

Git 【入門】

書籍名：開発効率をUpするGit逆引き入門

出版：C&R研究所

発表者：4年生の優しい先輩達

バージョン管理システム
へようこそ！



バージョン管理システム

- ファイルの作成日時， 変更日時， 変更点を保管する機能が売り
 - * 何度も変更を加えたファイルであっても， 過去の状態や変更内容を確認したり， 変更前の状態を復元することが容易に！
 - ▶ 複数の人間がファイルの編集に関わる状況を想定してます
- 様々なものを管理できる
 - * ソースコードの他に設定ファイルなども
- 間違って上書きしちゃった…でも
 - * 履歴から元に戻せます！！



バージョン管理システムの歴史

- 60年代：ソースコードがテキストファイルで保存できるようになった！
- 60年代：ソース管理部門に持って行くとスタッフが顧客の製品をビルド (人カバージョン管理)
- 70年代：SCCS；複数のバージョンを1つのファイルに保存！
- 80年代：複数のファイルを一括管理，複数人で同時に作業した結果をマージできるようになった！
- 2005年：**Gitの誕生；分散型バージョン管理システムが生まれた！**
- 2008年：GitHub；他人にソースを公開できる！

みんなでバージョン管理を！！



Gitの仕組み

リポジトリ

- ファイルやディレクトリの状態を記録する場所

* 保存した状態は、内容の変更履歴として格納



- 変更履歴を管理したいディレクトリをリポジトリの管理下に置くことで、そのディレクトリ内のファイルやディレクトリの変更履歴を記録できる

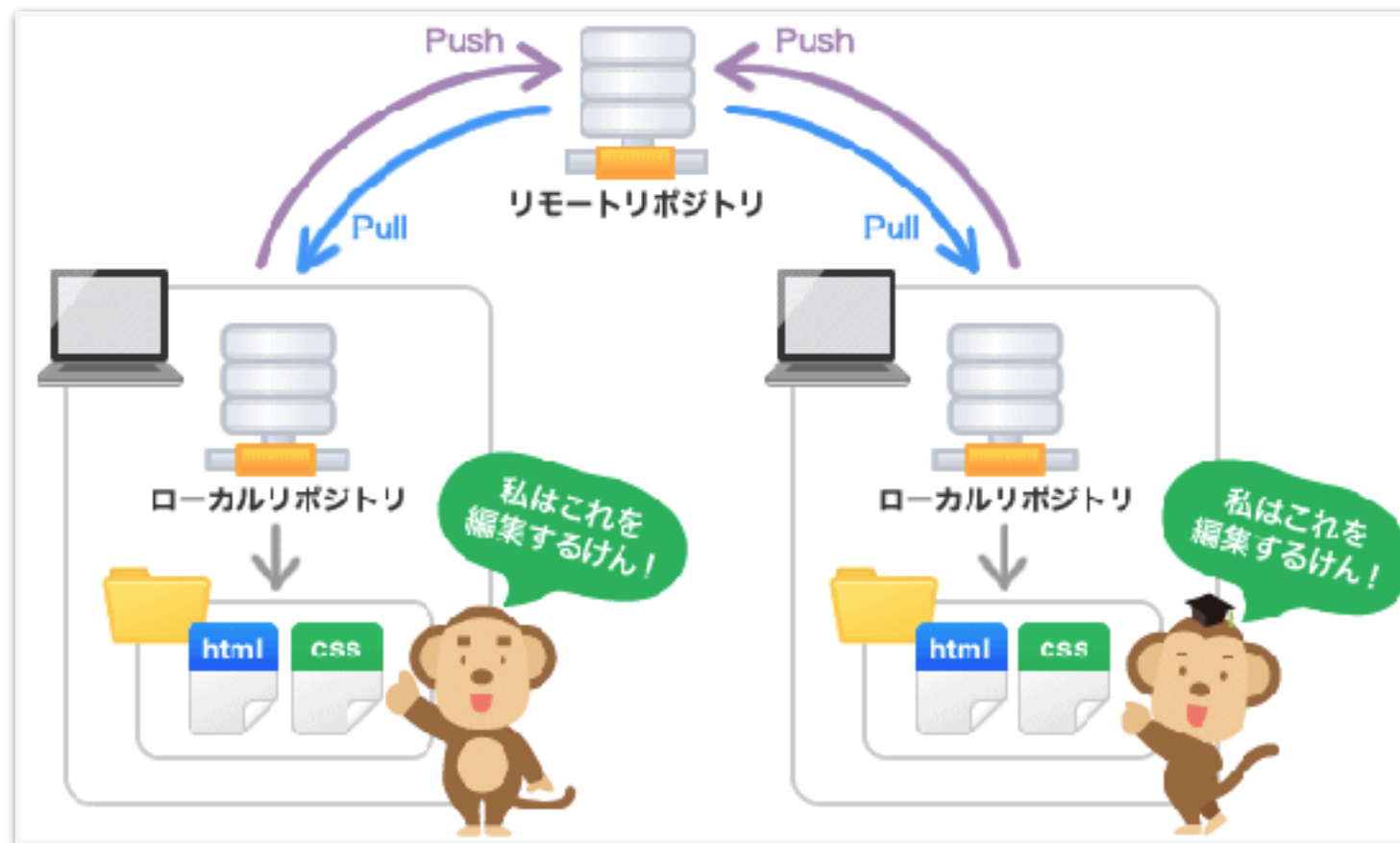
リモートリポジトリ・ローカルリポジトリ

- ・ リモートリポジトリ

- * 専用のサーバに配置して複数人で共有するためのリポジトリ

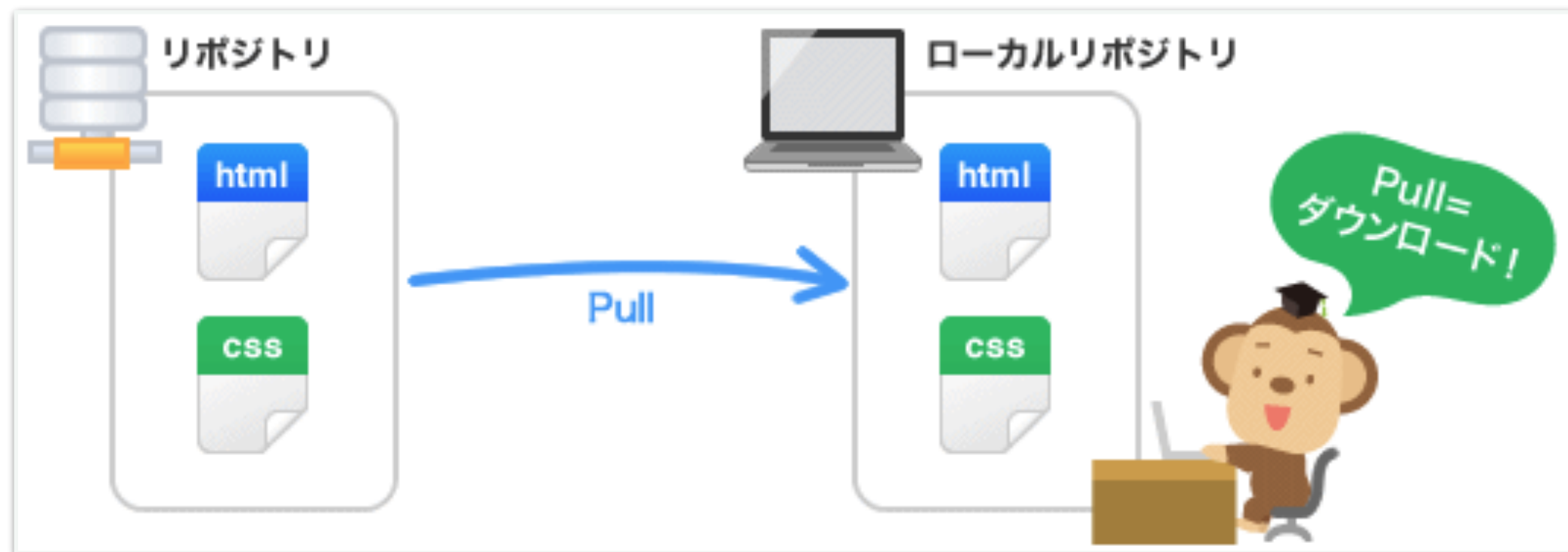
- ・ ローカルリポジトリ

- * ユーザ各人が利用するために、自分の手元のマシン上に配置するリポジトリ



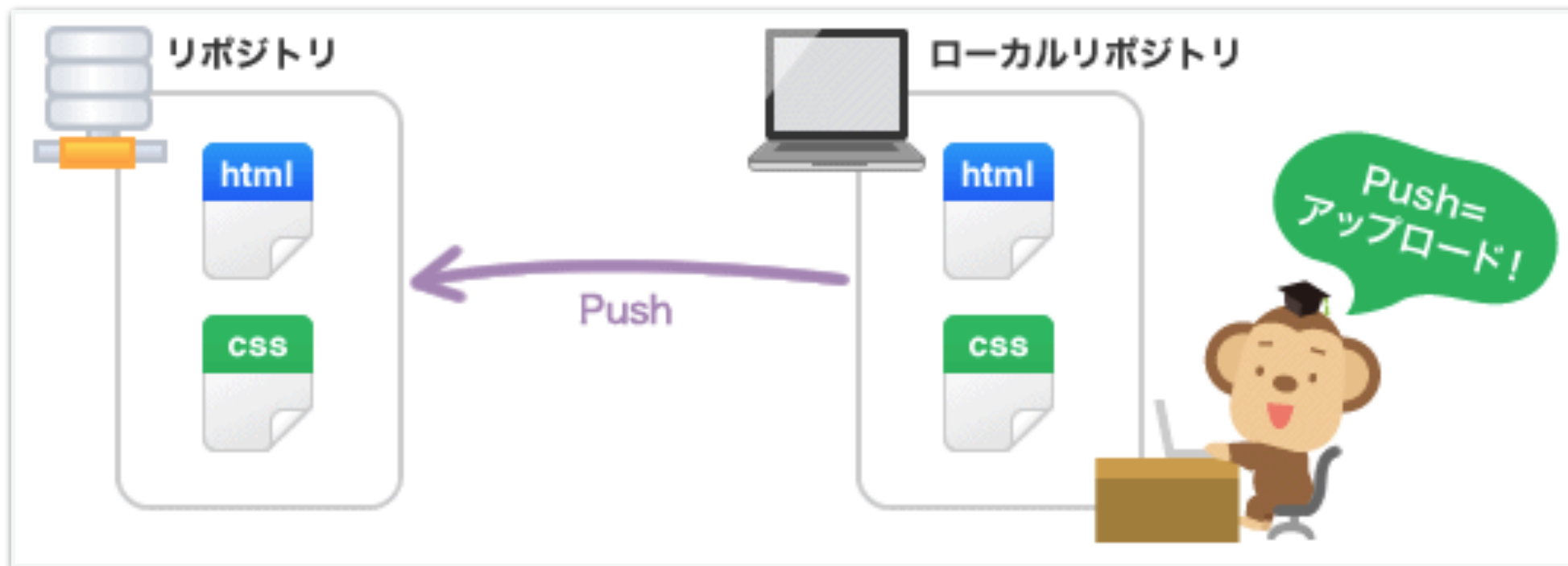
プル(pull)

- リモートリポジトリからローカルリポジトリを更新すること
- リモートリポジトリから最新の変更履歴をダウンロードして、自分のローカルリポジトリにその内容を取り込む



プッシュ(push)

- ローカルリポジトリ内の変更履歴をアップロードすること
- リモートリポジトリに自分の変更履歴がアップロードされて、リモートリポジトリ内の変更履歴がローカルリポジトリの変更履歴と同じ状態になる



クローン (clone)

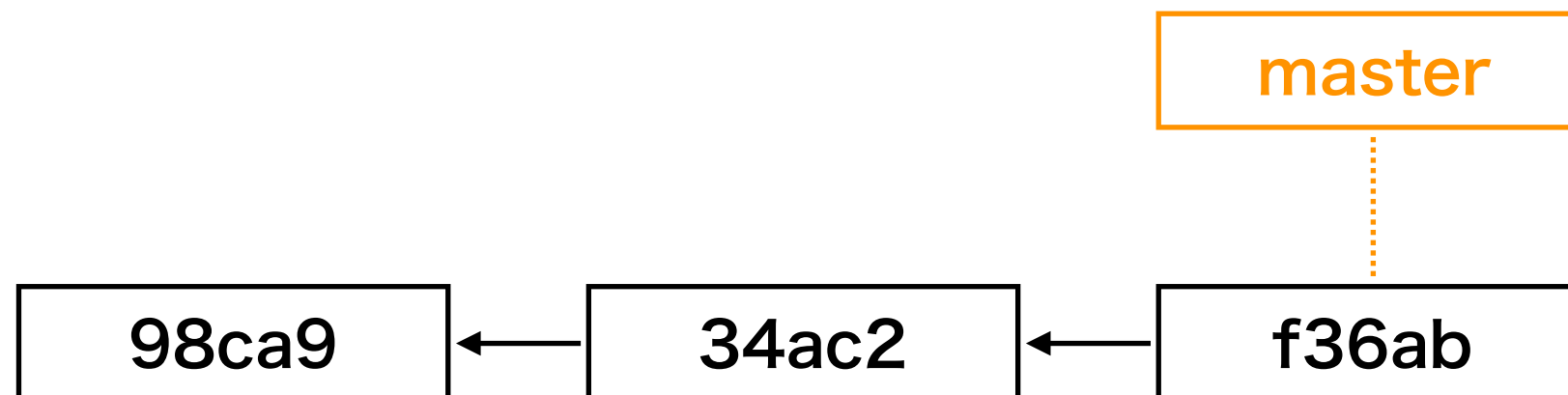
- リモートリポジトリを複製すること
- リモートリポジトリの内容をダウンロードして、別のマシンにローカルリポジトリとして作成できる

3つのデータ領域

- 作業ディレクトリ
 - * ユーザが実際に作業しているディレクトリ
- ステージング・エリア
 - * 作業ディレクトリ上で行った変更のうち、変更履歴として次のコミットに含めたい対象を管理する領域
- Gitディレクトリ
 - * Gitのリポジトリの全ての変更履歴を保持している領域

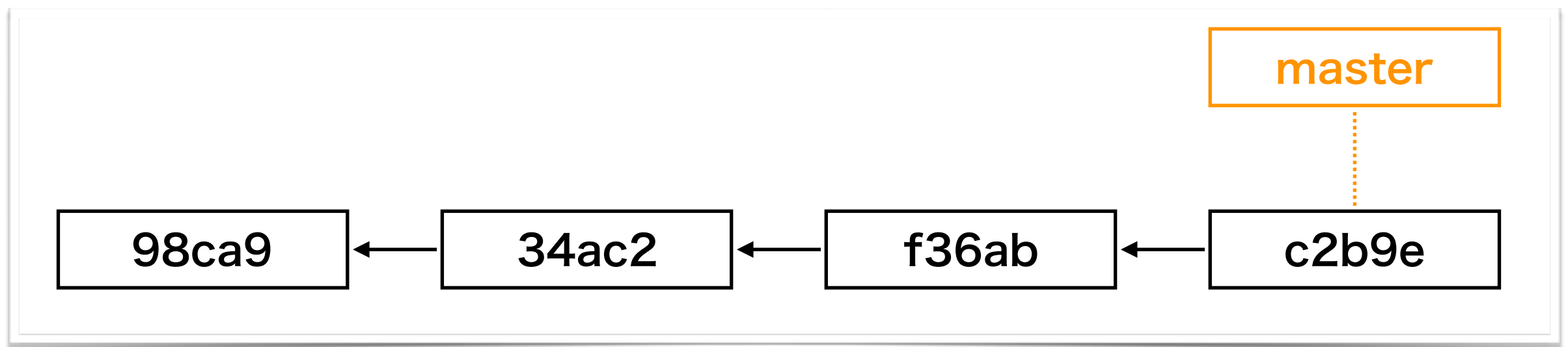
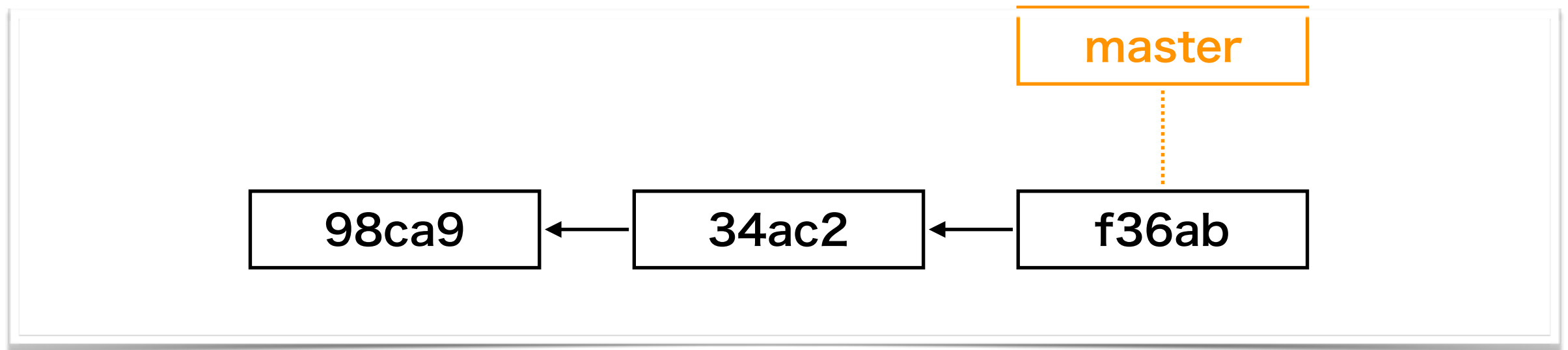
ブランチ

- メインの流れとは異なる流れで作業を続ける仕組み
 - * Gitでは、全ての流れはブランチ
 - ▶ 初期状態のGitリポジトリでは「**master**」という名前のブランチが用意されている



変更履歴の追加

- コミットされると， masterブランチは最新の変更履歴に移動する



それではGitを
インストールしましょう！

