



Getting distributed without losing your HEAD



CERN Spring Campus 2014

Gijón, Spain - 16th April 2014

[Benjamin Wolff](#) (CERN / GS-AIS)

# Agenda

---

# Agenda

---

1. What is Git?

# Agenda

---

1. What is Git?
2. Basic Concepts

# Agenda

---

1. What is Git?
2. Basic Concepts
3. Branching

# Agenda

---

1. What is Git?
2. Basic Concepts
3. Branching
4. Merging

# Agenda

---

1. What is Git?
2. Basic Concepts
3. Branching
4. Merging
5. Remotes

# What is Git?

---

Git is a free and open source, distributed version control system designed to handle everything from small to very large projects with speed and efficiency.

# What is Git?

---

Git is a **free and open source**, distributed version control system designed to handle everything from small to very large projects with speed and efficiency.

# What is Git?

---

Git is a free and open source, **distributed version control** system designed to handle everything from small to very large projects with speed and efficiency.

# What is Git?

---

Git is a free and open source, distributed version control system designed to handle everything from **small to very large** projects with speed and efficiency.

# What is Git?

---

Git is a free and open source, distributed version control system designed to handle everything from small to very large projects with **speed and efficiency**.

# What is Git?

---



# Some Facts

---

# Some Facts

---

- Created in **2005** by **Linus Torvalds**

# Some Facts

---

- Created in **2005** by **Linus Torvalds**
- Conceived as VCS system for the **Linux kernel**

# Some Facts

---

- Created in **2005** by **Linus Torvalds**
- Conceived as VCS system for the **Linux kernel**
- Strong support for **non-linear development**

# Some Facts

---

- Created in **2005** by **Linus Torvalds**
- Conceived as VCS system for the **Linux kernel**
- Strong support for **non-linear development**
- Almost all operations **run locally**

# Some Facts

---

- Created in **2005** by **Linus Torvalds**
- Conceived as VCS system for the **Linux kernel**
- Strong support for **non-linear development**
- Almost all operations **run locally**
- **Different** concepts: CVS, SVN, Microsoft TFS

# Some Facts

---

- Created in **2005** by **Linus Torvalds**
- Conceived as VCS system for the **Linux kernel**
- Strong support for **non-linear development**
- Almost all operations **run locally**
- **Different** concepts: CVS, SVN, Microsoft TFS
- **Similar** concepts: Mercurial, Bazaar, BitKeeper

# Some Facts

---

# Some Facts

---

- De facto a **standard** in open source

# Some Facts

---

- De facto a **standard** in open source
- Widely adopted in the **industry**

# Some Facts

---

- De facto a **standard** in open source
- Widely adopted in the **industry**
- Skills in IT Jobs demand: **itjobswatch.co.uk**

Category „Developments Application UK“

	2012	2014	Trend
SVN	20.06%	16.33%	-18.59%
Git	7.21%	16.88%	+234.12%

# What does Git mean?

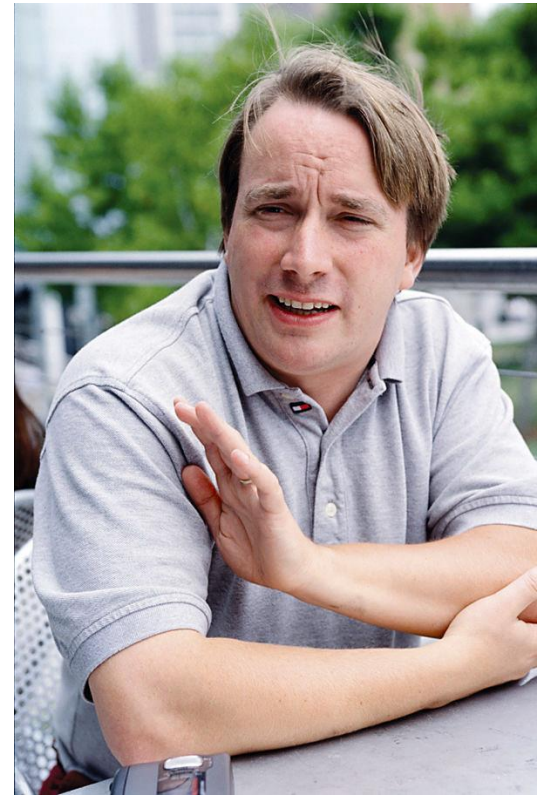
---

# What does Git mean?

---

*“I'm an egotistical bastard, and I name all my projects after myself. First Linux, now **git**.”*

- Linus Torvalds, 2007



# Basic Concepts

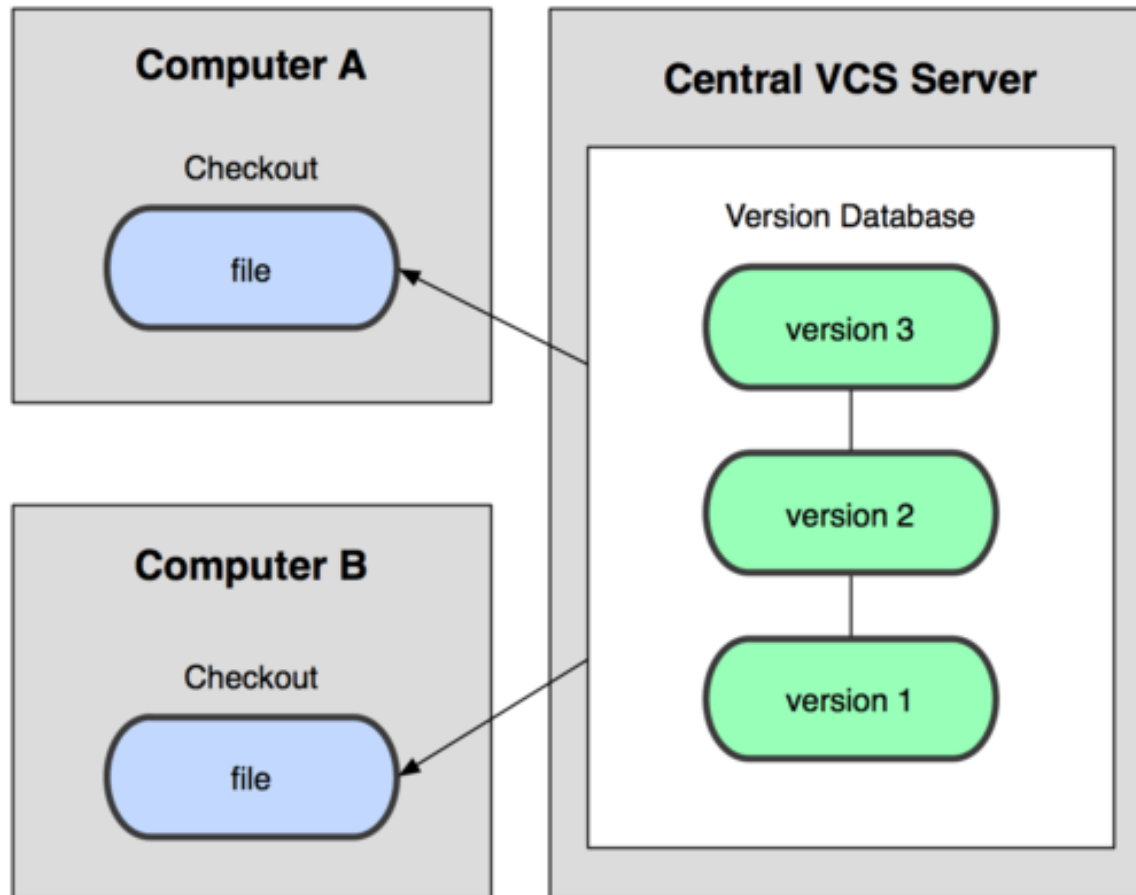
---

# Central vs. Distributed

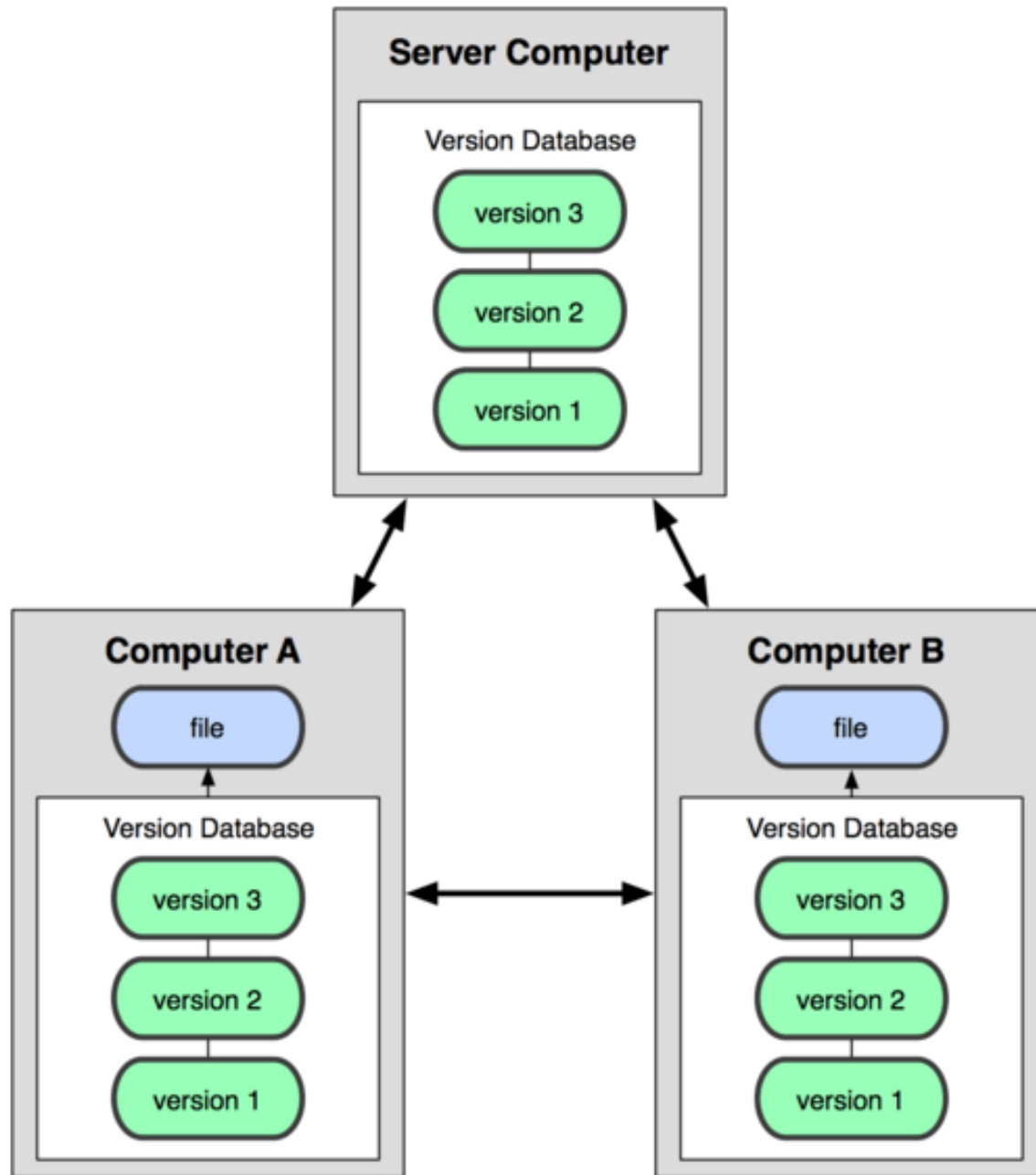
---

# Central vs. Distributed

---



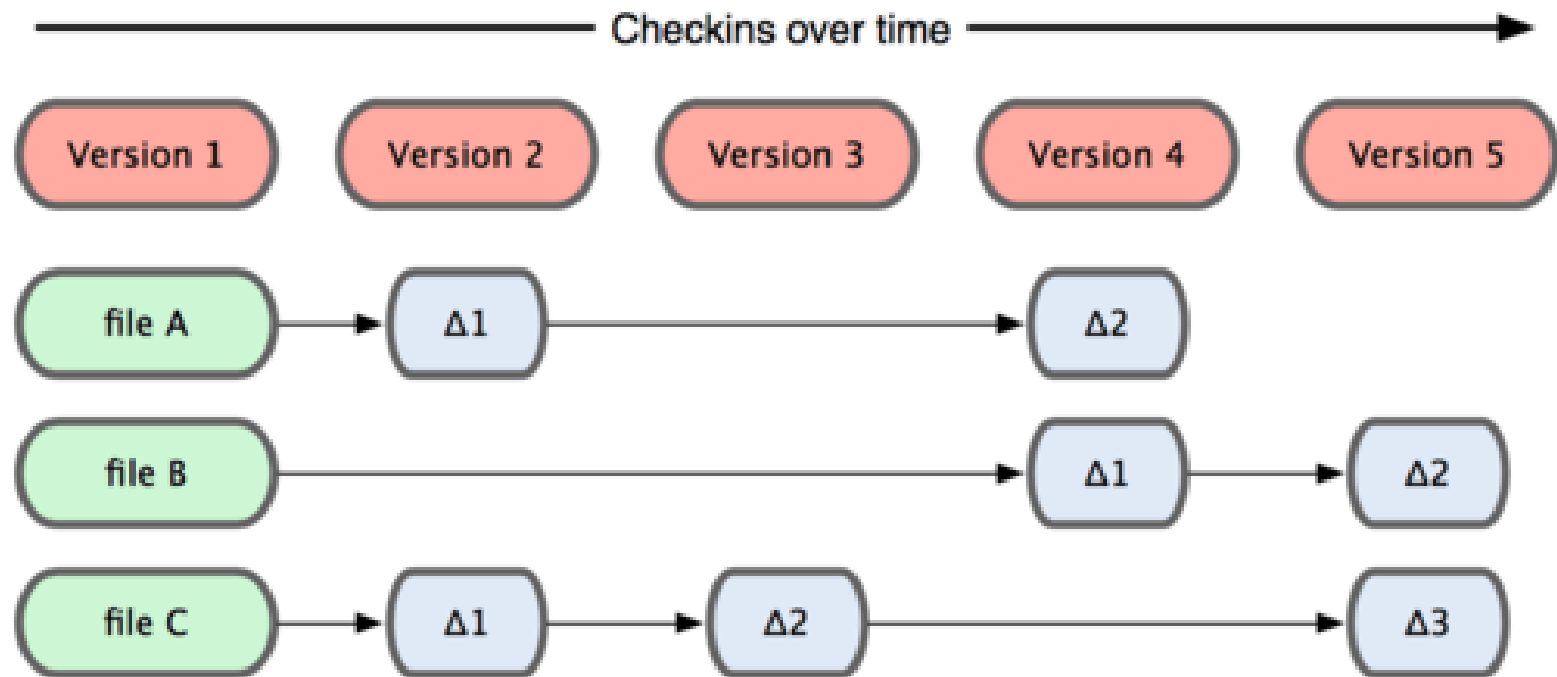




# Diffs vs. Snapshots

---

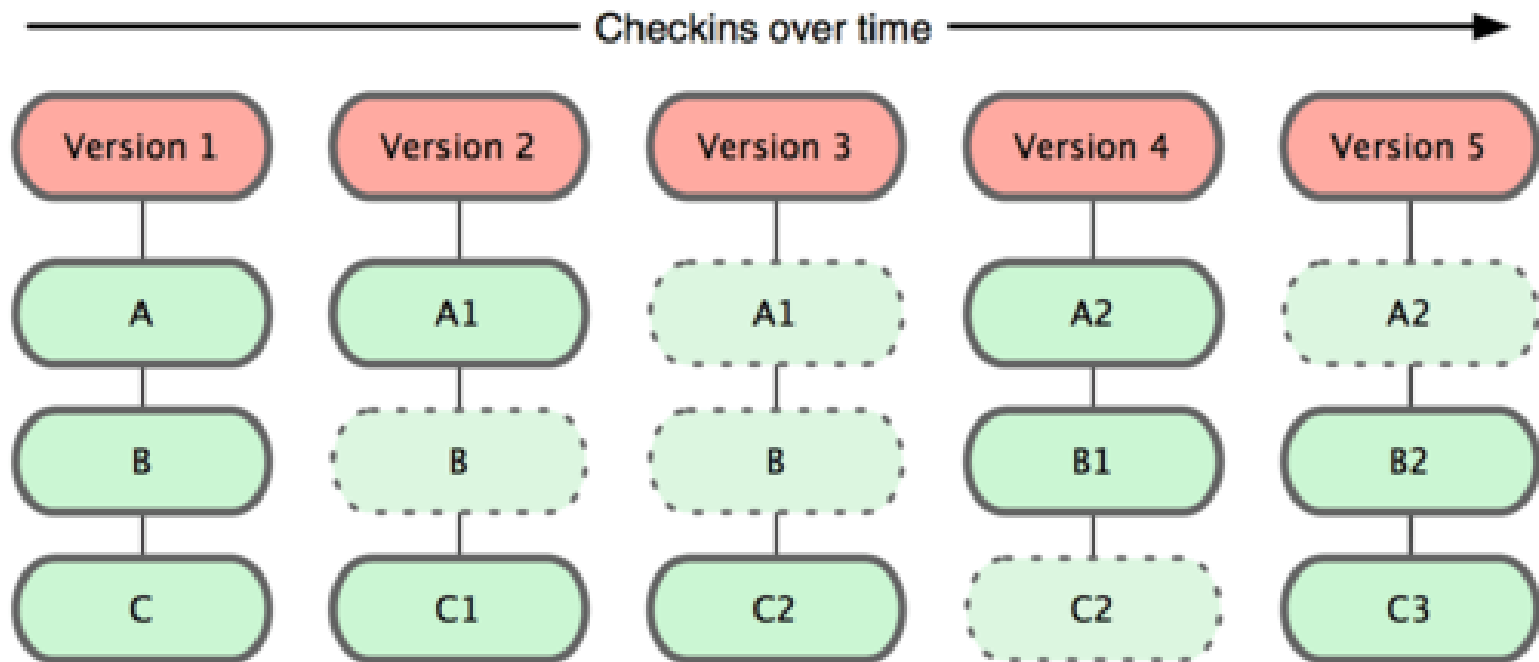
# Diffs vs. Snapshots



# Diffs vs. **Snapshots**

---

# Diffs vs. Snapshots



# Git Repository

---

# Git Repository

---

- Contains the Git **database, branch information, configuration, etc.**

# Git Repository

---

- Contains the Git **database, branch information, configuration**, etc.
- Located in a hidden **.git** folder in the **project root**

# Git Repository

---

- Contains the Git **database, branch information, configuration**, etc.
- Located in a hidden **.git** folder in the **project root**
- Creating a new repository:  
`git init`

# Git Repository

---

- Contains the Git **database, branch information, configuration**, etc.
- Located in a hidden **.git** folder in the **project root**
- Creating a new repository:  
`git init`
- More common:  
`git clone https://mycompany/myproject.git`

# The Staging Area

---

# The Staging Area

---

Working Directory

# The Staging Area

---

Working Directory

Repository

# The Staging Area

---

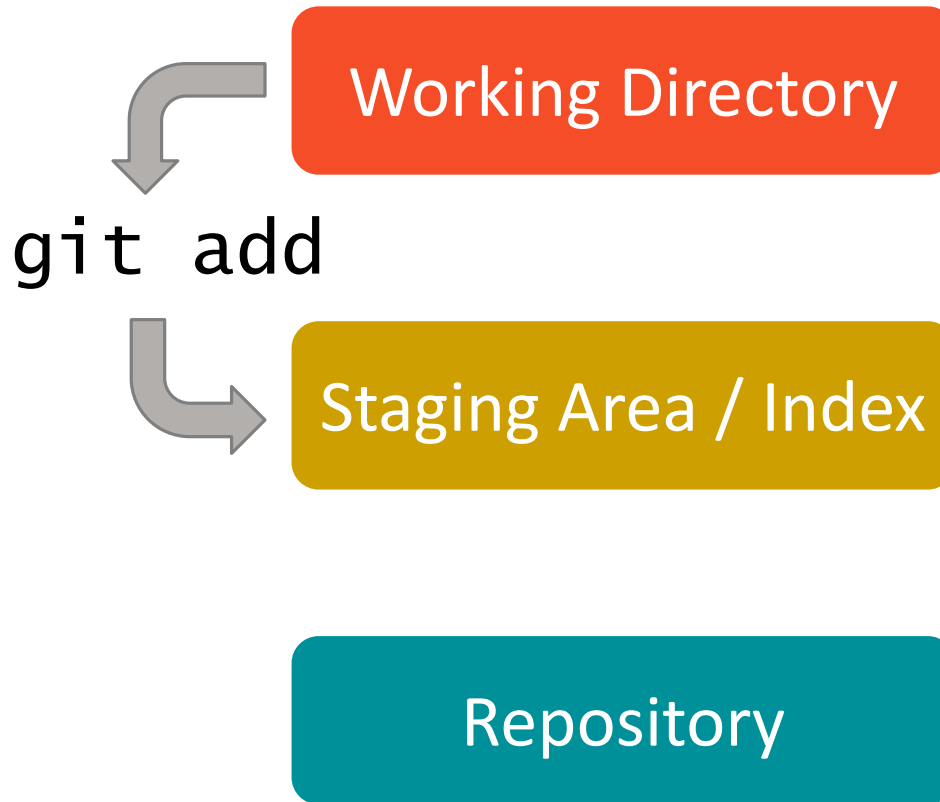
Working Directory

Staging Area / Index

Repository

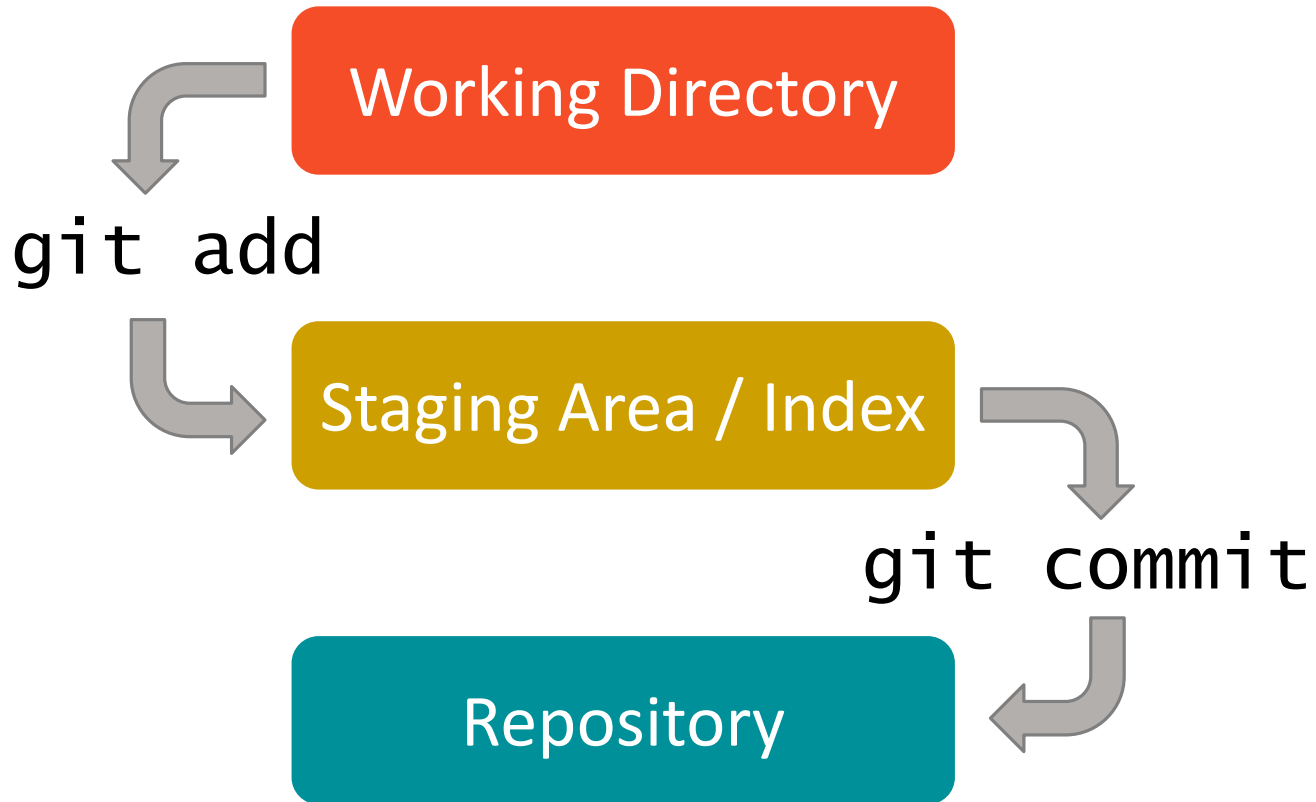
# The Staging Area

---



# The Staging Area

---



# The Staging Area

---

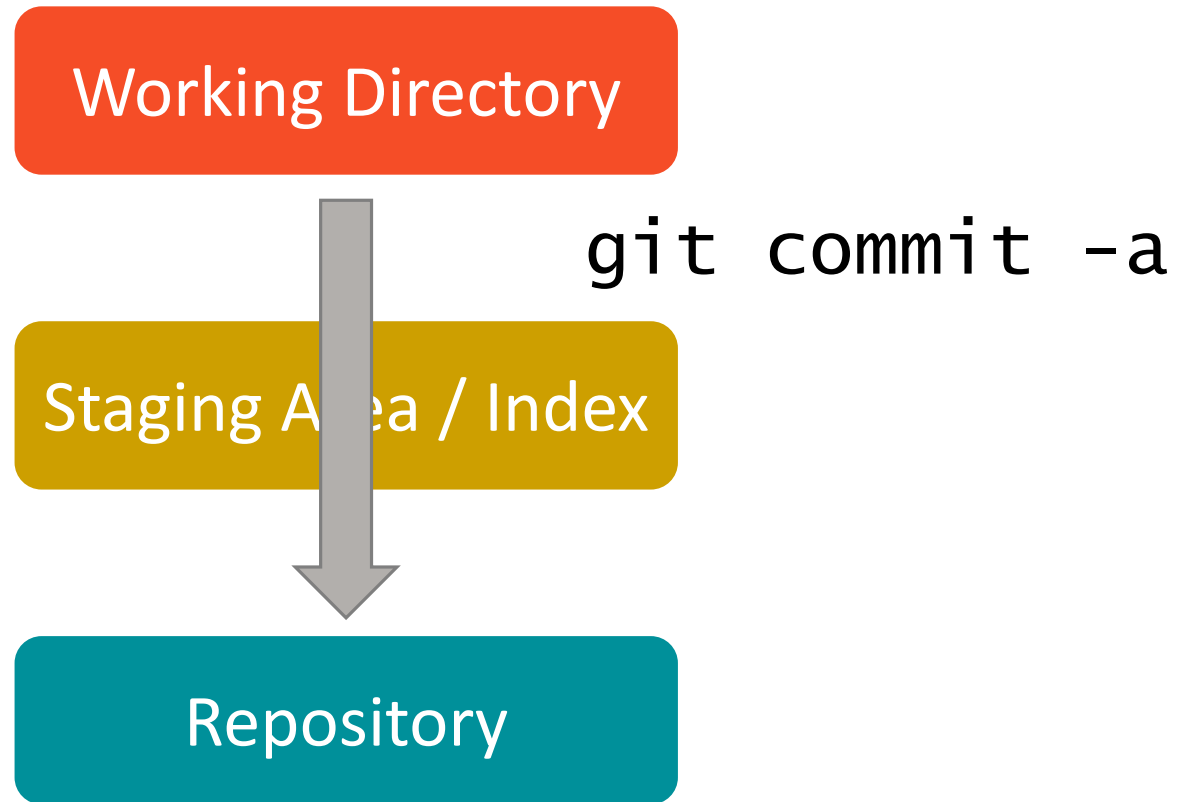
Working Directory

Staging Area / Index

Repository

# The Staging Area

---



# The Staging Area

---

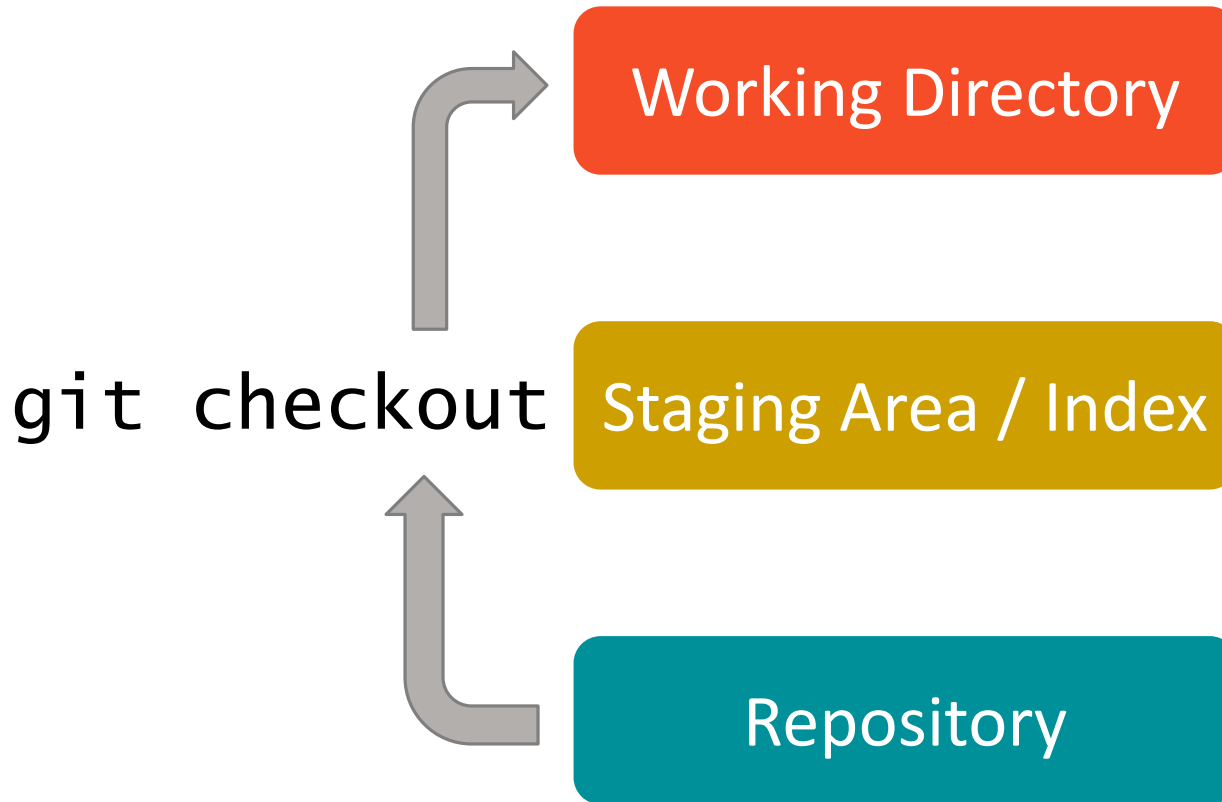
Working Directory

Staging Area / Index

Repository

# The Staging Area

---



# The Staging Area

---

Working Directory

Staging Area / Index

Repository

# The Staging Area

---

Working Directory

Staging Area / Index

Repository

`git status`

# The Staging Area

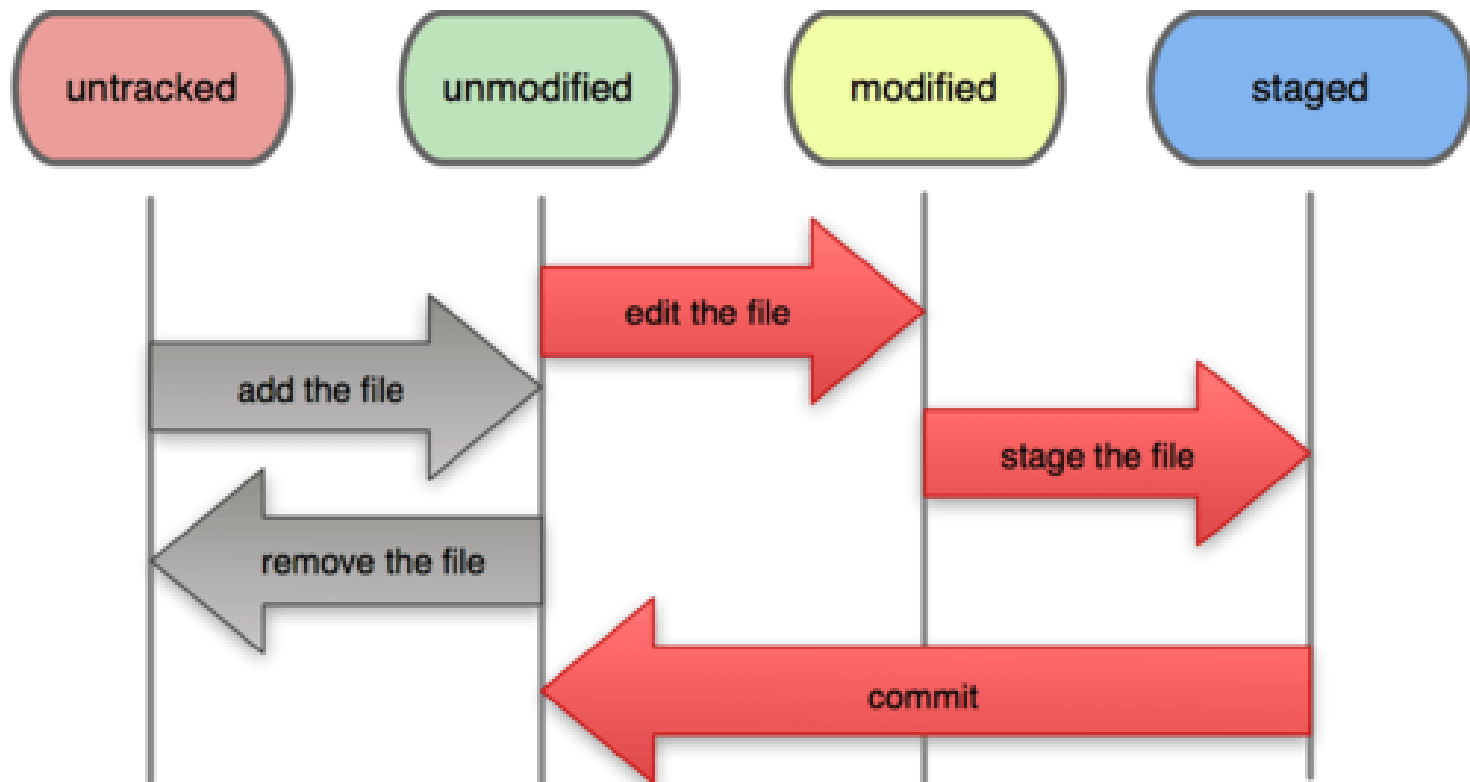
```
# On branch master
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
#       modified:   index.html
#       new file:   version.html
#
# Changes not staged for commit:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes in working directory)
#
#       modified:   error.html
#       modified:   help.html
#
# Untracked files:
#   (use "git add <file>..." to include in what will be committed)
#
#       about.html
#       css/
#       js/
```

`git status`

# File Status Lifecycle

---

# File Status Lifecycle



# Commits

---

# Commits

---

- Pointer to a **snapshot** (ref)

# Commits

---

- Pointer to a **snapshot** (ref)
- Stored as **objects** with **metadata**

# Commits

---

- Pointer to a **snapshot** (ref)
- Stored as **objects** with **metadata**
- Identified by a **SHA-1** hash

# Commits

---

- Pointer to a **snapshot** (ref)
- Stored as **objects** with **metadata**
- Identified by a **SHA-1 hash**
  - 24b9da6552252987aa493b52f8696cd6d3b00373

# Commits

---

- Pointer to a **snapshot** (ref)
- Stored as **objects** with **metadata**
- Identified by a **SHA-1 hash**
  - 24b9da6552252987aa493b52f8696cd6d3b00373
- Provide **cryptographic integrity** of the history

# Commits

---

- Pointer to a **snapshot** (ref)
- Stored as **objects** with **metadata**
- Identified by a **SHA-1 hash**
  - 24b9da6552252987aa493b52f8696cd6d3b00373
- Provide **cryptographic integrity** of the history
- Comparable to **revisions** in SVN

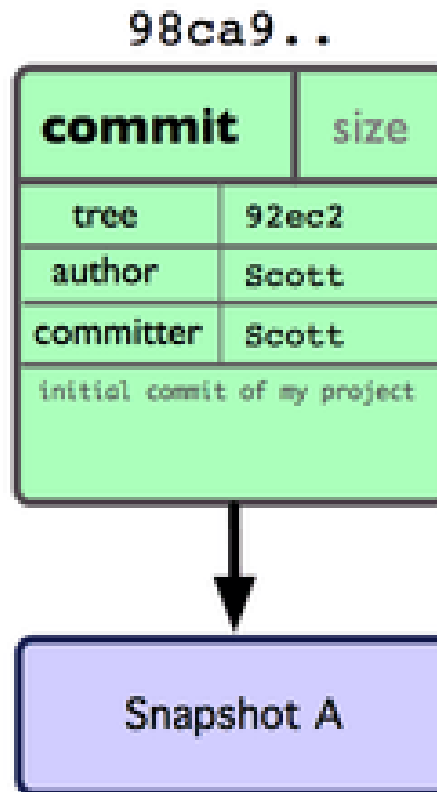
# Commits

---

- Pointer to a **snapshot** (ref)
- Stored as **objects** with **metadata**
- Identified by a **SHA-1 hash**
  - 24b9da6552252987aa493b52f8696cd6d3b00373
- Provide **cryptographic integrity** of the history
- Comparable to **revisions** in SVN
- Form a **Directed Acyclic Graph**, the full project history

# Commit Objects

---



# Commit Objects

---

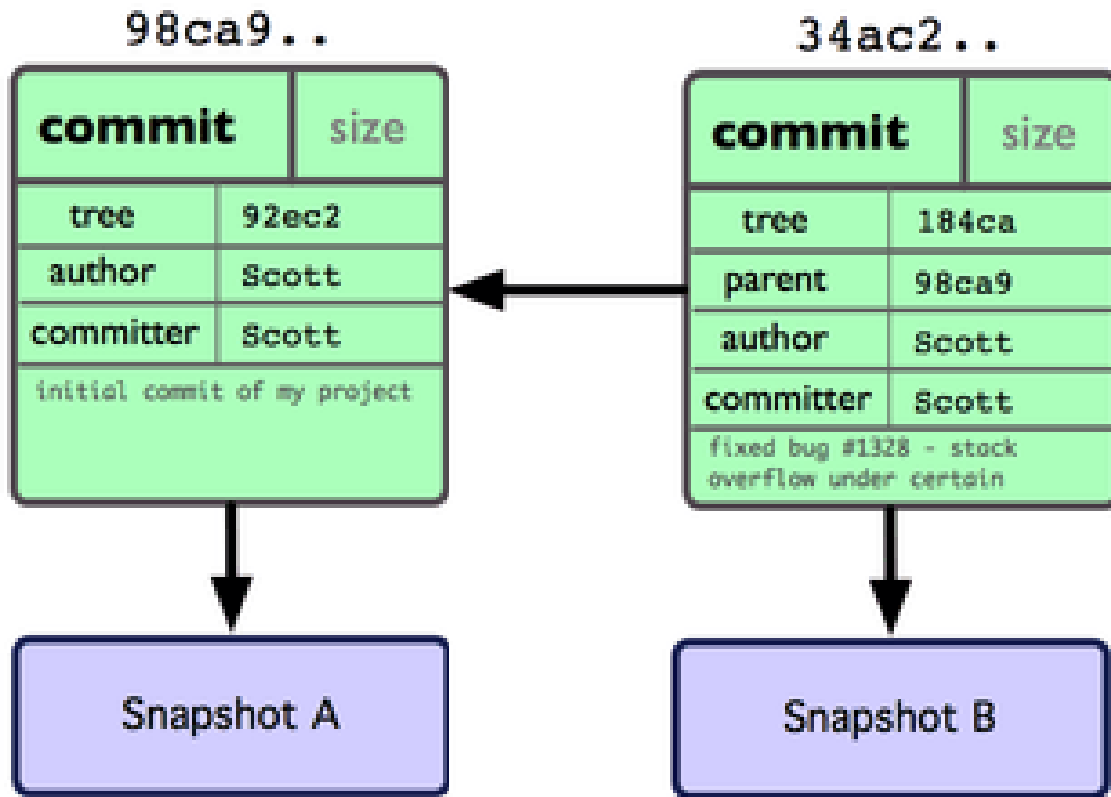
98ca9..

<b>commit</b>		size
tree	92ec2	
author	Scott	
committer	Scott	
initial commit of my project		

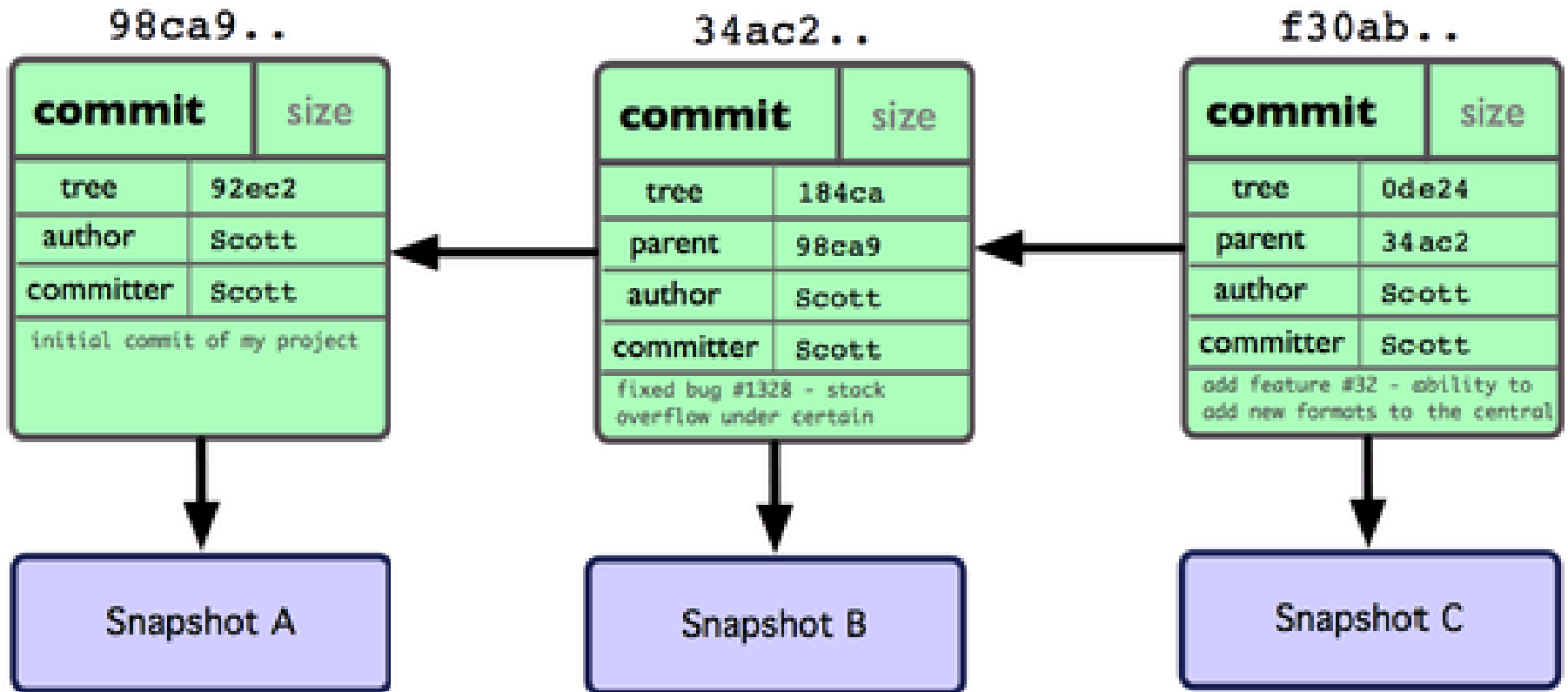


Snapshot A

# Commit Objects



# Commit Objects



# Tagging

---

# Tagging

---

- Constant pointer to a **specific commit**

# Tagging

---

- Constant pointer to a **specific commit**
- Often used to **mark a snapshot** of a **release** (v1.0)

# Tagging

---

- Constant pointer to a **specific commit**
- Often used to **mark a snapshot** of a **release** (v1.0)
- **Additional metadata** like creator, date, description

# Tagging

---

- Constant pointer to a **specific commit**
- Often used to **mark a snapshot** of a **release** (v1.0)
- **Additional metadata** like creator, date, description
- Can be **cryptographically signed** using GPG

# Tagging

---

- Constant pointer to a **specific commit**
- Often used to **mark a snapshot** of a **release** (v1.0)
- **Additional metadata** like creator, date, description
- Can be **cryptographically signed** using GPG
- Command: `git tag`

# Basic Branching

---

```
git branch <branch name>
```

```
git checkout <branch name>
```

```
git checkout -b <branch name>
```

# Branching with Git

---

# Branching with Git

---

- A named **pointer** to a **commit**

# Branching with Git

---

- A named **pointer** to a **commit**
- Very **lightweight** and **inexpensive** to create

# Branching with Git

---

- A named **pointer** to a **commit**
- Very **lightweight** and **inexpensive** to create
- **HEAD** is a special pointer to the current branch

# Branching with Git

---

- A named **pointer** to a **commit**
- Very **lightweight** and **inexpensive** to create
- **HEAD** is a special pointer to the current branch
- By convention **master** is the main branch (trunk)

# Branching with Git

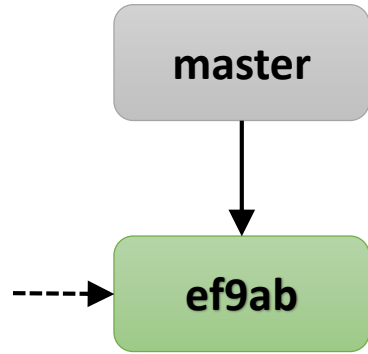
---

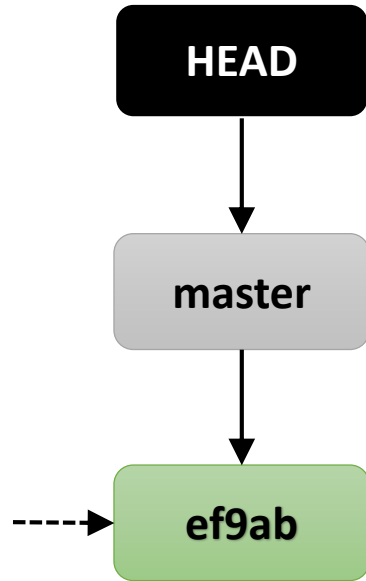
- A named **pointer** to a **commit**
- Very **lightweight** and **inexpensive** to create
- **HEAD** is a special pointer to the current branch
- By convention **master** is the main branch (trunk)
- Used for new **features**, **bugfixes**, **maintenance**, etc.

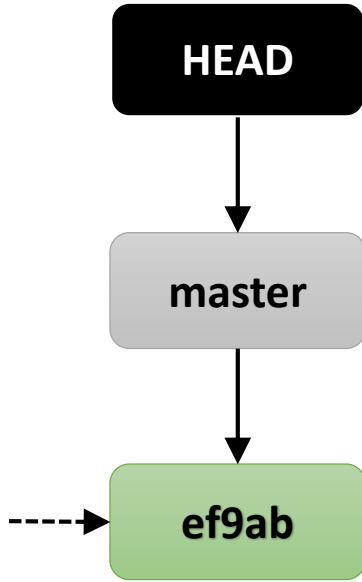
# Branching with Git

---

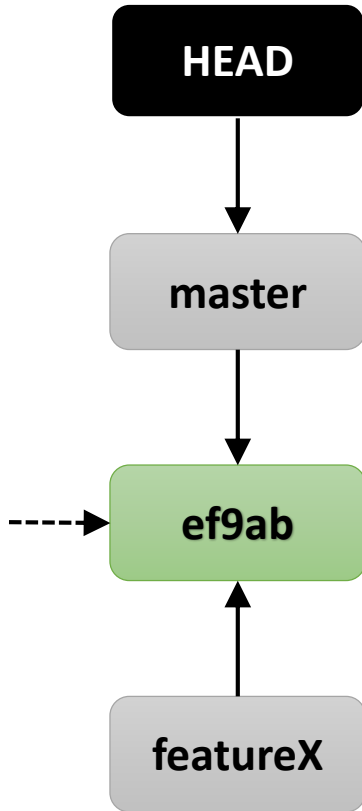
- A named **pointer** to a **commit**
- Very **lightweight** and **inexpensive** to create
- **HEAD** is a special pointer to the current branch
- By convention **master** is the main branch (trunk)
- Used for new **features**, **bugfixes**, **maintenance**, etc.
- Can be **short-lived**, very good for **trying ideas**



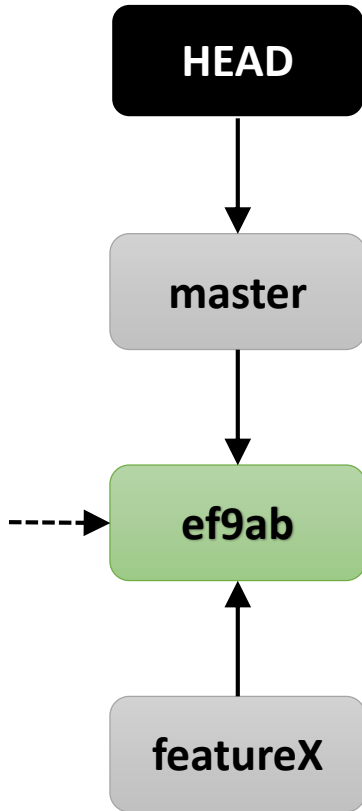




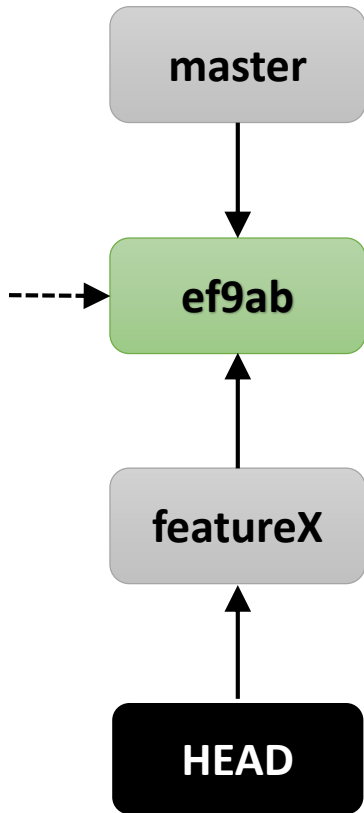
git branch featureX



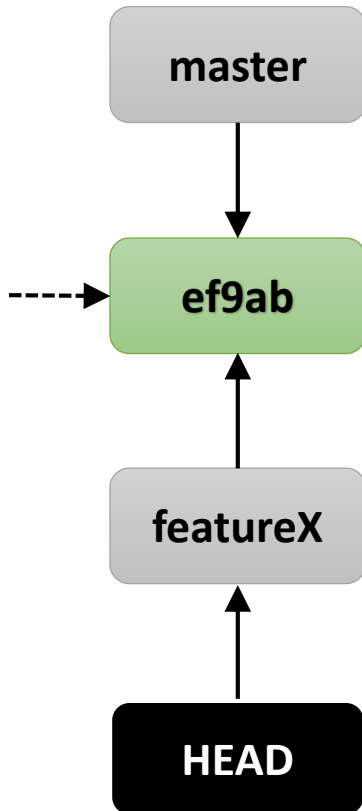
git branch featureX



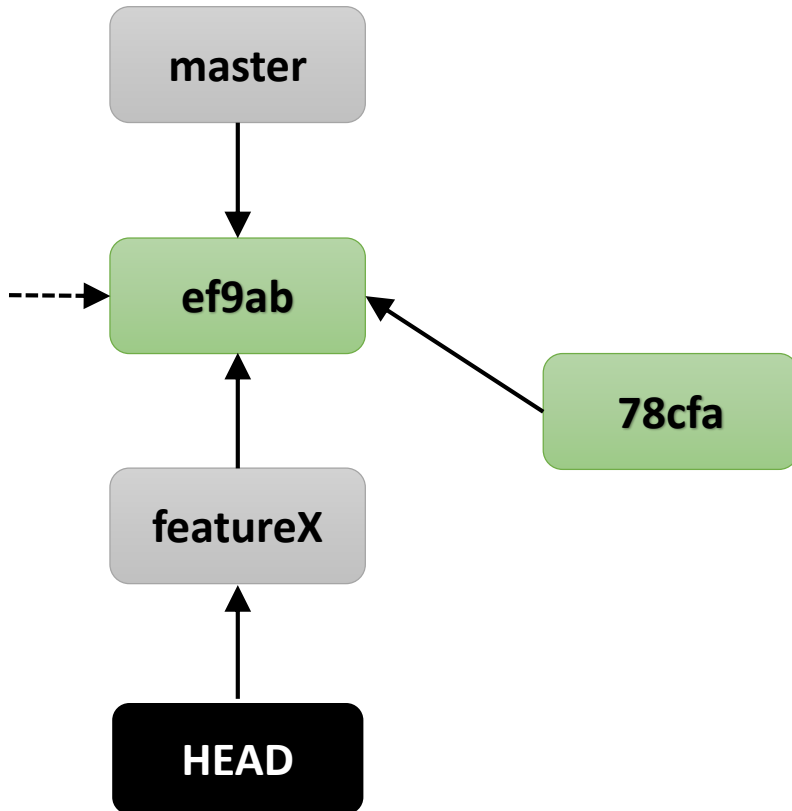
git checkout featureX



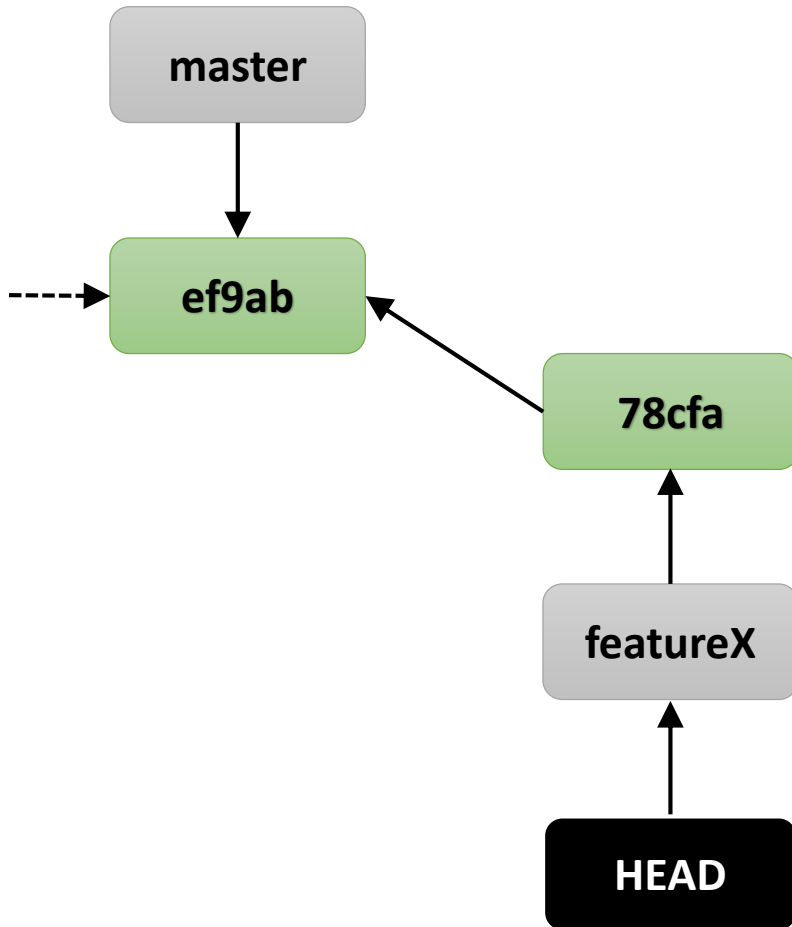
git checkout featureX



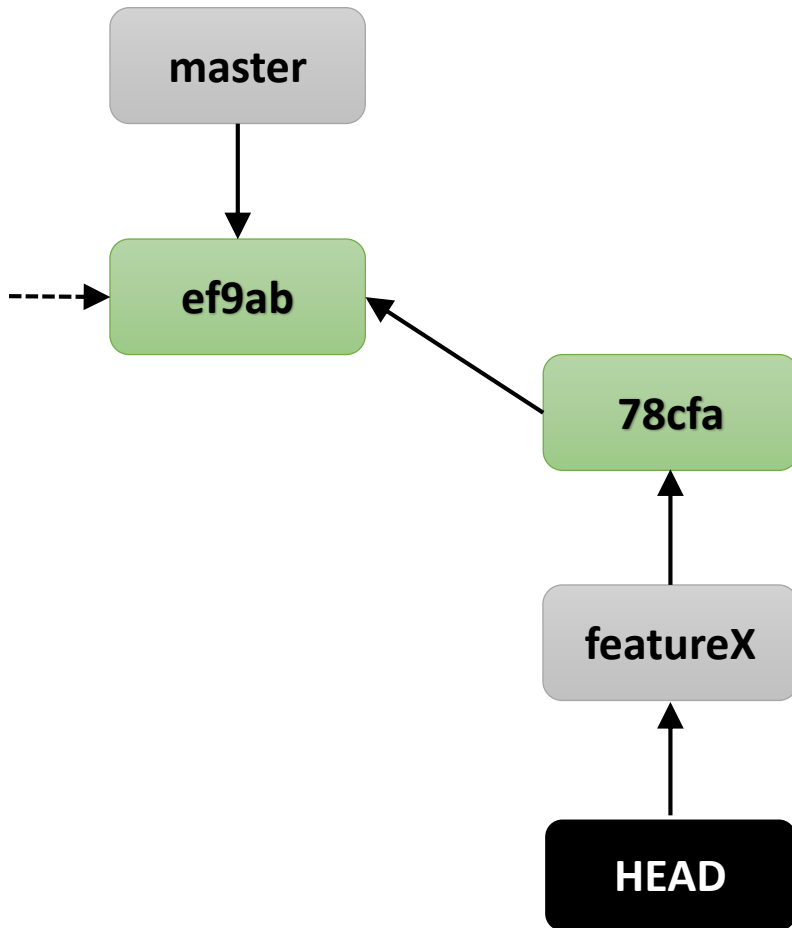
```
git commit -m "Some changes"
```



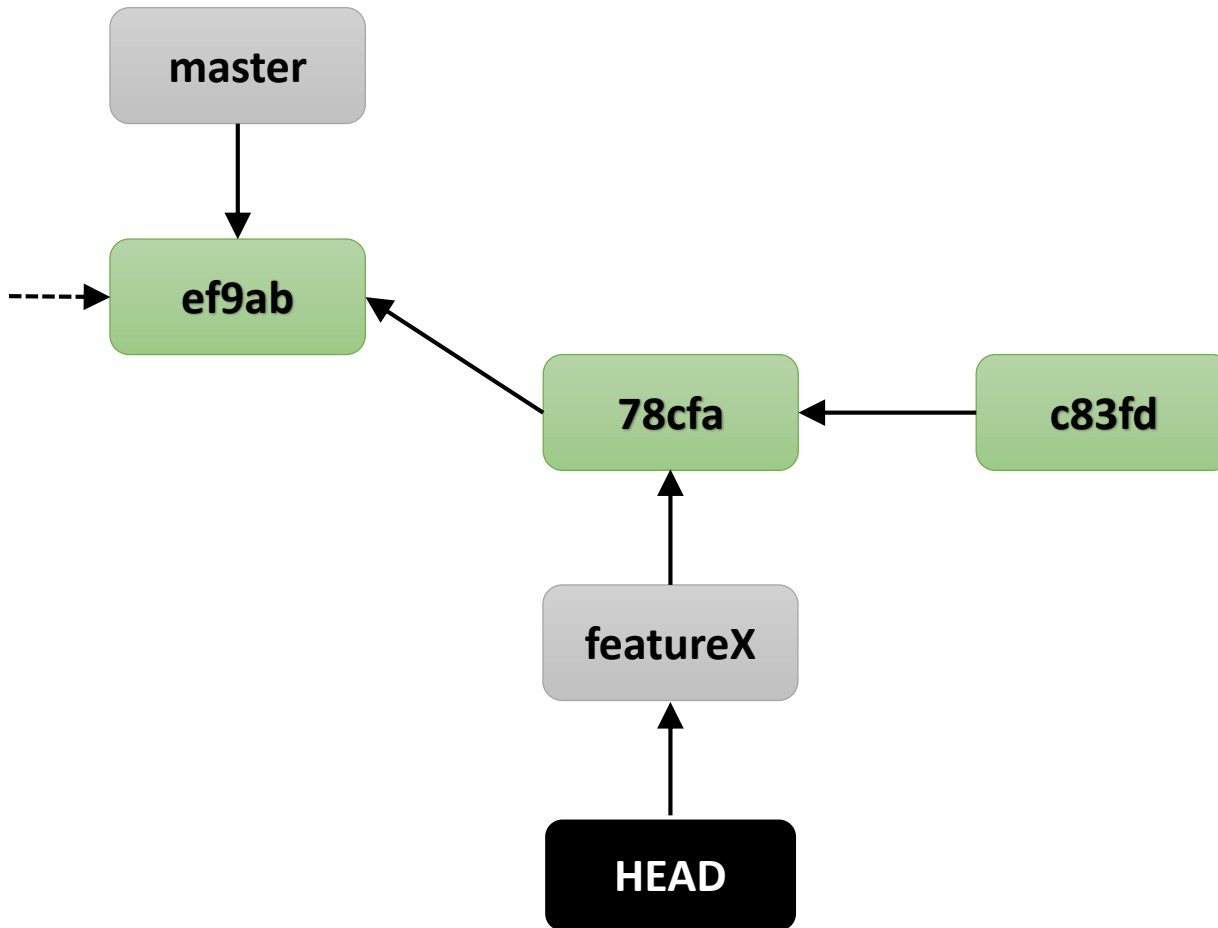
```
git commit -m "Some changes"
```



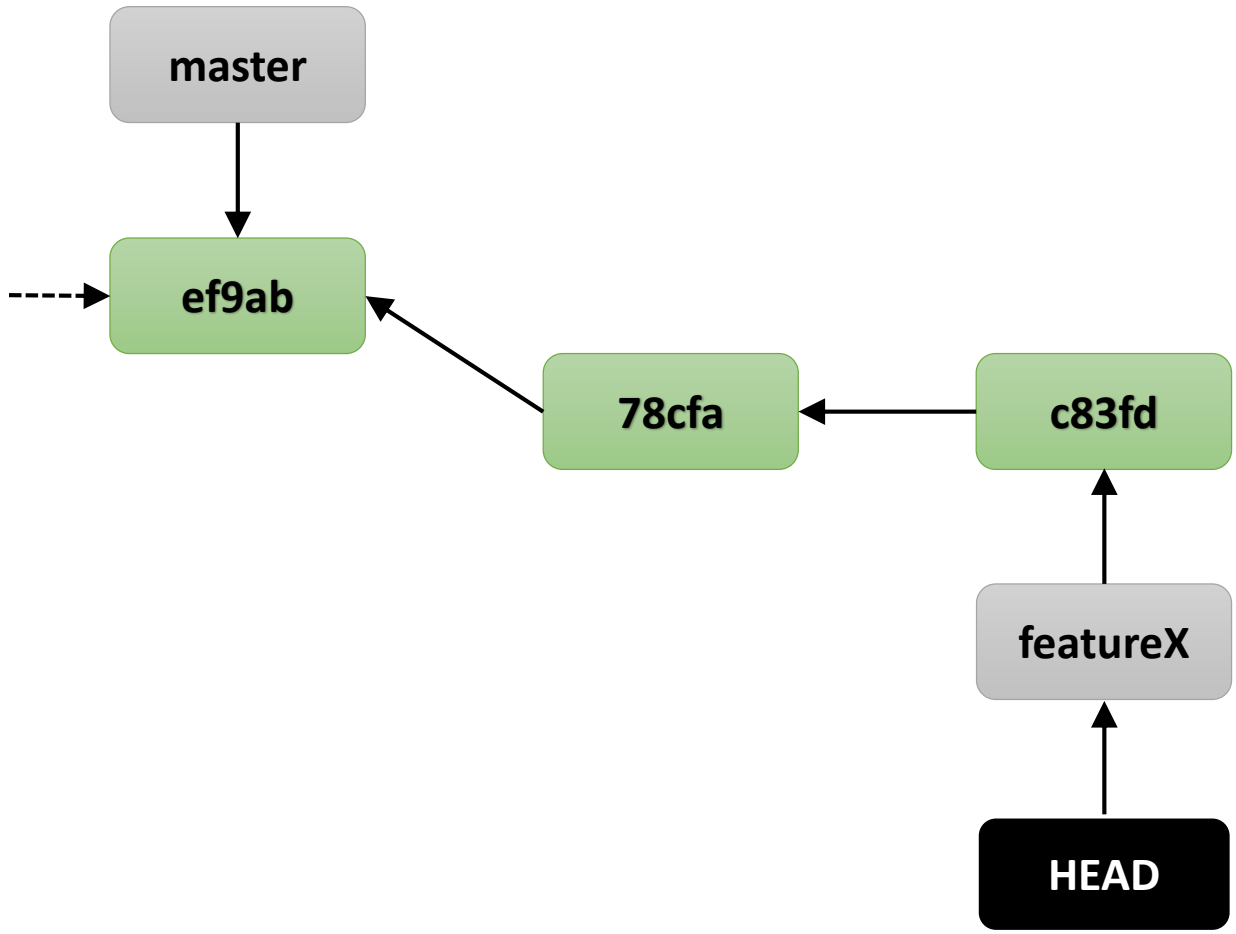
`git commit -m "Some changes"`



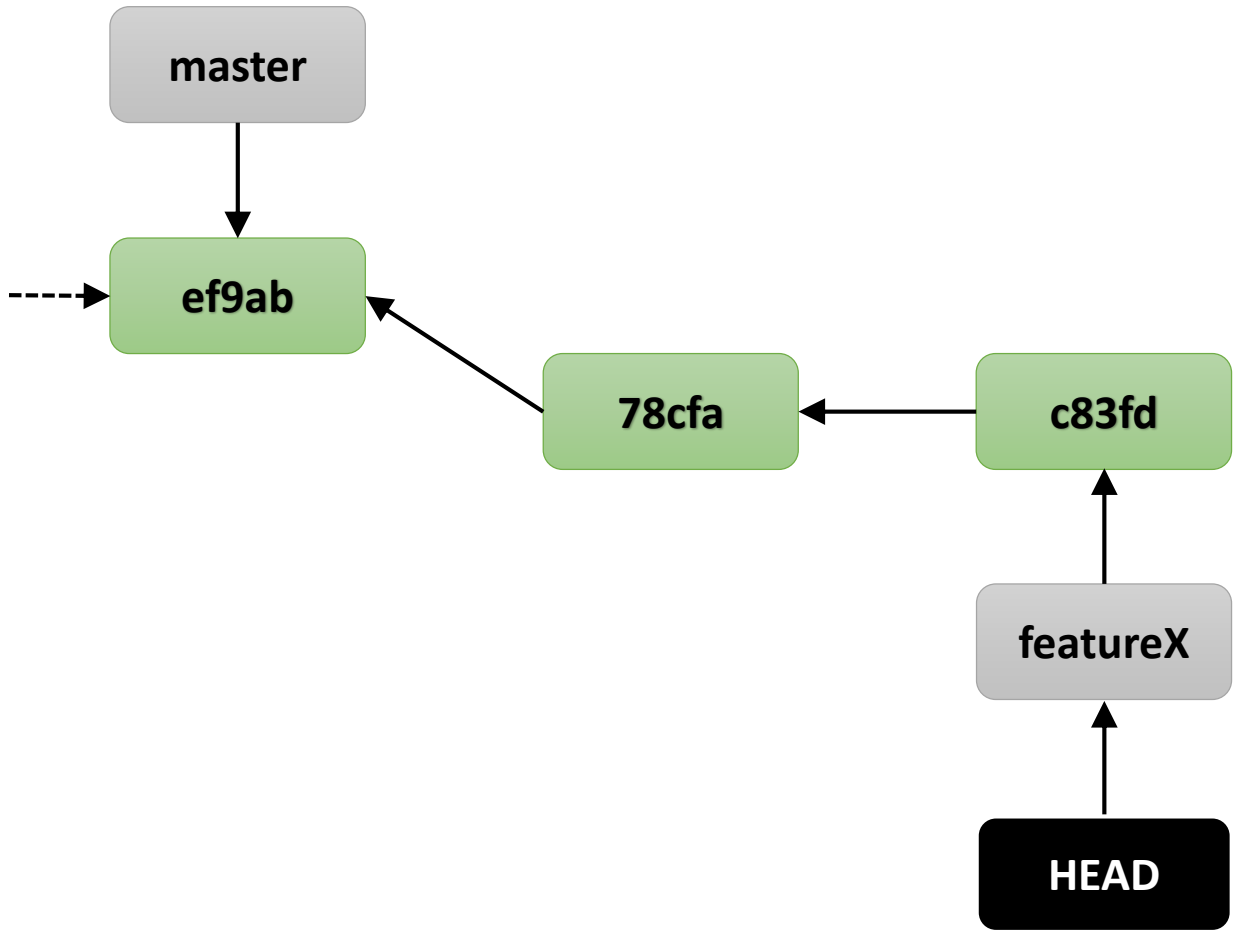
`git commit -m "Some more changes"`



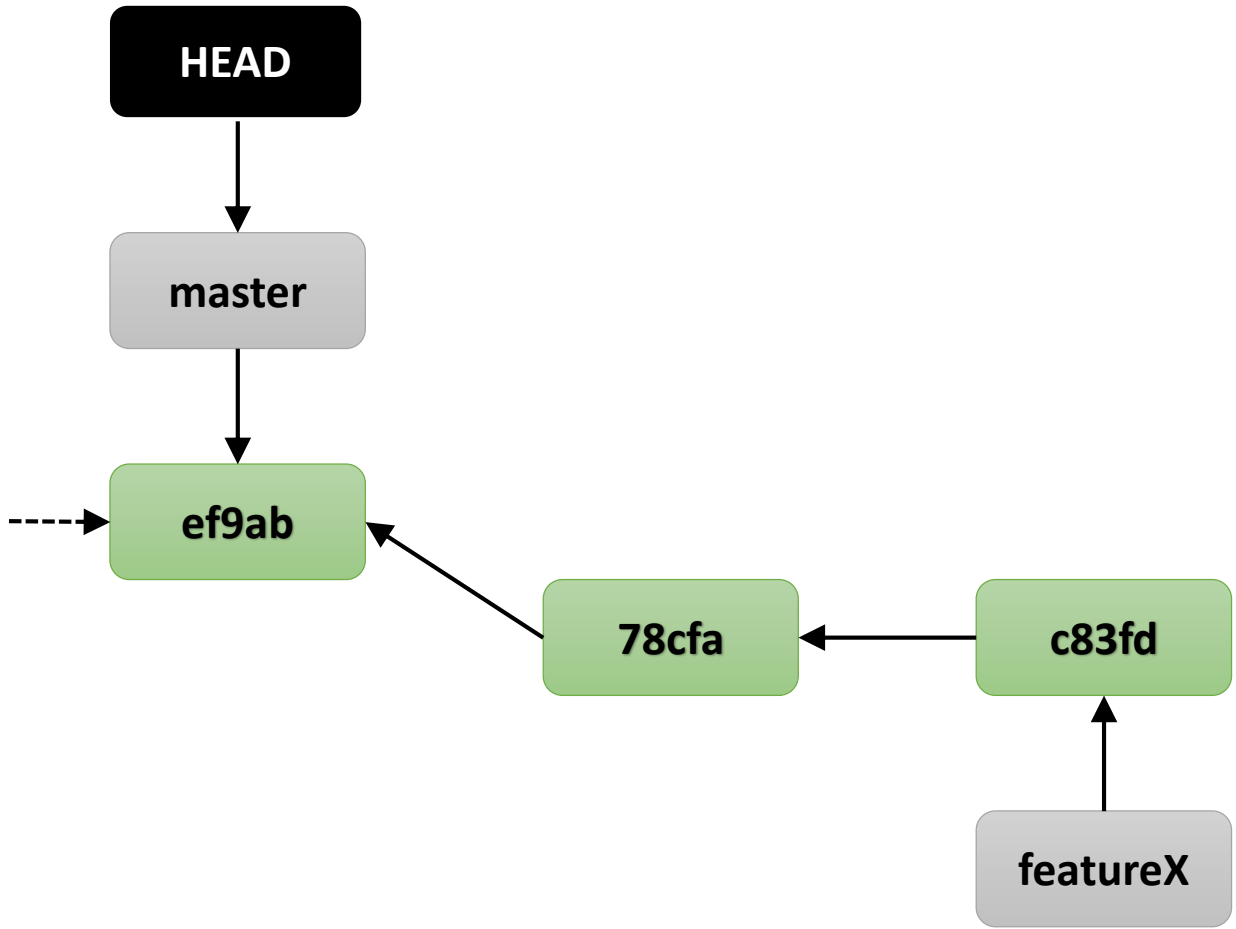
`git commit -m "Some more changes"`



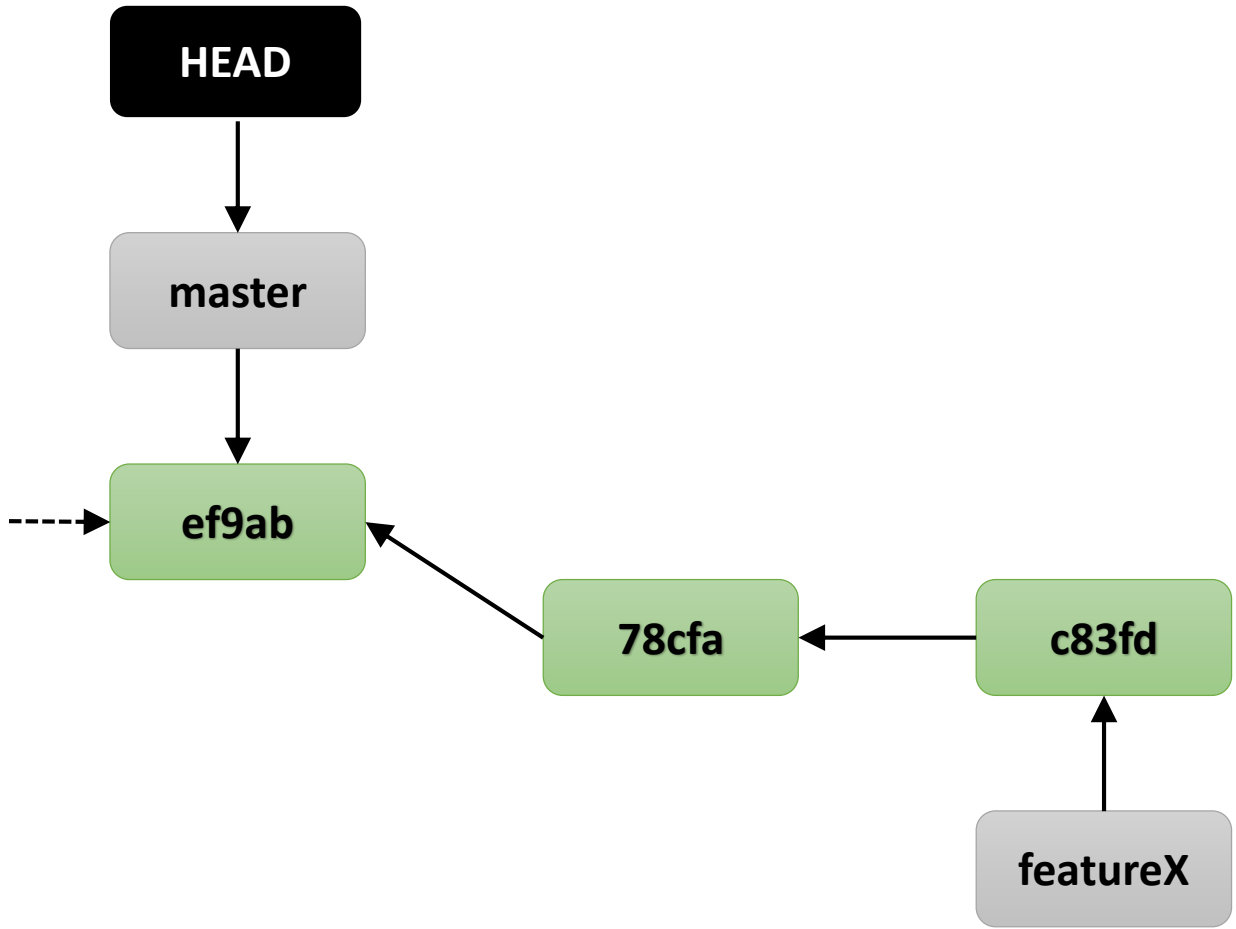
```
git commit -m "Some more changes"
```



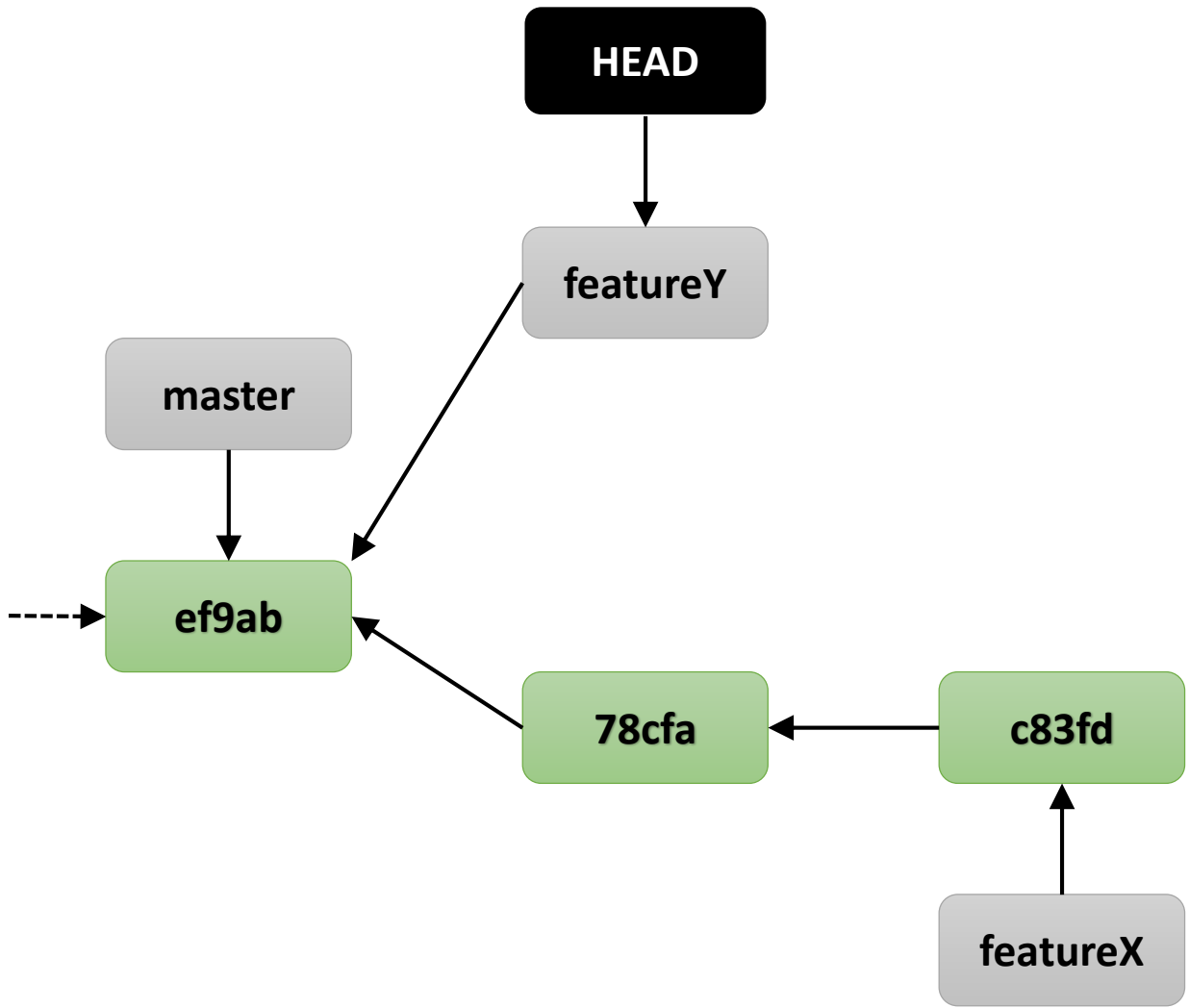
git checkout master



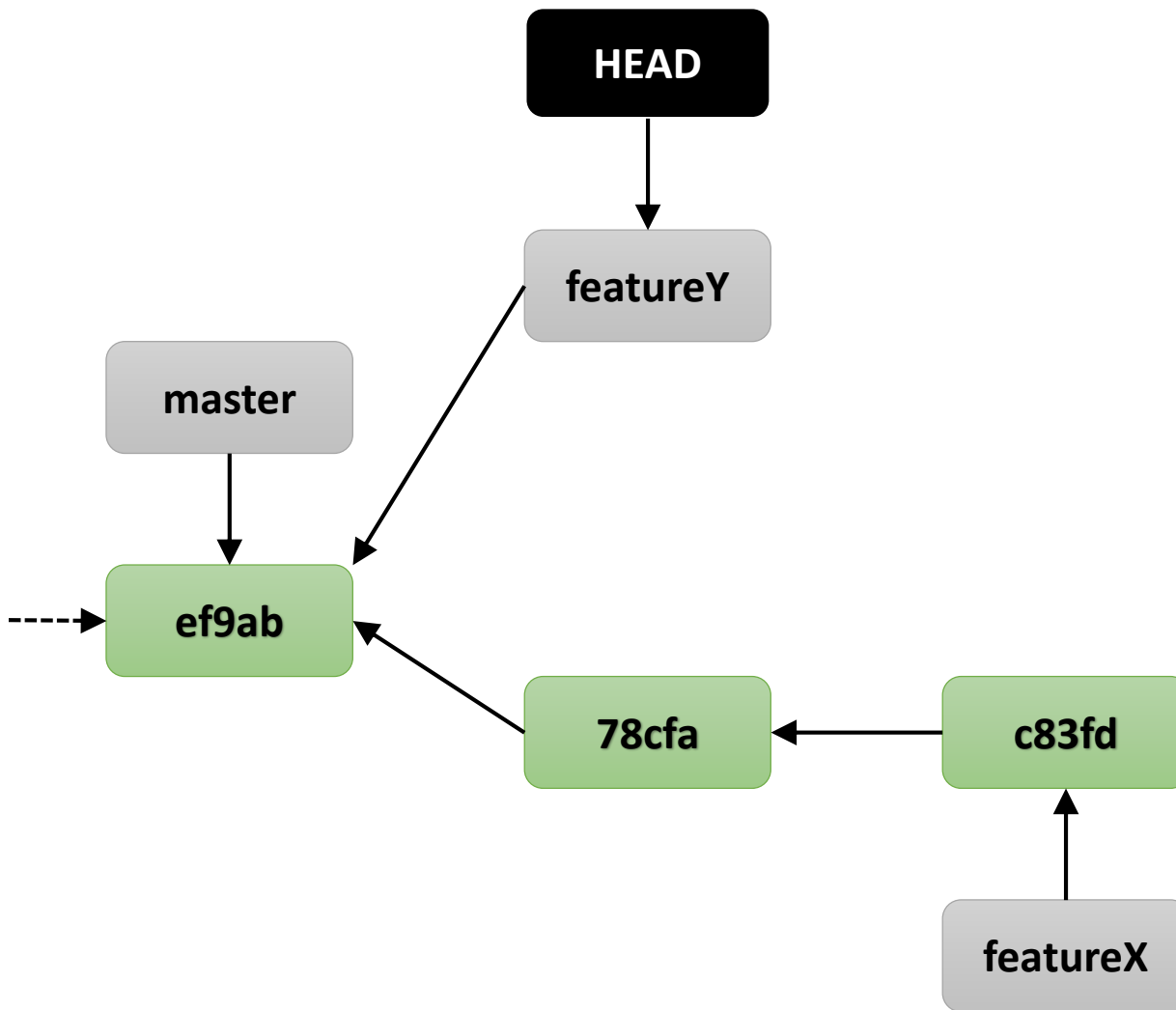
git checkout master



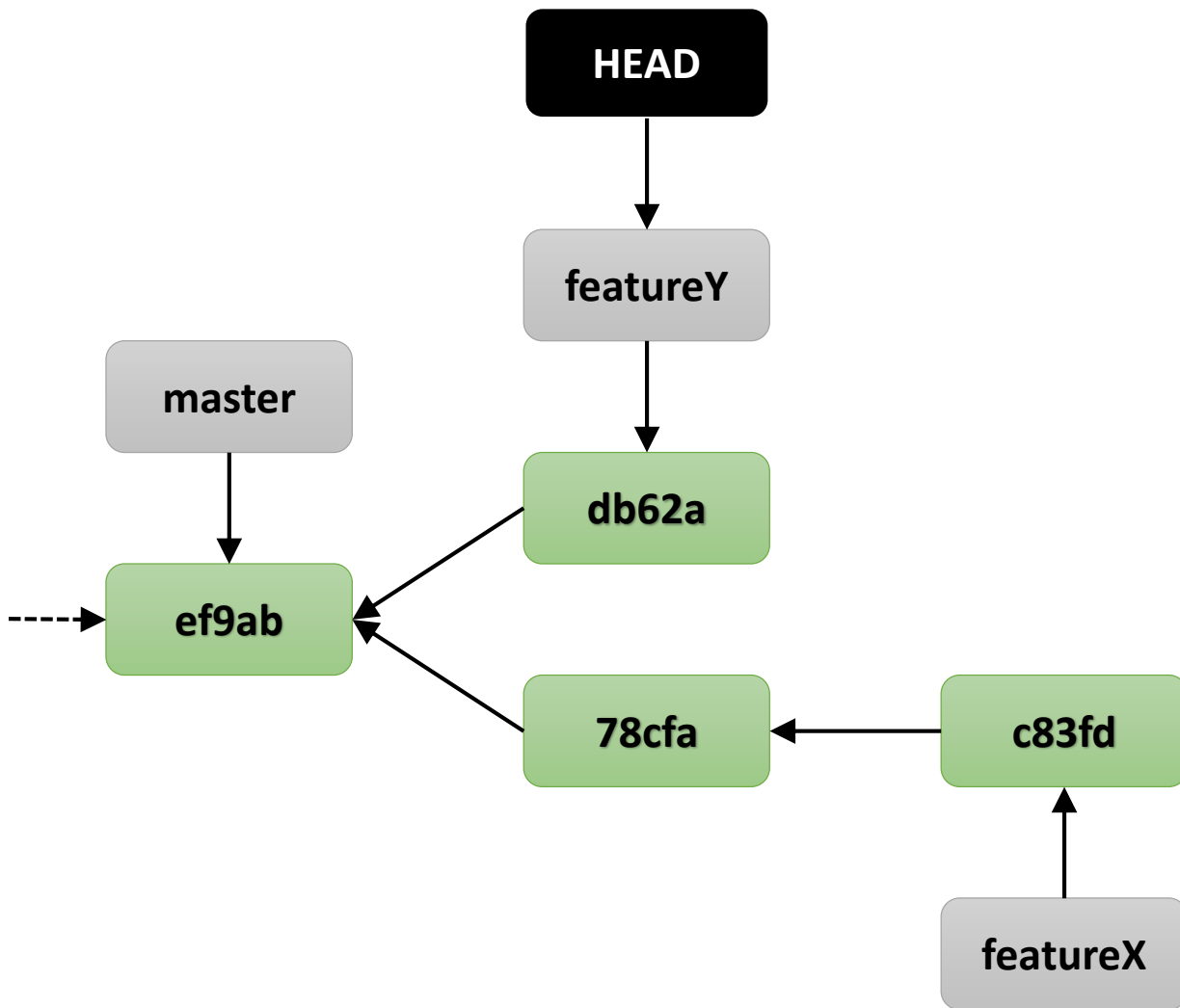
`git checkout -b featureY`



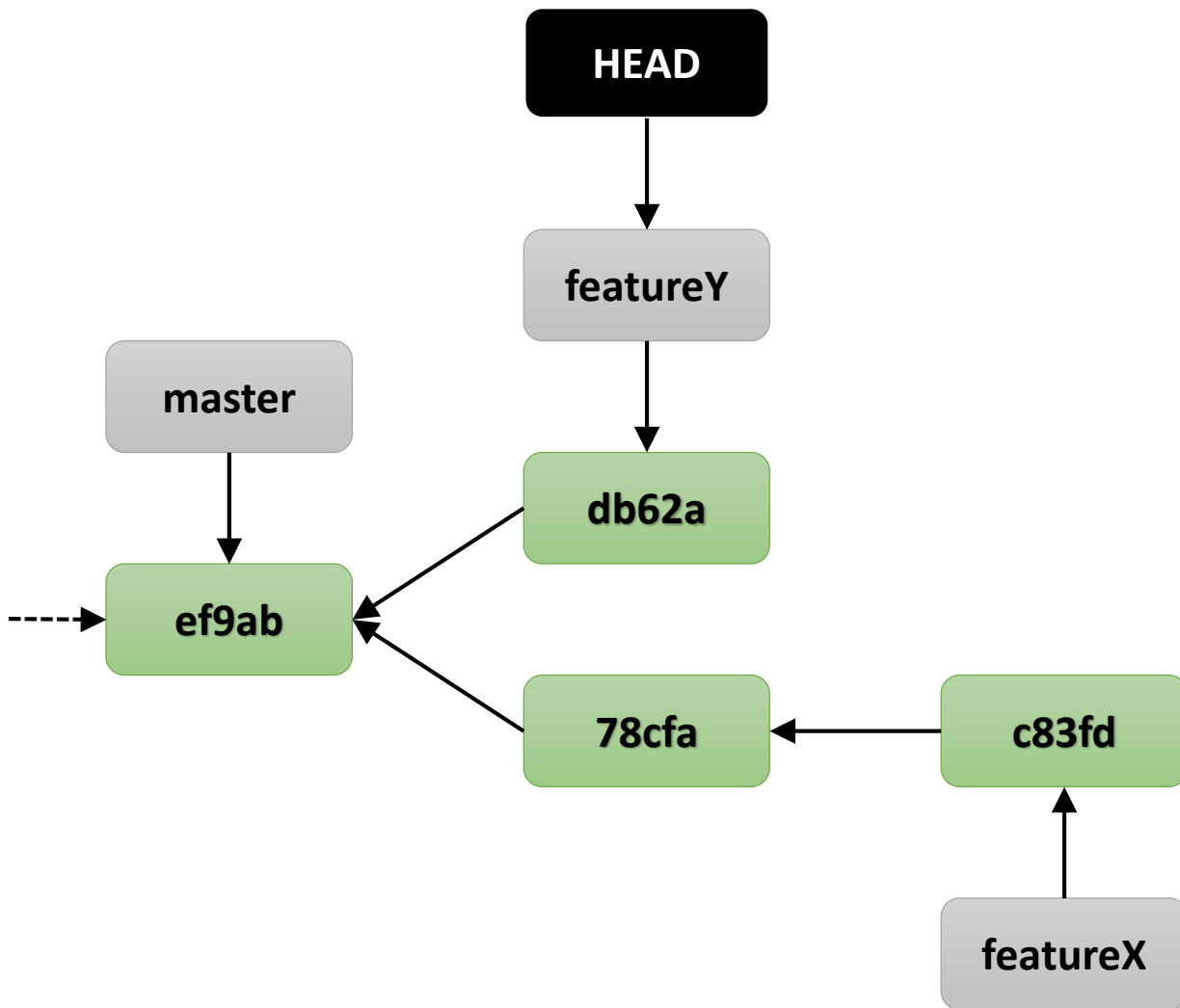
`git checkout -b featureY`



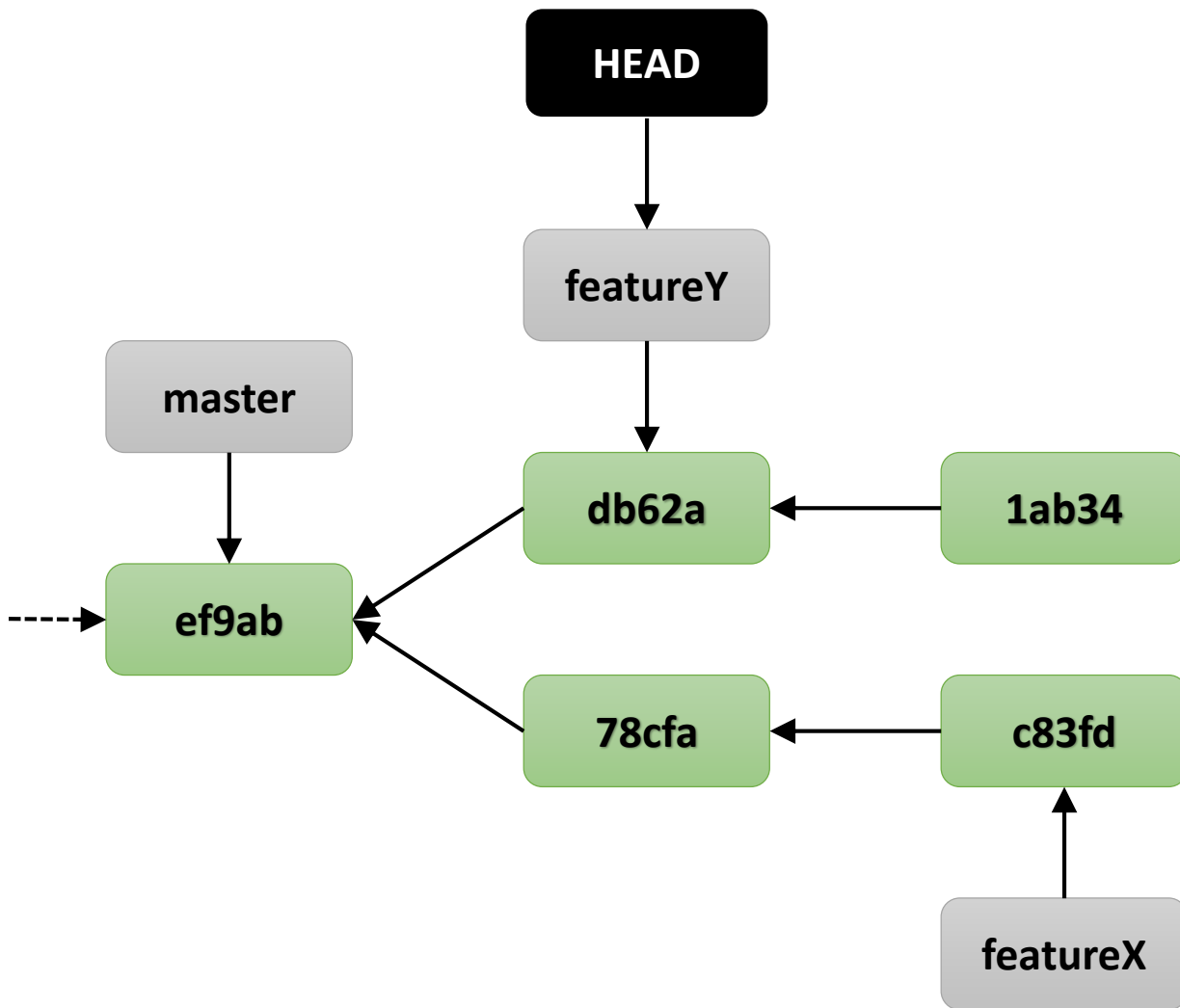
`git commit -m "Some changes"`



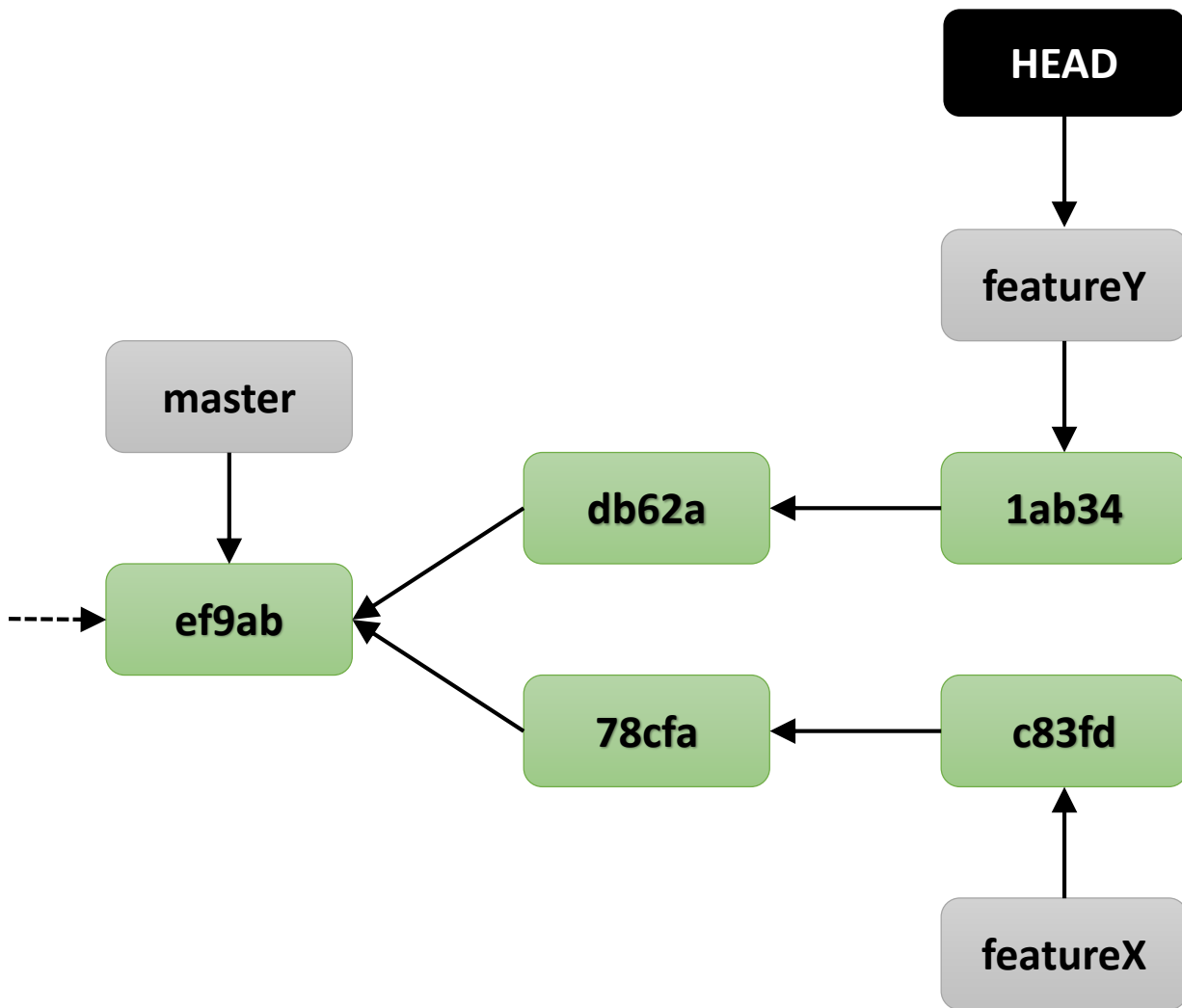
`git commit -m "Some changes"`



`git commit -m "Some more changes"`



`git commit -m "Some more changes"`



`git commit -m "Some more changes"`

# Basic Merging

---

```
git merge <source branch name>
```

# Merging with Git

---

# Merging with Git

---

- **Reintegration** of one branch into another

# Merging with Git

---

- **Reintegration** of one branch into another
- **Snapshot storage** of Git makes it very performant

# Merging with Git

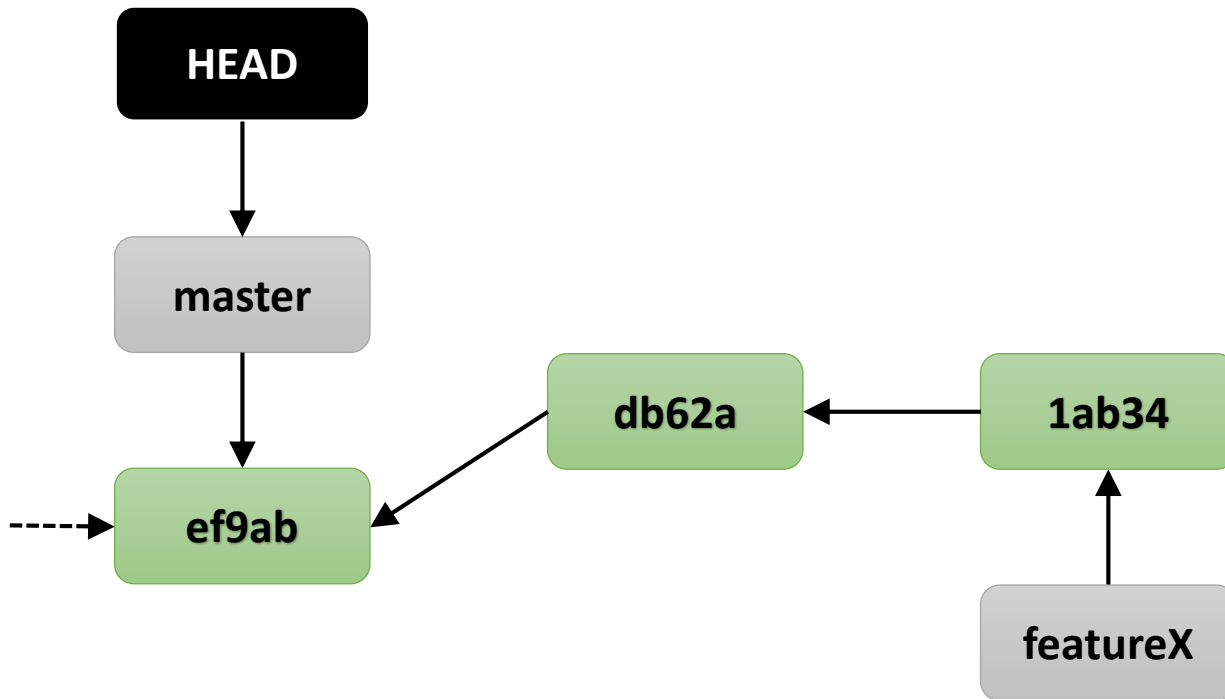
---

- **Reintegration** of one branch into another
- **Snapshot storage** of Git makes it very performant
- Very good **auto-merge** strategies

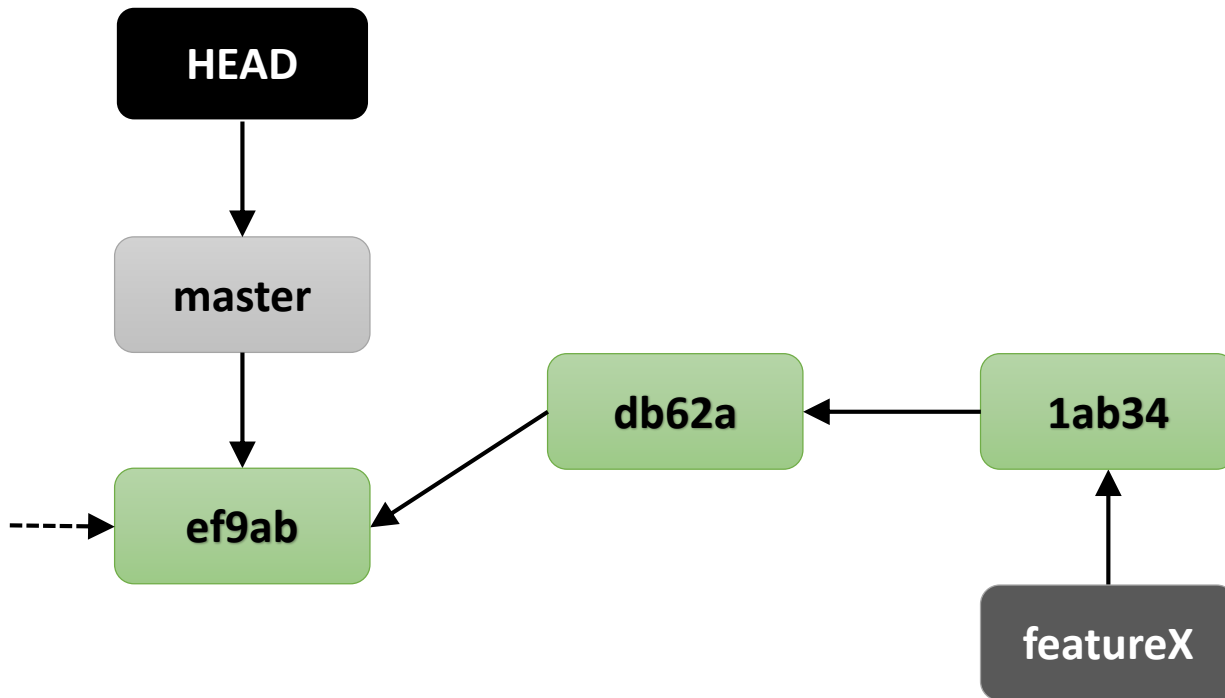
# Merging with Git

---

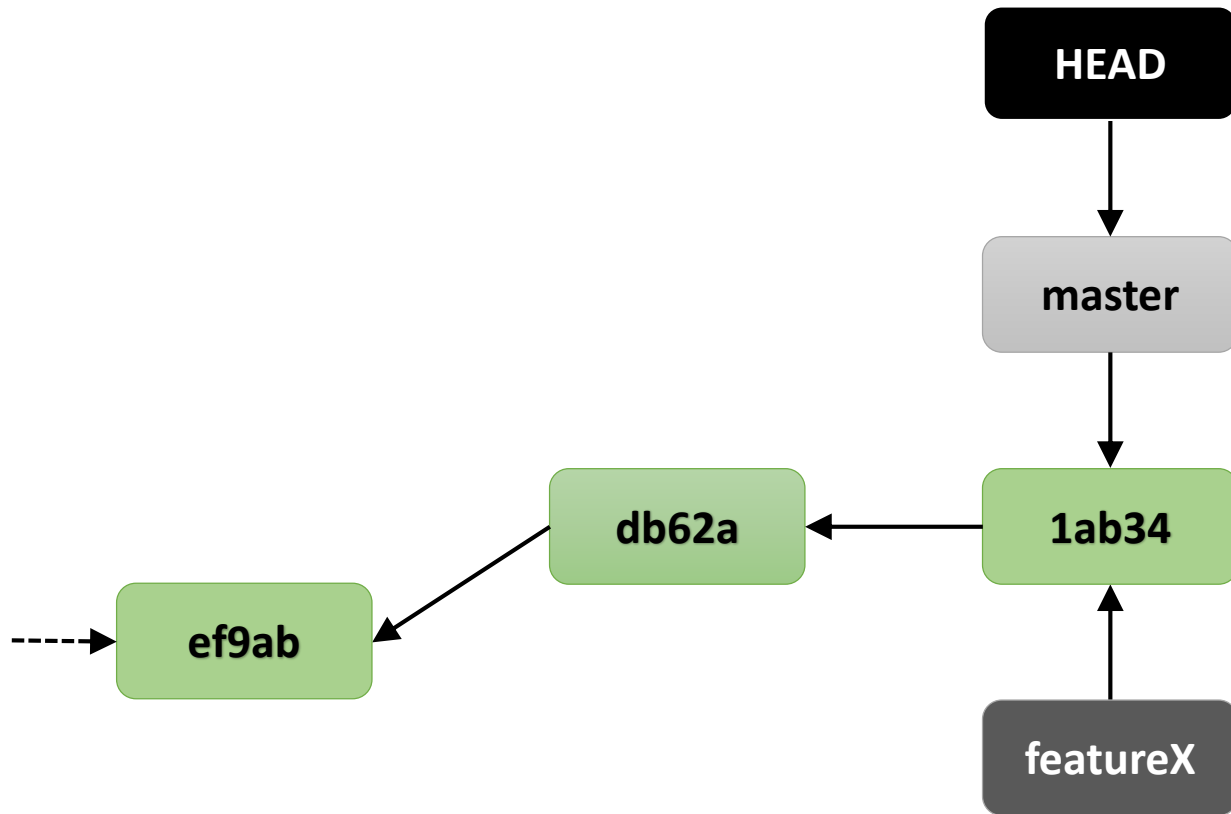
- **Reintegration** of one branch into another
- **Snapshot storage** of Git makes it very performant
- Very good **auto-merge** strategies
- But: conflicting files may require **manual resolving**



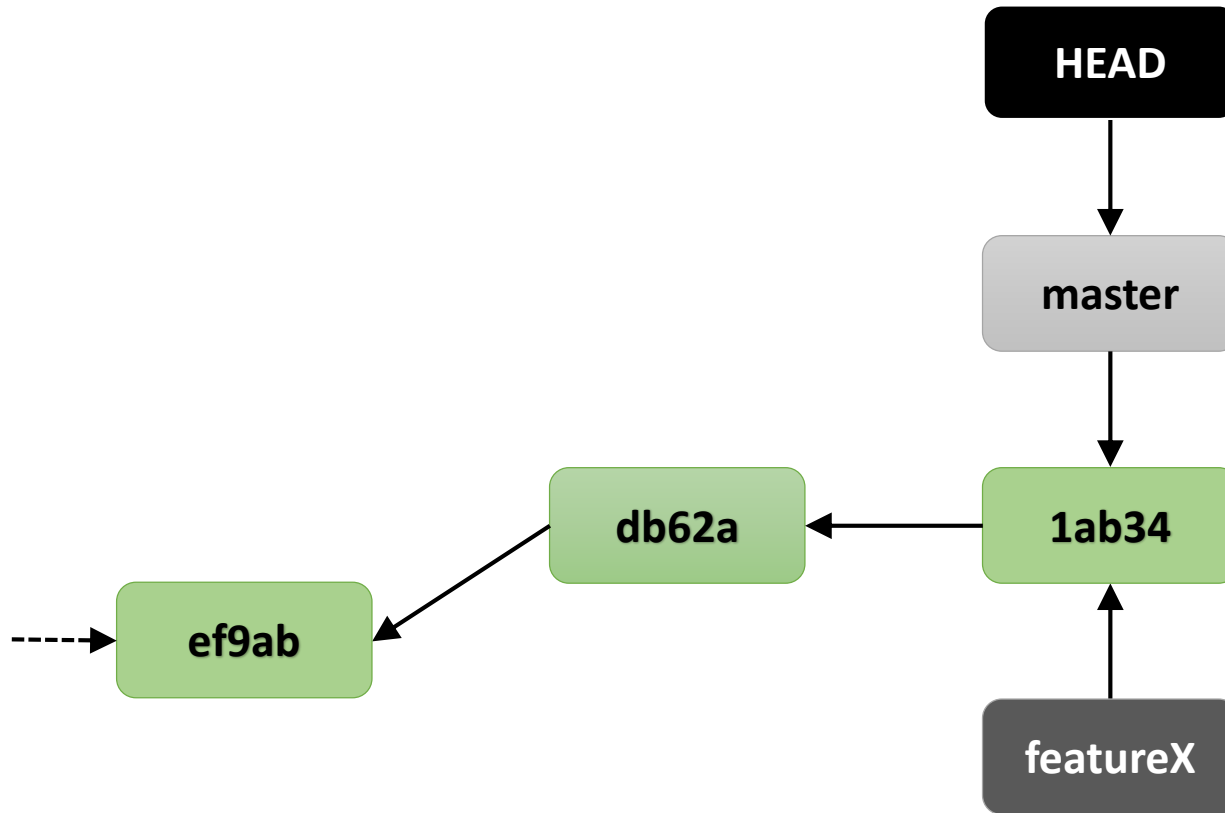
git checkout master



`git merge featureX`

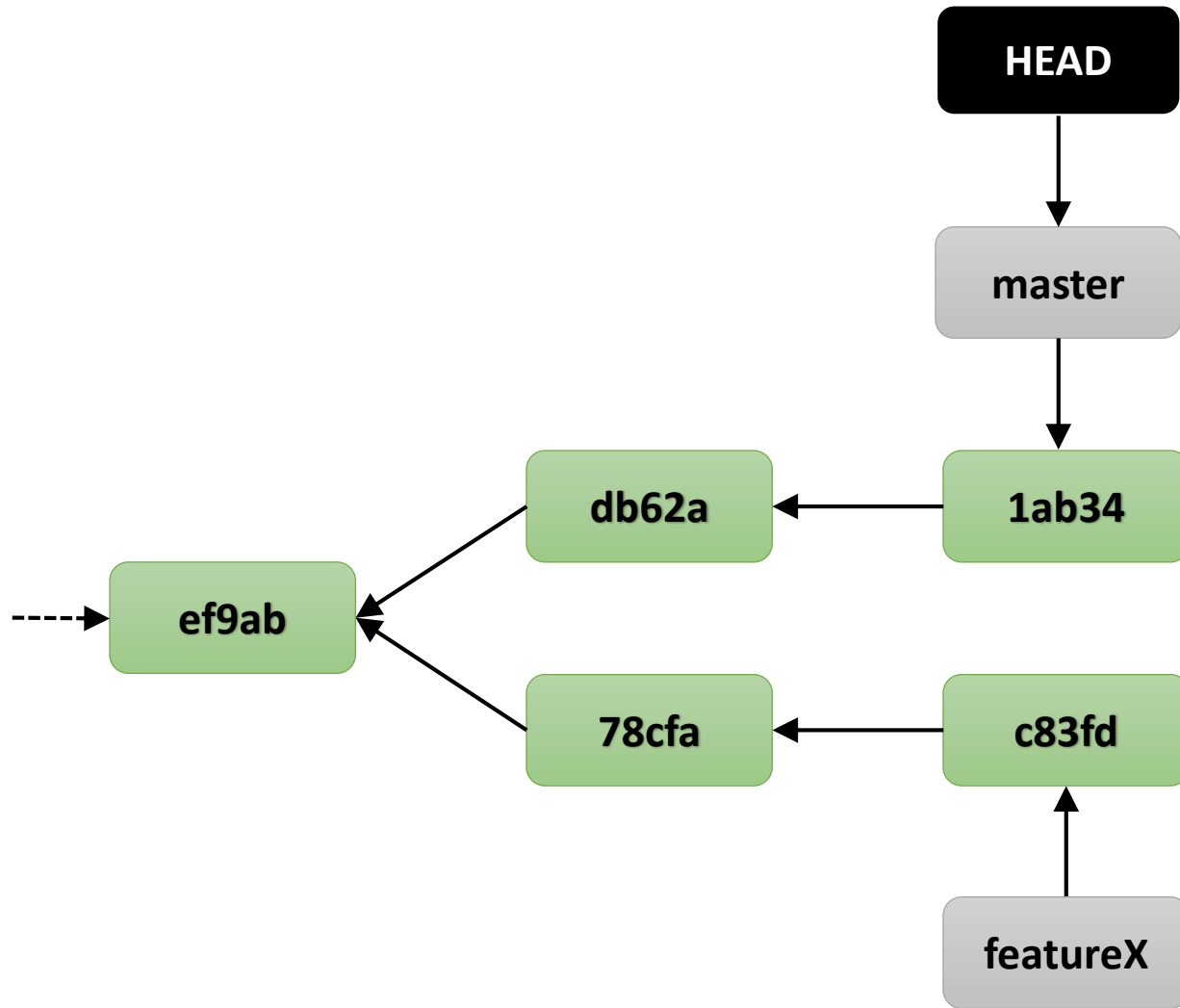


`git merge featureX`

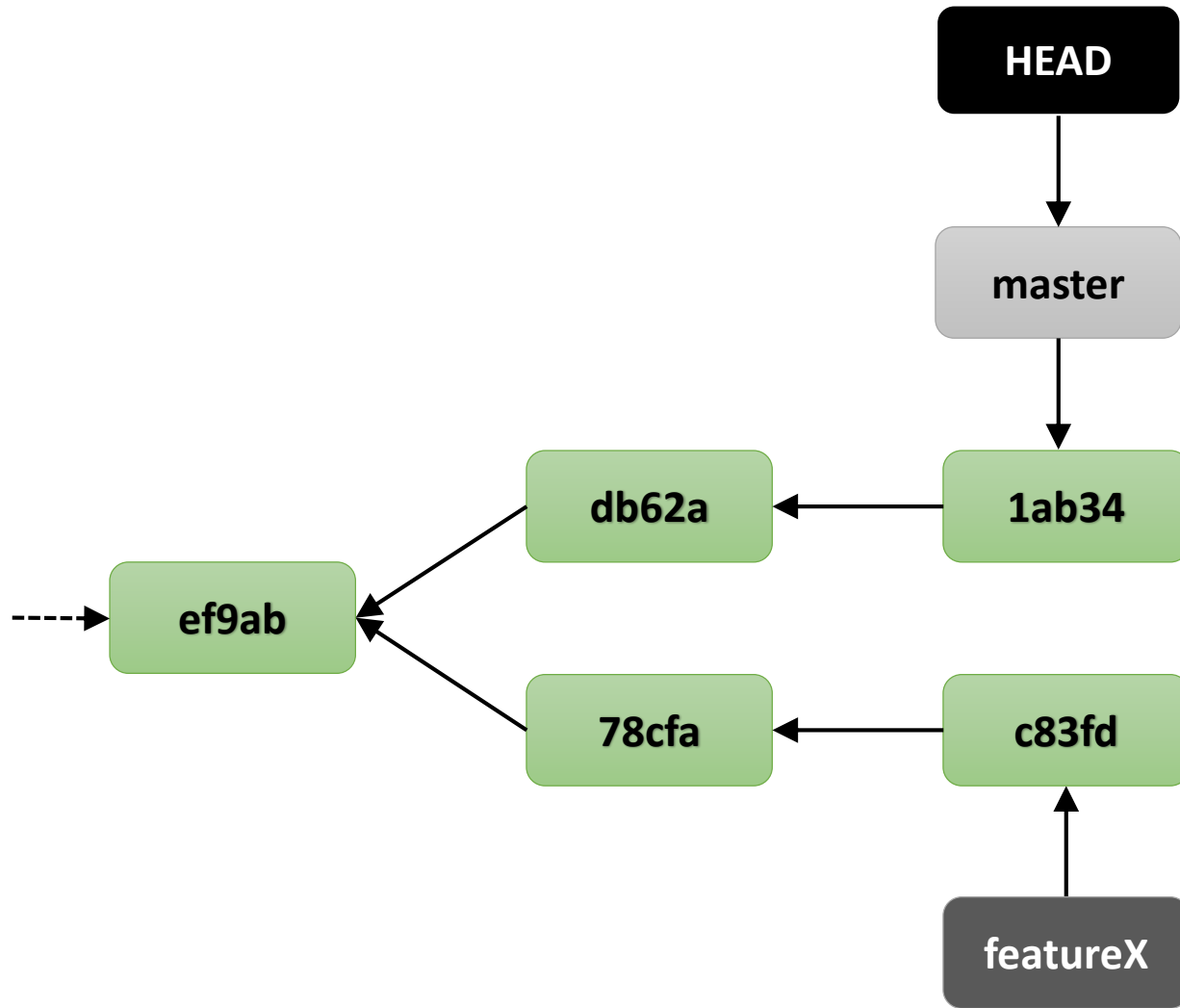


“Fast Forward”

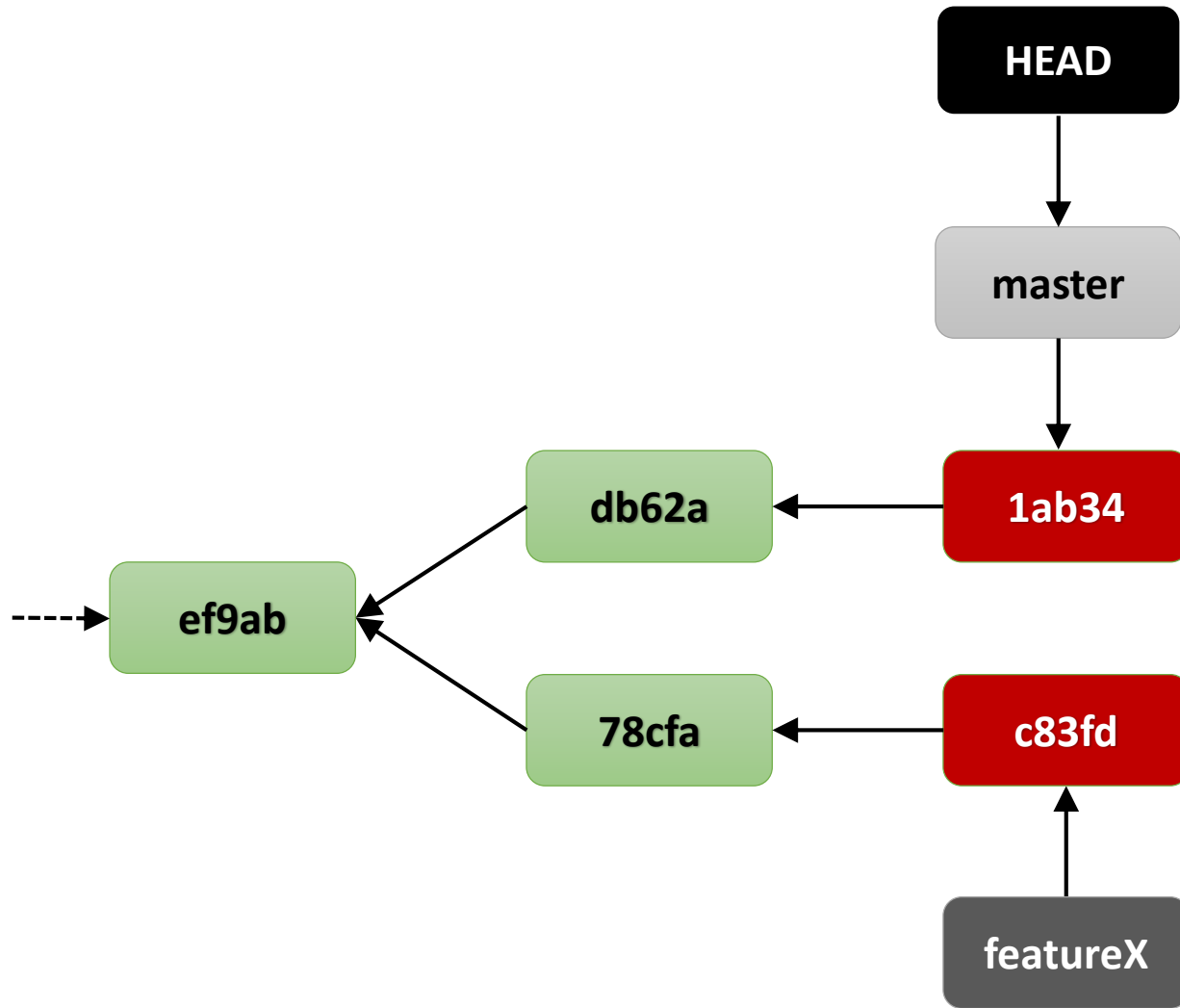
`git merge featureX`



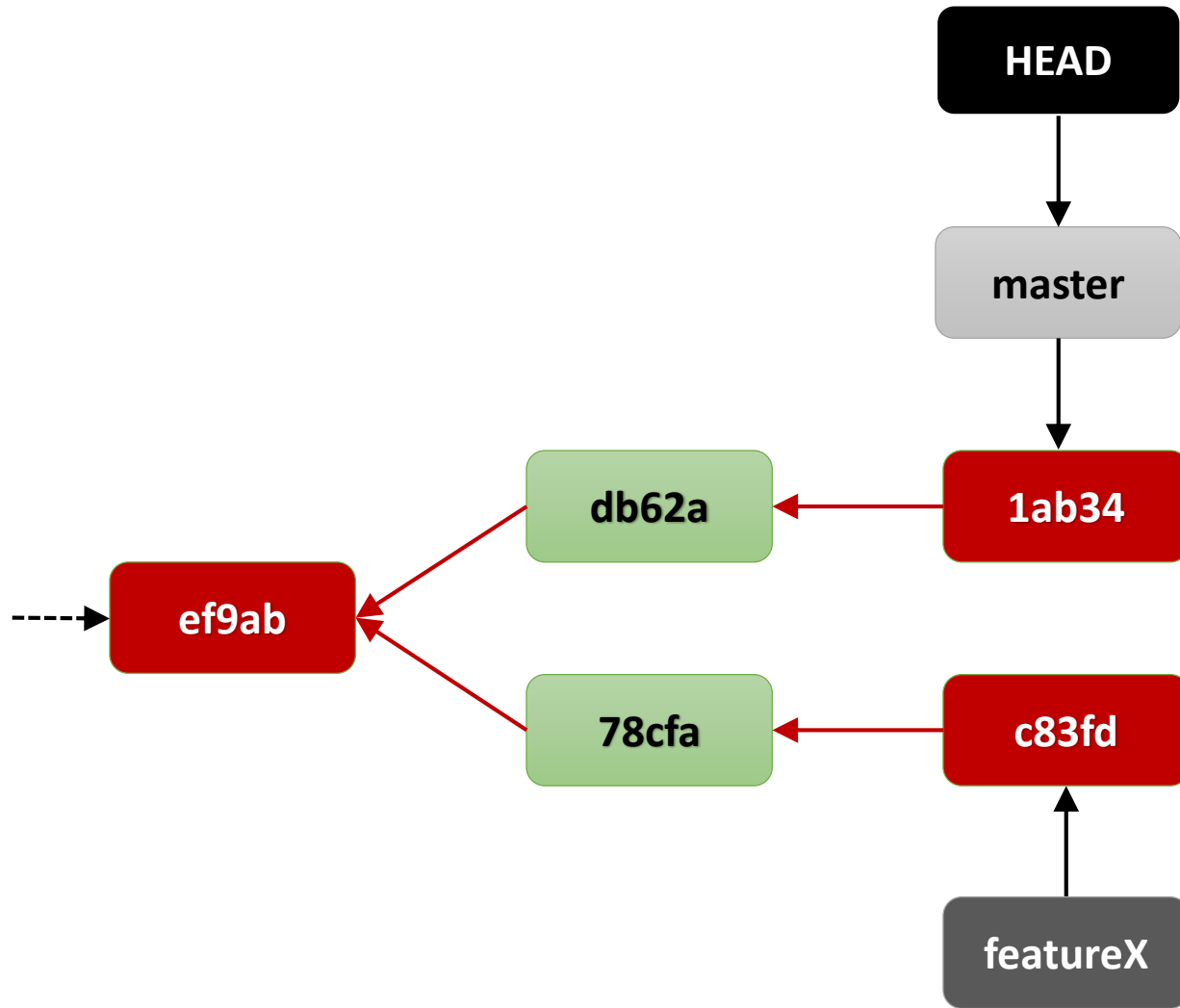
git checkout master



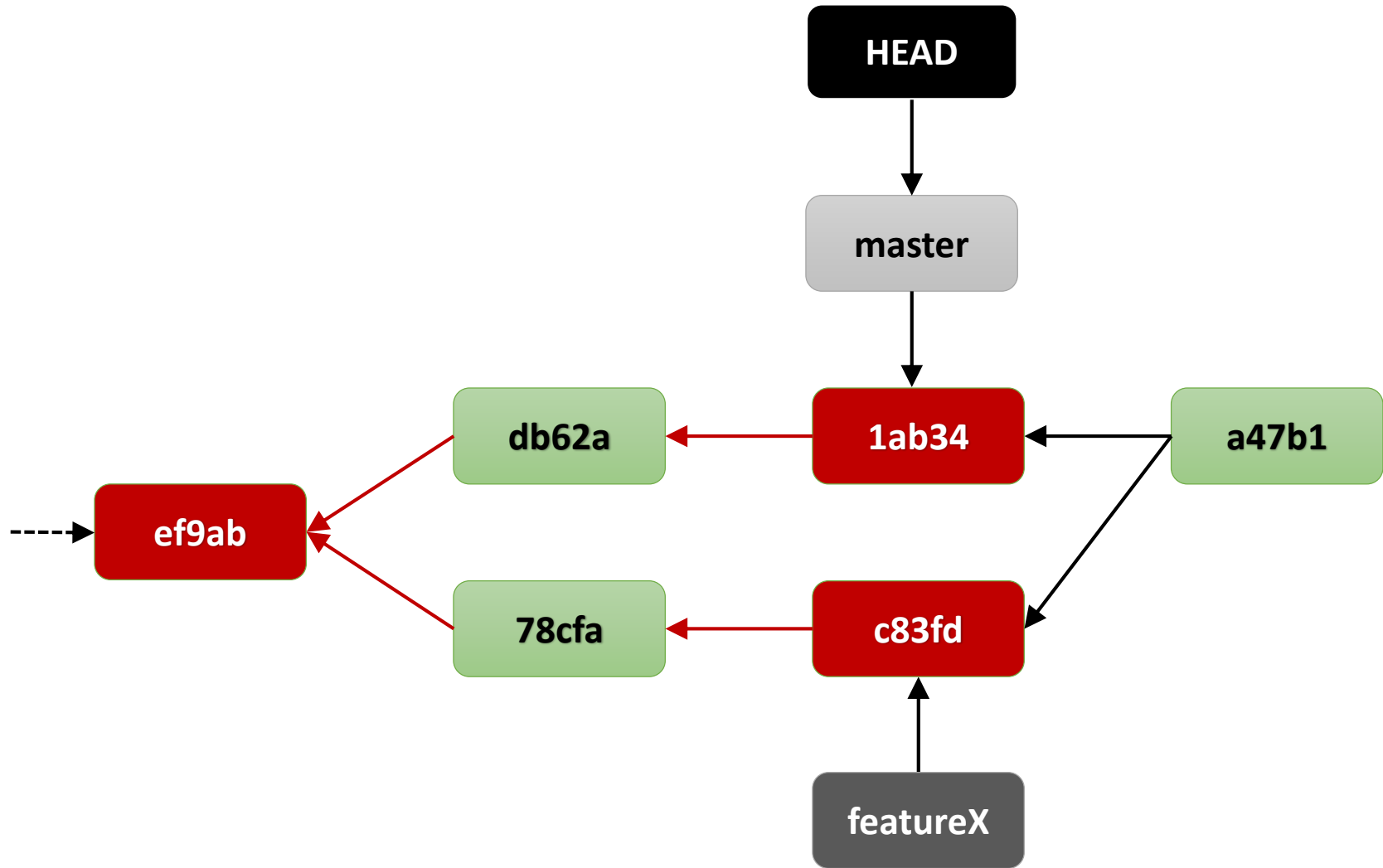
git merge featureX



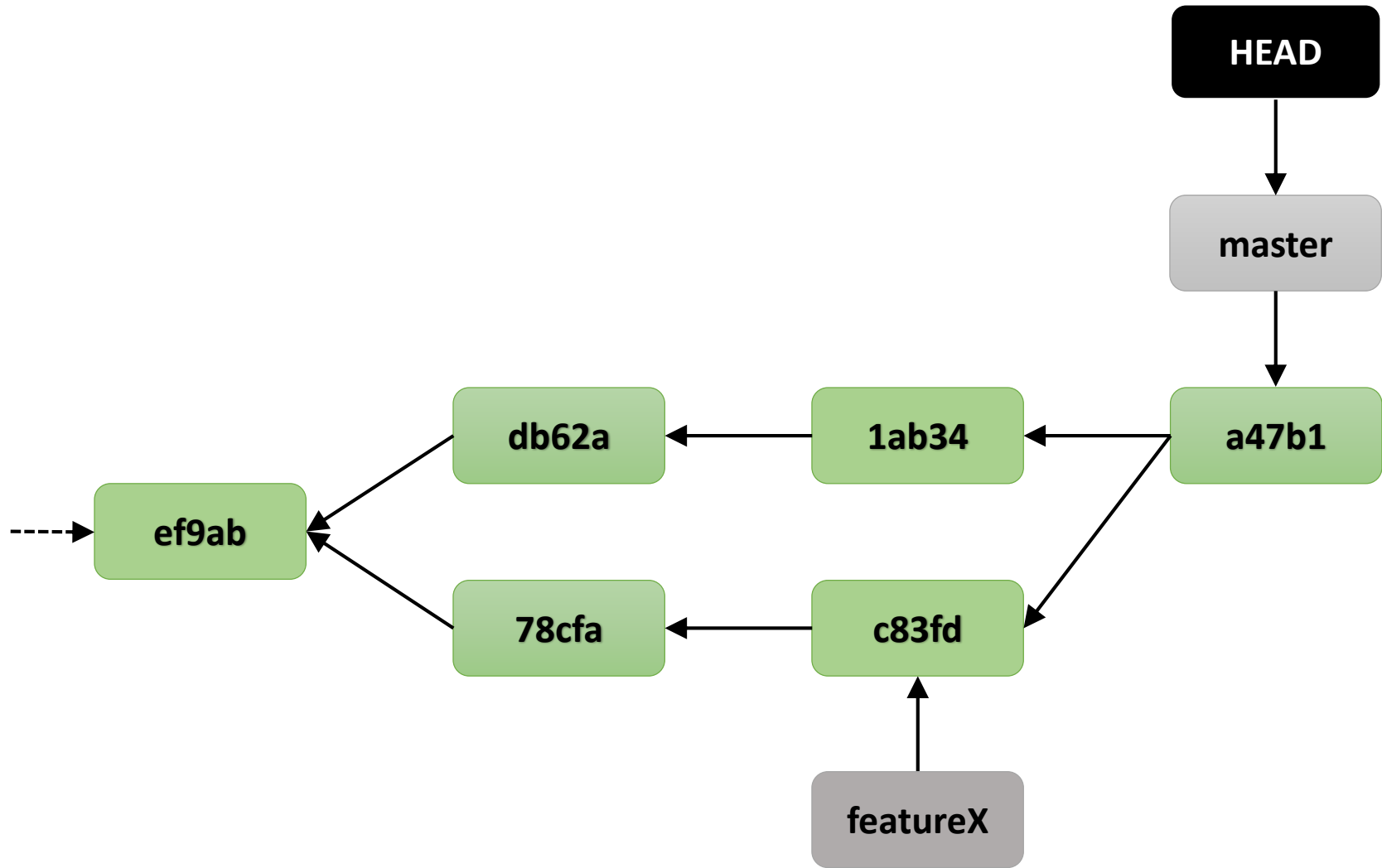
git merge featureX



`git merge featureX`

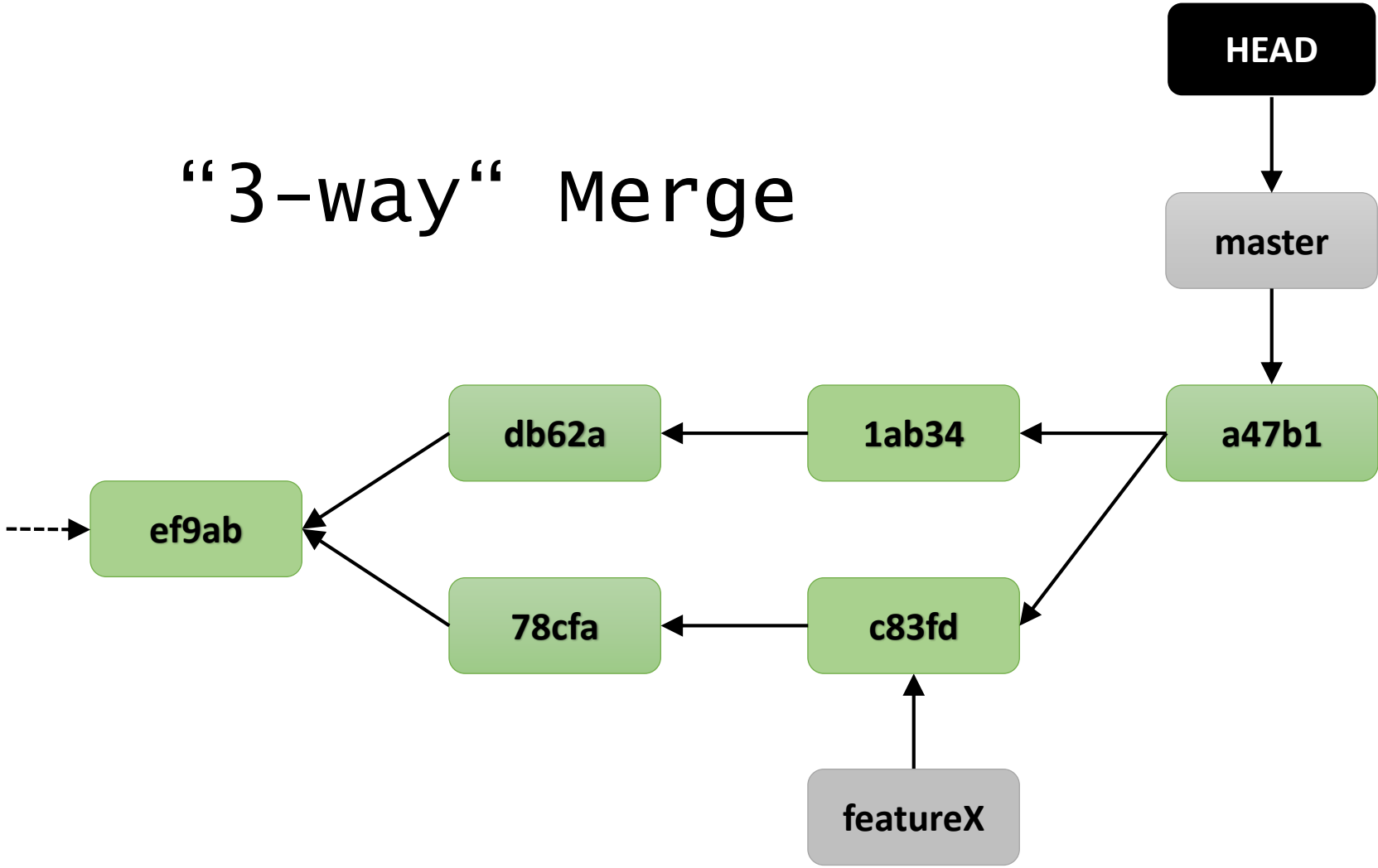


git merge featureX



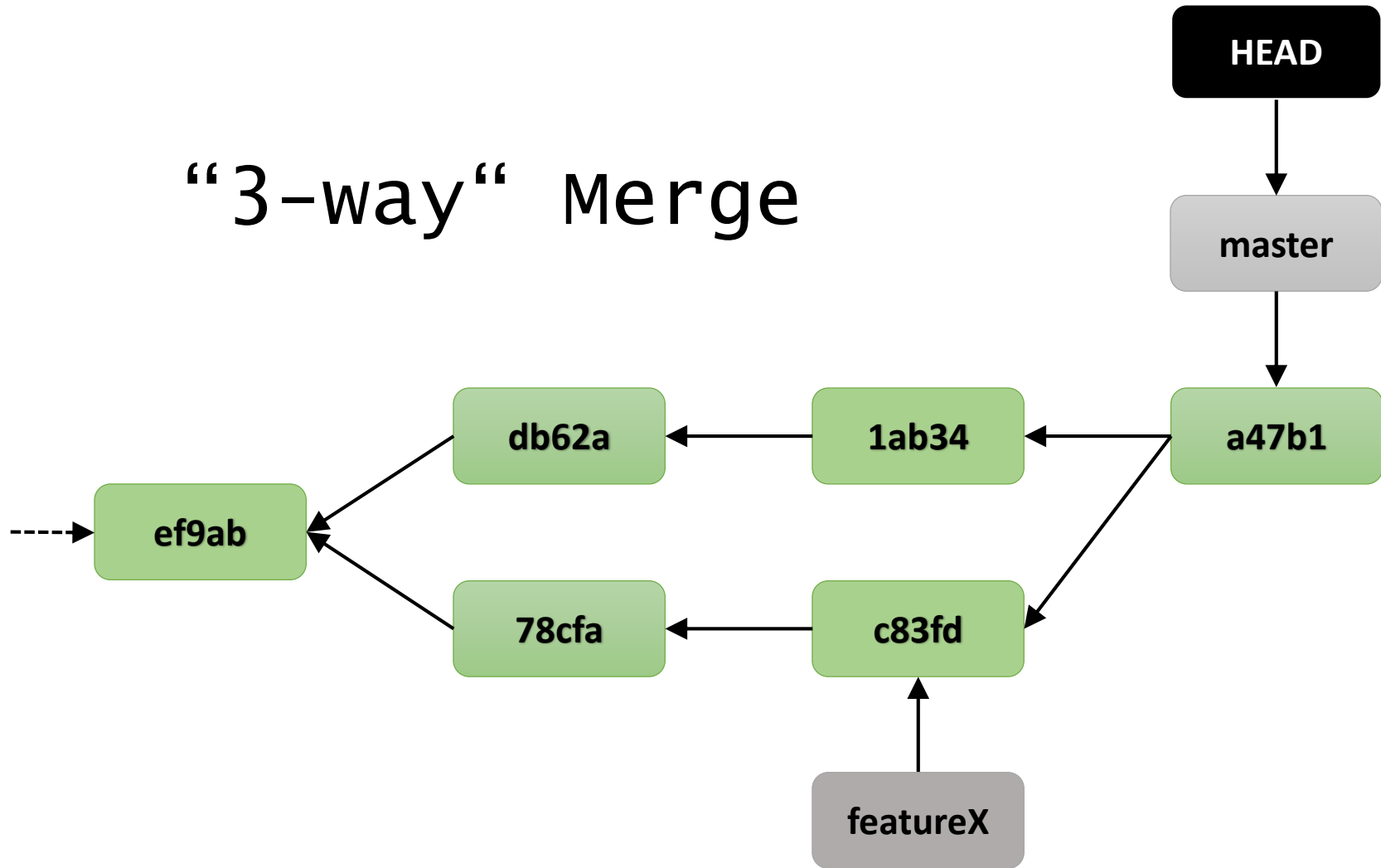
git merge featureX

# “3-way” Merge



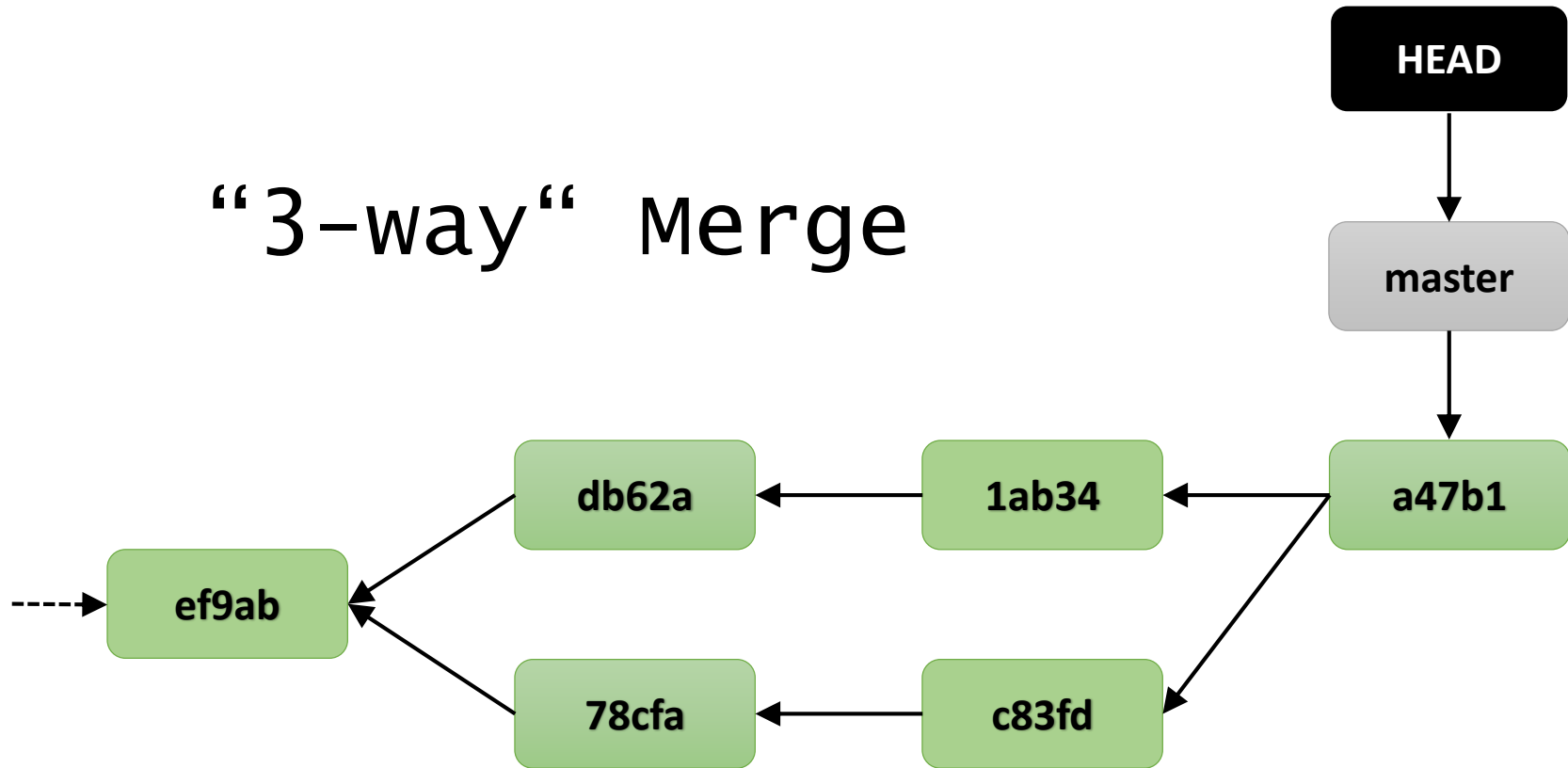
git merge featureX

# “3-way” Merge



`git branch -d featureX`

# “3-way” Merge



`git branch -d featureX`

# Resolving Conflicts

---

# Resolving Conflicts

---



# Resolving Conflicts

---

# Resolving Conflicts

---

```
1  
2 <<<<<<< HEAD  
3 This is the original change.  
4 =====  
5 This is a modified change.  
6 >>>>>>> example  
7
```

# Resolving Conflicts

Merge Revisions for /home/ben/git\_example/conflict.html

Local Changes

This is a simple modified line.	X >>	1
		2
Now I made some changes here in the master branch.	X >>	3
		4
		5
		6
This is an unmodified line.		7
Here is another line.	X >>	8
		9
		10
And I added a line here.		11

Merge Result

1	This is a simple modified line.
2	
3	This is a text block has also been modified by the other branch.
4	
5	
6	
7	This is an unmodified line.
8	
9	
10	

Changes from Server (revision 239ea9e9...)

1 << X	Modified this is a simple line.
2	
3 << X	This is a text block has also been modified by the other branch.
4	
5	
6	
7	This is an unmodified line.
8 << X	Modified another line.
9	
10	

No changes. 6 conflicts

Deleted Changed Inserted Conflict

Apply Cancel

# Remotes

---

```
git push <remote name> <branch name>  
git fetch <remote name> <branch name>  
git pull <remote name> <branch name>
```

# Remotes

---

# Remotes

---

- Basis for **collaboration** and **work sharing**

# Remotes

---

- Basis for **collaboration** and **work sharing**
- **Non-local** repositories of the same project

# Remotes

---

- Basis for **collaboration** and **work sharing**
- **Non-local** repositories of the same project
- Usually accessed via **HTTP(S)** or **SSH**

# Remotes

---

- Basis for **collaboration** and **work sharing**
- **Non-local** repositories of the same project
- Usually accessed via **HTTP(S)** or **SSH**
- A project can have **multiple remotes**

# Remotes

---

- Basis for **collaboration** and **work sharing**
- **Non-local** repositories of the same project
- Usually accessed via **HTTP(S)** or **SSH**
- A project can have **multiple remotes**
- Default remote of a cloned repository: **origin**

# Remotes

---

- Basis for **collaboration** and **work sharing**
- **Non-local** repositories of the same project
- Usually accessed via **HTTP(S)** or **SSH**
- A project can have **multiple remotes**
- Default remote of a cloned repository: **origin**
- Remote branches are **tracked**: origin/master

# Remote Interaction

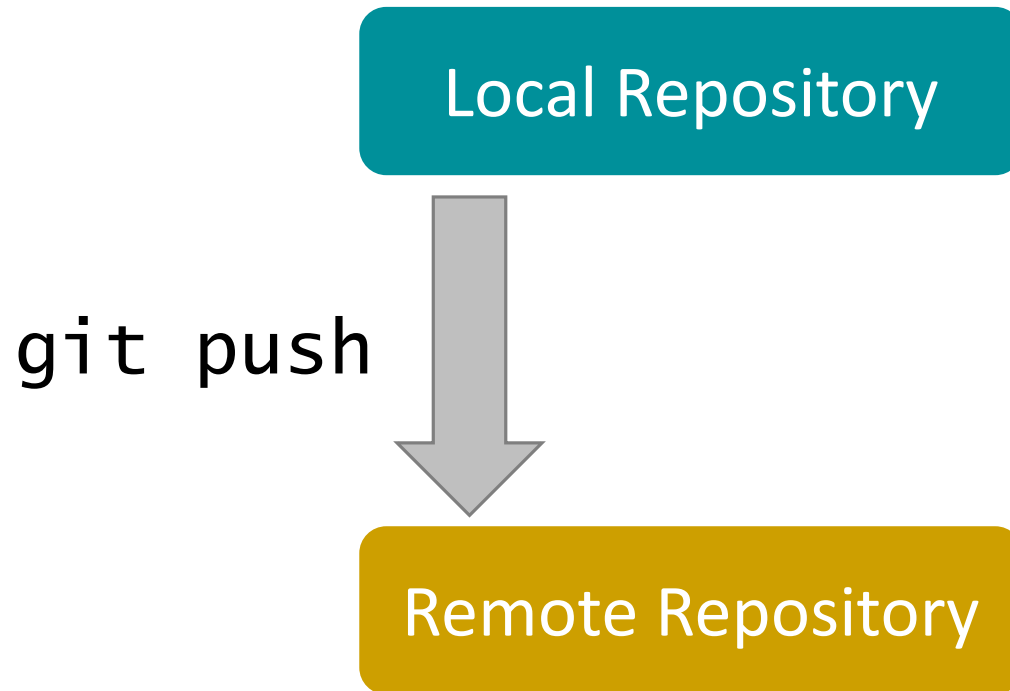
---

Local Repository

Remote Repository

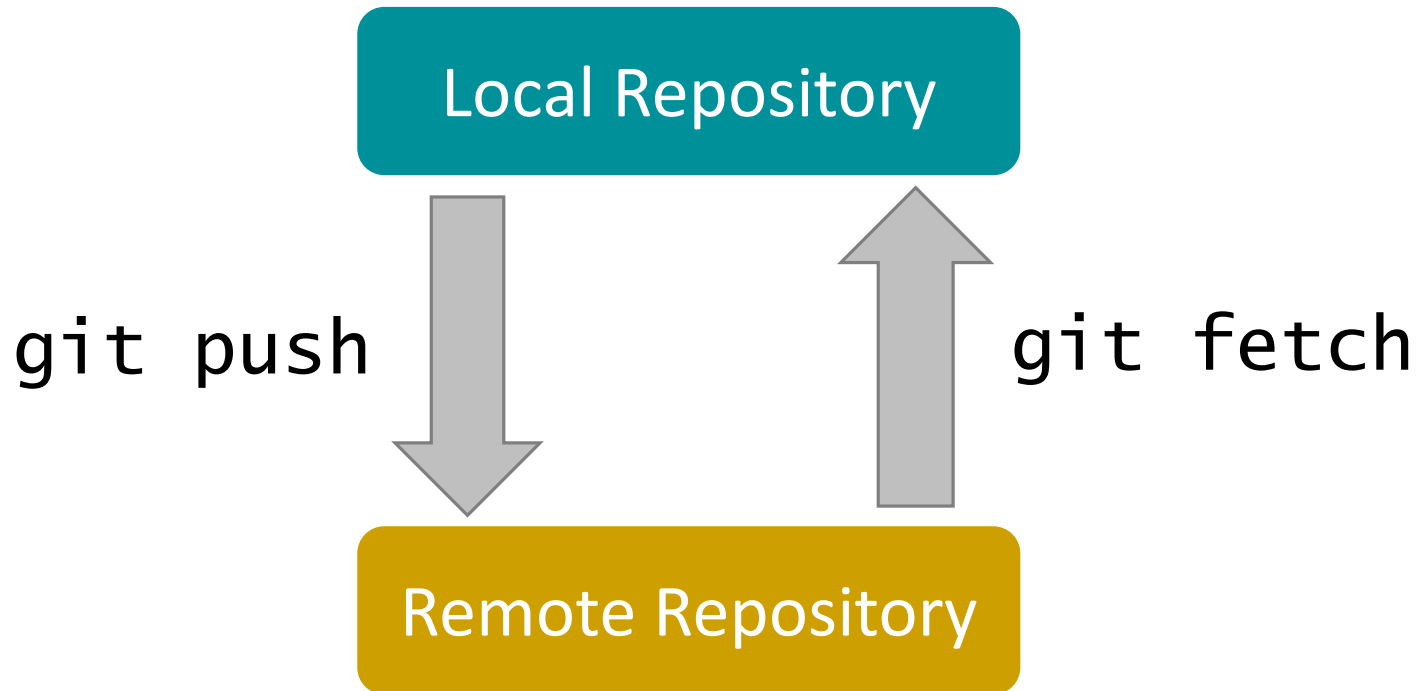
# Remote Interaction

---



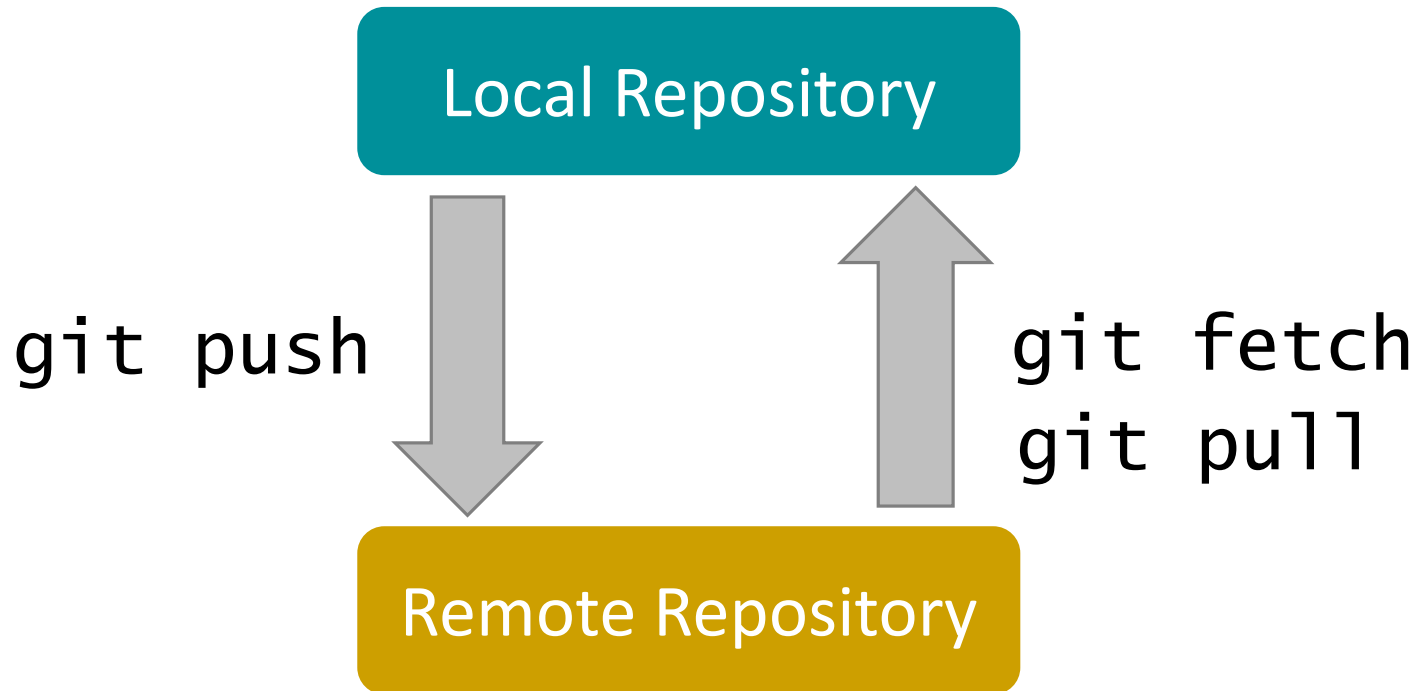
# Remote Interaction

---



# Remote Interaction

---



origin

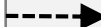
featureX



db62a



ef9ab

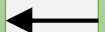


local

featureX



db62a



ef9ab



origin/  
featureX



origin

featureX



db62a



ef9ab



local

featureX



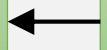
c83fd



1ab34



db62a

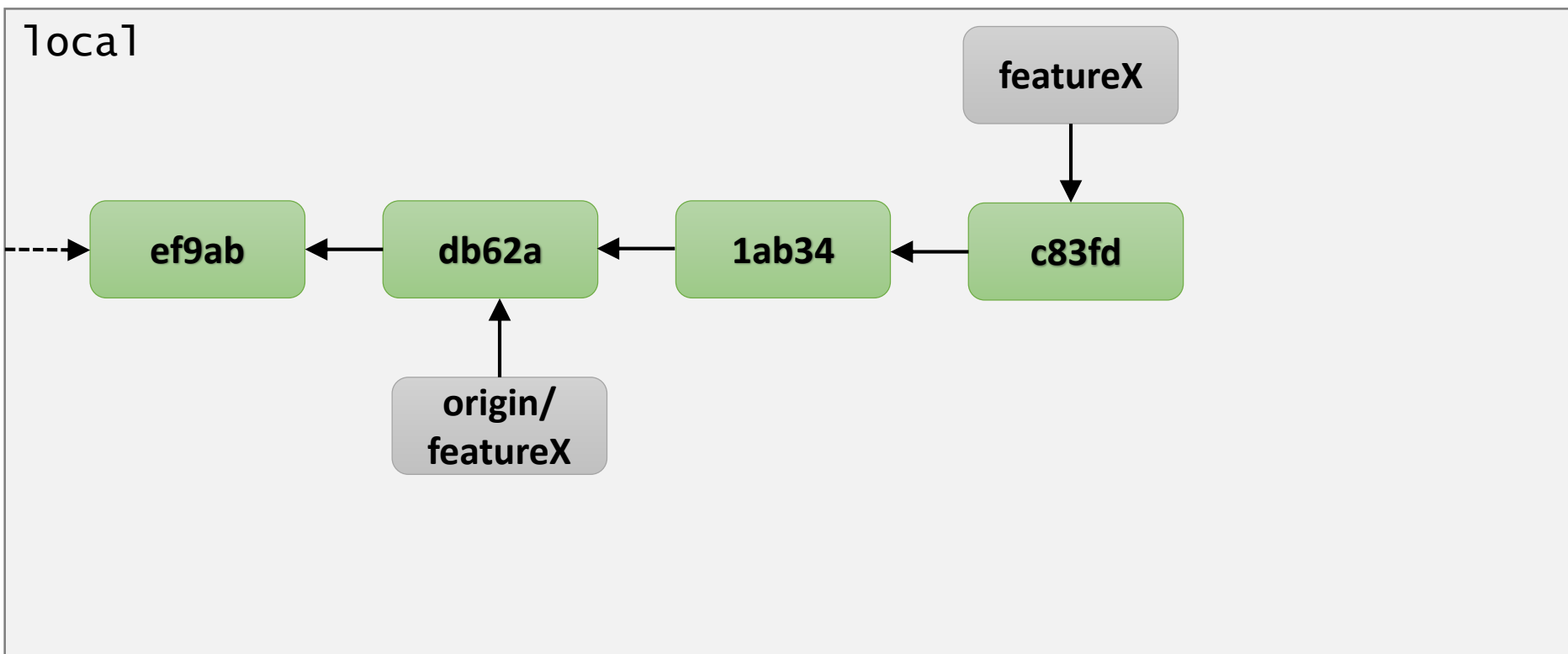
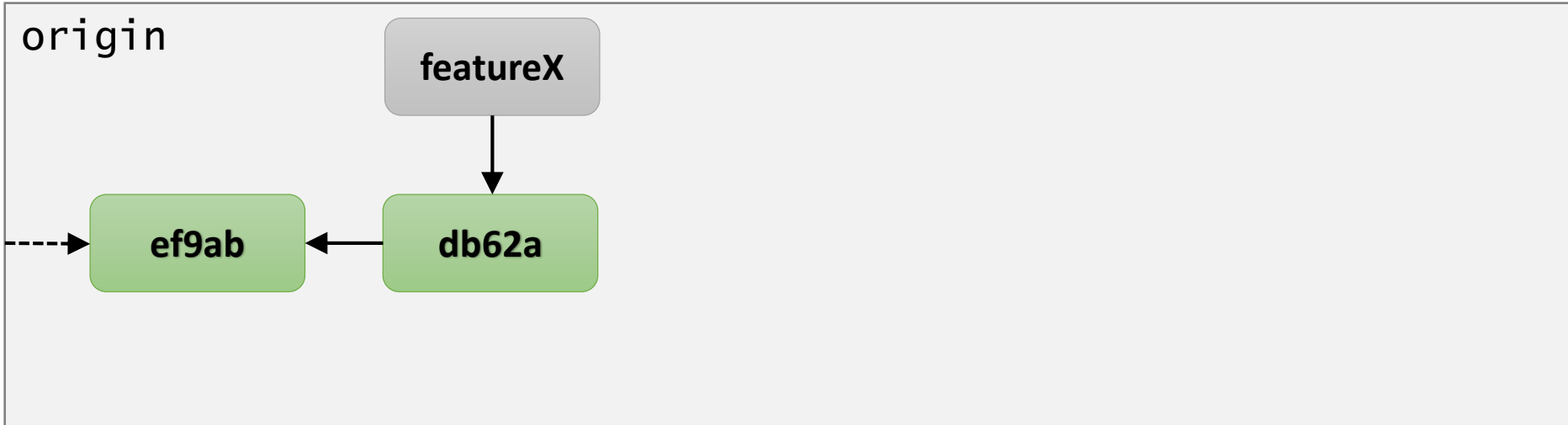


ef9ab

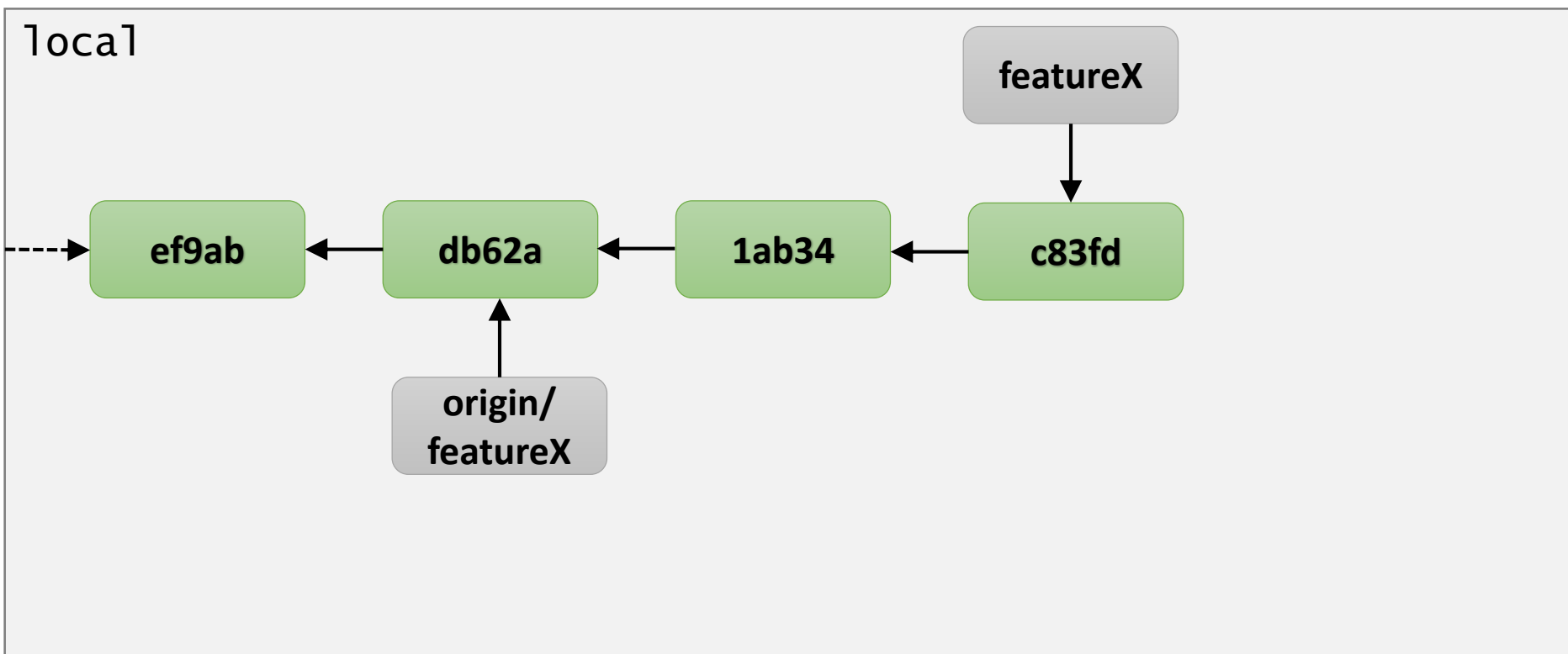
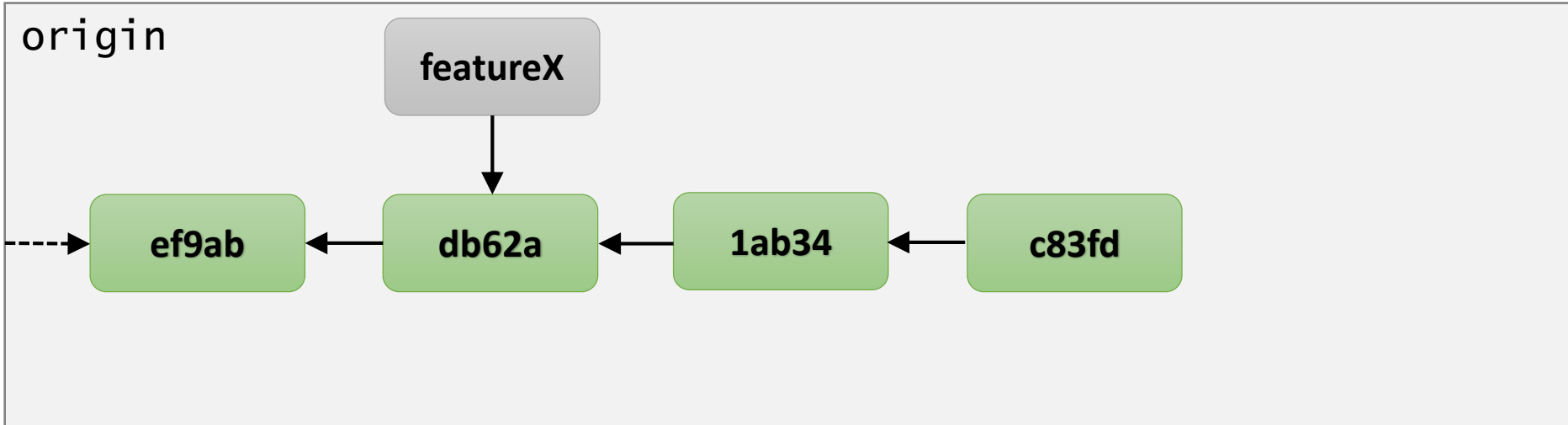


origin/  
featureX

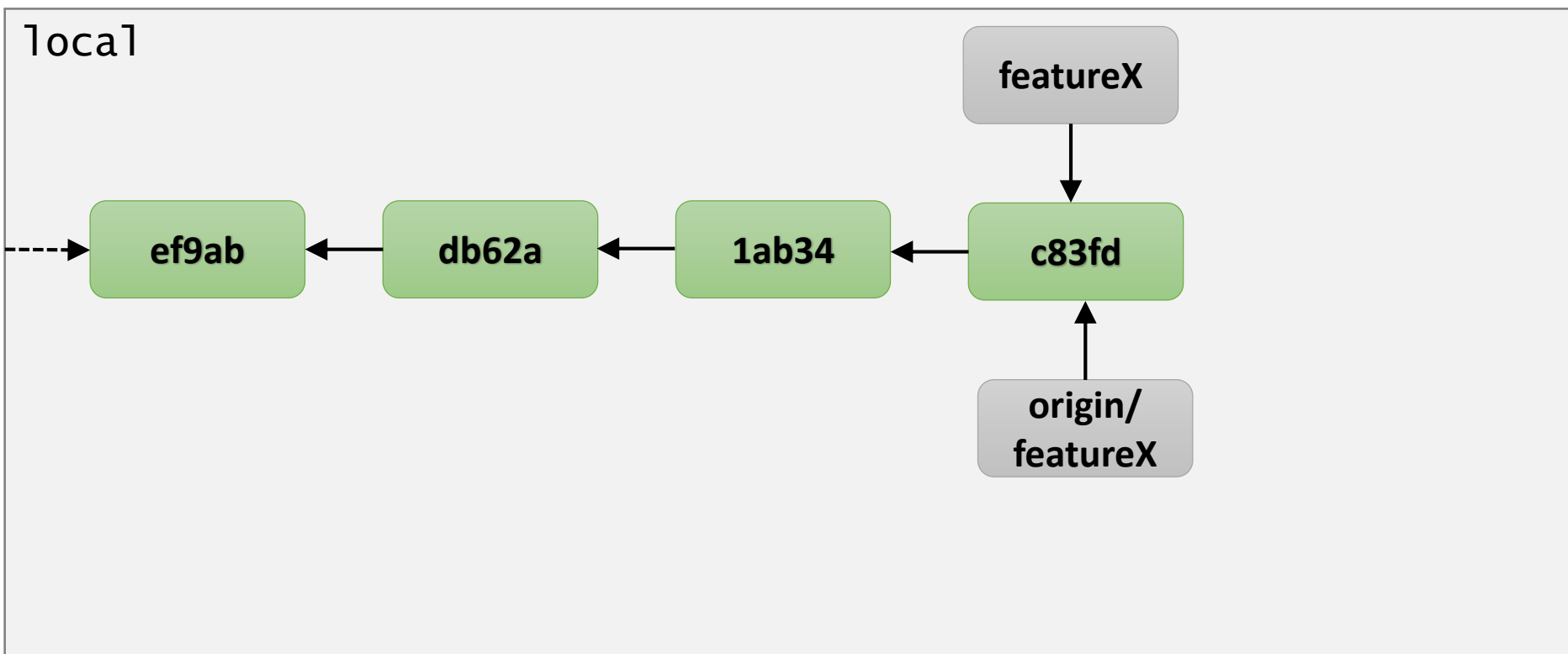
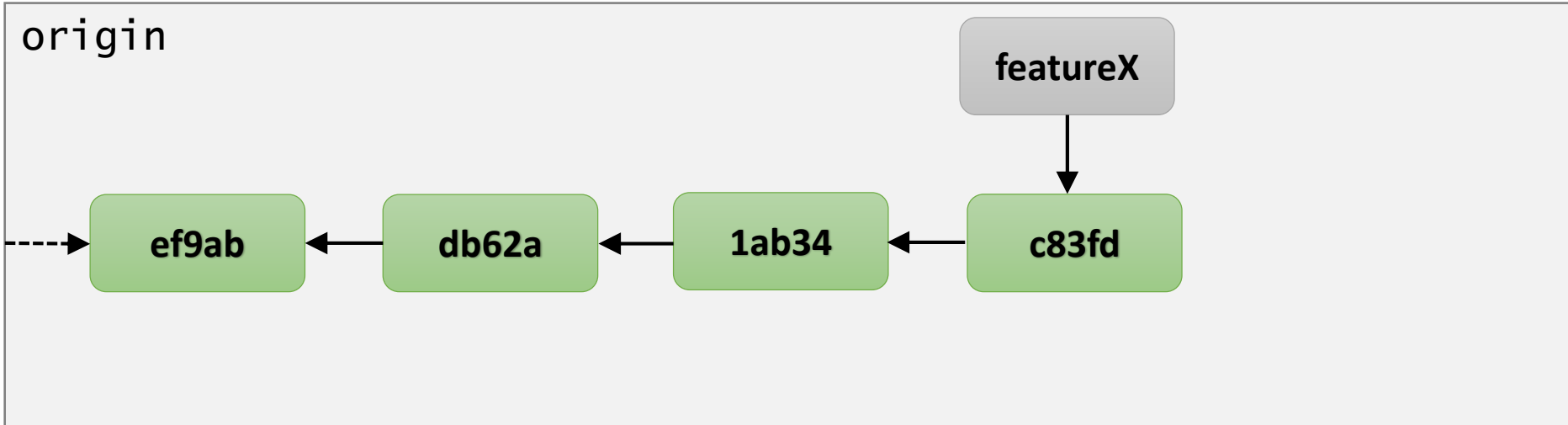




git push origin featureX



git push origin featureX



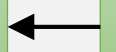
git push origin featureX

origin

featureX



db62a



ef9ab

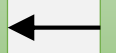


local

featureX



db62a



ef9ab



origin/  
featureX



origin

featureX



db62a



ef9ab

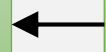


local

featureX



c83fd



1ab34



db62a



ef9ab



origin/  
featureX



origin

featureX

ef9ab

db62a

a47b1

fe12a

local

featureX

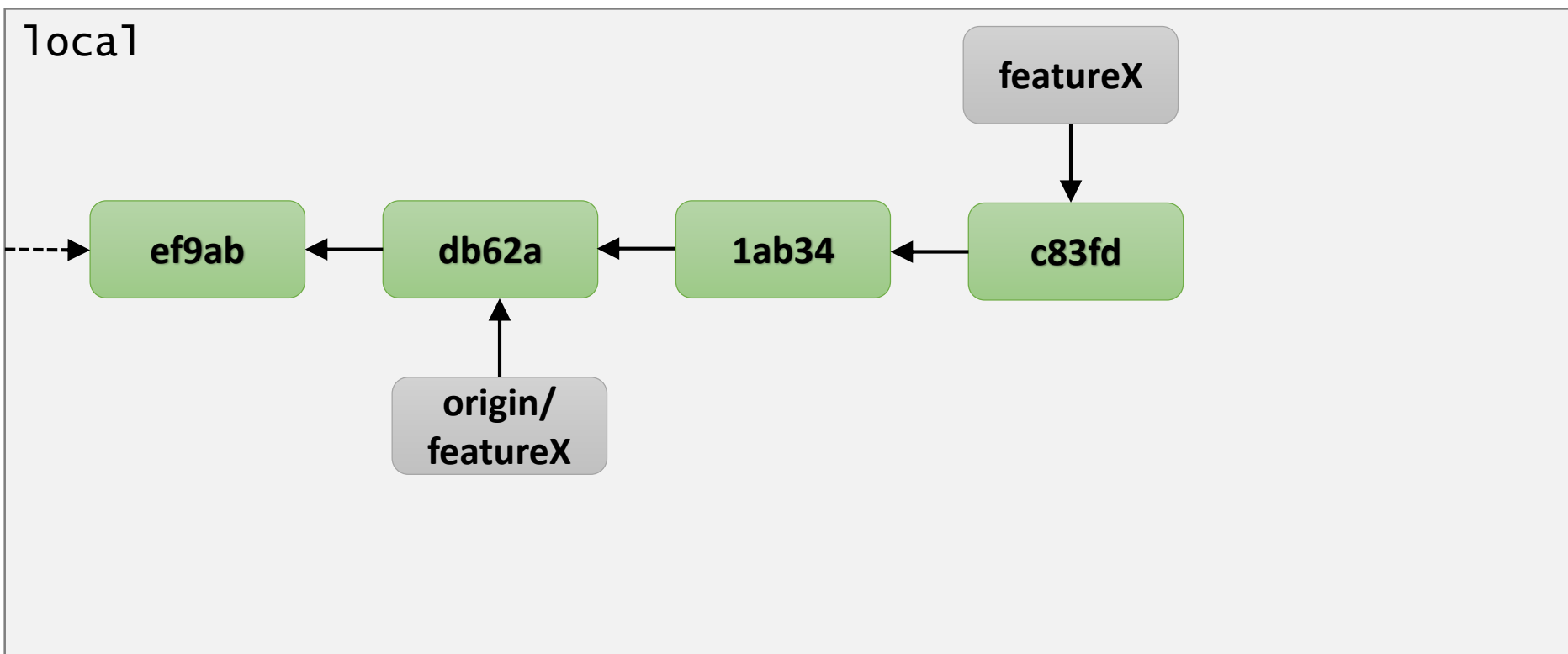
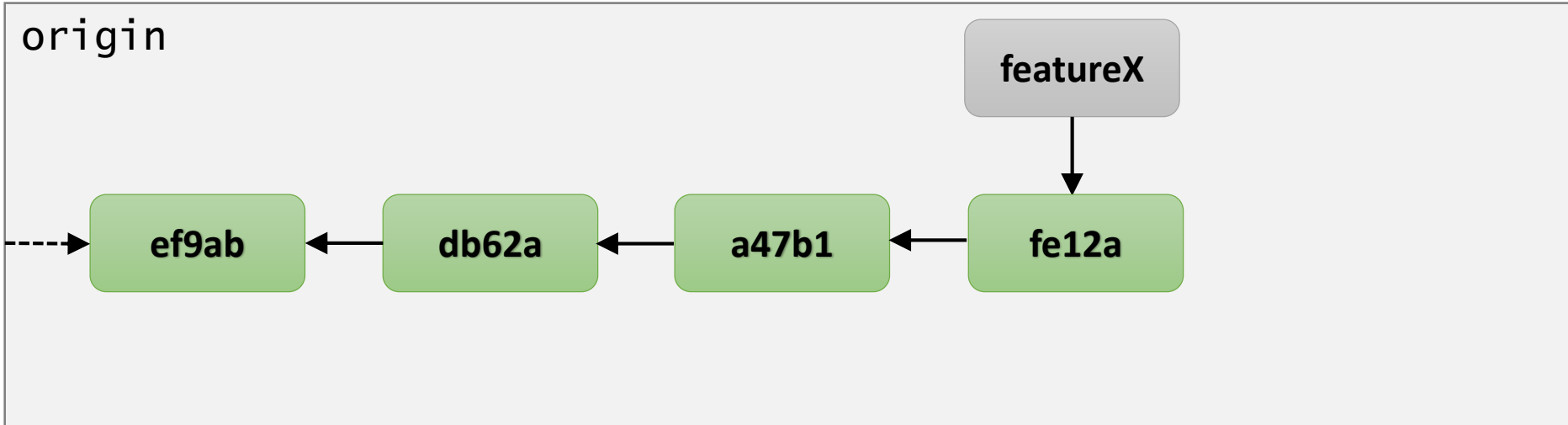
ef9ab

db62a

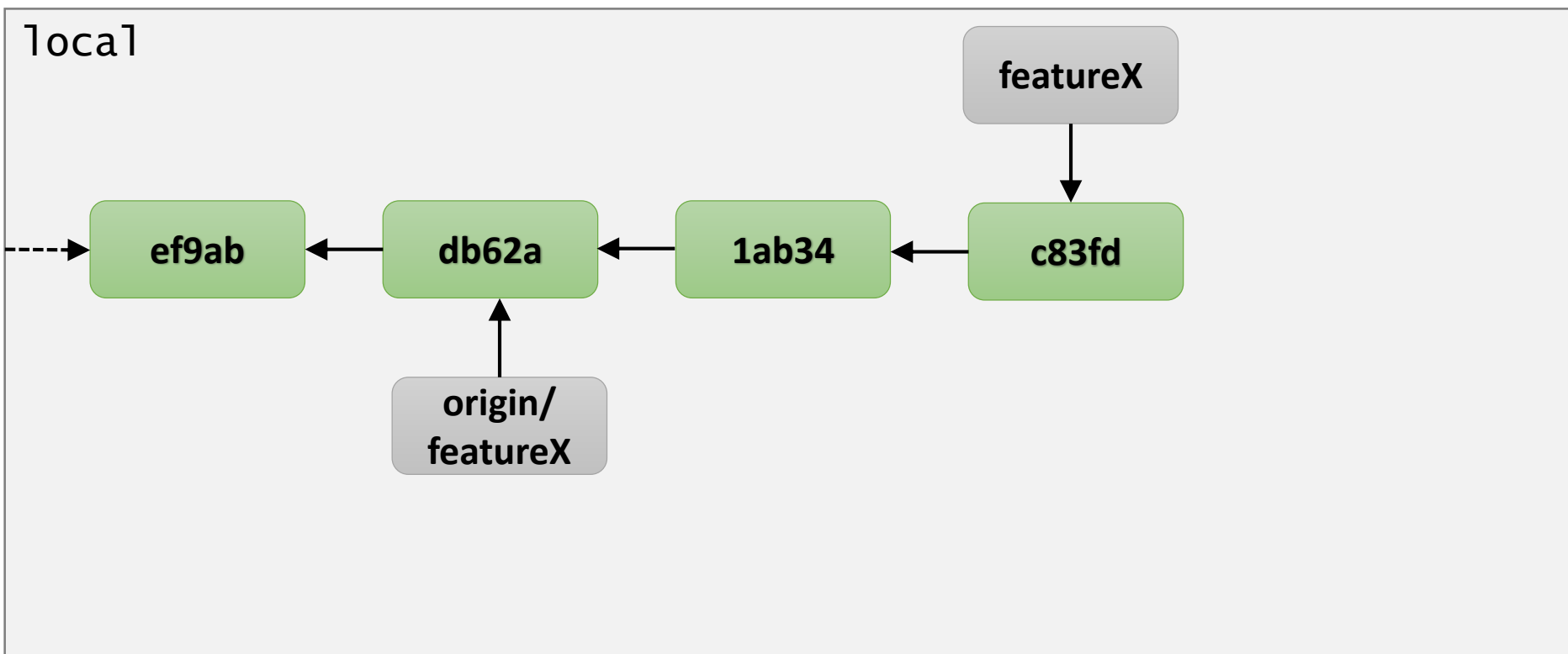
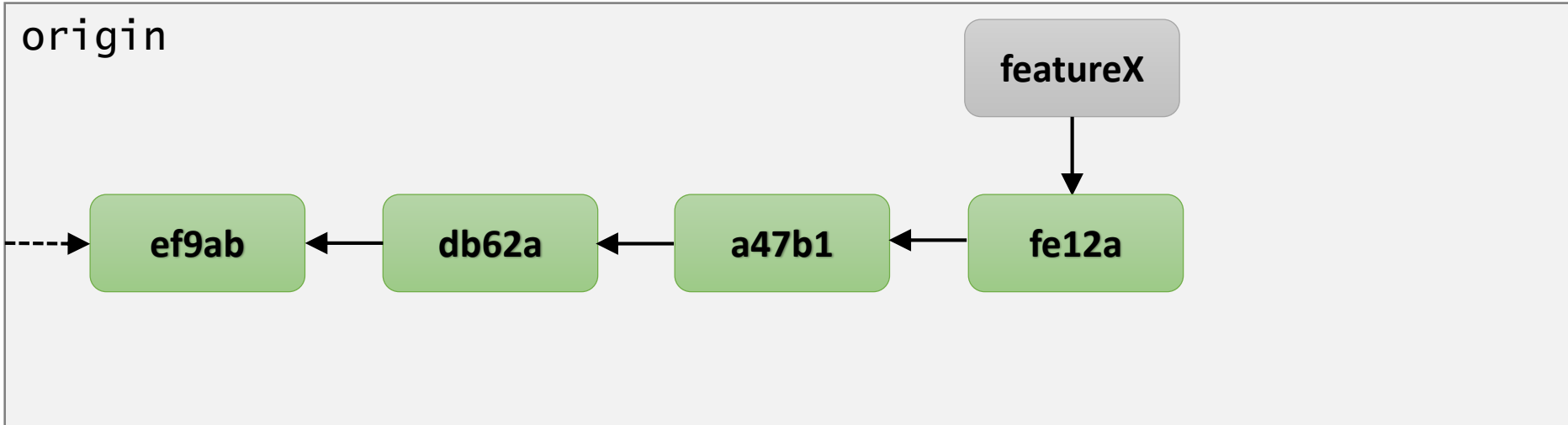
1ab34

c83fd

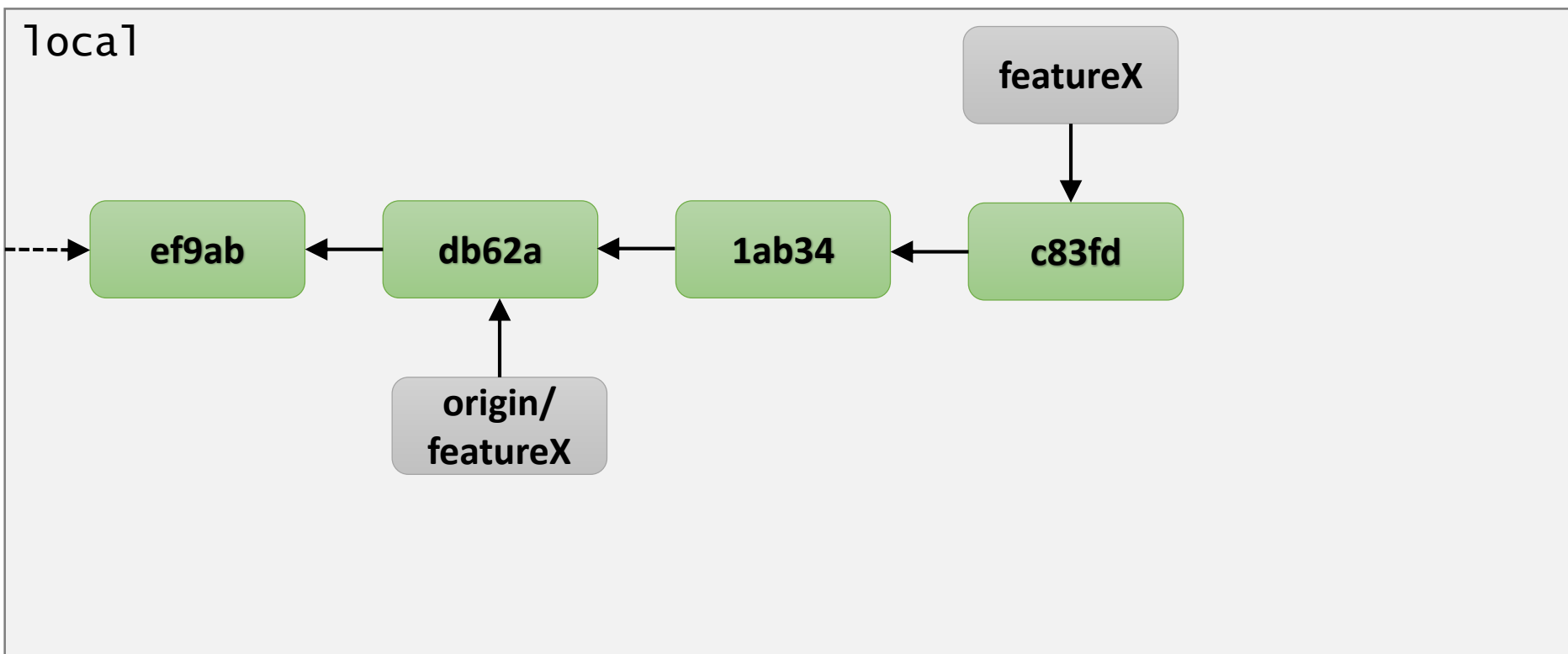
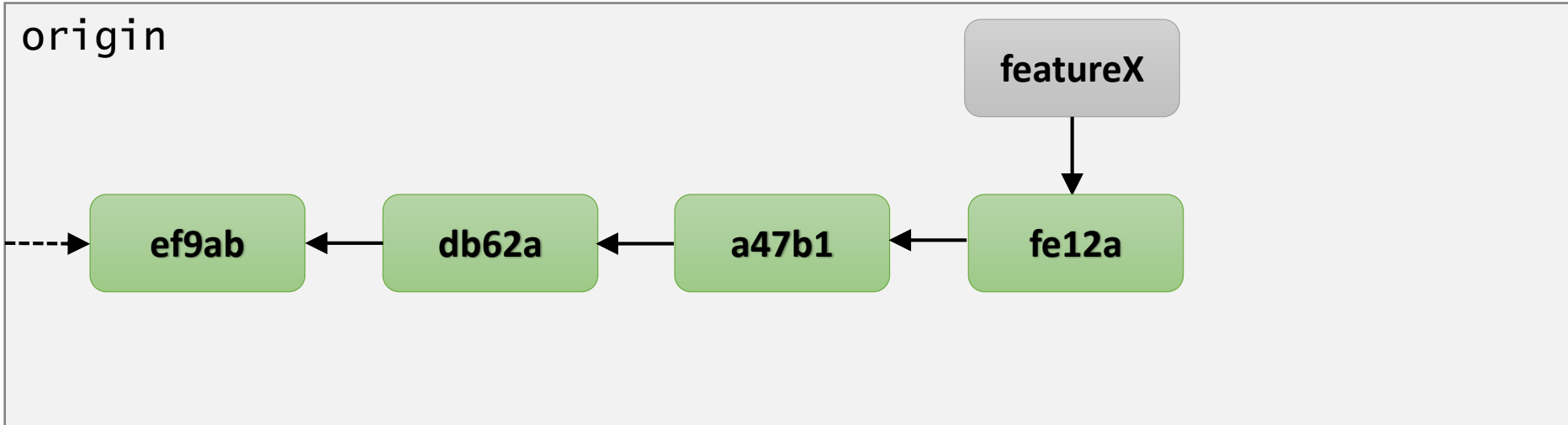
origin/  
featureX



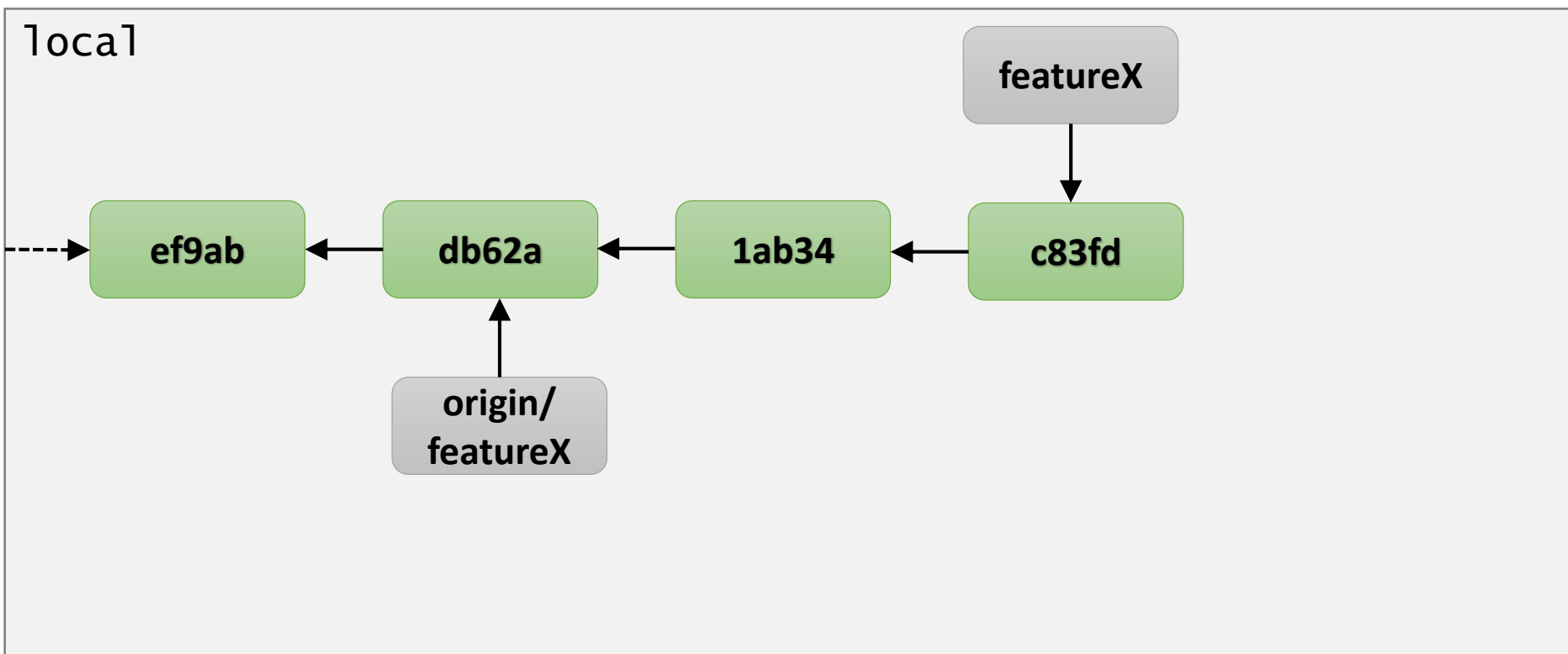
git push origin featureX



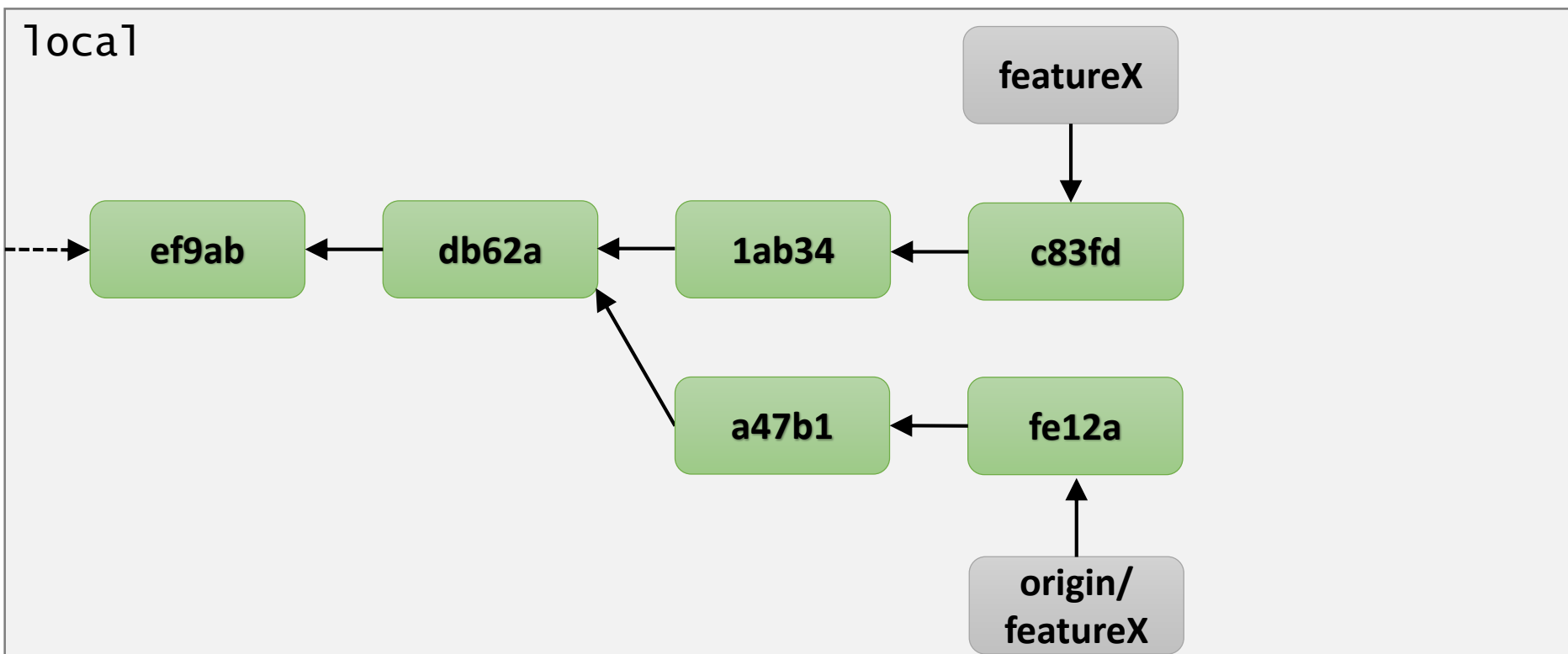
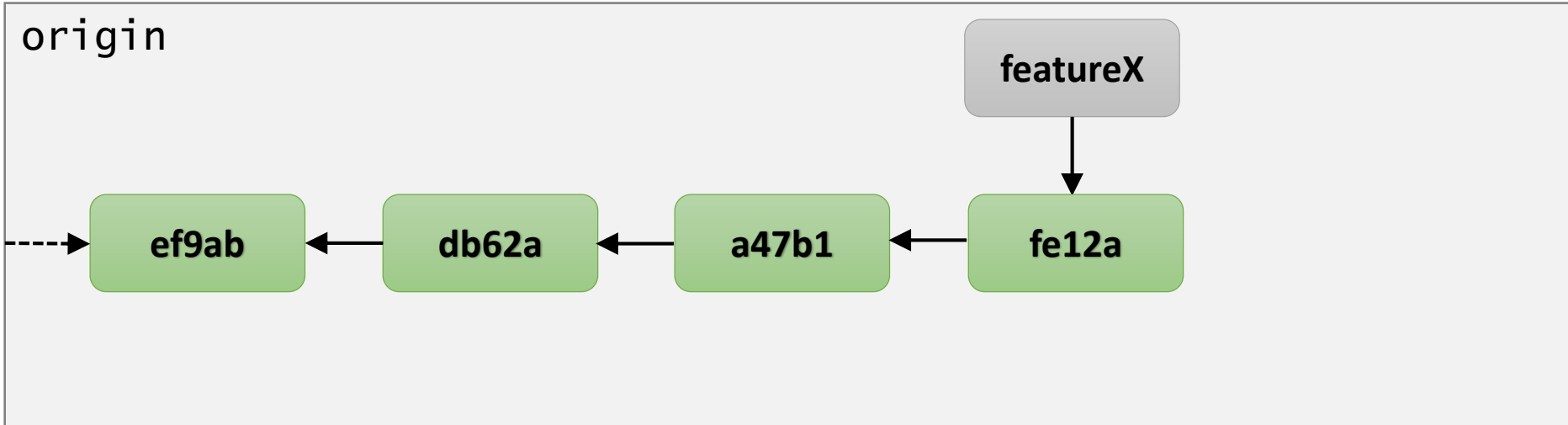
git push origin featureX



git push origin featureX 

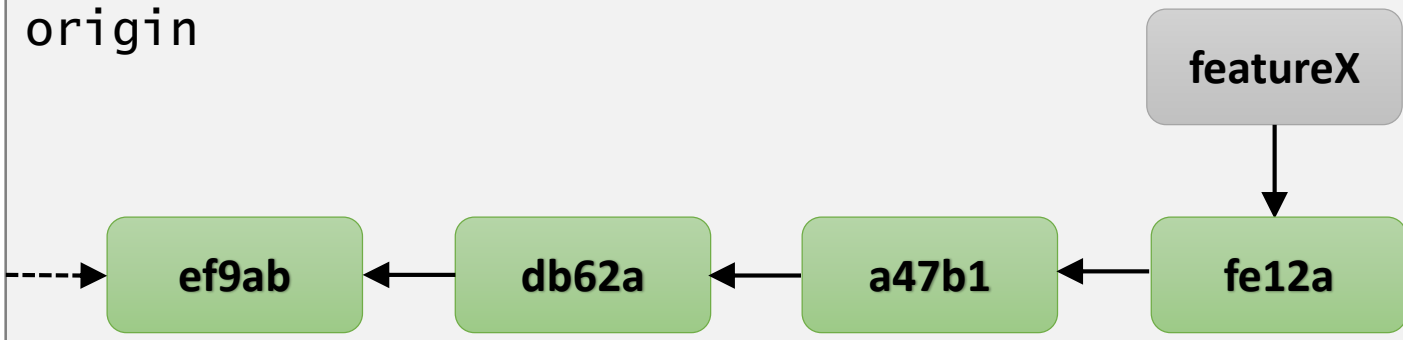


git fetch origin

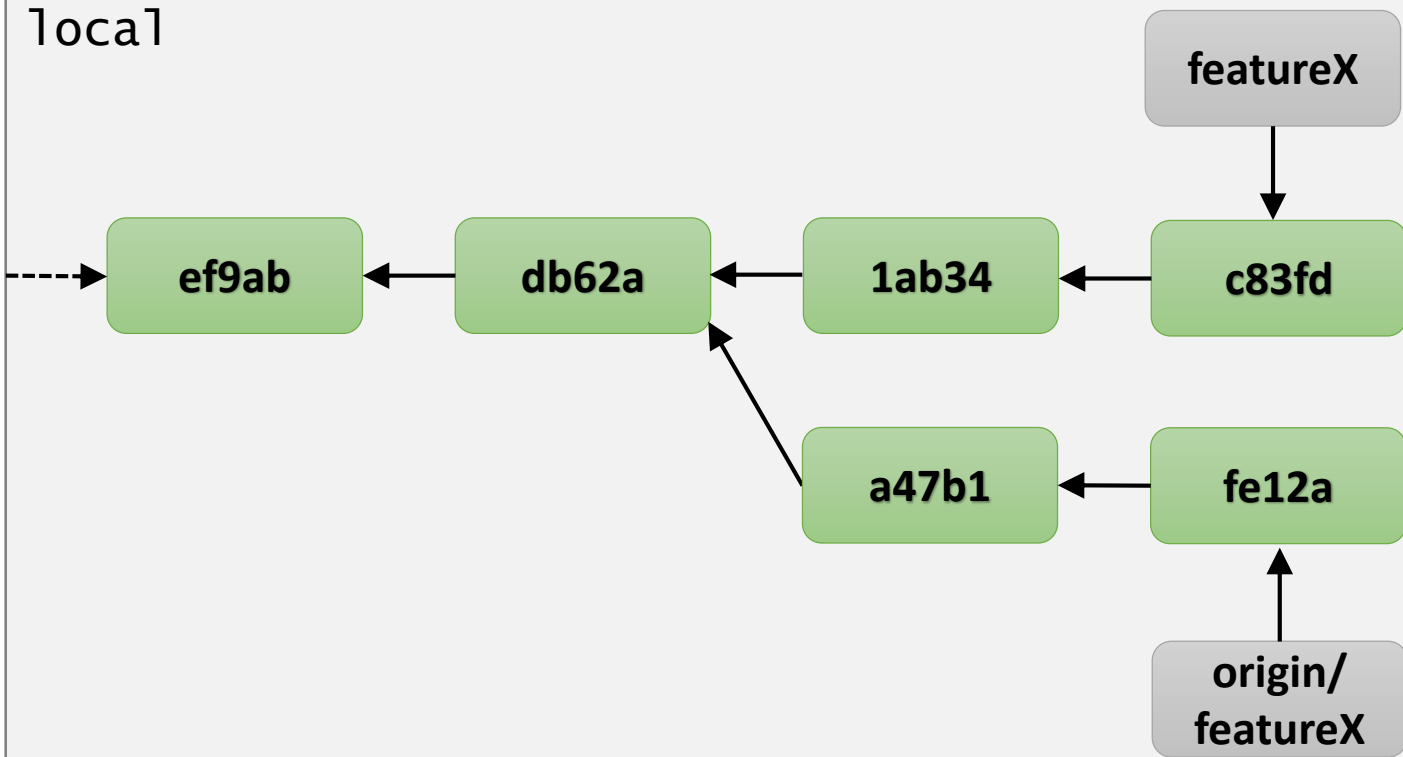


git fetch origin

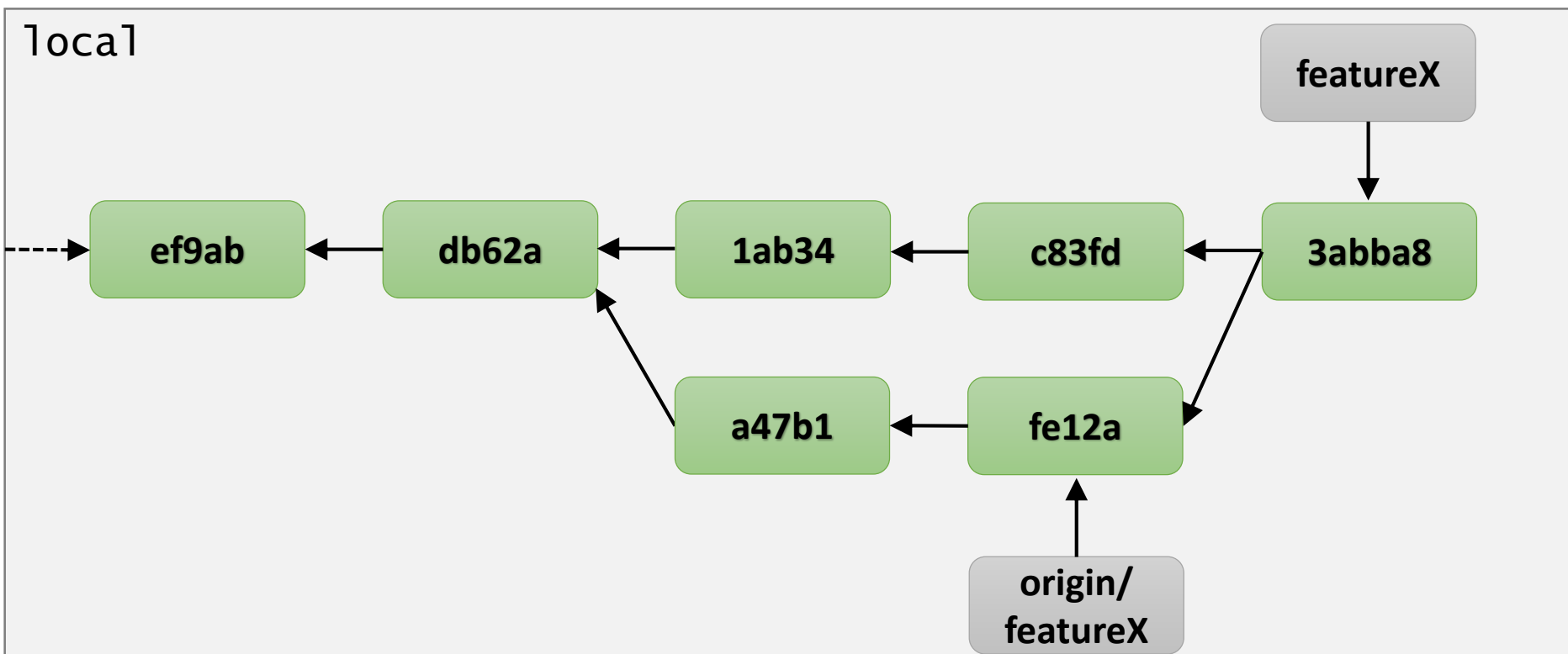
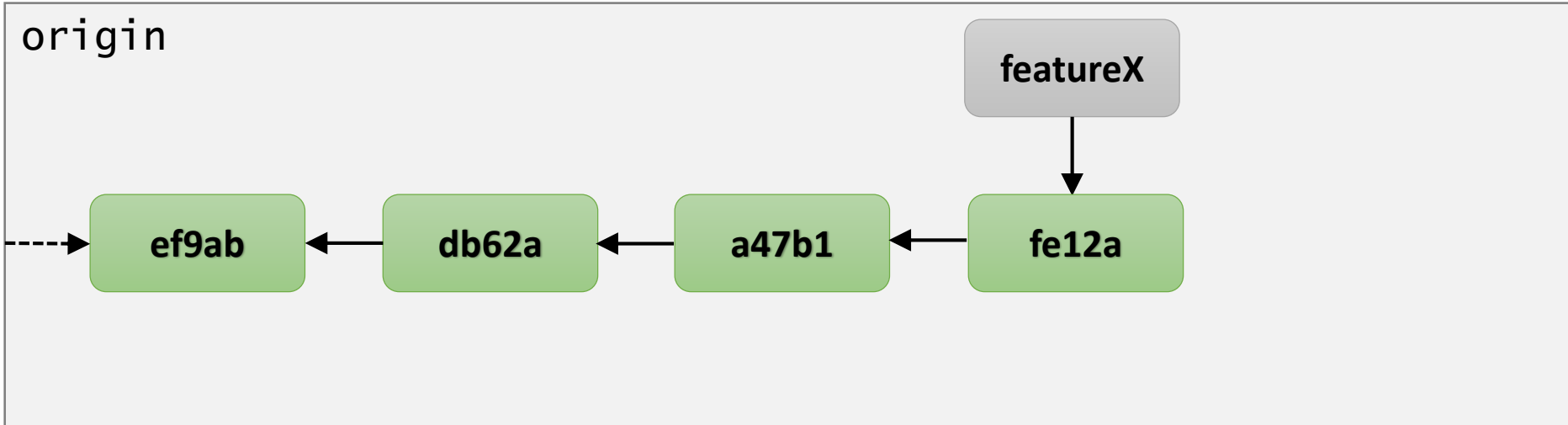
origin



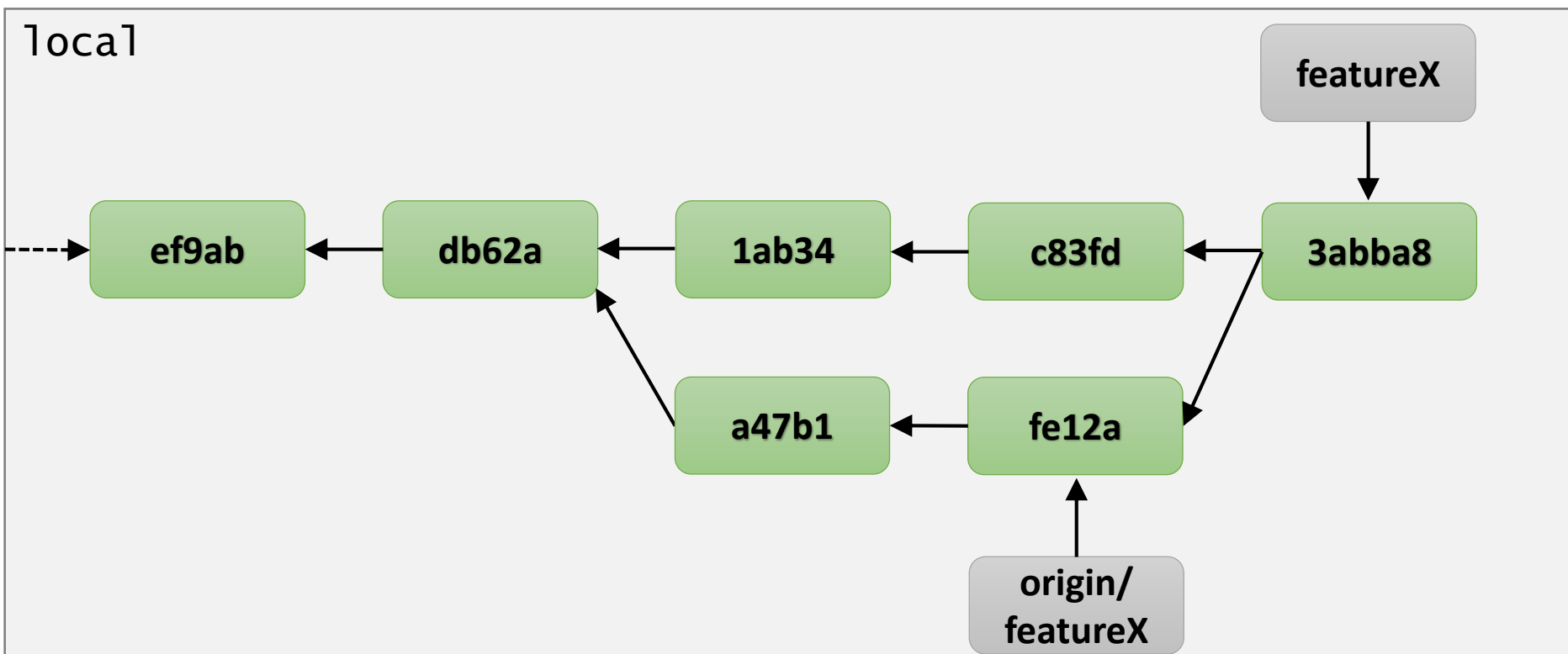
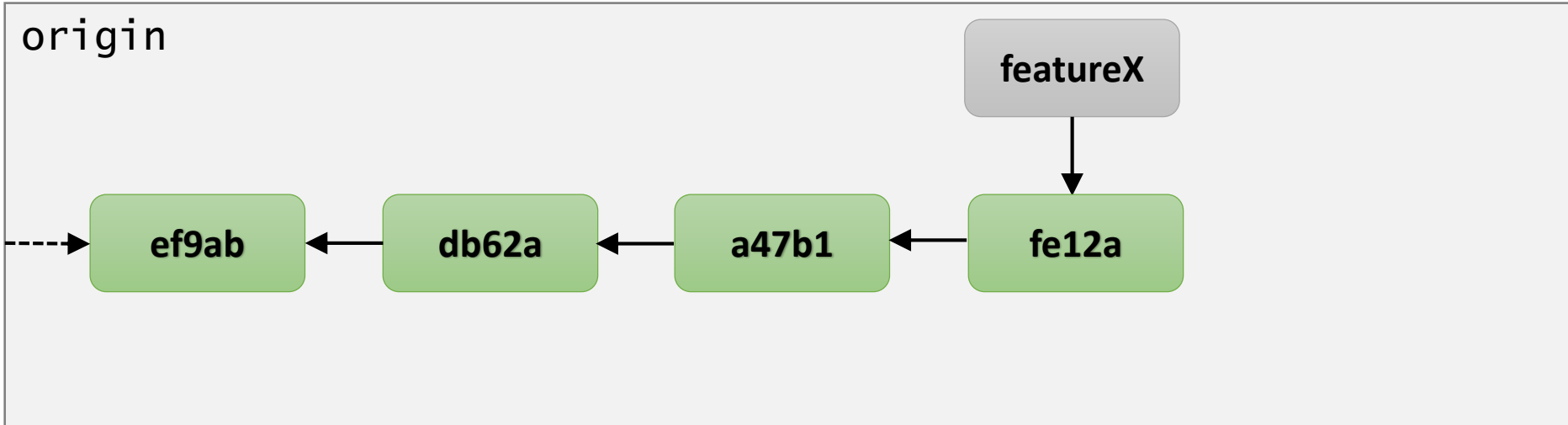
local



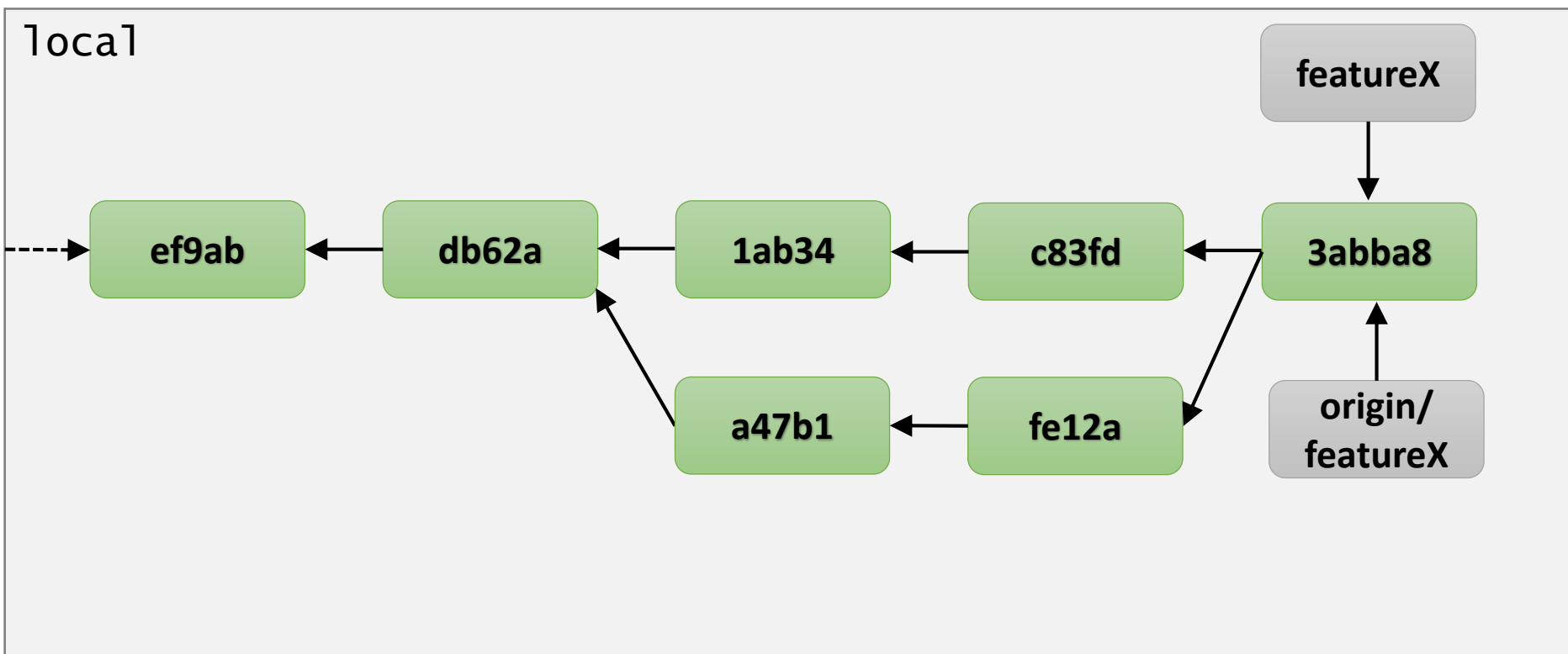
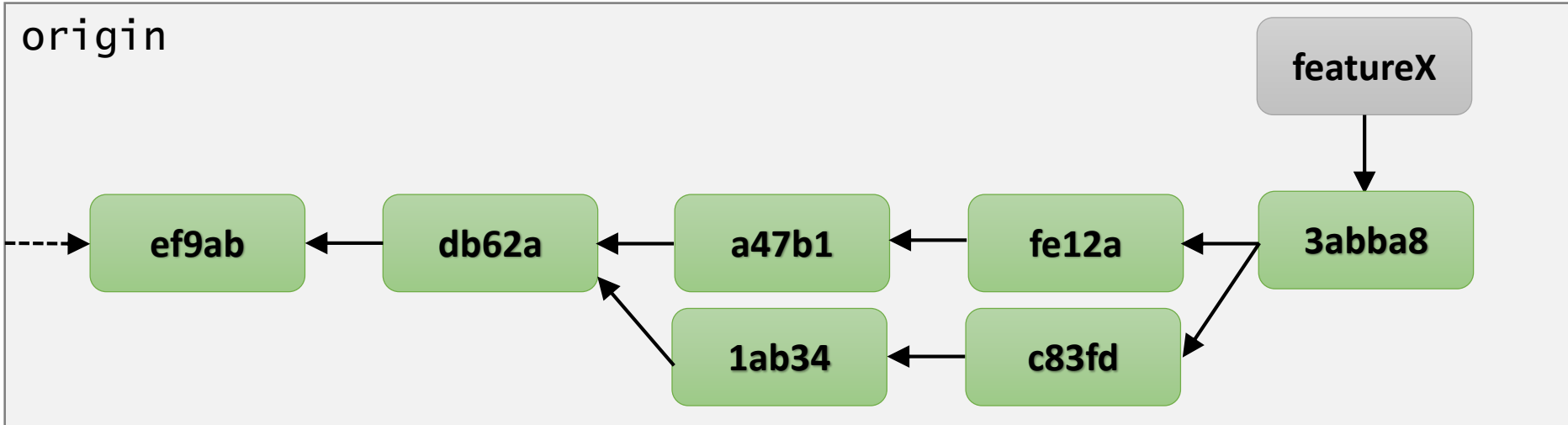
git merge origin/featureX



git merge origin/featureX



git push origin featureX



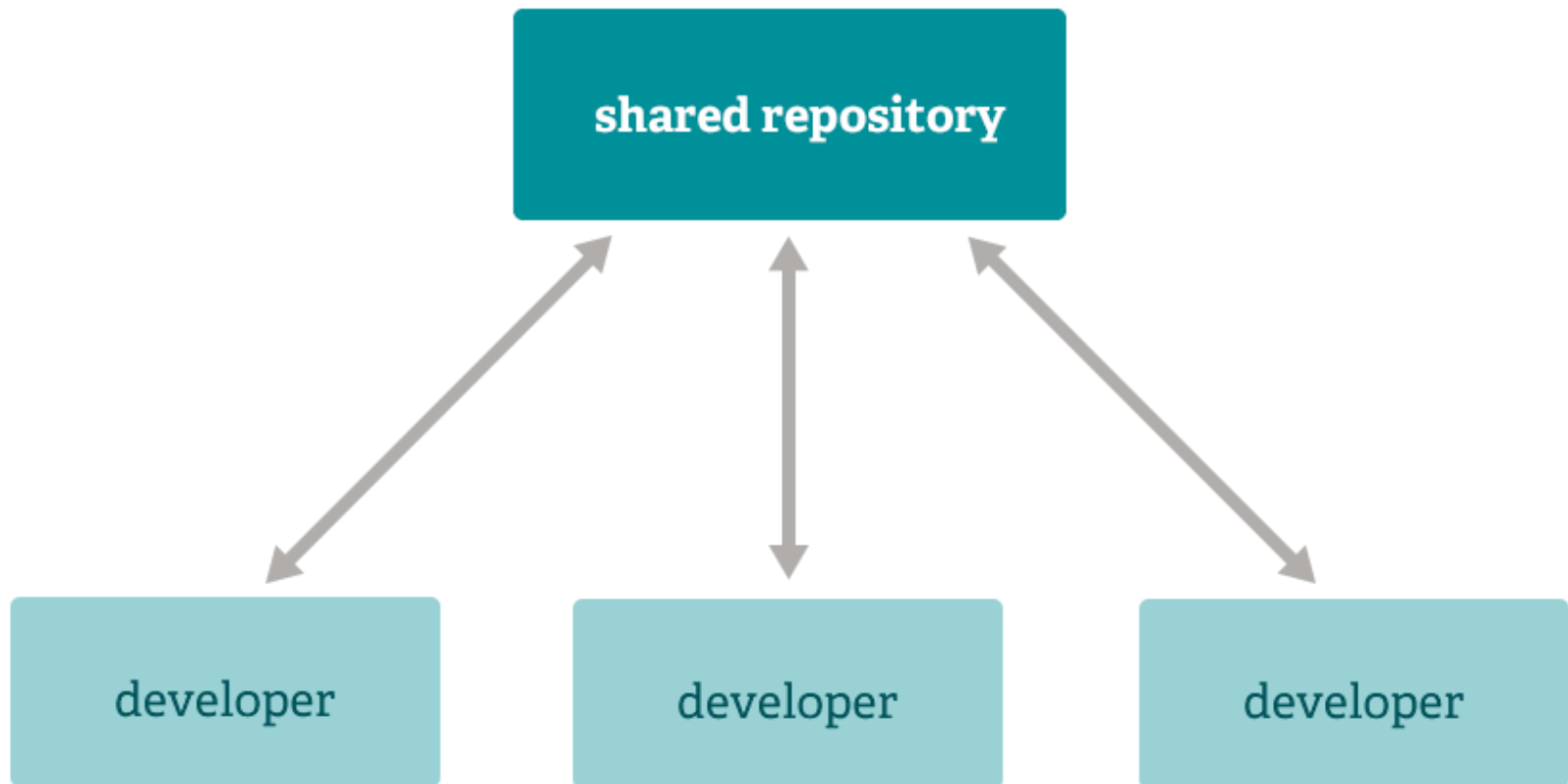
git push origin featureX 

# Central Repository Workflow

---

# Central Repository Workflow

---

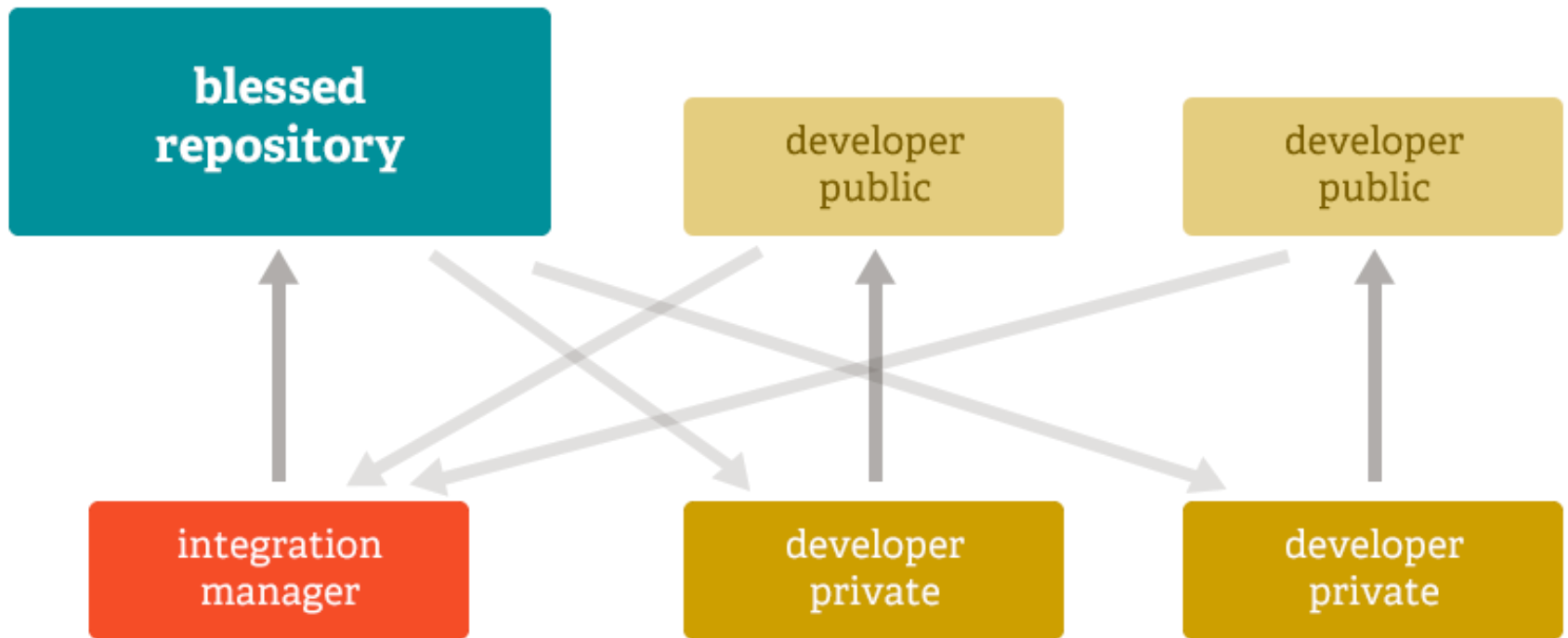


# Integration Manager Workflow

---

# Integration Manager Workflow

---

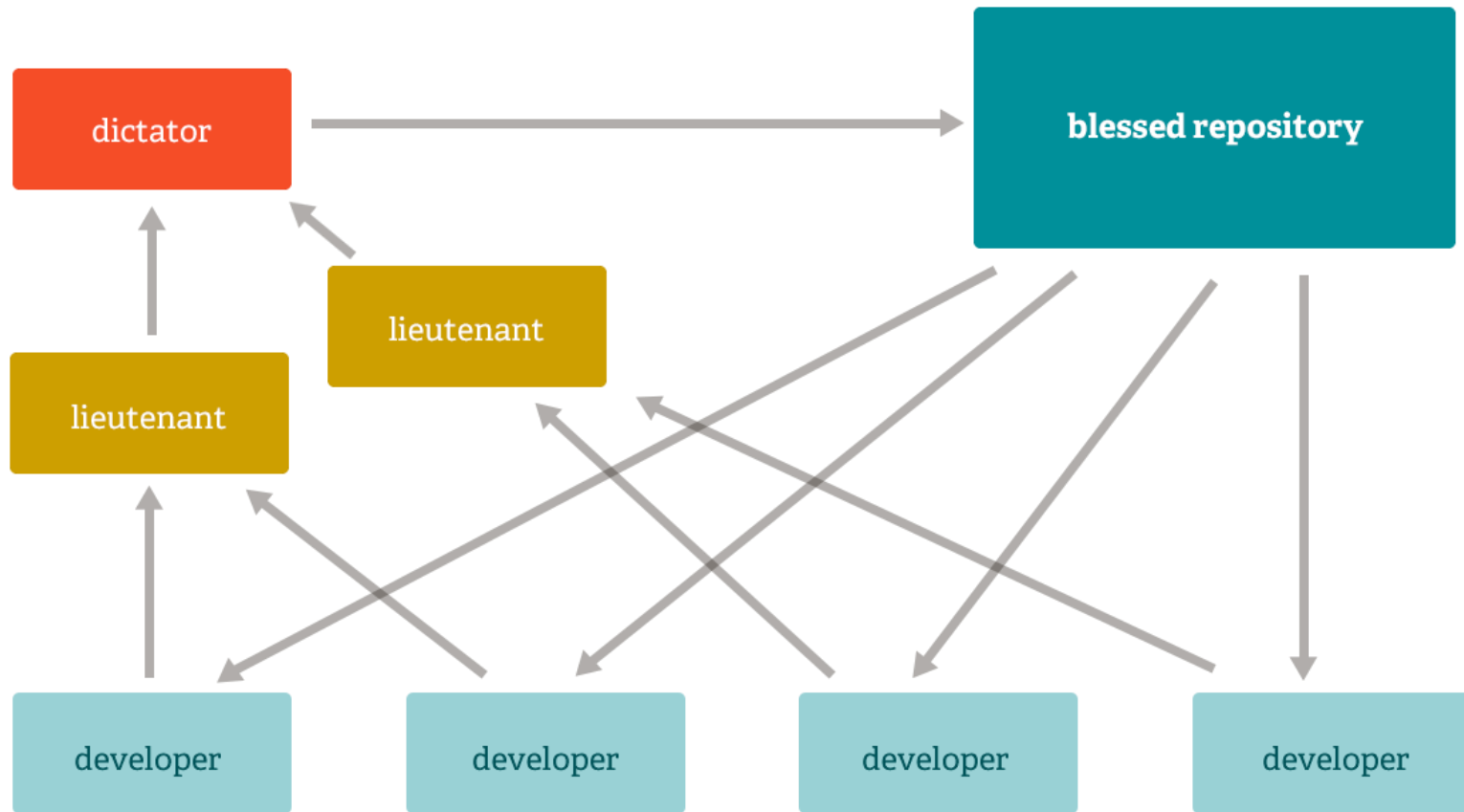


# Dictator-Lieutenants Workflow

---

# Dictator-Lieutenants Workflow

---



# Personal Recommendations

---

# Personal Recommendations

---

- **Forget** what you know about version control

# Personal Recommendations

---

- **Forget** what you know about version control
- Try to use the **command line** at the beginning

# Personal Recommendations

---

- **Forget** what you know about version control
- Try to use the **command line** at the beginning
- Read the Git **command output** carefully

# Personal Recommendations

---

- **Forget** what you know about version control
- Try to use the **command line** at the beginning
- Read the Git **command output** carefully
- Best way to learn: **set up** a repo and **play** with it

# Personal Recommendations

---

- **Forget** what you know about version control
- Try to use the **command line** at the beginning
- Read the Git **command output** carefully
- Best way to learn: **set up** a repo and **play** with it
- Hang in there, **it's worth it!**

# Resources

---

- <http://git-scm.com/> (Excellent docs, tutorials, and resources)
- <http://git-scm.com/book/> (Pro Git, a free Git book)
- <http://gitref.org/> (Quick reference for Git commands)
- <https://www.atlassian.com/en/git/> (Nice visual tutorials & workflows)
- <http://try.github.io/> (Small live tutorial)
- <http://www.youtube.com/> „Git“

So Git?

---

# So Git?

---

**¡Gracias por vuestra atención!**

¿Preguntas?

---

# Backup Slides

---

# GitHub



---

Social Coding

[www.github.com](https://github.com)

# GitHub Workflow

---

# GitHub Workflow

---

Main  
Repository

Owner

# GitHub Workflow

---

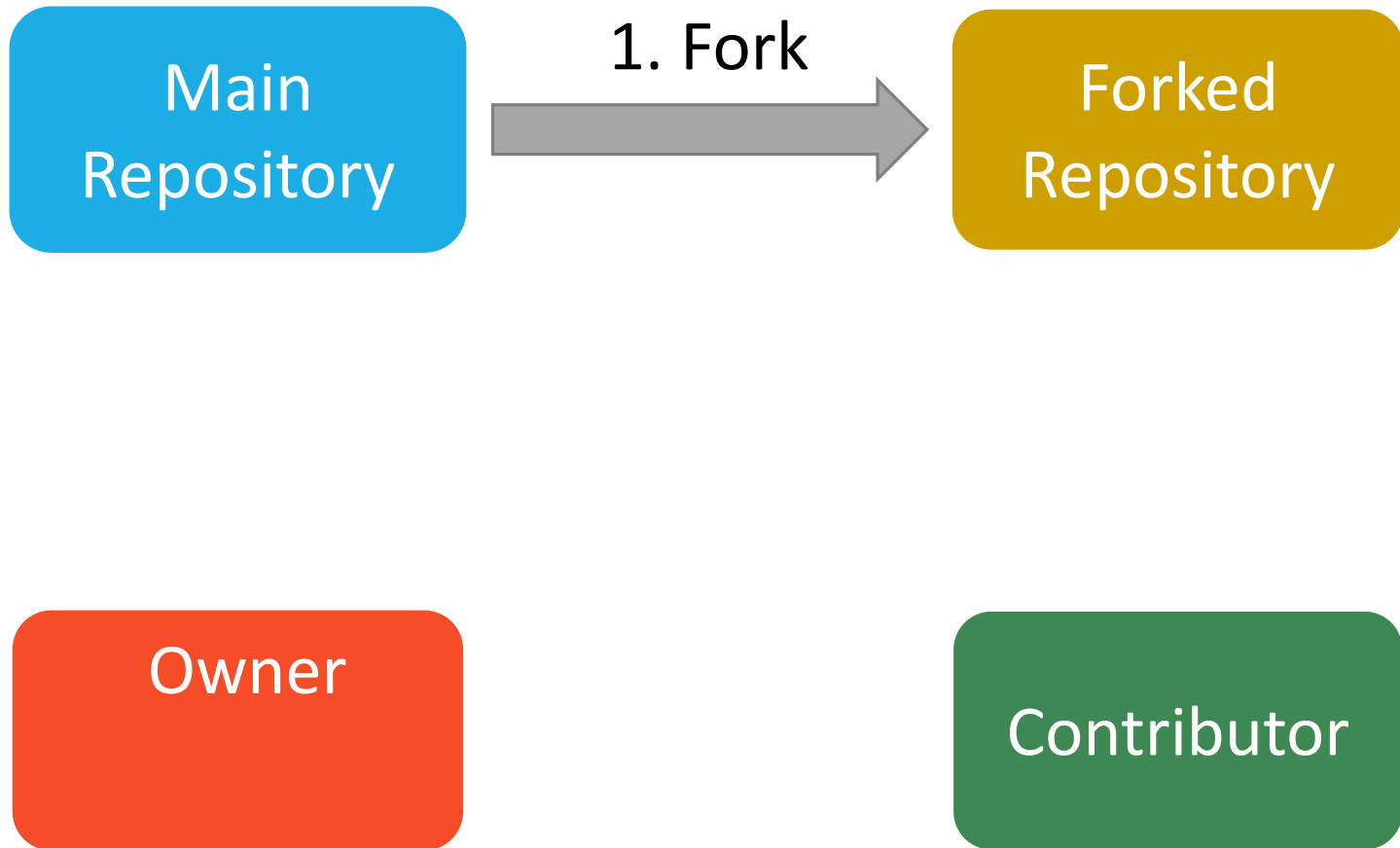
Main  
Repository

Owner

Contributor

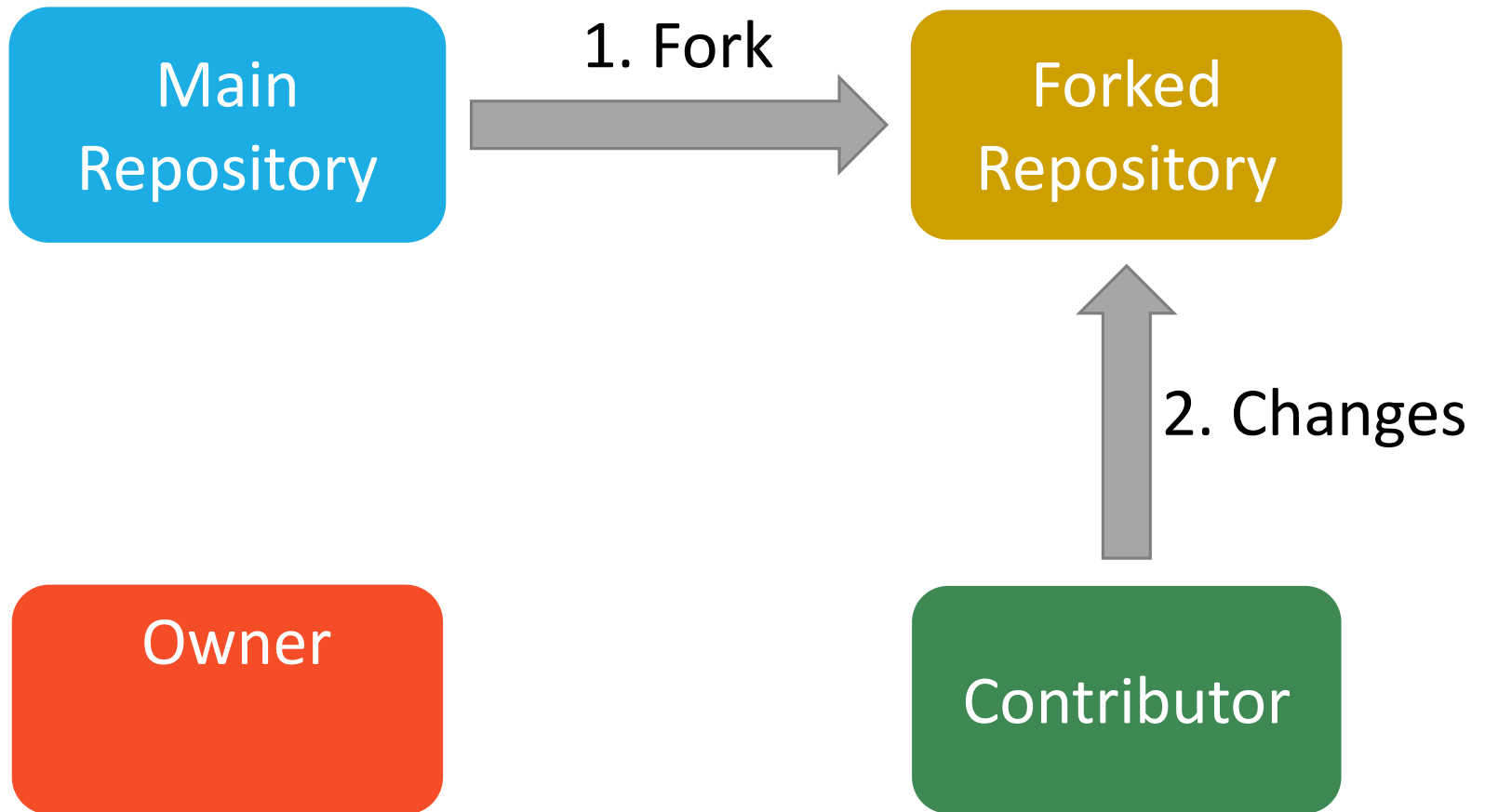
# GitHub Workflow

---



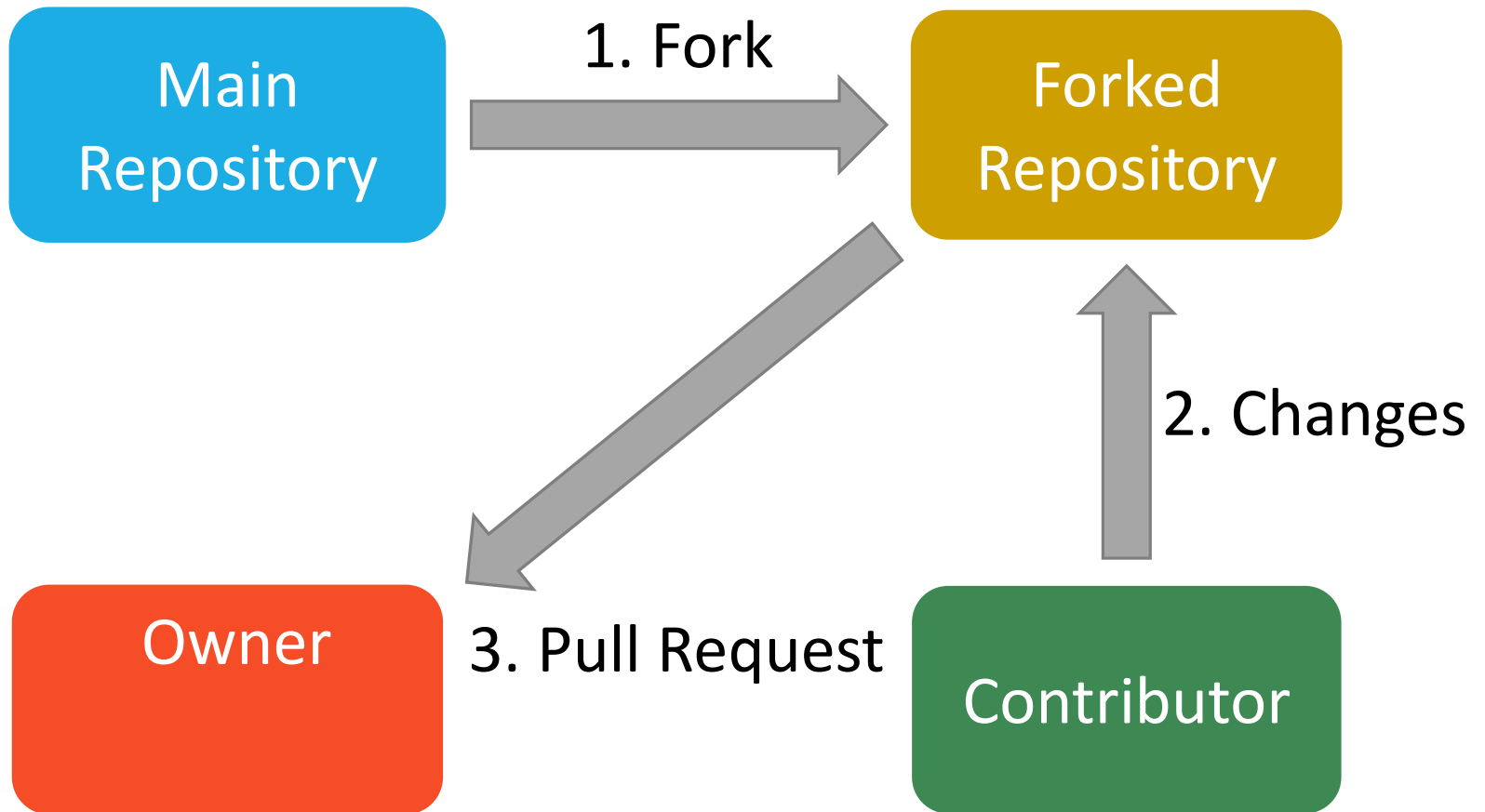
# GitHub Workflow

---



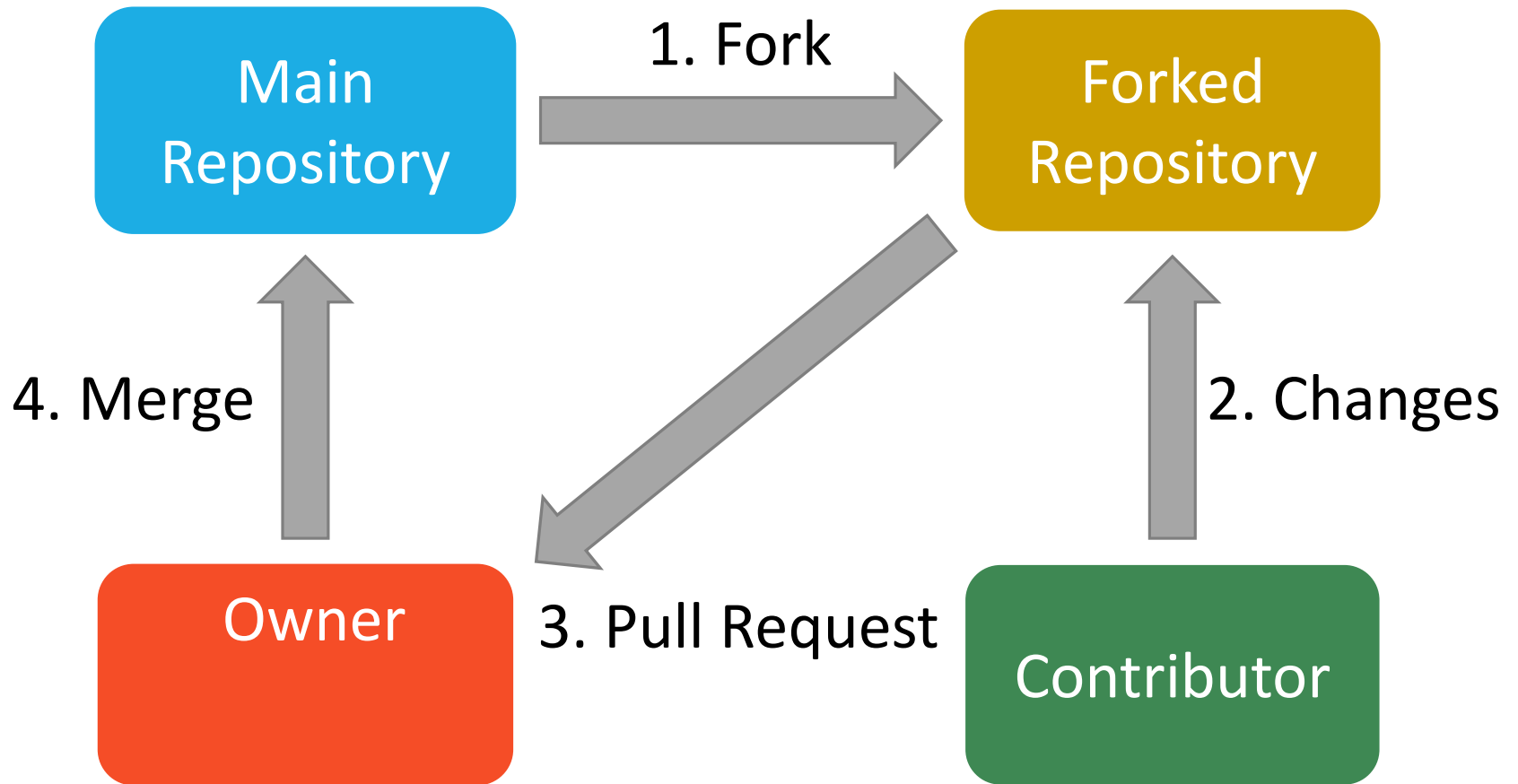
# GitHub Workflow

---



# GitHub Workflow

---







This repository ▾

Search or type a command 🔍

Explore Gist Blog Help

bwofff + - 🔧 📄

PUBLIC



bwofff / grails-rwt

Unwatch ▾ 2

Star 2

Fork 2

branch: master ▾

grails-rwt / src / groovy / grails / plugins / rwt / RwtClasspathResourceLoader.groovy 📄

bwofff a year ago Changes towards release 1.0

1 contributor

file | 47 lines (38 sloc) | 1.5 kb

Open Edit Raw Blame History Delete

```
1 /*****
2  * Copyright 2012-2013 Benjamin Wolff
3  *
4  * Licensed under the Apache License, Version 2.0 (the "License");
5  * you may not use this file except in compliance with the License.
6  * You may obtain a copy of the License at
7  *
8  * http://www.apache.org/licenses/LICENSE-2.0
9  *
10 * Unless required by applicable law or agreed to in writing, software
11 * distributed under the License is distributed on an "AS IS" BASIS,
12 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
13 * See the License for the specific language governing permissions and
14 * limitations under the License.
15 *
16 * Contributors:
17 * Benjamin Wolff - initial API and implementation
18 *****/
19 package grails.plugins.rwt
20
21 import java.io.IOException
22 import java.io.InputStream;
23
24 import org.eclipse.rap.rwt.service.ResourceLoader;
25
26 class RwtClasspathResourceLoader implements ResourceLoader {
27
28     final String resourcePath
29 }
```

# Stashing

---

# Stashing

---

Working Directory

Staging Area / Index

# Stashing

---

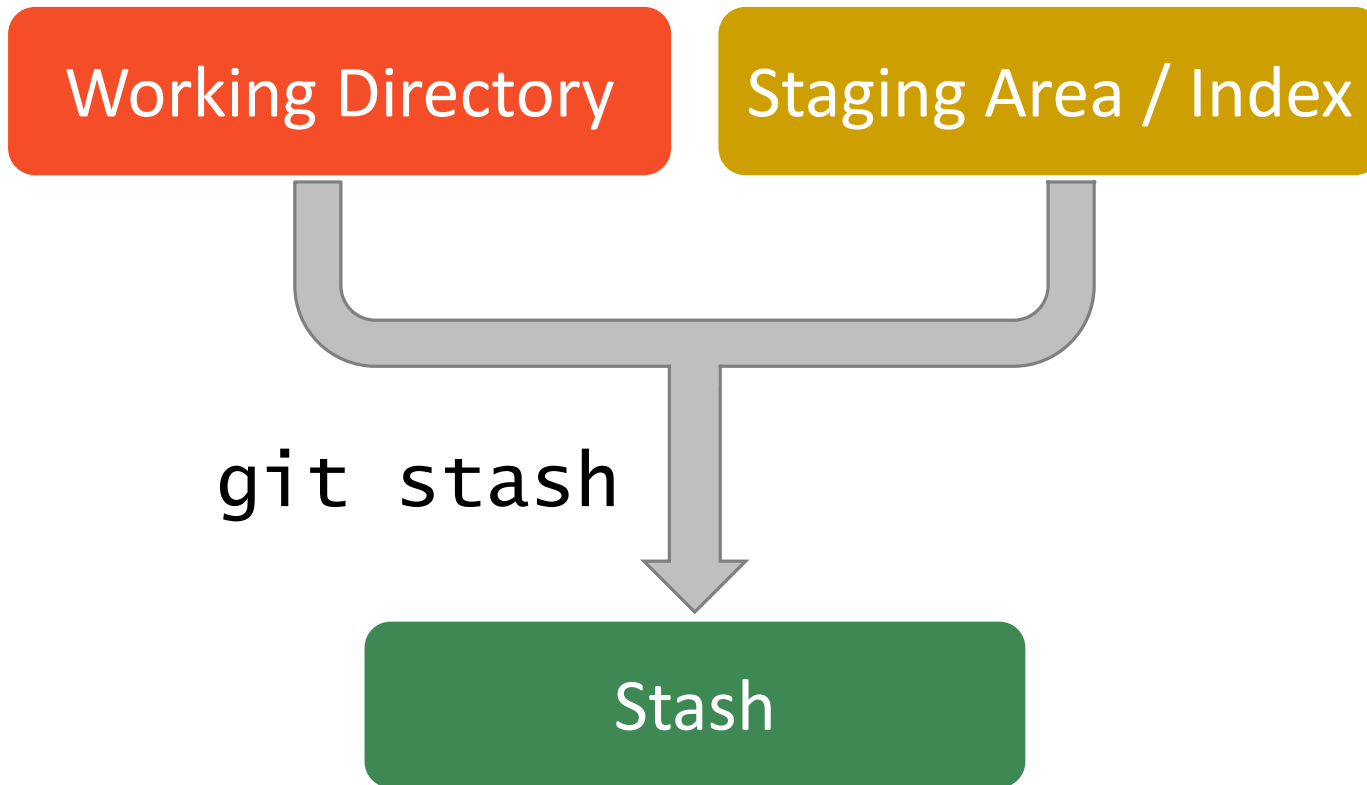
Working Directory

Staging Area / Index

Stash

# Stashing

---



# Stashing

---

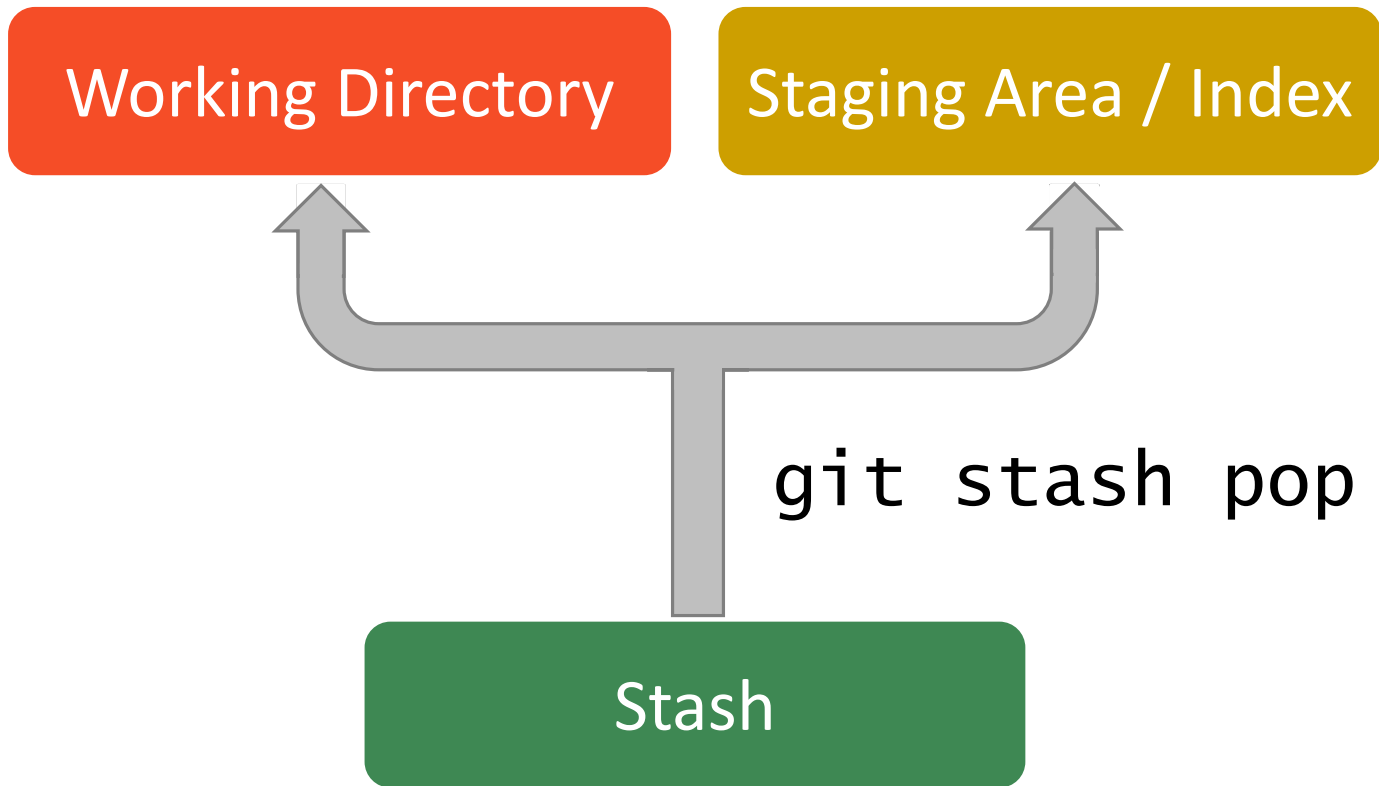
Working Directory

Staging Area / Index

Stash

# Stashing

---



98ca9..

<b>commit</b>	<b>size</b>
tree	0de24
parent	nil
author	Scott
committer	Scott
my commit message goes here and it is really, really cool	

```
$>tree
```

```
.  
|-- README  
`-- lib  
    |-- inc  
    |   `-- tricks.rb  
    `-- mylib.rb
```

```
2 directories, 3 files
```

98ca9..

commit		size
tree	0de24	
parent	nil	
author	Scott	
committer	Scott	
my commit message goes here and it is really, really cool		

0de24..

tree		size
blob	e8455	README
tree	10af9	lib

10af9..

tree		size
blob	bc52a	mylib.rb
tree	b70f8	inc

b70f8..

tree		size
blob	0ad1a	tricks.rb

```
$>tree
```

```
.  
|-- README  
`-- lib  
    |-- inc  
    |   |-- tricks.rb  
    |-- mylib.rb
```

```
2 directories, 3 files
```

98ca9..

commit		size
tree	0de24	
parent	nil	
author	Scott	
committer	Scott	
my commit message goes here and it is really, really cool		

0de24..

tree		size
blob	e8455	README
tree	10af9	lib

10af9..

tree		size
blob	bc52a	mylib.rb
tree	b70f8	inc

b70f8..

tree		size
blob	0ad1a	tricks.rb

```
$>tree
```

```
.  
|-- README  
`-- lib  
    |-- inc  
    |   |-- tricks.rb  
    |-- mylib.rb
```

```
2 directories, 3 files
```

98ca9..

commit		size
tree	0de24	
parent	nil	
author	Scott	
committer	Scott	
my commit message goes here and it is really, really cool		

0de24..

tree		size
blob	e8455	README
tree	10af9	lib

10af9..

tree		size
blob	bc52a	mylib.rb
tree	b70f8	inc

b70f8..

tree		size
blob	0ad1a	tricks.rb

```
$>tree
```

```
.  
|-- README  
`-- lib  
    |-- inc  
    |   |-- tricks.rb  
    |-- mylib.rb
```

```
2 directories, 3 files
```

98ca9..

commit		size
tree	0de24	
parent	nil	
author	Scott	
committer	Scott	
my commit message goes here and it is really, really cool		

0de24..

tree		size
blob	e8455	README
tree	10af9	lib

10af9..

tree		size
blob	bc52a	mylib.rb
tree	b70f8	inc

b70f8..

tree		size
blob	0ad1a	tricks.rb

```
$>tree
```

```
.  
|-- README  
`-- lib  
    |-- inc  
    |   |-- tricks.rb  
    `-- mylib.rb
```

```
2 directories, 3 files
```

98ca9..

commit		size
tree	0de24	
parent	nil	
author	Scott	
committer	Scott	
my commit message goes here and it is really, really cool		

0de24..

tree		size
blob	e8455	README
tree	10af9	lib

10af9..

tree		size
blob	bc52a	mylib.rb
tree	b70f8	inc

b70f8..

tree		size
blob	0ad1a	tricks.rb

```
$>tree
```

```
.  
|-- README  
`-- lib  
    |-- inc  
    |   |-- tricks.rb  
    |-- mylib.rb
```

```
2 directories, 3 files
```

98ca9..

commit		size
tree	0de24	
parent	nil	
author	Scott	
committer	Scott	
my commit message goes here and it is really, really cool		

0de24..

tree		size
blob	e8455	README
tree	10af9	lib

e8455..

blob	size
<pre> == LICENSE: (The MIT License)  Copyright (c) 2007 Tom Preston- Permission is hereby granted, f ree of charge, to any person ob </pre>	

10af9..

tree		size
blob	bc52a	mylib.rb
tree	b70f8	inc

bc52a..

blob	size
<pre> require 'grit/index' require 'grit/status'  module Grit   class &lt;&lt; self     attr_accessor :debug </pre>	

b70f8..

tree		size
blob	0ad1a	tricks.rb

0ad1a..

blob	size
<pre> require 'grit/git-ruby/reposi require 'grit/git-ruby/file_i  module Grit   module Tricks </pre>	

```
$>tree
```

```

.
|-- README
`-- lib
    |-- inc
    |   `-- tricks.rb
    `-- mylib.rb

```

```
2 directories, 3 files
```

98ca9..

commit		size
tree	0de24	
parent	nil	
author	Scott	
committer	Scott	
my commit message goes here and it is really, really cool		

0de24..

tree		size
blob	e8455	README
tree	10af9	lib

e8455..

blob	size
<pre> == LICENSE: (The MIT License)  Copyright (c) 2007 Tom Preston- Permission is hereby granted, f ree of charge, to any person ob </pre>	

10af9..

tree		size
blob	bc52a	mylib.rb
tree	b70f8	inc

bc52a..

blob	size
<pre> require 'grit/index' require 'grit/status'  module Grit   class &lt;&lt; self     attr_accessor :debug </pre>	

b70f8..

tree		size
blob	0ad1a	tricks.rb

0ad1a..

blob	size
<pre> require 'grit/git-ruby/reposi require 'grit/git-ruby/file_i  module Grit   module Tricks </pre>	

```

$>tree
.
|-- README
`-- lib
    |-- inc
    |   |-- tricks.rb
    |-- mylib.rb

```

2 directories, 3 files

98ca9..

commit		size
tree	0de24	
parent	nil	
author	Scott	
committer	Scott	
my commit message goes here and it is really, really cool		

0de24..

tree		size
blob	e8455	README
tree	10af9	lib

e8455..

blob	size
<pre> == LICENSE:  (The MIT License)  Copyright (c) 2007 Tom Preston- Permission is hereby granted, f ree of charge, to any person ob </pre>	

10af9..

tree		size
blob	bc52a	mylib.rb
tree	b70f8	inc

bc52a..

blob	size
<pre> require 'grit/index' require 'grit/status'  module Grit   class &lt;&lt; self     attr_accessor :debug </pre>	

b70f8..

tree		size
blob	0ad1a	tricks.rb

0ad1a..

blob	size
<pre> require 'grit/git-ruby/reposi require 'grit/git-ruby/file_i  module Grit   module Tricks </pre>	

```

$>tree
.
|-- README
`-- lib
    |-- inc
    |   `-- tricks.rb
    `-- mylib.rb

2 directories, 3 files

```

98ca9..

commit		size
tree	0de24	
parent	nil	
author	Scott	
committer	Scott	
my commit message goes here and it is really, really cool		

0de24..

tree		size
blob	e8455	README
tree	10af9	lib

e8455..

blob	size
<pre> == LICENSE: (The MIT License)  Copyright (c) 2007 Tom Preston- Permission is hereby granted, f ree of charge, to any person ob </pre>	

10af9..

tree		size
blob	bc52a	mylib.rb
tree	b70f8	inc

bc52a..

blob	size
<pre> require 'grit/index' require 'grit/status'  module Grit   class &lt;&lt; self     attr_accessor :debug </pre>	

b70f8..

tree		size
blob	0ad1a	tricks.rb

0ad1a..

blob	size
<pre> require 'grit/git-ruby/reposi require 'grit/git-ruby/file_i  module Grit   module Tricks </pre>	

```

$>tree
.
|-- README
`-- lib
    |-- inc
    |   |-- tricks.rb
    |-- mylib.rb

```

2 directories, 3 files