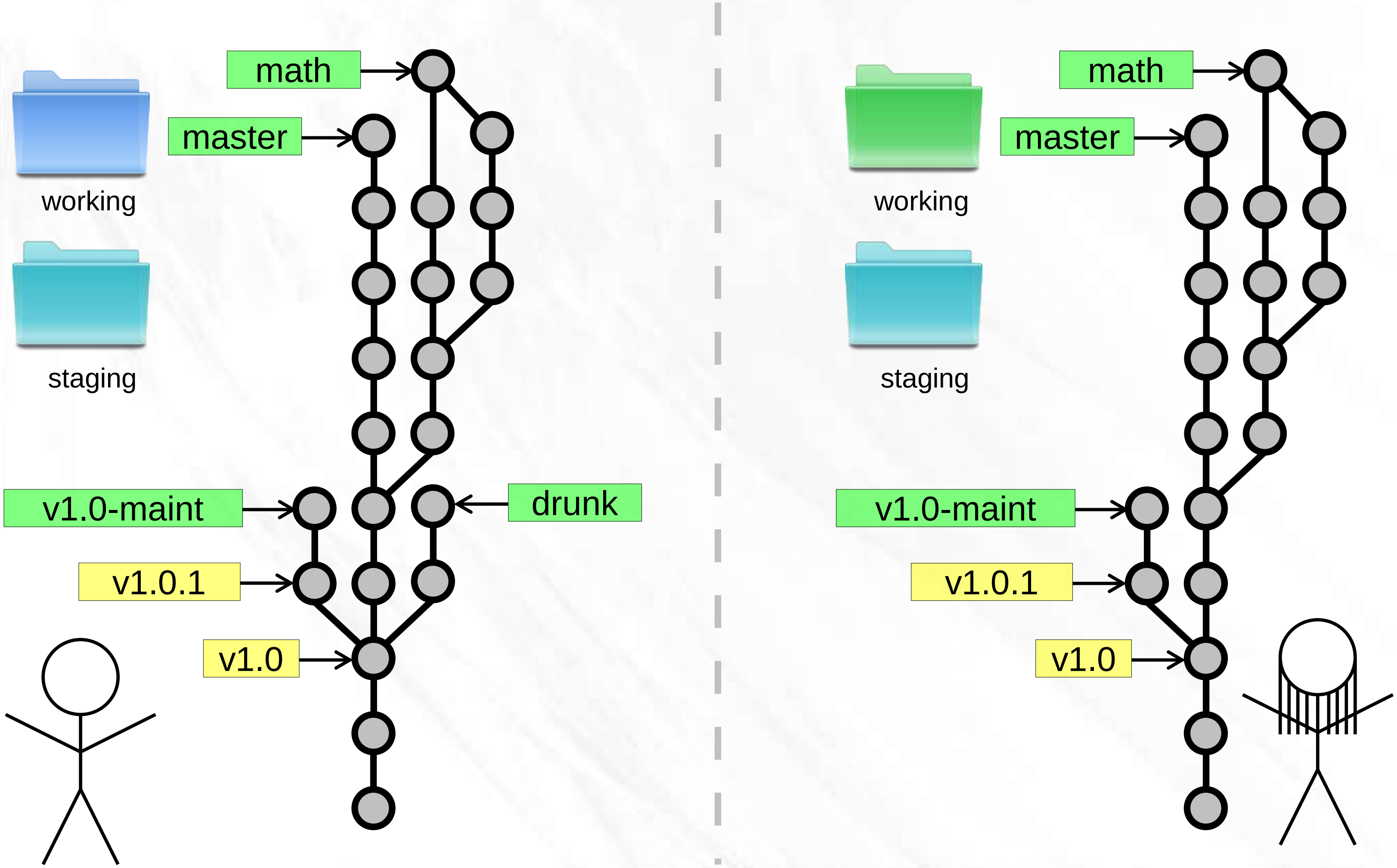
Diffs



```
--- 90b69eb...
+++ 5fa5d28...

 foo
 bar
-xyzzy
+lorem ipsum dolor
+sit amet, ...
```

Eliminating Duplication



math

master

working

staging

v1.0-maint

drunk

v1.0.1

v1.0

math

master

working

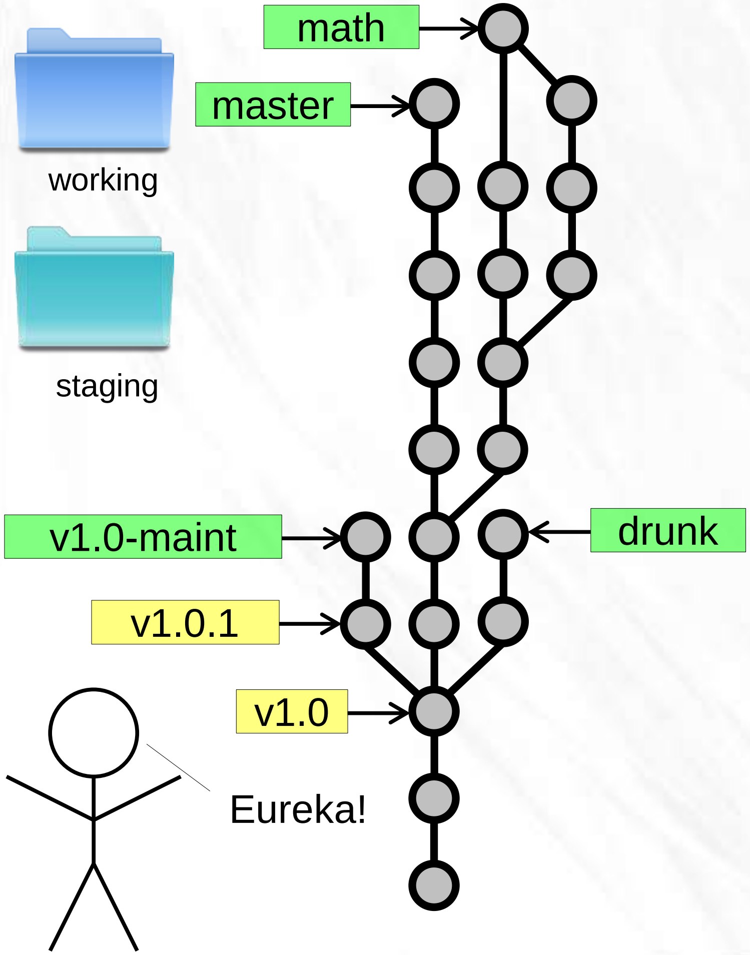
staging

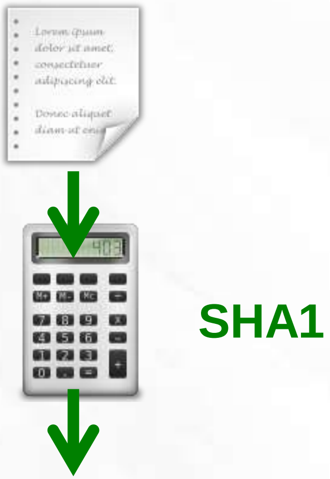
v1.0-maint

v1.0.1

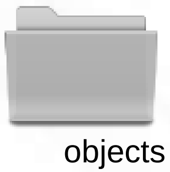
v1.0

Eliminating Duplication

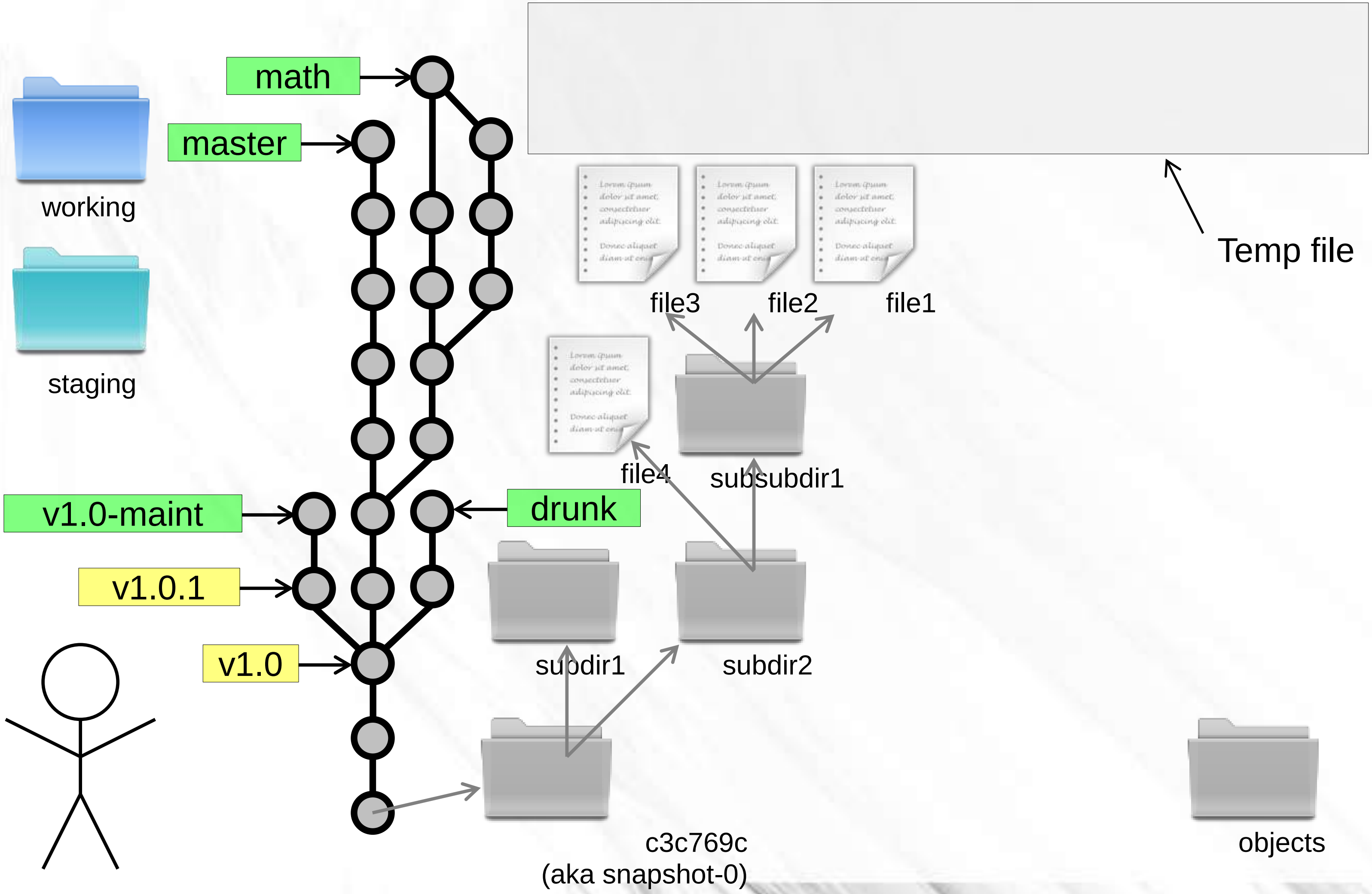




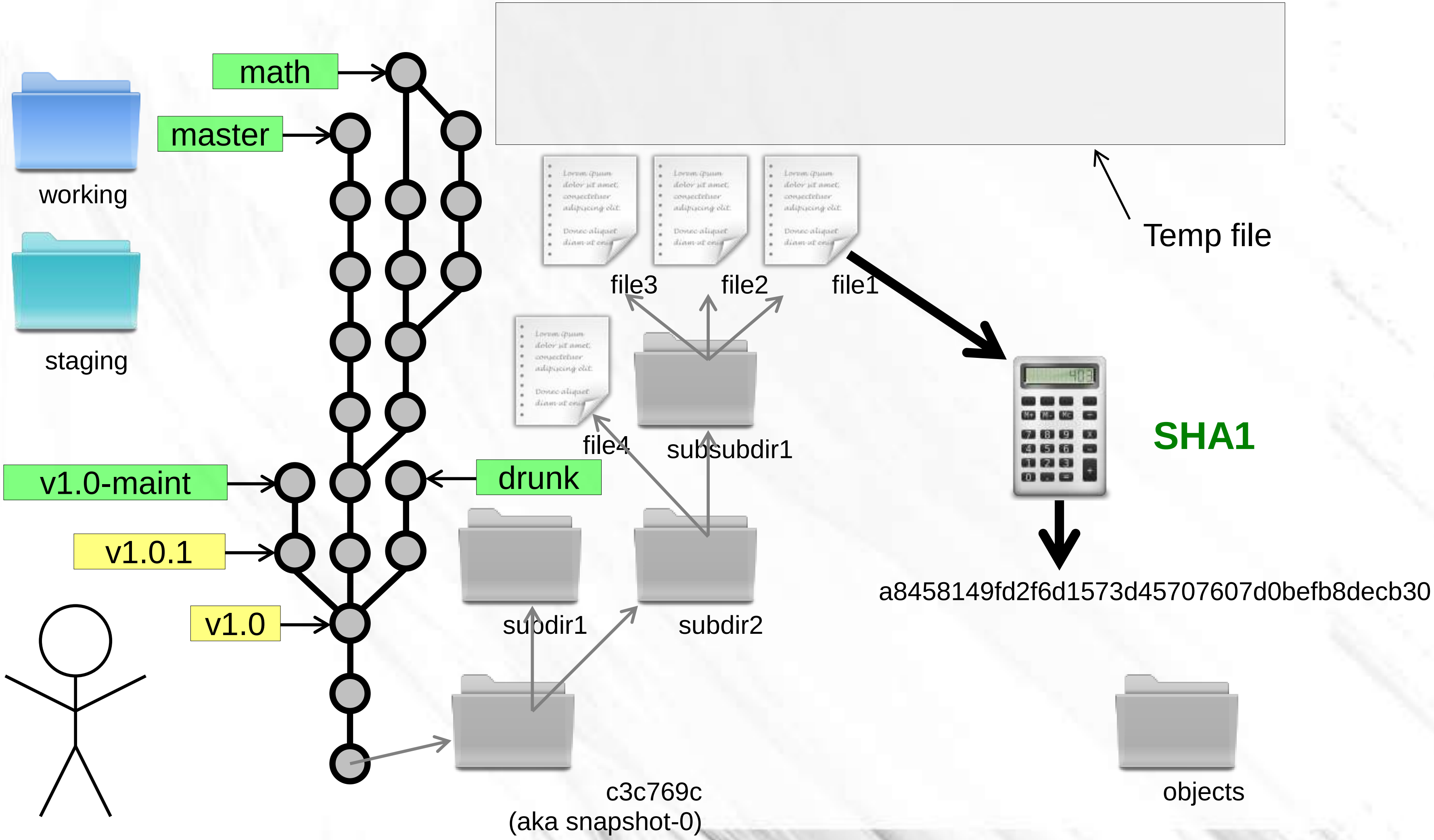
2804133755c3ed396d162028c7b30a1cbcfecded



Eliminating Duplication

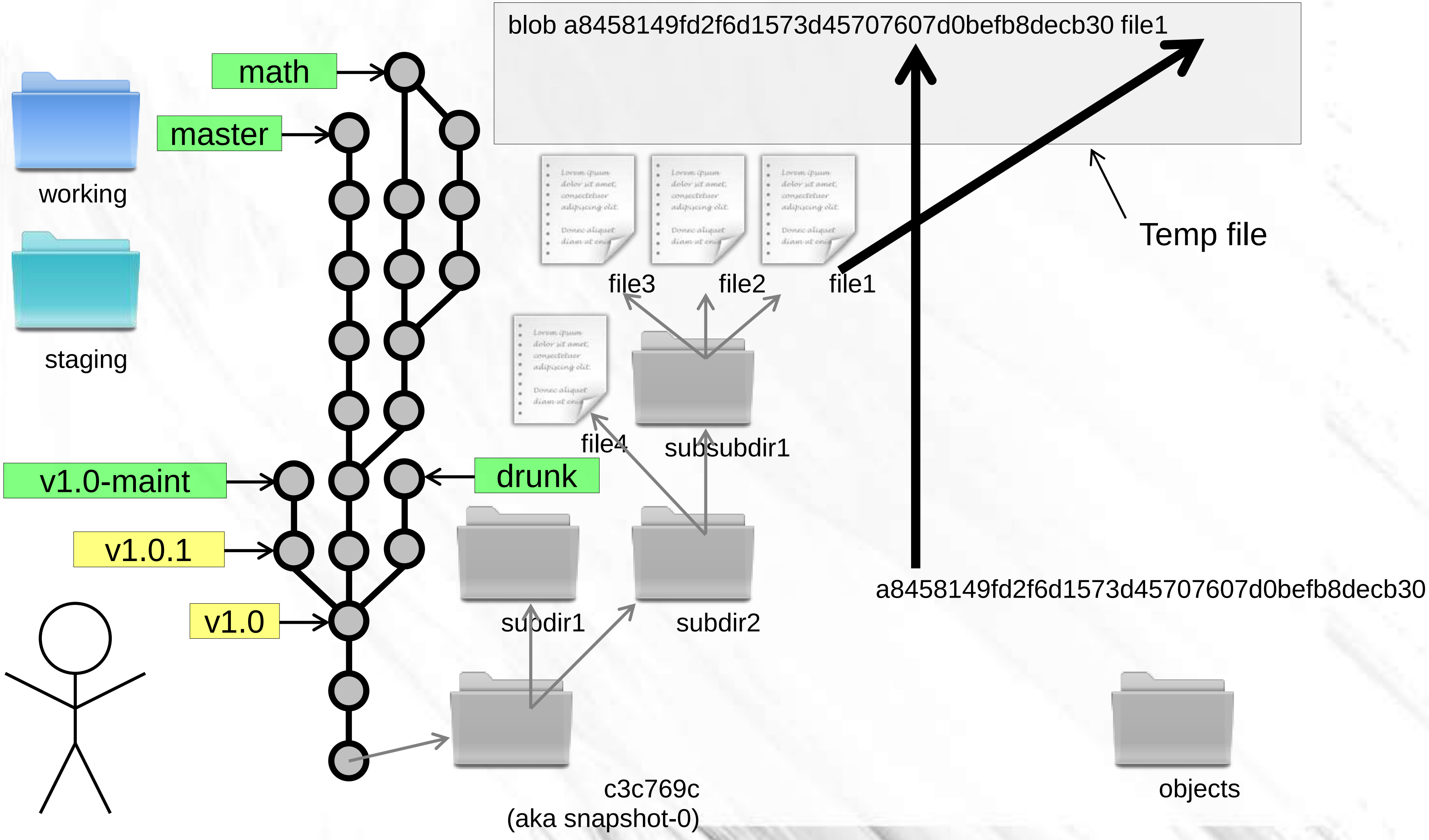


Eliminating Duplication

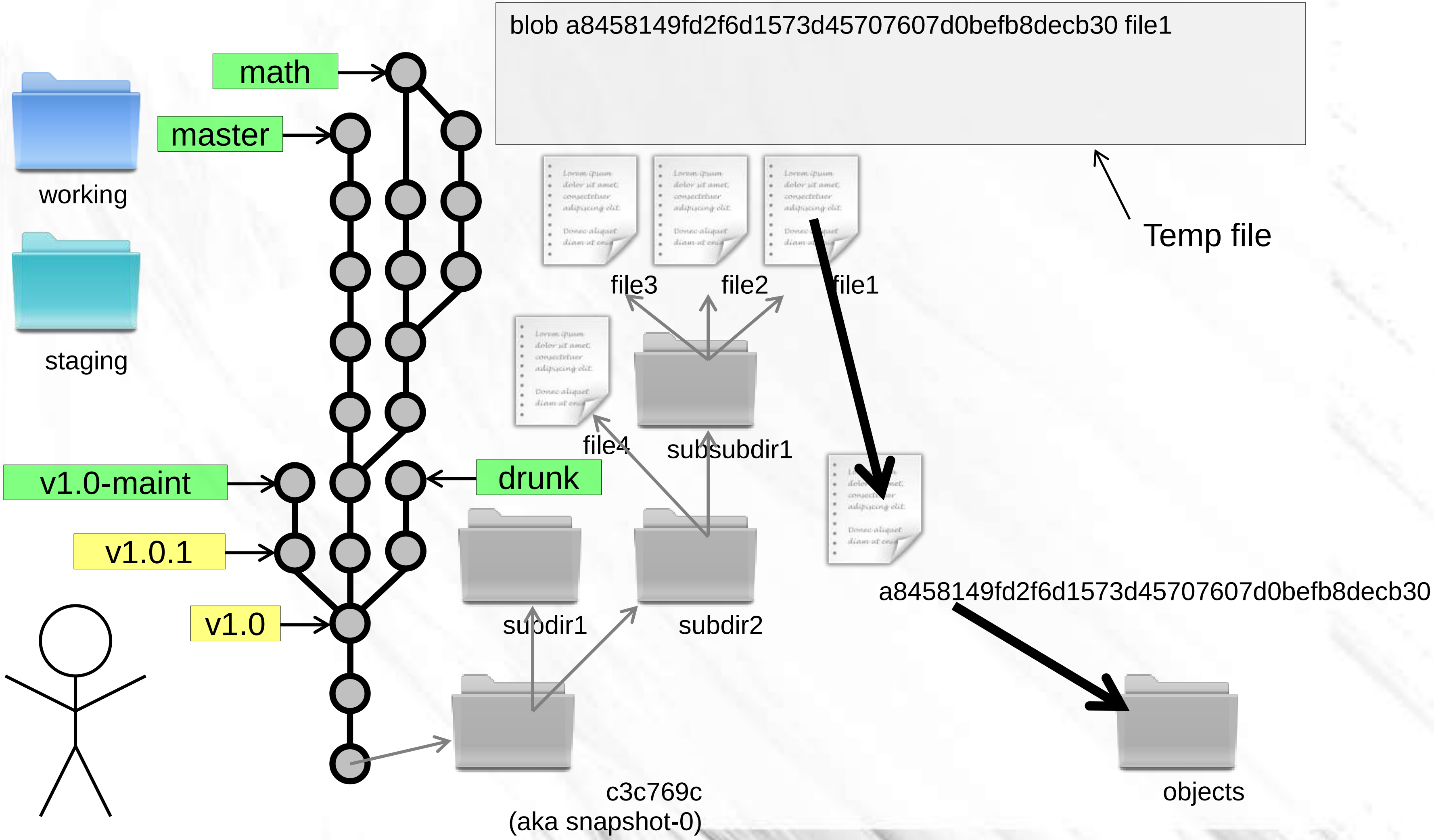


Eliminating Duplication

blob a8458149fd2f6d1573d45707607d0befb8decb30 file1

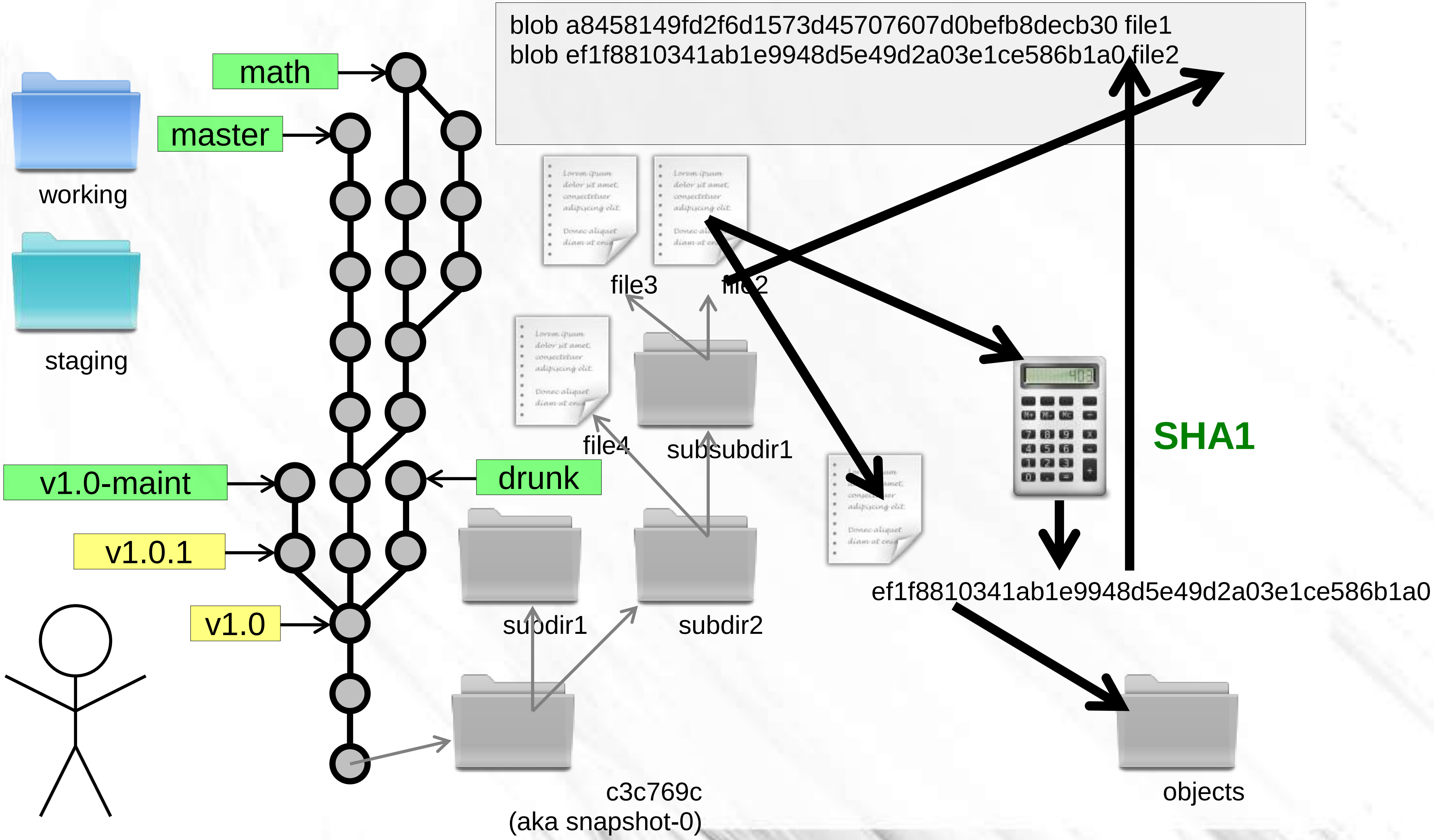


Eliminating Duplication



blob a8458149fd2f6d1573d45707607d0befb8decb30 file1

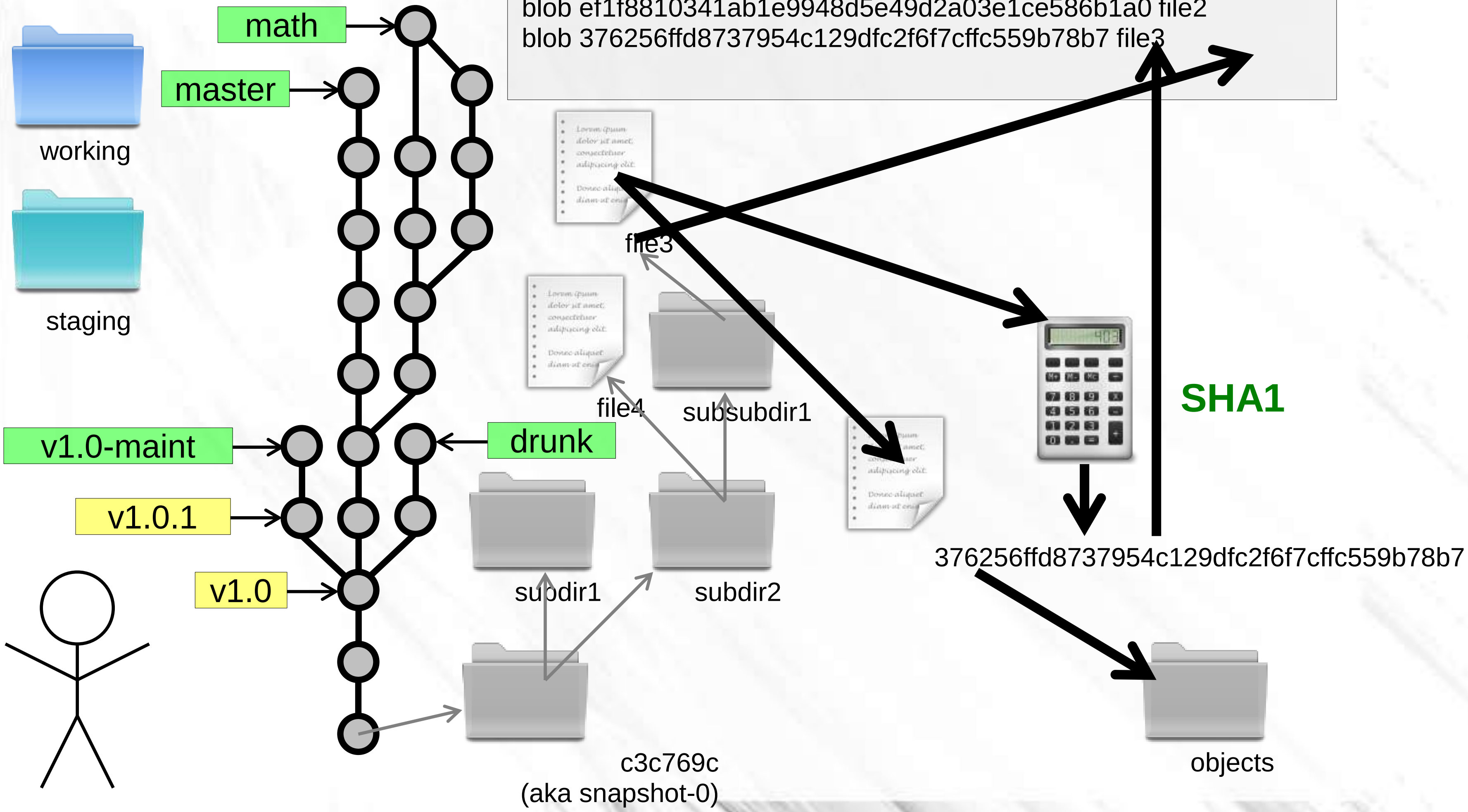
Eliminating Duplication



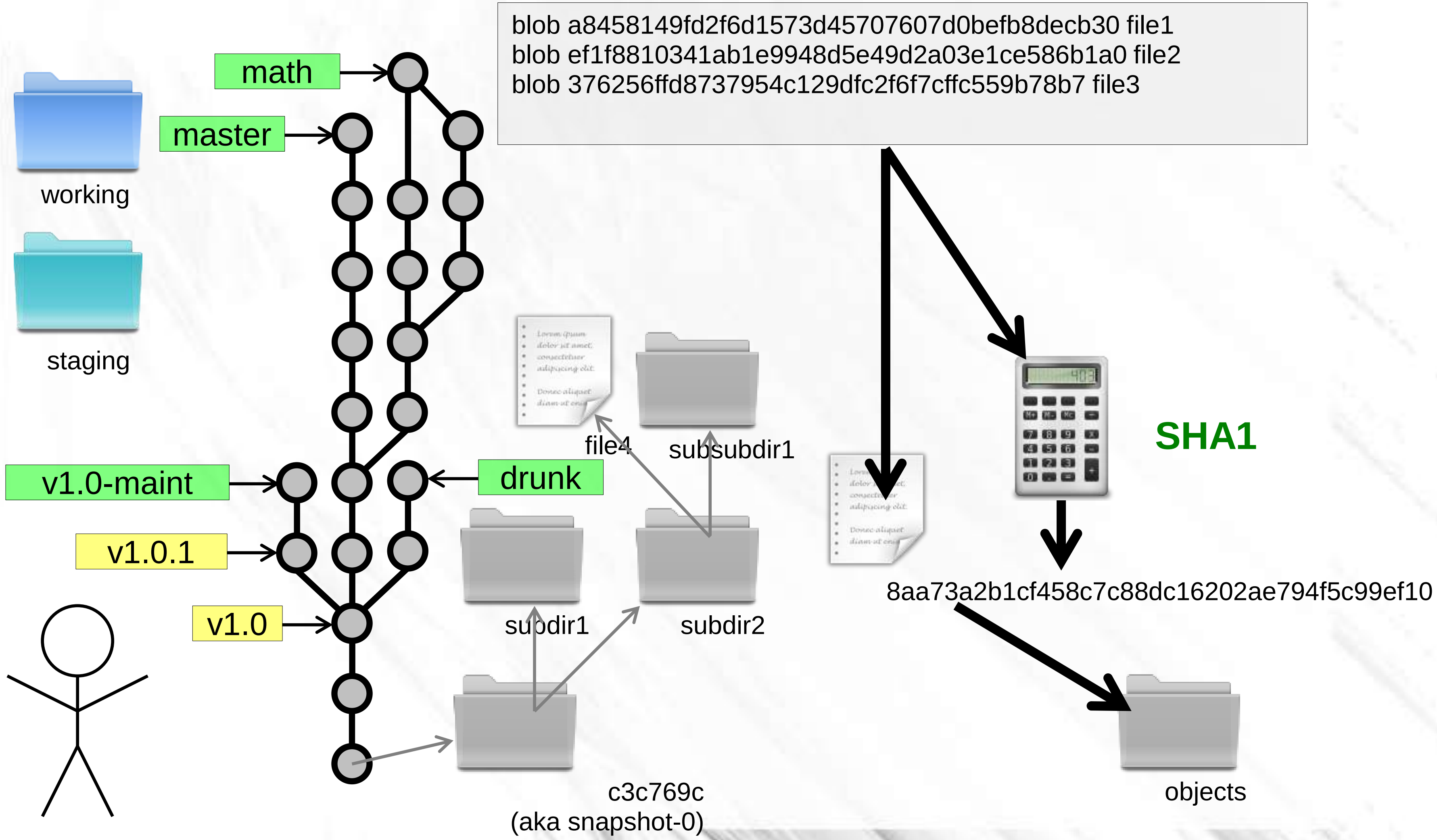
```
blob a8458149fd2f6d1573d45707607d0befb8decb30 file1
blob ef1f8810341ab1e9948d5e49d2a03e1ce586b1a0 file2
```

Eliminating Duplication

```
blob a8458149fd2f6d1573d45707607d0befb8decb30 file1
blob ef1f8810341ab1e9948d5e49d2a03e1ce586b1a0 file2
blob 376256ffd8737954c129dfc2f6f7cffc559b78b7 file3
```

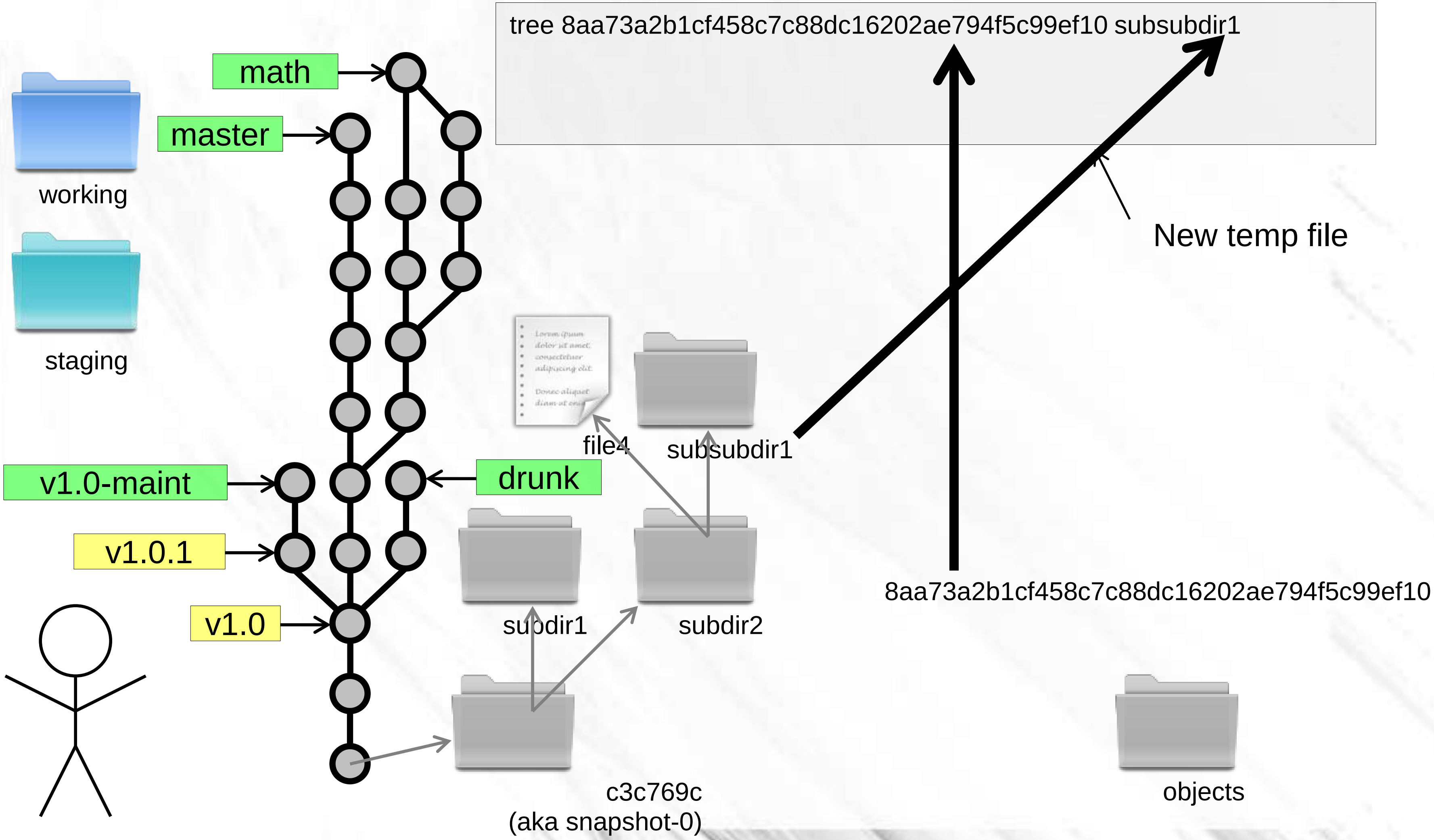


Eliminating Duplication



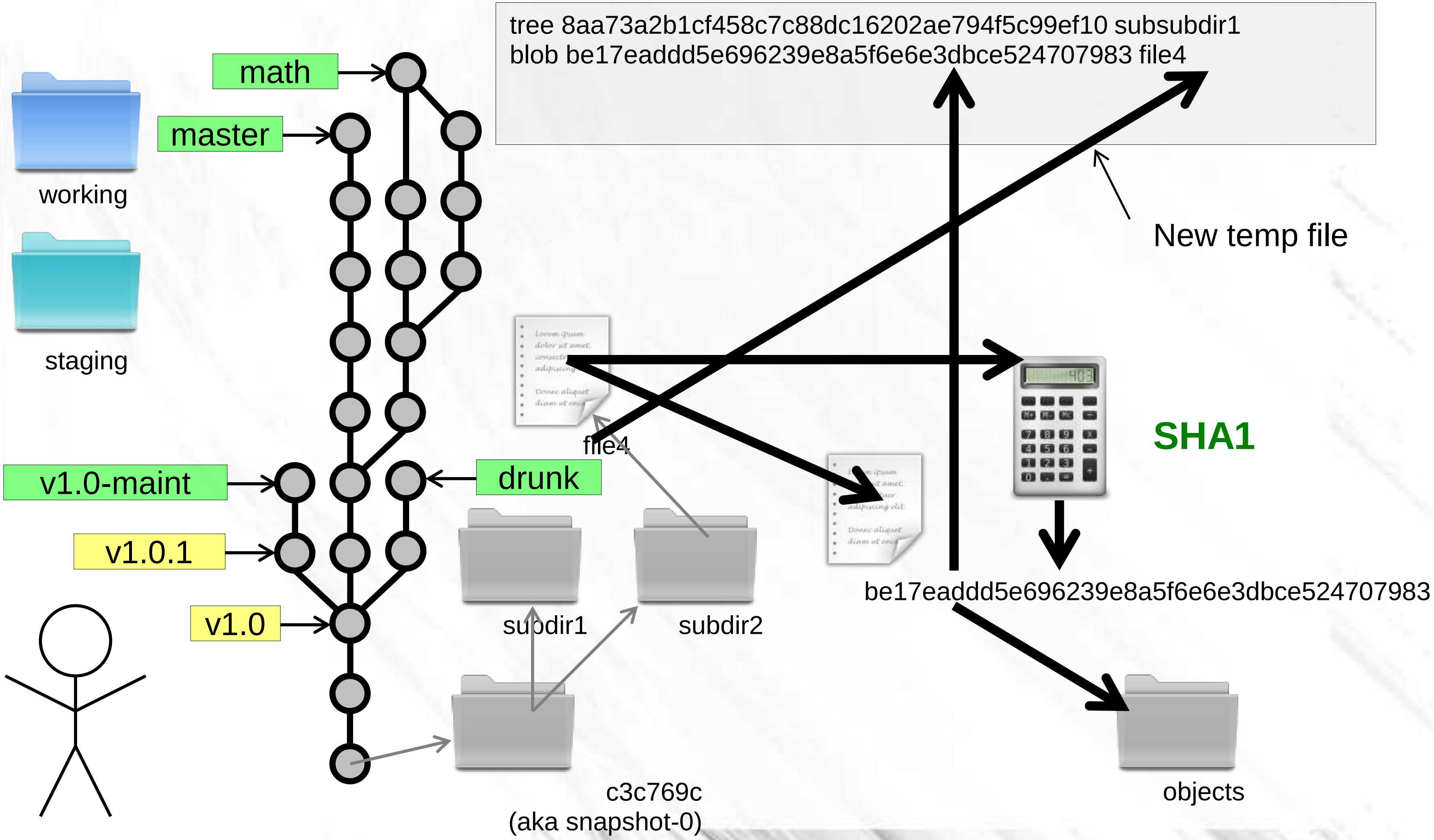
blob a8458149fd2f6d1573d45707607d0befb8decb30 file1
blob ef1f8810341ab1e9948d5e49d2a03e1ce586b1a0 file2
blob 376256ffd8737954c129dfc2f6f7cffc559b78b7 file3

Eliminating Duplication



```
tree 8aa73a2b1cf458c7c88dc16202ae794f5c99ef10 subsubdir1
```

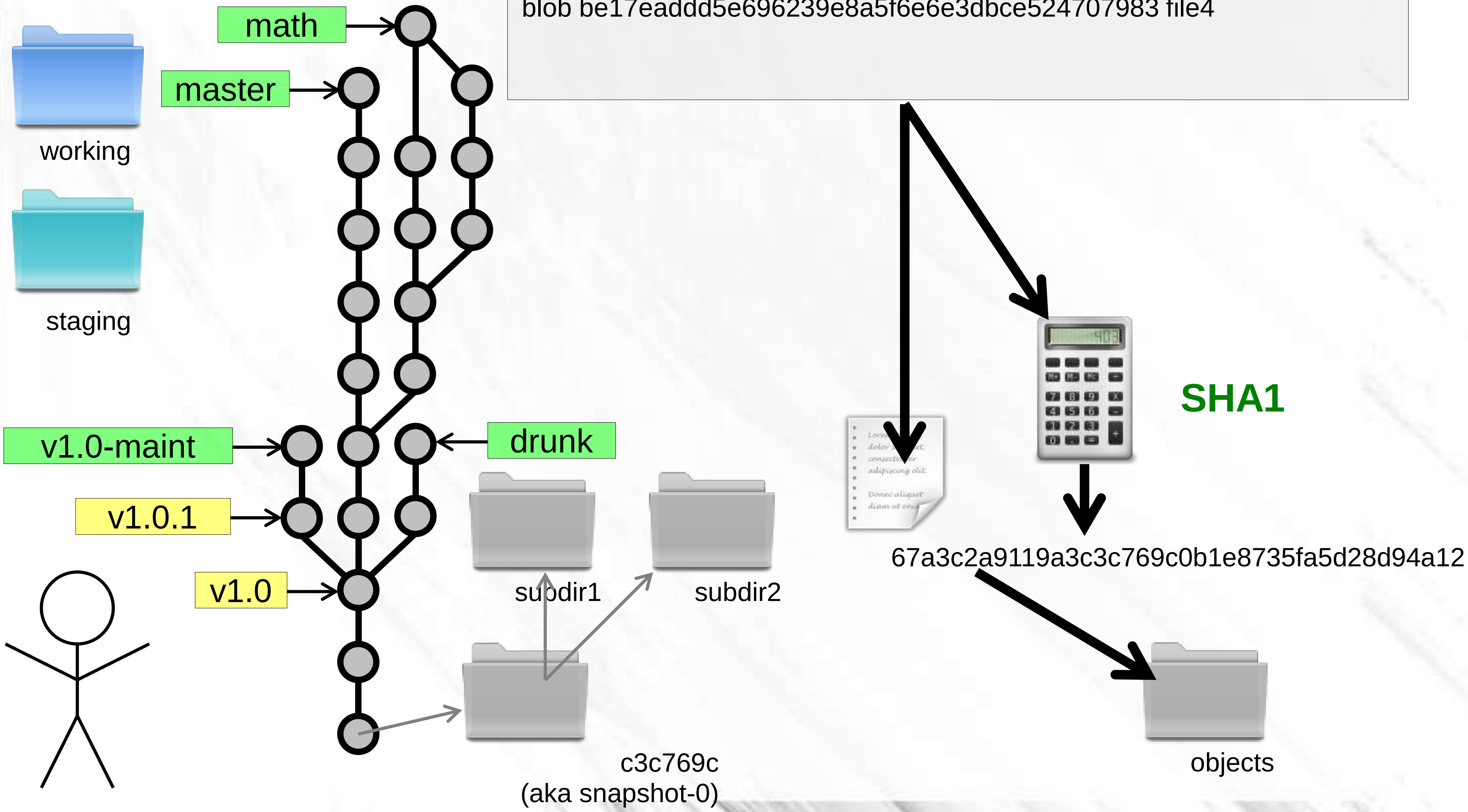
Eliminating Duplication



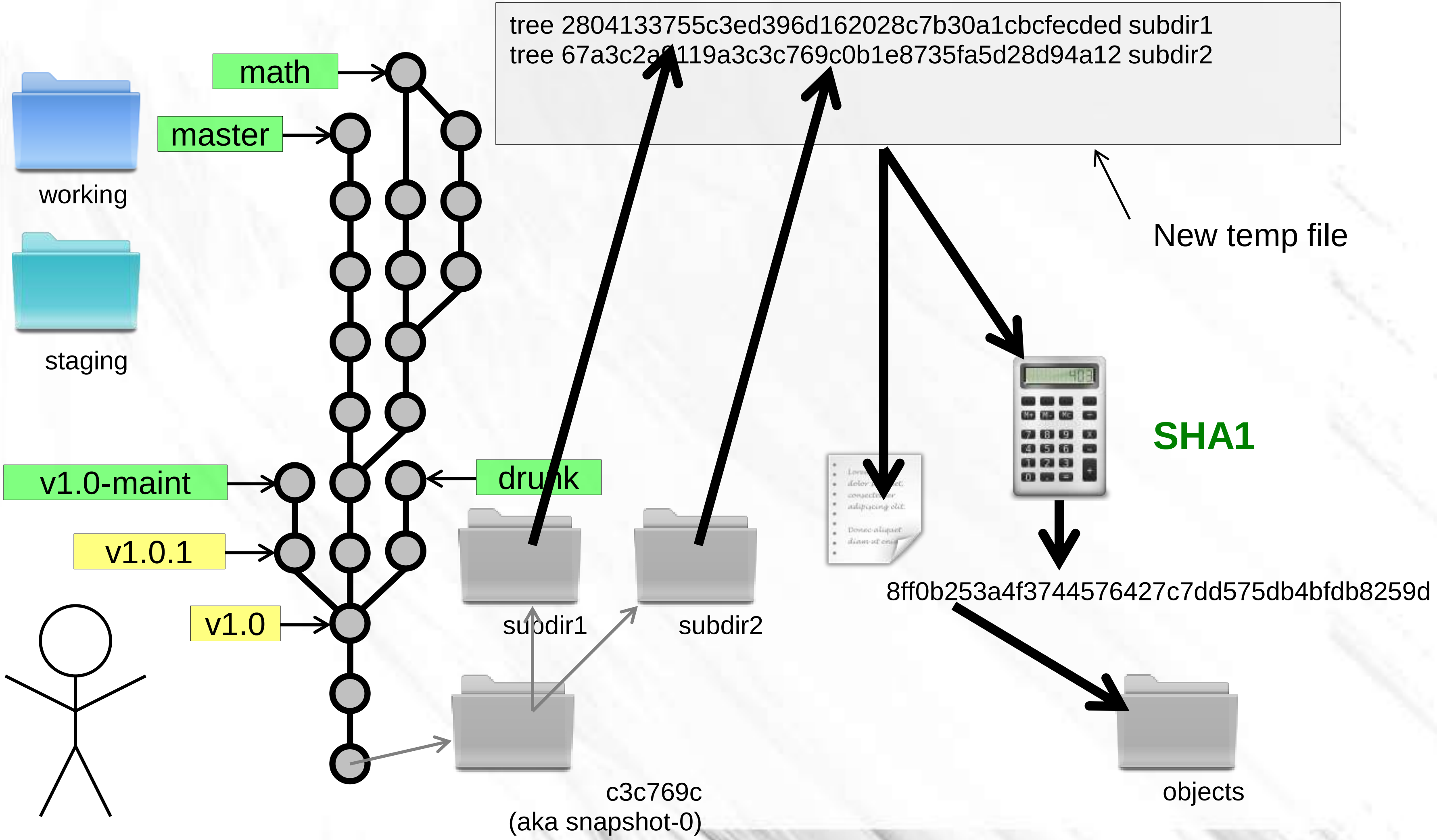
tree 8aa73a2b1cf458c7c88dc16202ae794f5c99ef10 subsubdir1
blob be17eaddd5e696239e8a5f6e6e3dbce524707983 file4

Eliminating Duplication

```
tree 8aa73a2b1cf458c7c88dc16202ae794f5c99ef10 subsubdir1
blob be17eaddd5e696239e8a5f6e6e3dbce524707983 file4
```



Eliminating Duplication



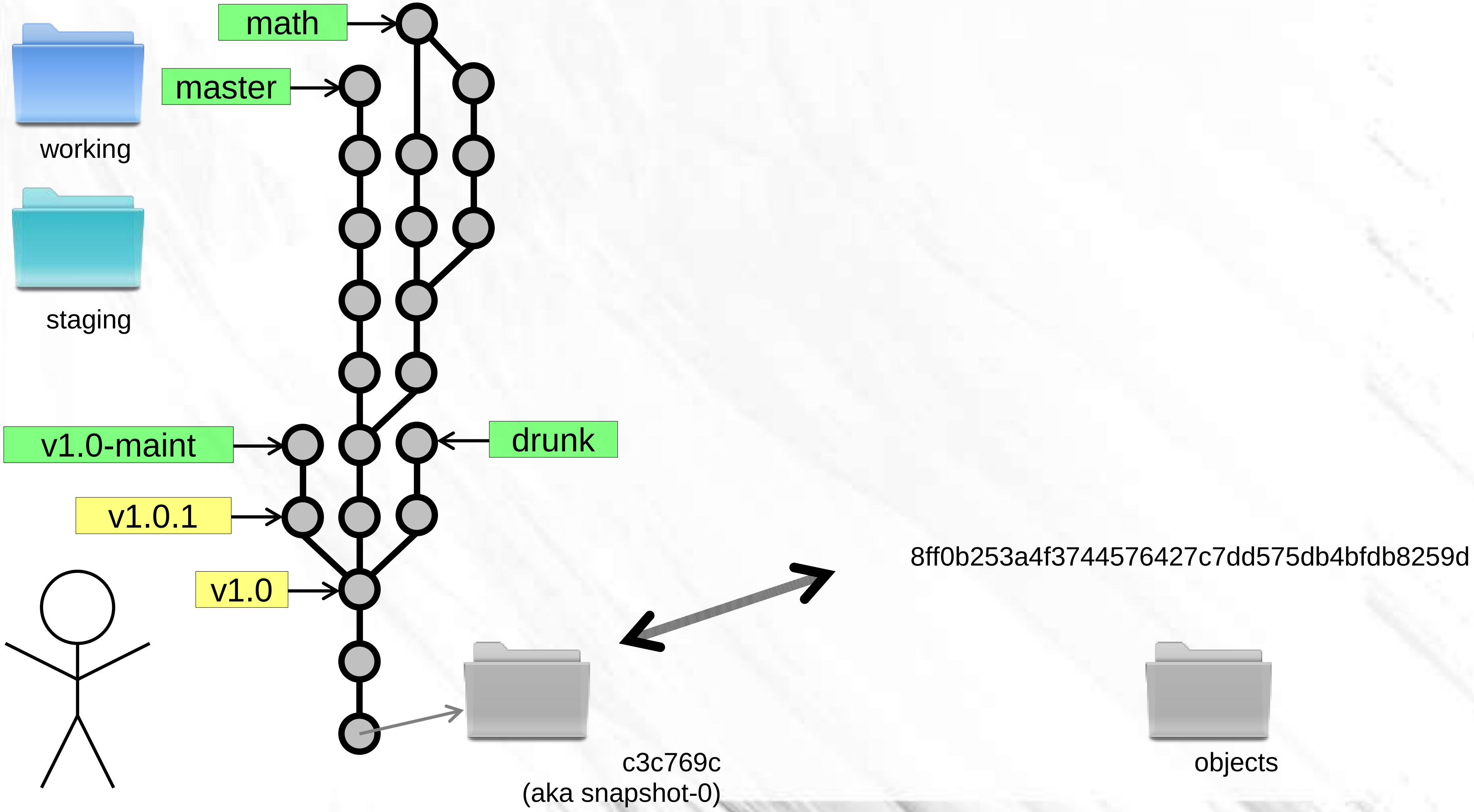
tree 2804133755c3ed396d162028c7b30a1cbcfecded subdir1
tree 67a3c2af119a3c3c769c0b1e8735fa5d28d94a12 subdir2

New temp file

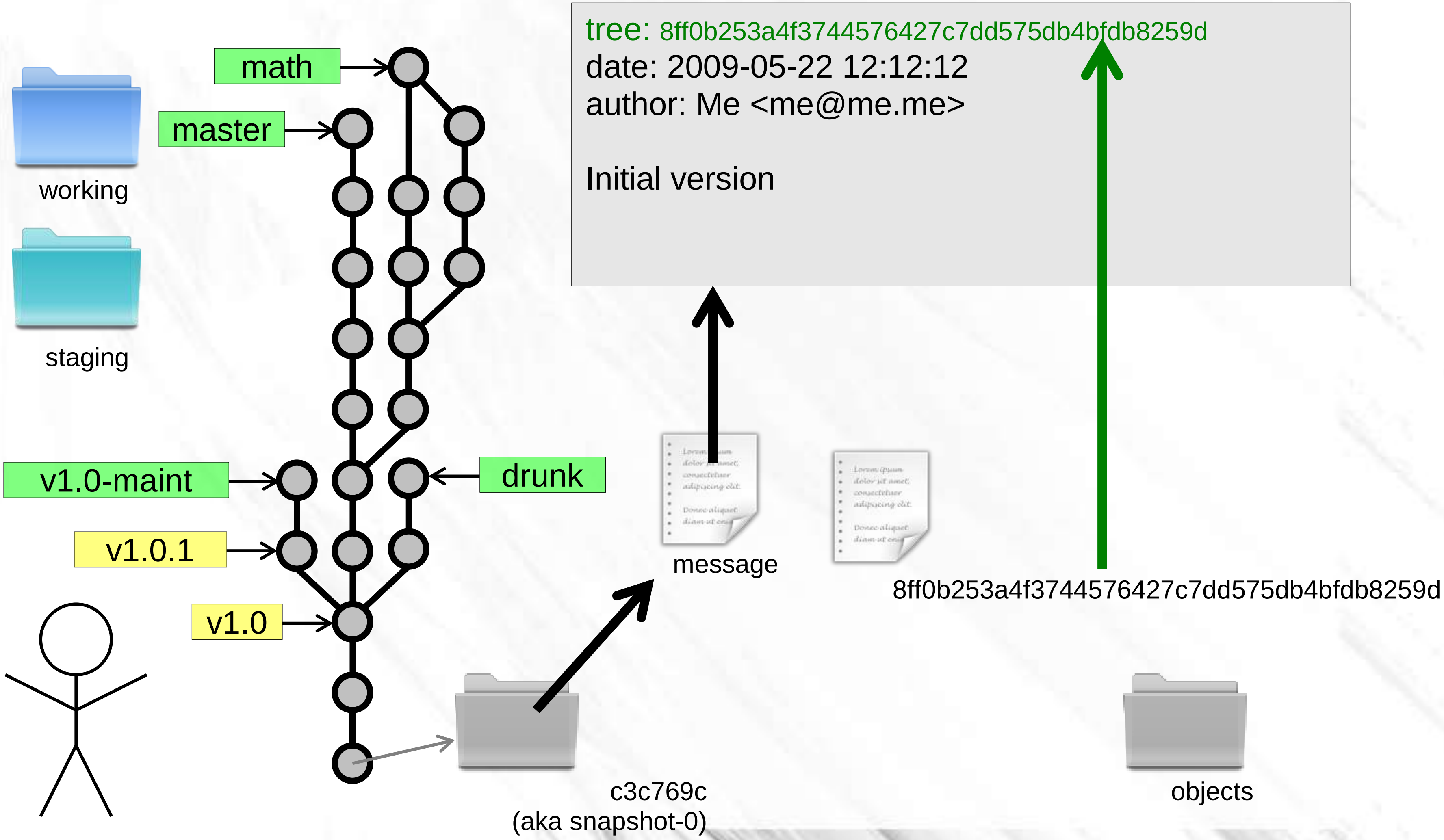
SHA1

8ff0b253a4f3744576427c7dd575db4bfdb8259d

Eliminating Duplication



Eliminating Duplication



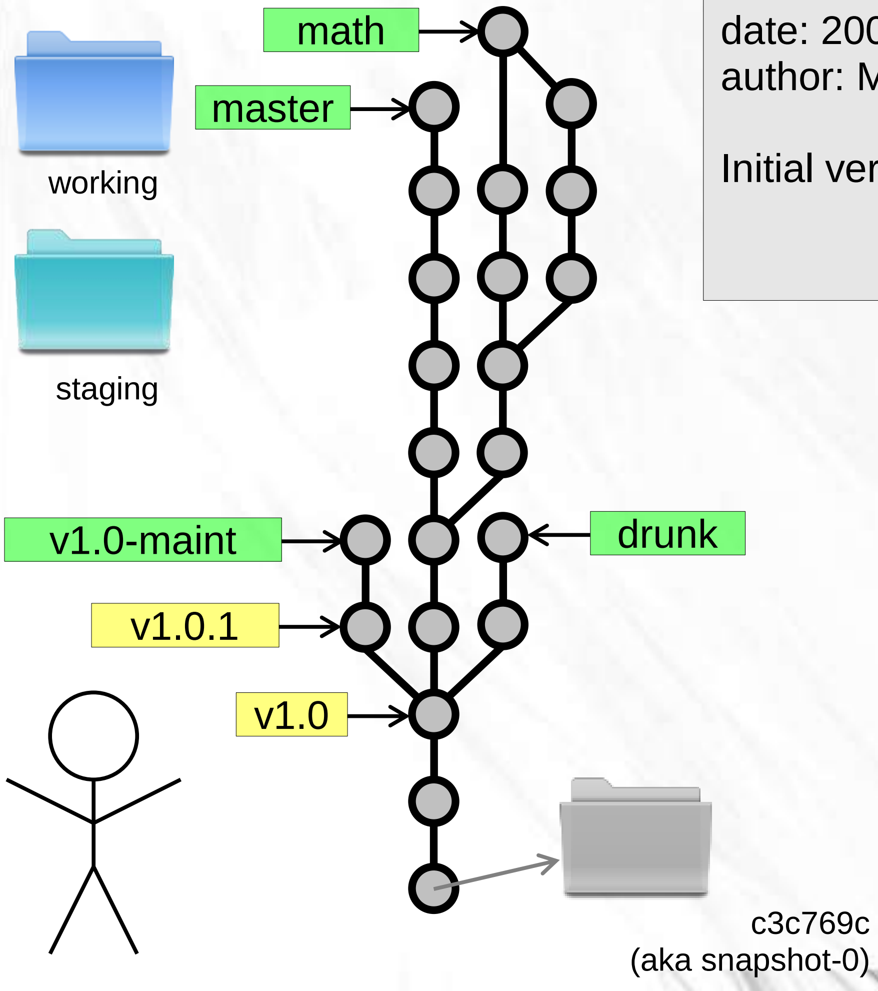
tree: 8ff0b253a4f3744576427c7dd575db4bfdb8259d
date: 2009-05-22 12:12:12
author: Me <me@me.me>

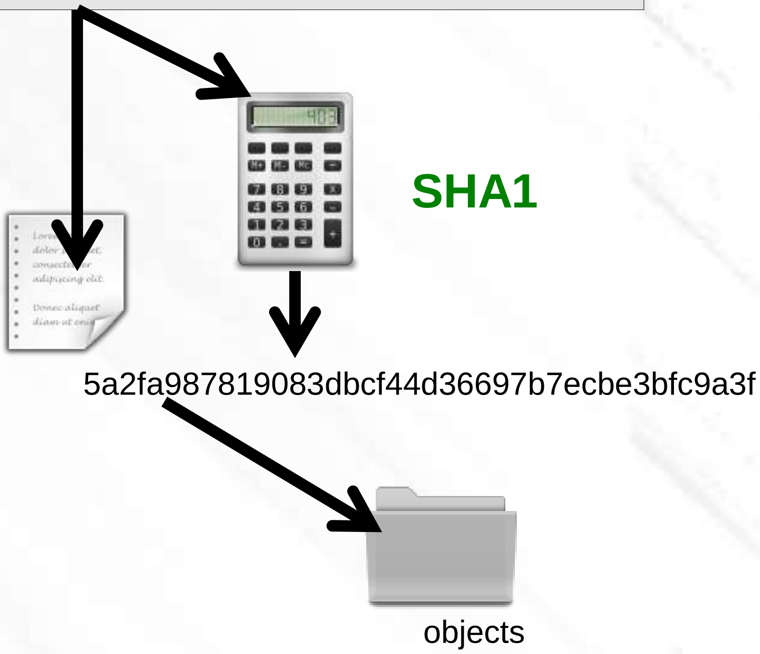
Initial version

Eliminating Duplication

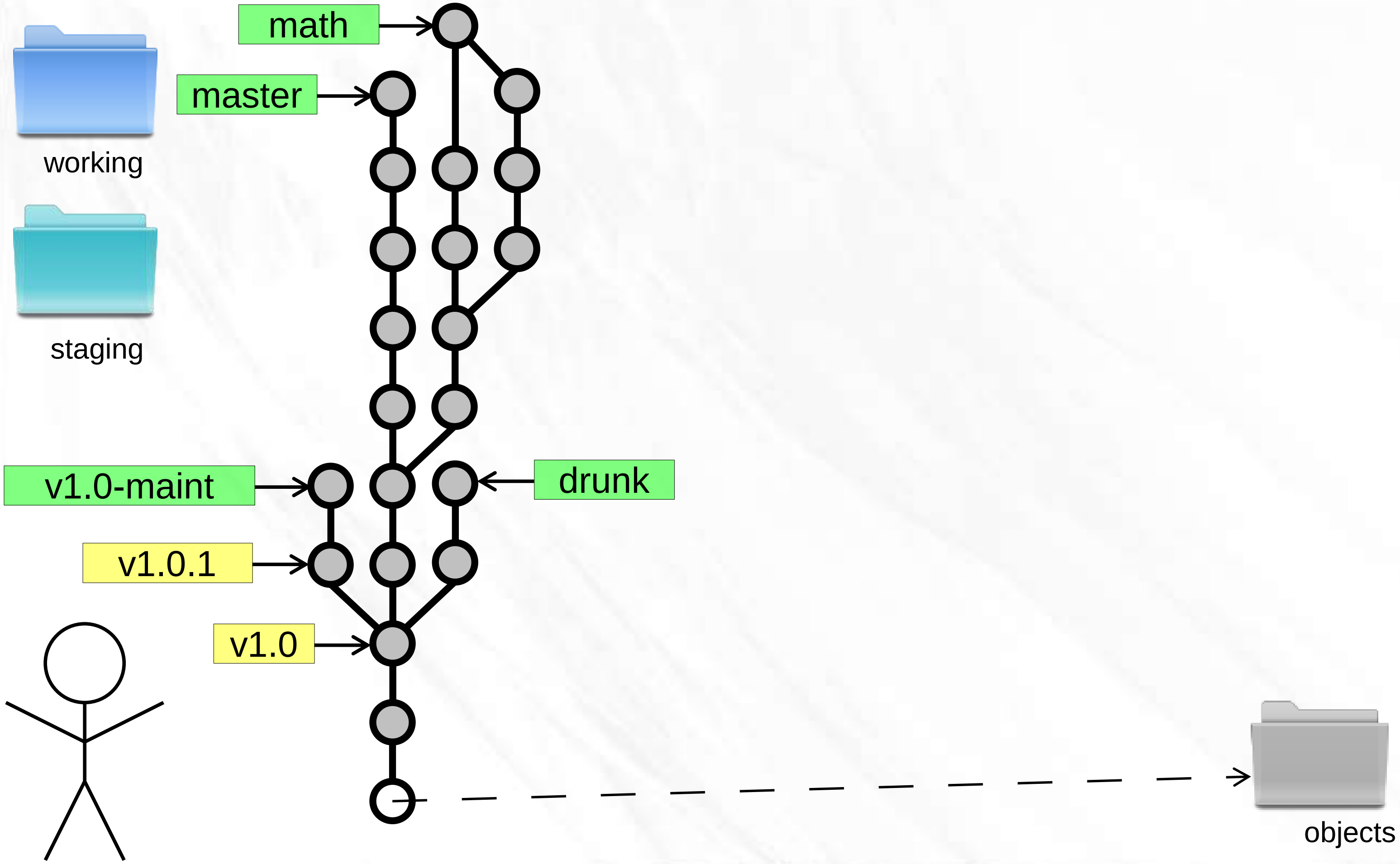
tree: 8ff0b253a4f3744576427c7dd575db4bfdb8259d
date: 2009-05-22 12:12:12
author: Me <me@me.me>

Initial version

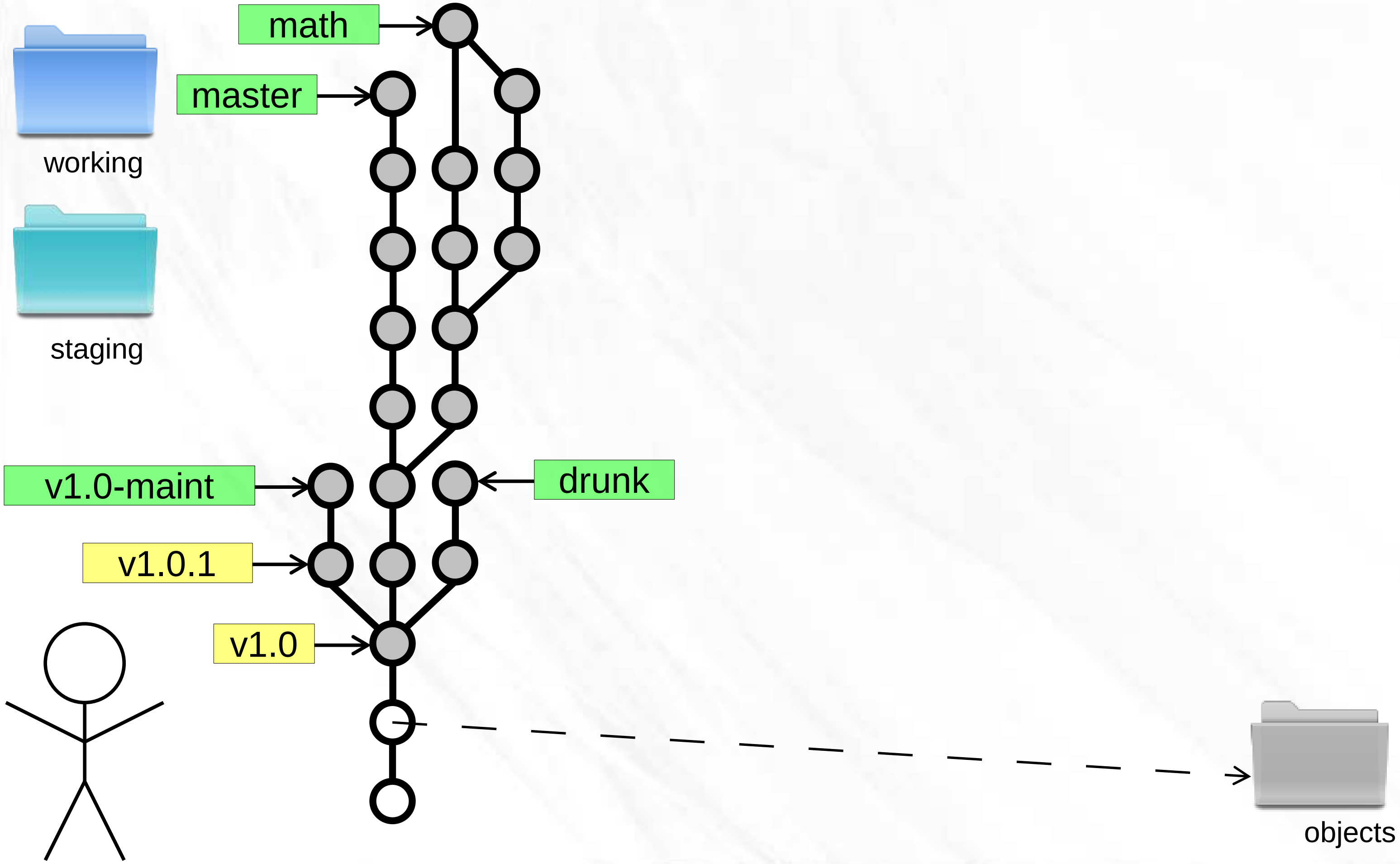




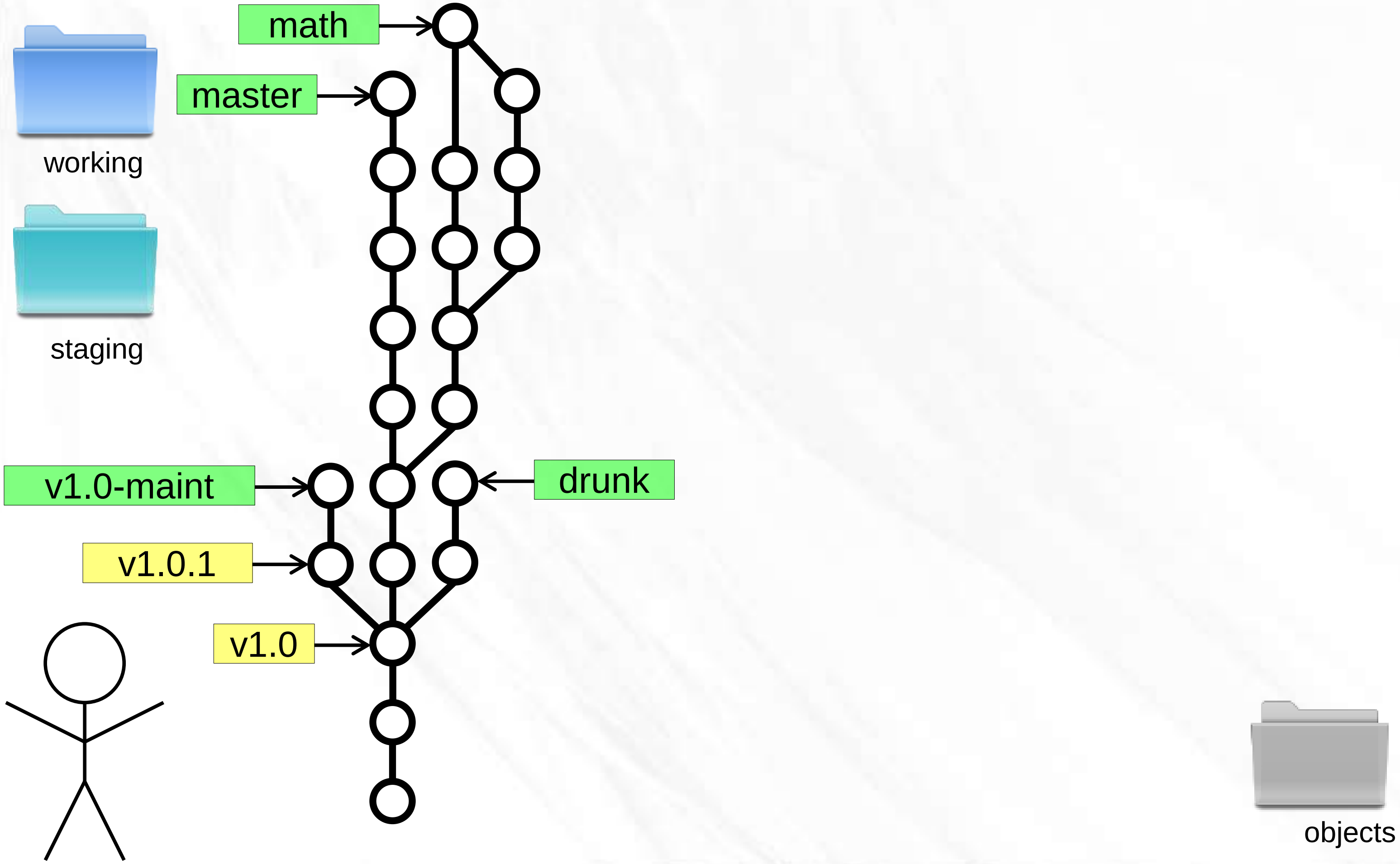
Eliminating Duplication



Eliminating Duplication



Eliminating Duplication



math

master

working

staging

v1.0-maint

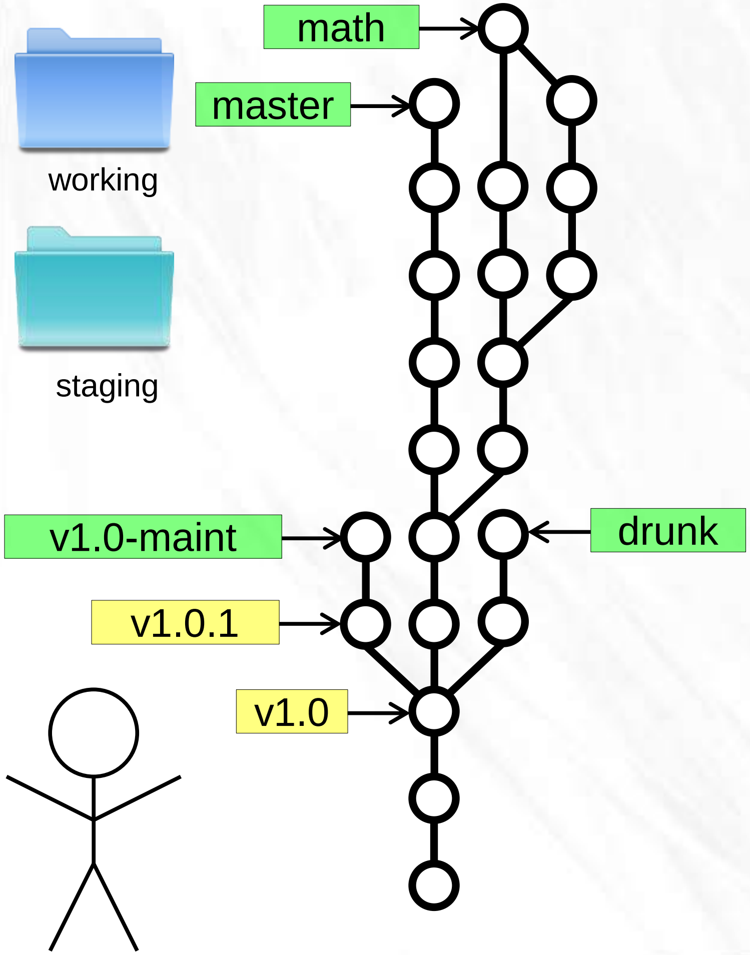
drunk

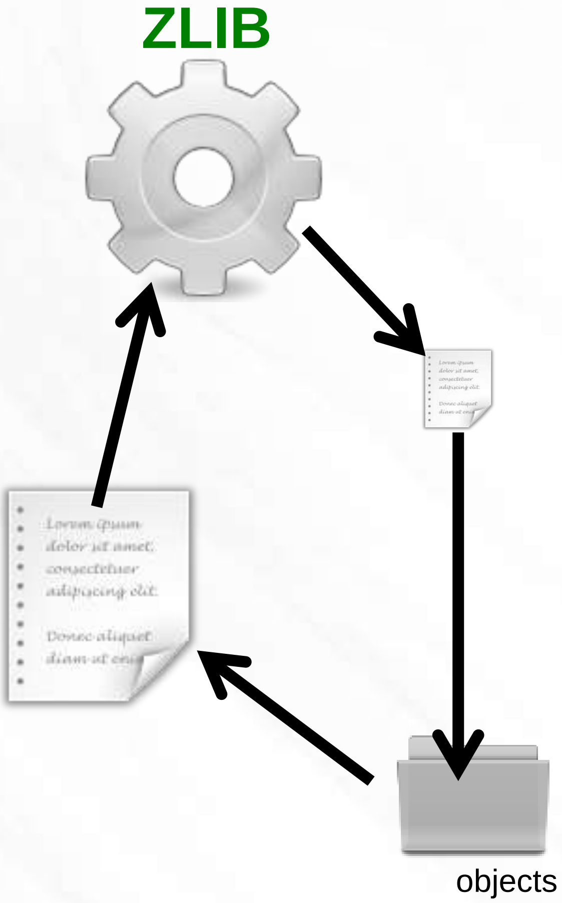
v1.0.1

v1.0

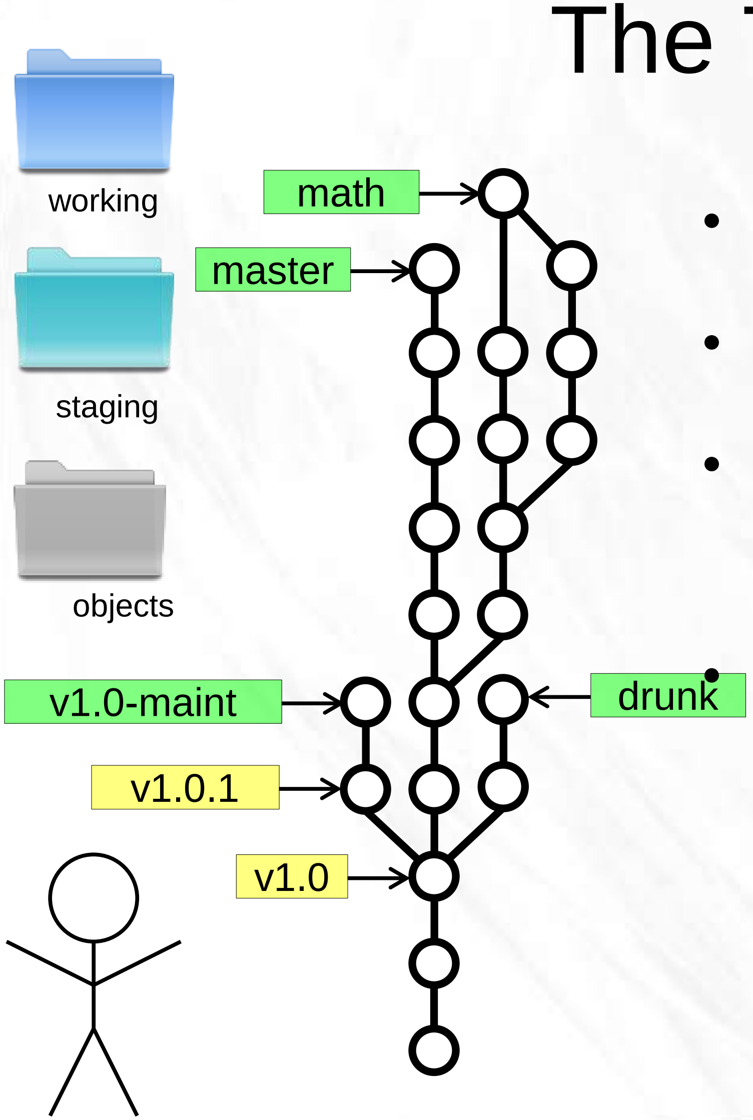
objects

Compressing Blobs





The True Git

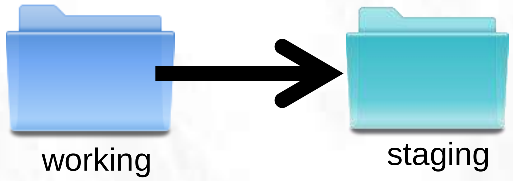


- TADAA!
- This is pretty much Git
- Nicer command line tools for all these operations
- Many, many other tools

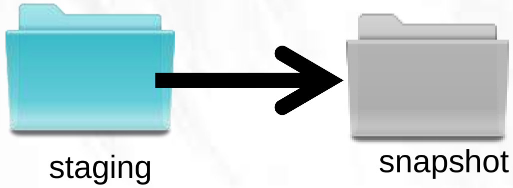
Commands: Getting Started

- First, tell Git who you are:
  - git config --global user.name “My Name”
  - git config --global user.email “my@email.address”
- Get help:
  - git <command> -h
  - git help <command>
- Start a new Git repository:
  - git init

Commands: Making snapshots

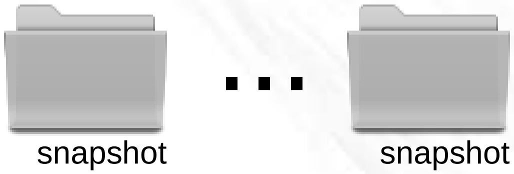


- git add

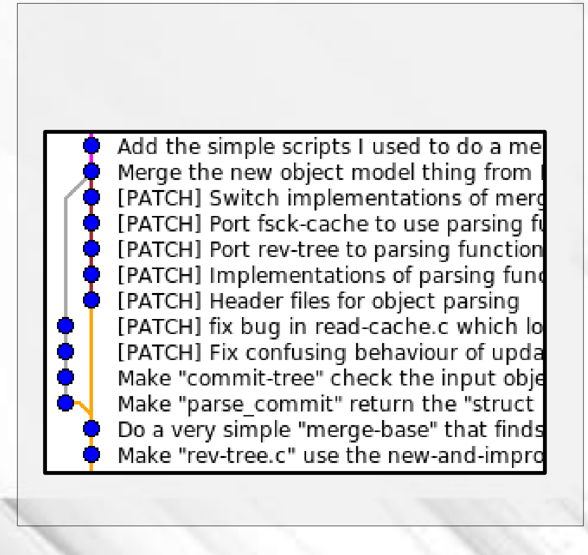


- git commit

} git commit -a

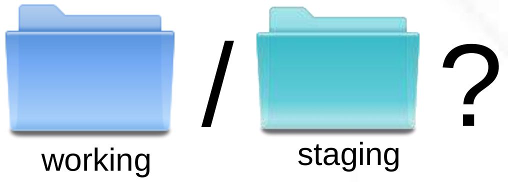


- git log



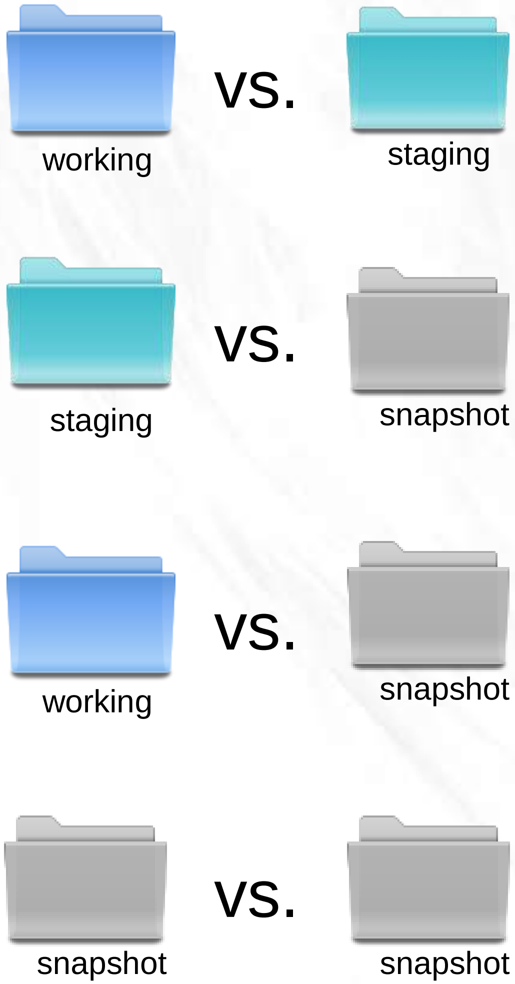
```
Add the simple scripts I used to do a me
Merge the new object model thing from I
[PATCH] Switch implementations of merg
[PATCH] Port fsck-cache to use parsing fu
[PATCH] Port rev-tree to parsing function
[PATCH] Implementations of parsing func
[PATCH] Header files for object parsing
[PATCH] fix bug in read-cache.c which lo
[PATCH] Fix confusing behaviour of upda
Make "commit-tree" check the input obje
Make "parse_commit" return the "struct
Do a very simple "merge-base" that finds
Make "rev-tree.c" use the new-and-impro
```

gitk



- git status

Commands: Diffing



- git diff
- git diff --staged
- git diff HEAD
- git diff <from> <to>

Commands: Branches & Tags

- git branch
- git branch <branch>
- git checkout <branch>
- git tag -l
- git tag <tag>

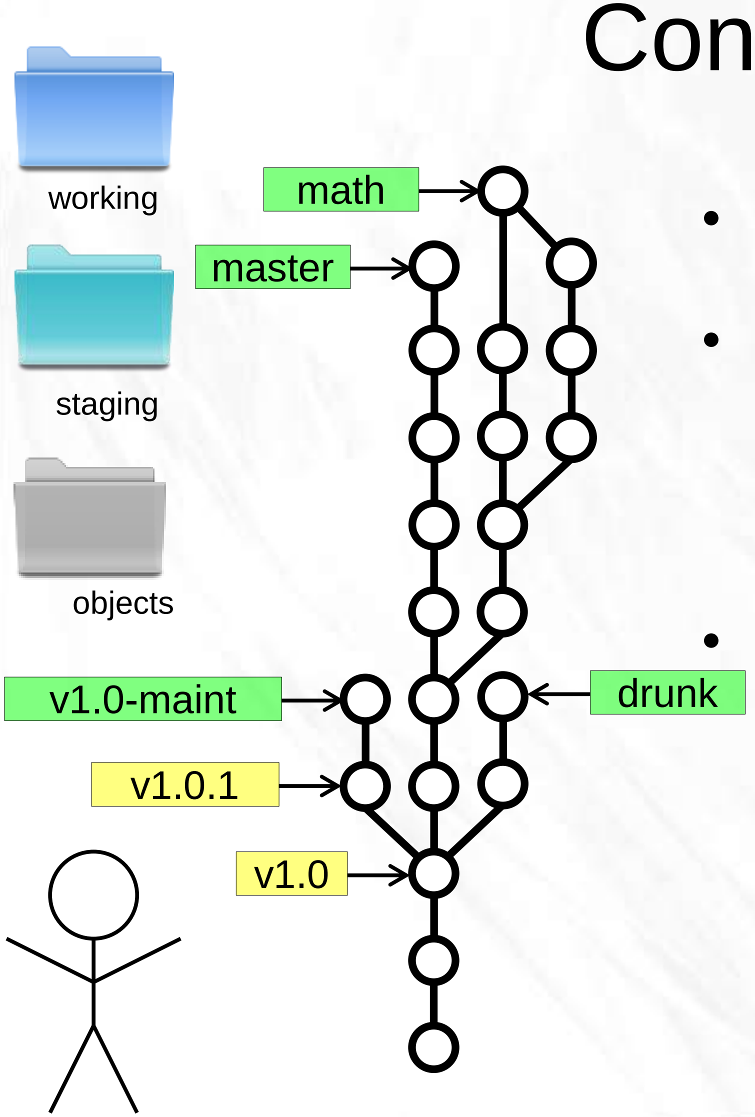
} git checkout -b ...

Commands: Fetching & Merging

- git remote add <name> <URL>
- git fetch <name>
- git merge <name>/<branch>

} git pull

Conclusion



- Keep this parable in mind
- Git is simple and powerful

- One more thing:

git reflog

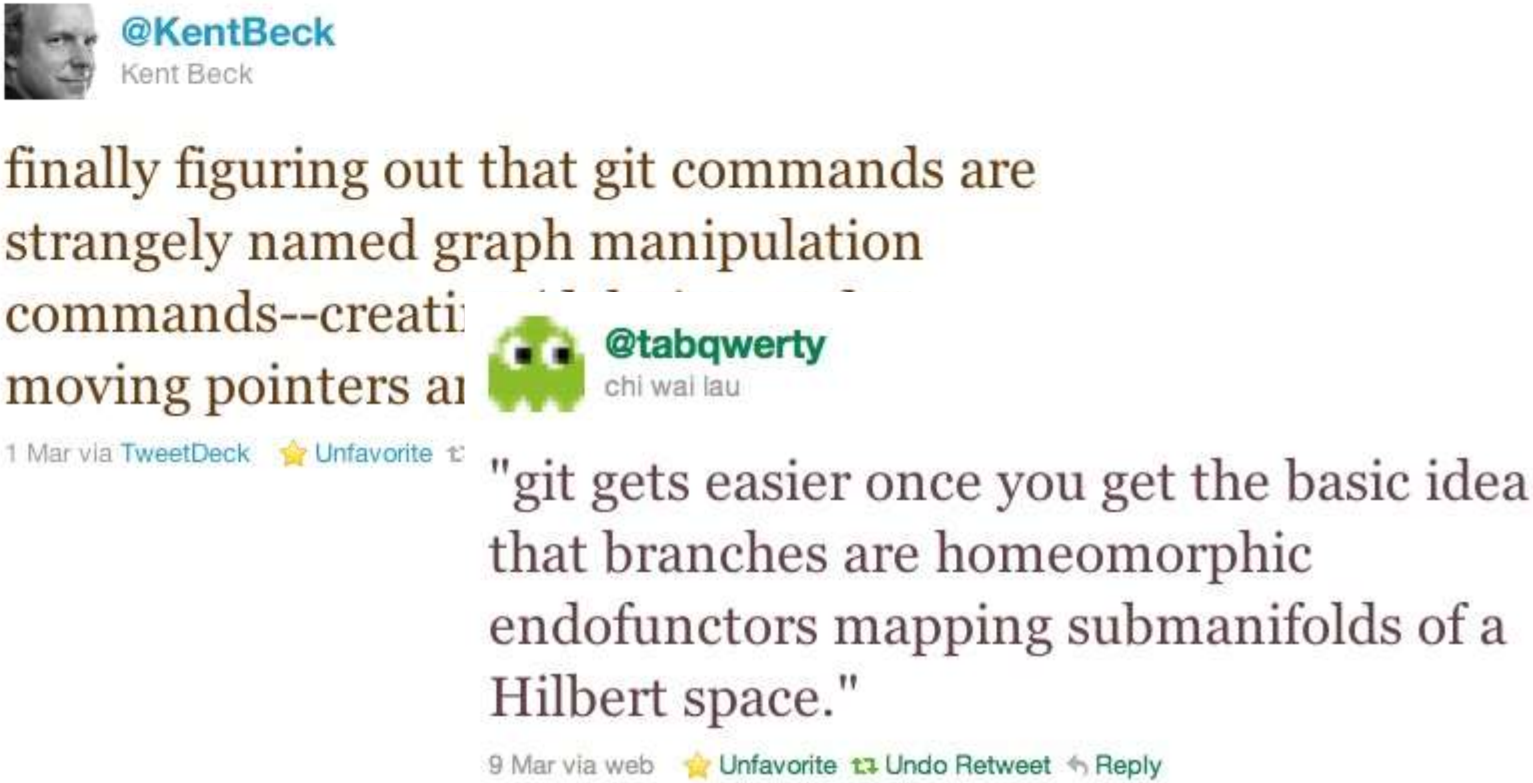
Where to go next?

- Git homepage: http://git-scm.com
- Pro Git: http://git-scm.com/book
- Git Reference: http://gitref.org
- GitHub: http://github.com
- Gitorious: http://gitorious.org

Questions?

- Thanks for your attention!
- These slides are available at:
  https://github.com/jherland/git_parable
- Reach me at <johan@herland.net>

Git makes more sense when you understand [...]



@KentBeck
Kent Beck

finally figuring out that git commands are strangely named graph manipulation commands--creati
moving pointers a

1 Mar via TweetDeck ☆ Unfavorite

@tabqwerty
chi wai lau

"git gets easier once you get the basic idea that branches are homeomorphic endofunctors mapping submanifolds of a Hilbert space."

9 Mar via web ☆ Unfavorite Undo Retweet Reply

Git clients (Windows)

- CLI Shell: Git Bash
- Windows Explorer Shell
- Github for Windows (+powershell)
- IDE Integration
  - Visual Studio (VS2012U2+ native)
  - Eclipse Egit

הדגמה- טיפים

- http://learn.github.com/, http://gitimmersion.com/, etc.
- http://help.github.com/create-a-repo/
  Local user settings:
  git config user.name <user>
  git config user.email user@example.com
- git pull
- git add: add / stage
- git commit –a == add+commit

Git Flow

- במיוחד חשוב בקוד פתוח
- Git (hub) flow
- דוגמא: תהליך העבודה בפרויקט Nuget:
Contributing a Bug Fix or Feature
- ראו ויקי

בפעם הבאה

- פרויקט: פיתוח בסבבים
- בדיקות תוכנה
  - בקרת גרסאות II – תרחישים נוספים עם git ?
- משימה אישית מס' 2:
  קורס git מקוון קצר try.github.io
  להגשה: קישור לתגית סיום הקורס (דוגמא)
  אפשר להגיע אל התגית ע"י כניסה לאתר codeschool עם חשבון github->
  My Report Card <- Make Public <- Earned Badges ולחיצה על התגית

לסיכום

- בקרת גרסאות