

Intro to version control

Version control with git

Gineke ten Holt
Target Holding



Overview

- Why version control?
- The diff (Unix, 1974)
- Snapshots
- Commit history
- Branches

Target Holding

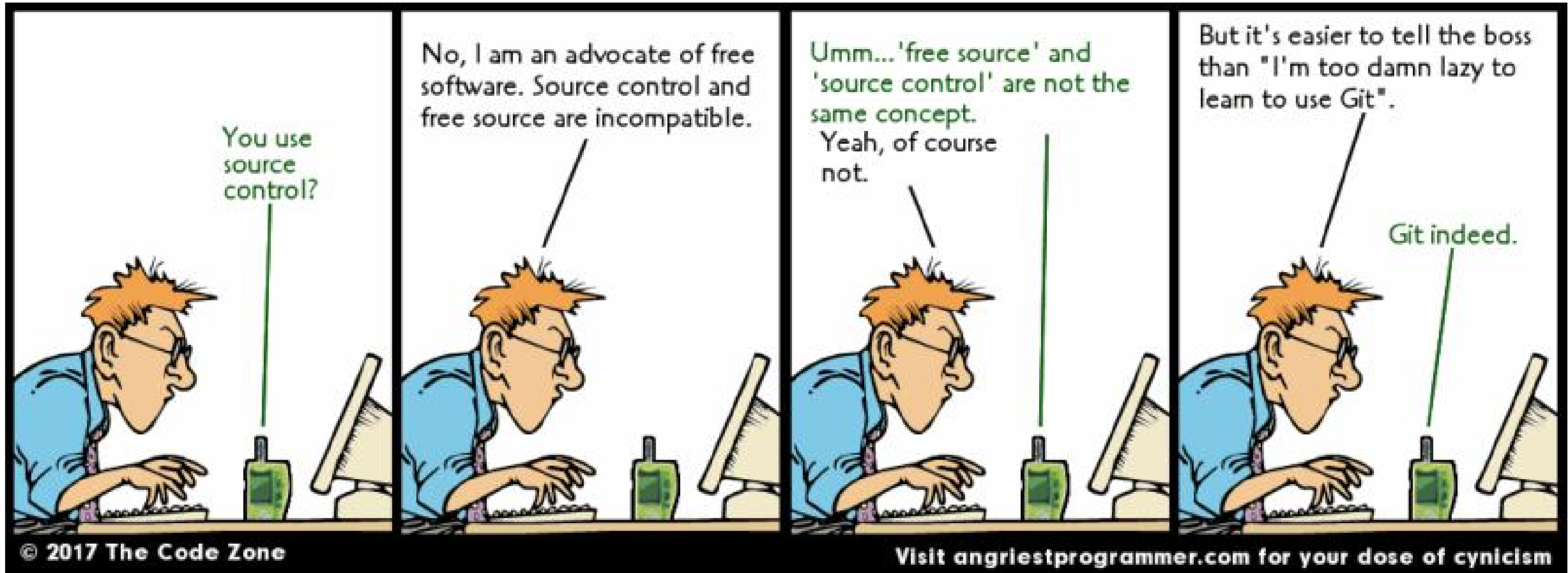
- Pioneering AI to solve business-critical problems
- Fast-growing, young, no-nonsense
- Focus: Financial Risk & Compliance, Scientific Publishing, R&D

and... we're hiring!

- backend
- frontend
- testers



Why version control?



The Diff

36 ■■■■ src/material-examples/dialog-overview/dialog-overview-example.ts

... @@ -1,24 +1,46 @@

```
1 -import {Component} from '@angular/core';
2 -import {MdDialog} from '@angular/material';
```

```
3
4  /**
5   * @title Dialog Overview
6   */
```

```
7  @Component({
8    selector: 'dialog-overview-example',
9    - templateUrl: 'dialog-overview-example.html',
10  })
11  export class DialogOverviewExample {
```

```
12    constructor(public dialog: MdDialog) {}
13
```

```
1 +import {Component, Inject} from '@angular/core';
2 +import {MdDialog, MdDialogRef, MD_DIALOG_DATA} from '@angular/material';
```

```
3
4  /**
5   * @title Dialog Overview
6   */
```

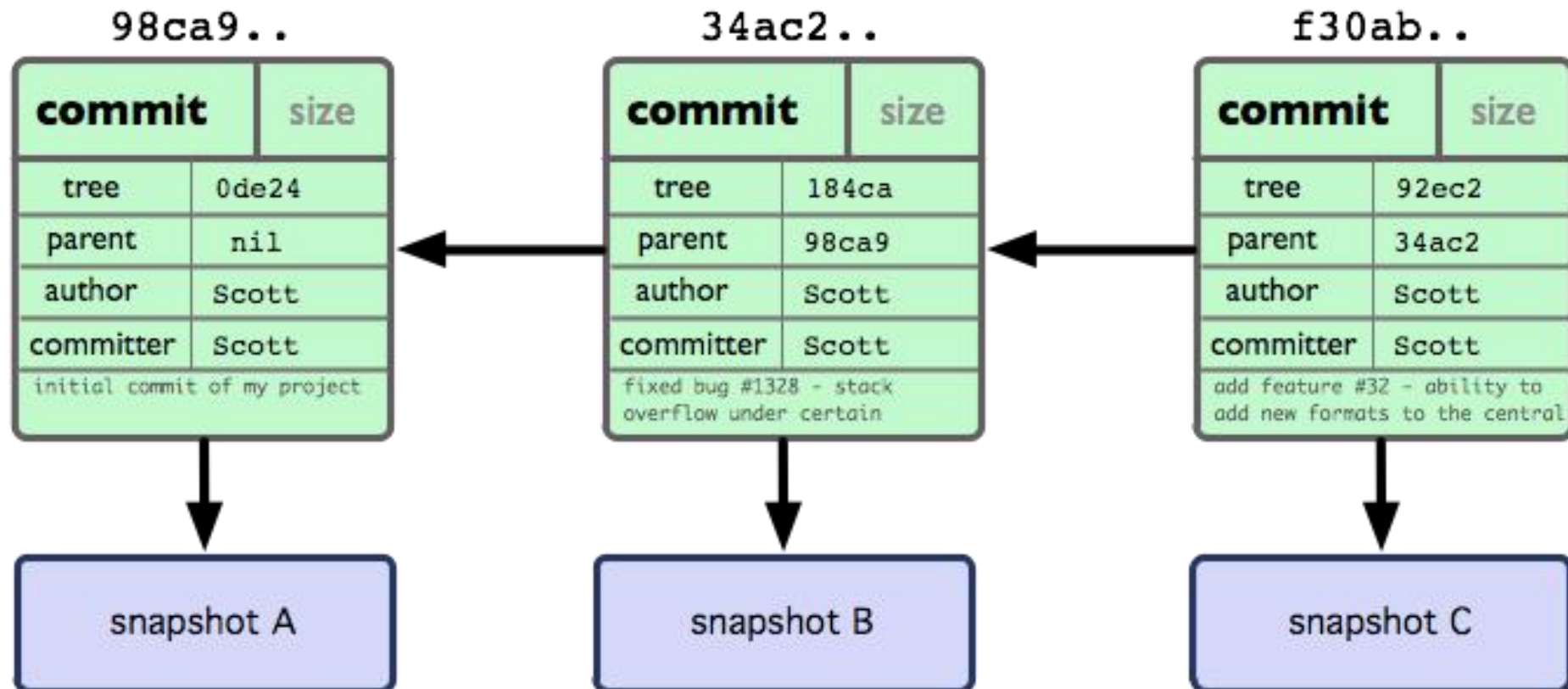
```
7  @Component({
8    selector: 'dialog-overview-example',
9    + templateUrl: 'dialog-overview-example.html'
10  })
11  export class DialogOverviewExample {
```

```
12  +
13  +  animal: string;
14  +  name: string;
15  +
```

```
16    constructor(public dialog: MdDialog) {}
17
```

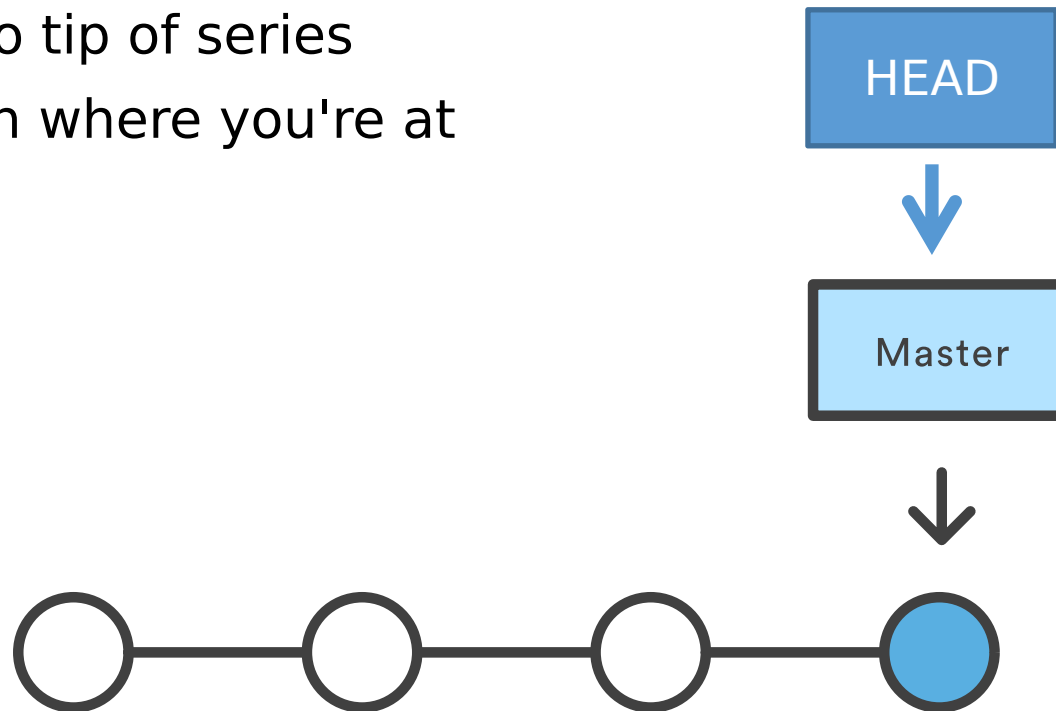
Snapshots

- The state your code was in at the moment of commit
- Repacking saves space



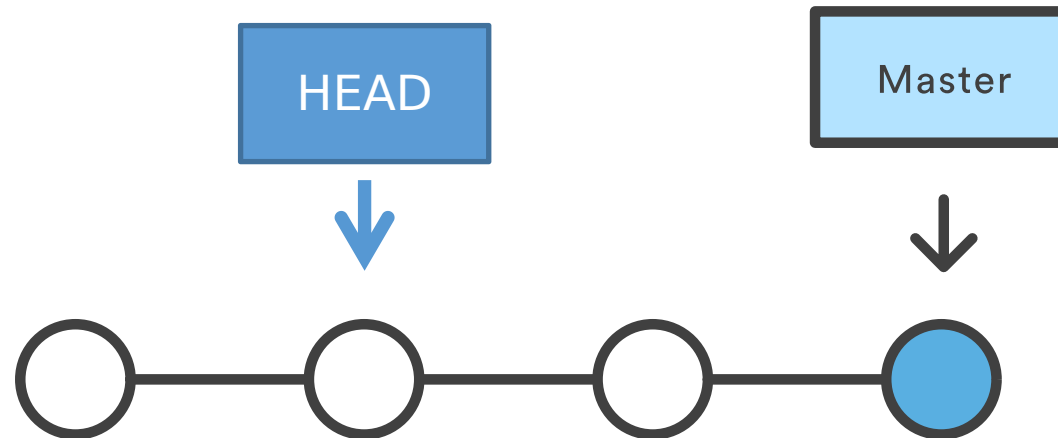
Commit history

- Series of commits / snapshots
- Branch name points to tip of series
- HEAD points at branch where you're at



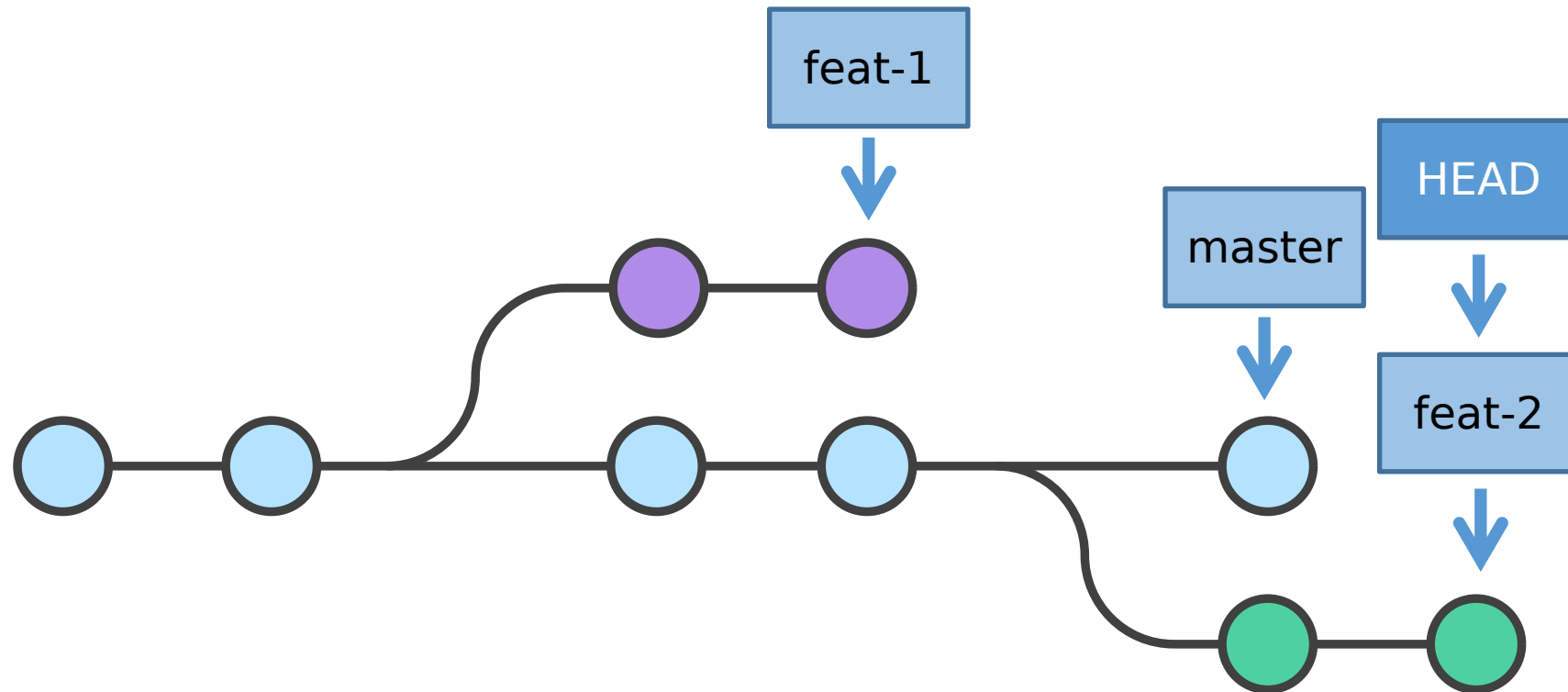
Commit history

- Easily go back to a previous state



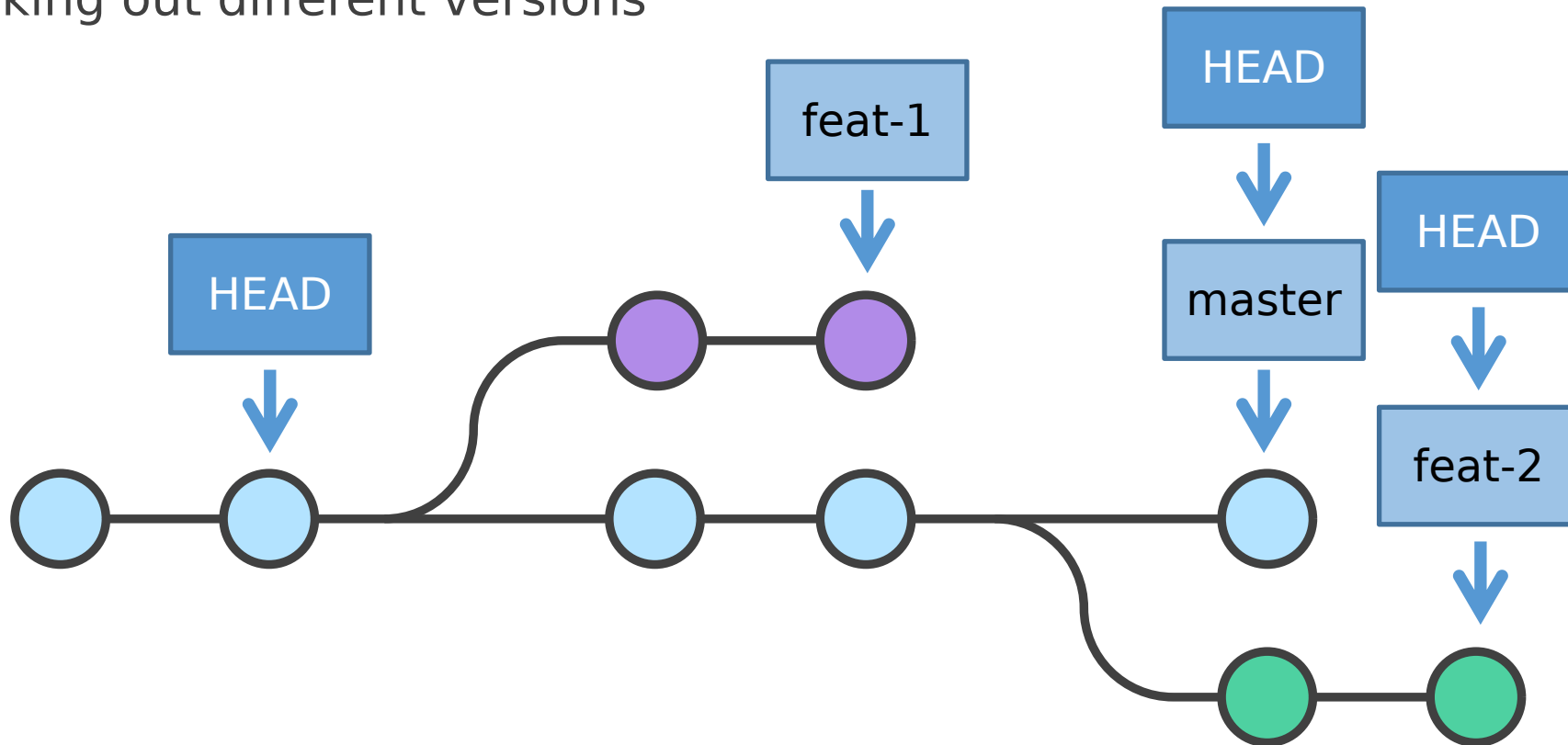
Branching

- A powerful way of keeping things organized



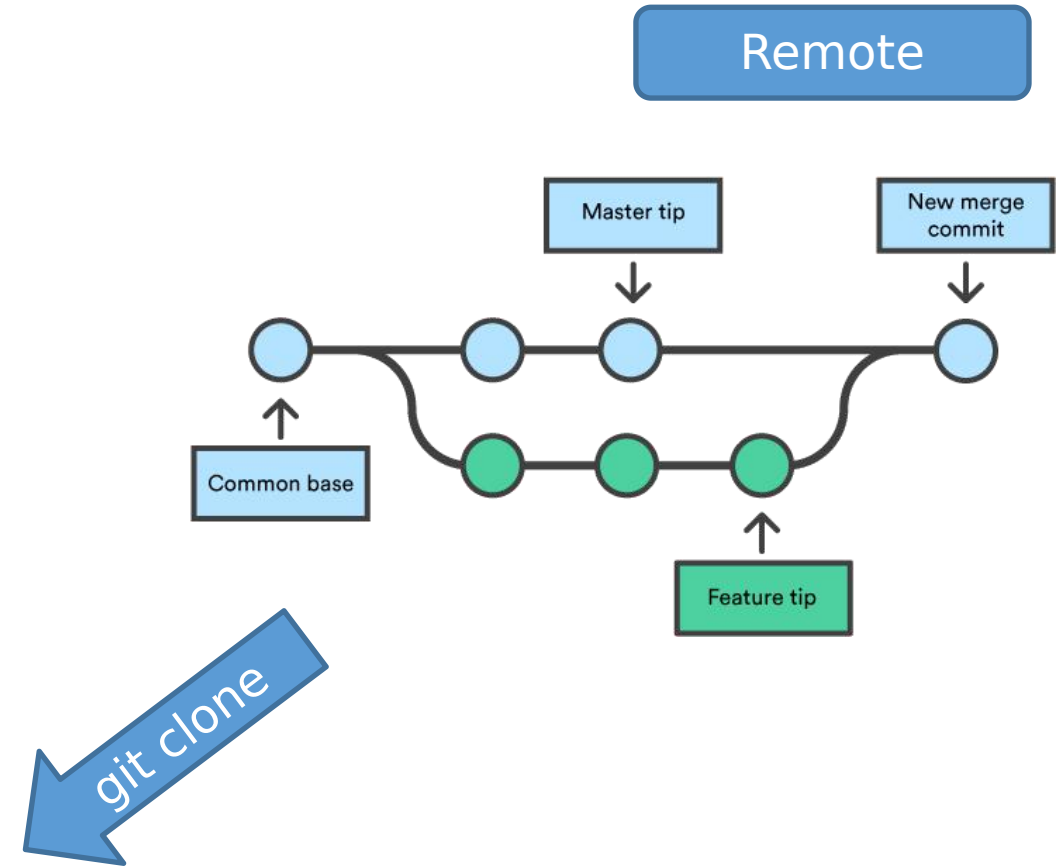
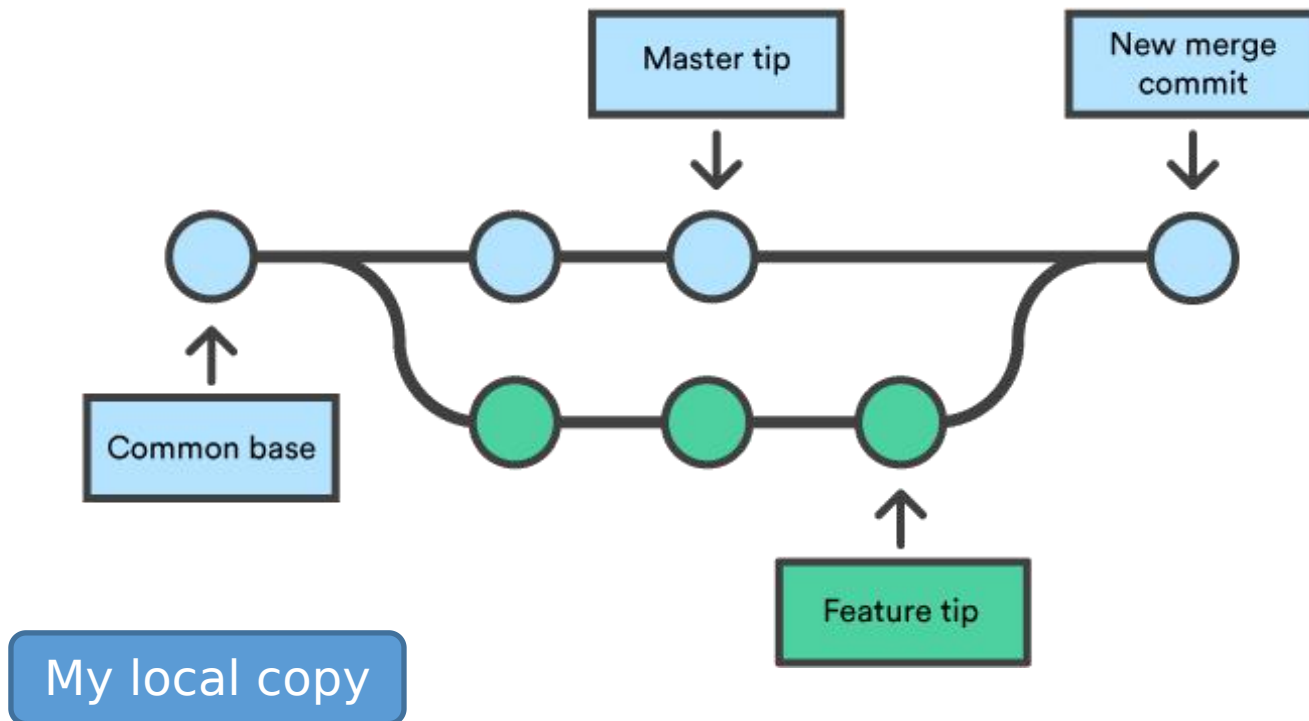
Commit history

- keeping track of coherent states of your code
- checking out different versions



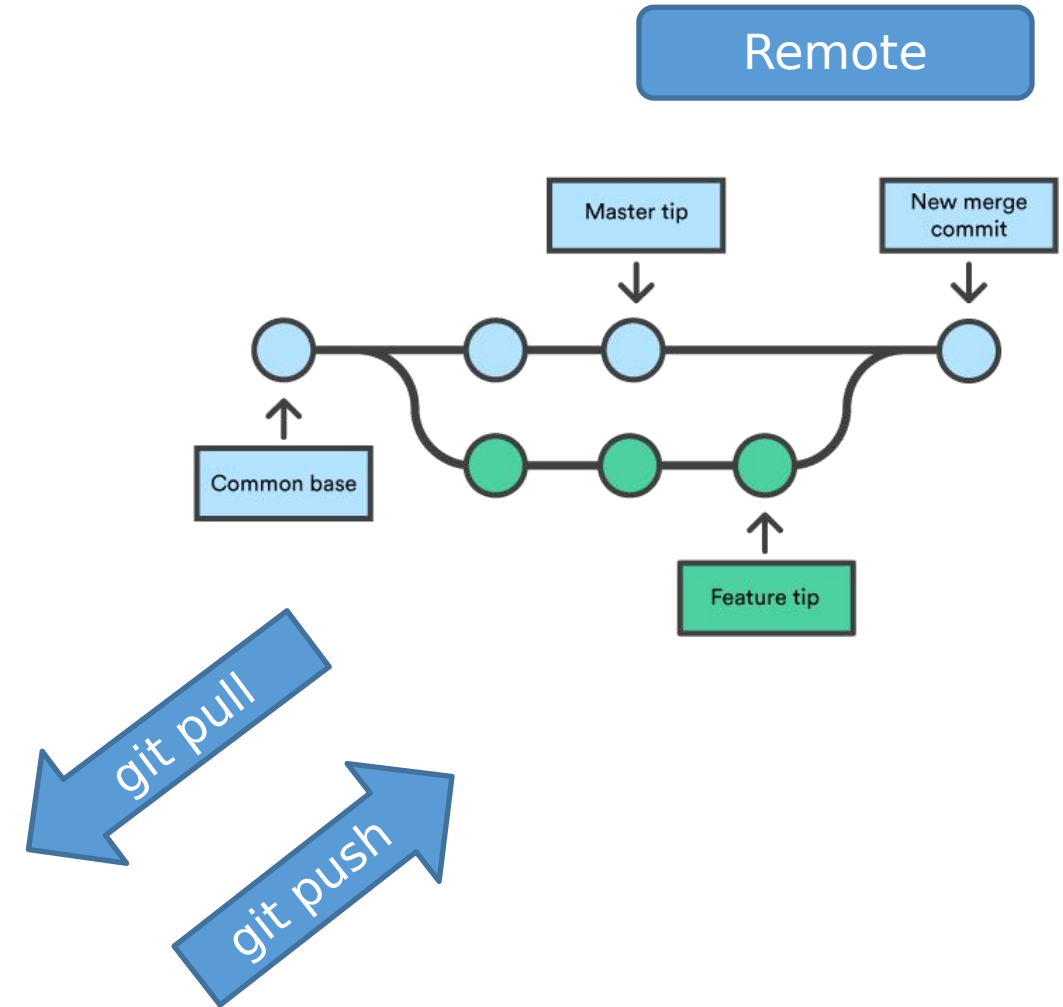
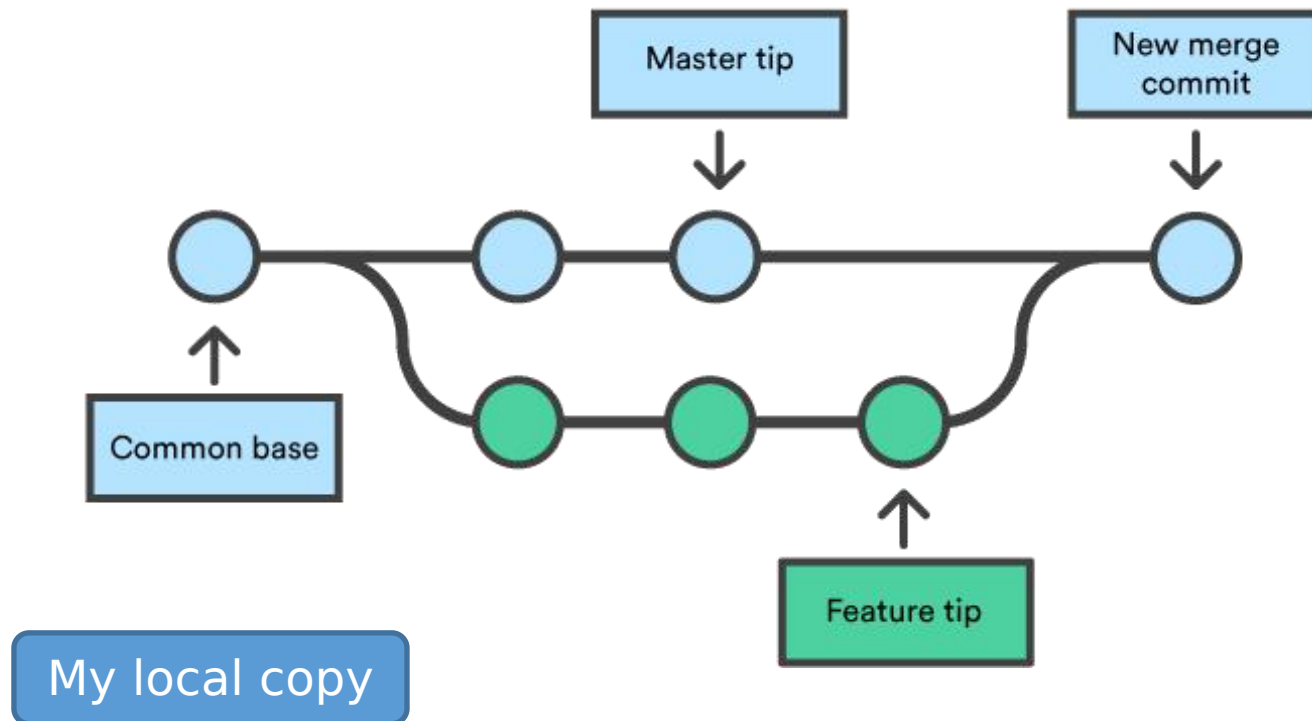
Distributed Version Control

- A Local and a Remote version of your repo
- *Clone* to get a local copy



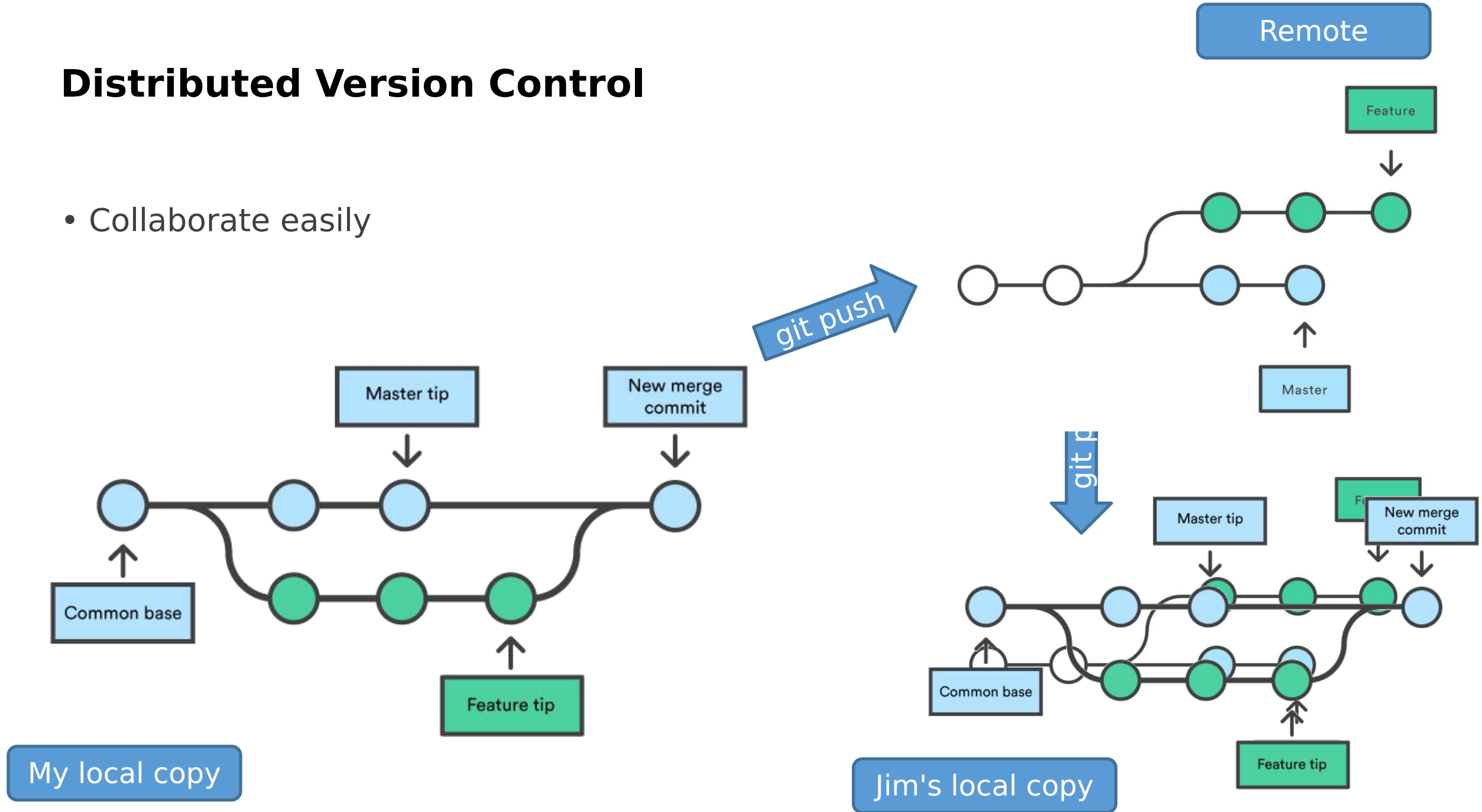
Distributed Version Control

- *Push* and *pull* to share and receive commits



Distributed Version Control

- Collaborate easily



Resources & Examples

- Git tutorial :
<https://git-scm.com/docs/gittutorial>
- Pro Git book :
<https://git-scm.com/book/en/v2>
- The Git Parable :
<http://tom.preston-werner.com/2009/05/19/the-git-parable.html>
- Try it in game form with Githug! :
[https://www.tobscore.com/githug-what-is-it/\(ruby program\)](https://www.tobscore.com/githug-what-is-it/(ruby%20program))
- Back to the future in 10 git actions :
<http://www.mattluedke.com/back-git-history/>

