

バージョン管理基礎演習

第1回

導入編

①成績評価について

PDFファイル「バージョン管理基礎演習_成績評価.pdf」
の内容に沿って説明

②テキストについて

水色のテキスト『いちばんやさしいGit&GitHubの教本 第2版
人気講師が教えるバージョン管理&共有入門』
が手元にある方は用意

前提として、授業で扱うバージョン管理ソフトは『Git』と『GitHub』の2つです。

『Git』と『GitHub』は世界的に使われているバージョン管理ソフトの代表格であり、インターネット上に情報が多く存在する為、参考書が無くても学習が充分可能です。



定価 ¥2,420

授業の進行はテキストの内容を基準に進めますが、基本的に資料はデータとして配布し、実際の操作を通じて学ぶ時間がメインになるので、「テキストと一緒に読みましょう！」のような時間はほとんど無いです。

ただし、せっかくお金払って買っているので、最初だけテキストを使用します。

(以降、予習として読んでおくことを推奨します。)

③基礎知識を学ぶ

水色のテキスト『いちばんやさしいGit&GitHubの教本 第2版

人気講師が教えるバージョン管理&共有入門』

【Chapter1:Gitの基本を学ぼう】(P14～28)の内容に沿って、説明

1. バージョン管理

我々は普段、様々なファイルを作成または編集し、その構成や内容を変更している
(資料作成、プログラミングなど)

このようにアップデートするにつれて変化するファイルの状態を『バージョン』と呼ぶ



「編集前の内容を後から参照したい」

「間違った変更をしても元に戻せるようにしておきたい」

⇒ ファイルをコピーしてバックアップを取得

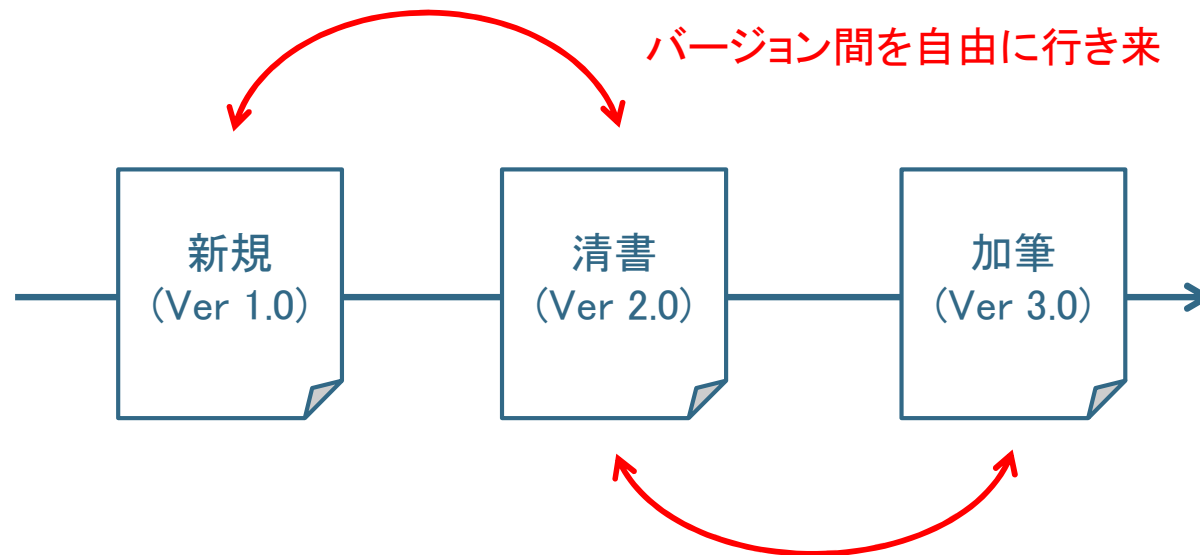
⇒ 同じファイルの複数のバージョンを保持することを『バージョン管理』と呼ぶ

2. バージョン管理システム

『バージョン管理システム』では、

ファイルの変更をバージョンとして記録し、記録した時点へいつでも戻れる仕組みを提供

(例: Git、Subversion、etc..)



3. バージョン管理システムの必要性

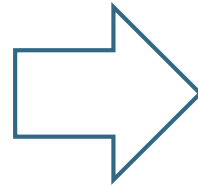
原始的なバージョン管理の手法として、
「ファイルをコピーしてバックアップを取り続ける」という方法がある

File_20240405.txt

File_20240413_1.txt

File_20240413_2.txt

File_20240513.txt



- ・ストレージ容量の圧迫
- ・毎回コピーするのが手間
- ・変更内容の把握不可
- ・上書きミス等による事故のリスク大

バージョン管理システムを使用することで、
一貫したルールに基づいた管理ができ、ファイルの最新状態や変更履歴を保つことが可能

4. Git



2005年にLinuxカーネルのソースコード管理システムとして、誕生。

大規模プロジェクトを高速に、且つ複数人で並行して扱える仕組みが大きな特徴。

ソースコードの管理として使われることが多いが、

Wikiやブログ等の日本語で書くようなテキストも管理できる。

昨今ではドイツが法令をGitで管理している事例もある。

⇒ <https://github.com/bundestag/gesetze>

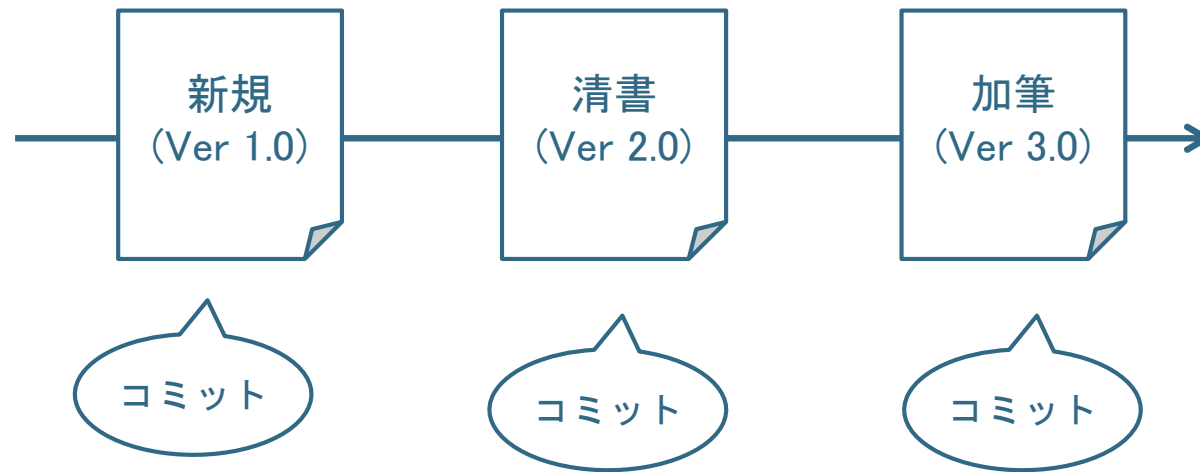
5. GitHub



GitHubは、Gitの仕組みを利用して、
インターネット上でのスムーズな共同作業を可能にしたWebサービス。
いつでもどこでも効率よく開発が行えるメリットから、広く普及している。

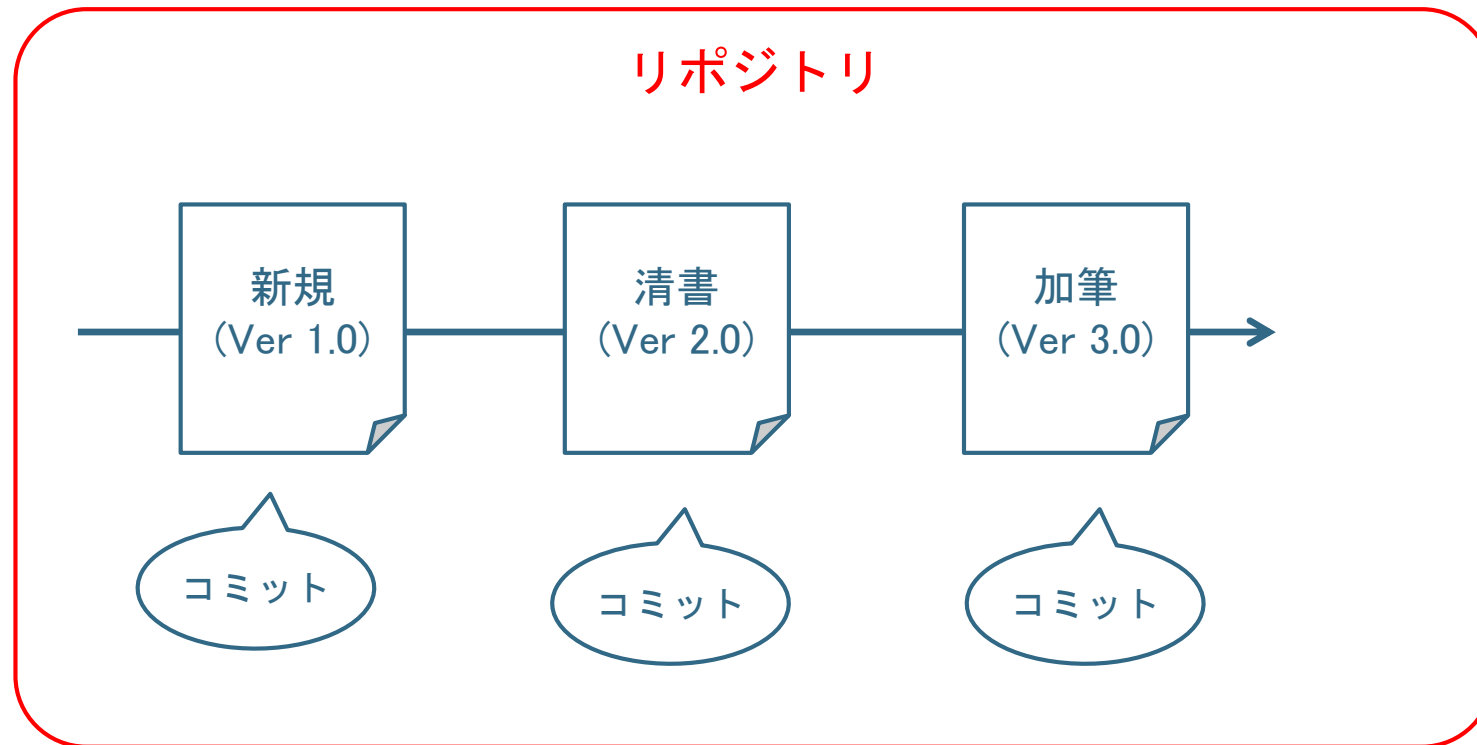
GitHubの勢いによる助けもあり、
Gitは広範囲な普及を成し遂げている。

6. コミット(commit)



Gitでは、変更の履歴として、管理対象となっている全ファイルのその時点の状態を保持していく。ユーザが任意のタイミングで変更記録を保存する操作を『**コミット**』と呼ぶ。このコミットを繰り返すことでファイルの変更履歴がわかるようになる。この仕組みにより「いつ」「誰が」「どんな変更を」したかが、誰が見ても分かるようになる。

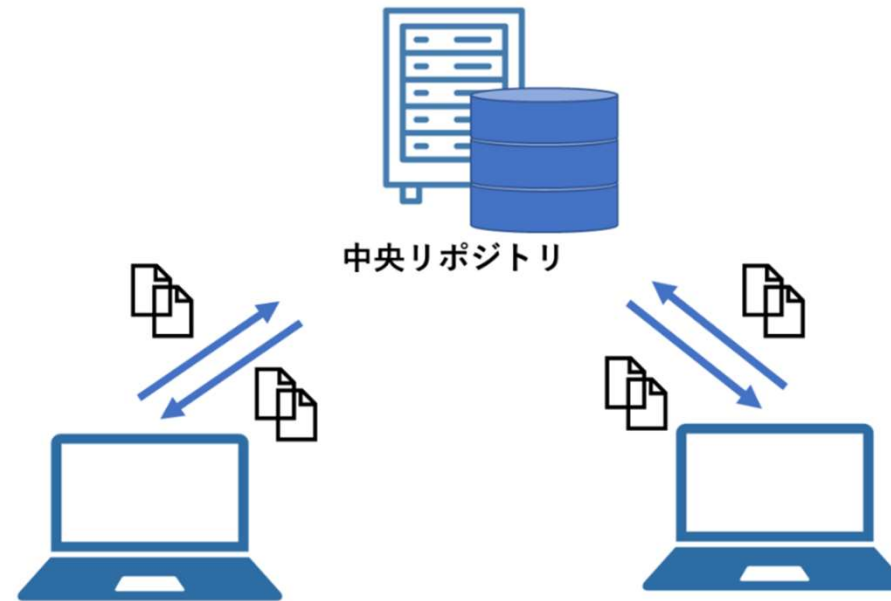
7. リポジトリ(repository)



コミットを貯めていく場所を『**リポジトリ**』と呼ぶ。

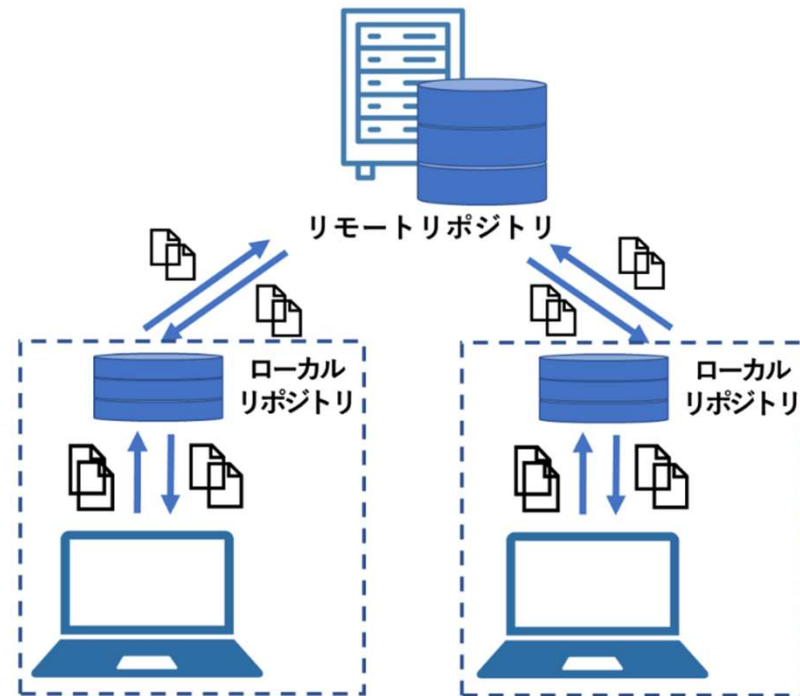
ファイルやフォルダの1つ1つの情報を保存しておく保存先(保存場所)。

8. 集中管理方式



専用のサーバー(リポジトリ)を使い、ファイルのバージョンを一元管理する方式。
ユーザーがサーバー上に設けられた中央リポジトリに、変更履歴を直接記録する。
(例: CVS、Apache Subversion: SVN)

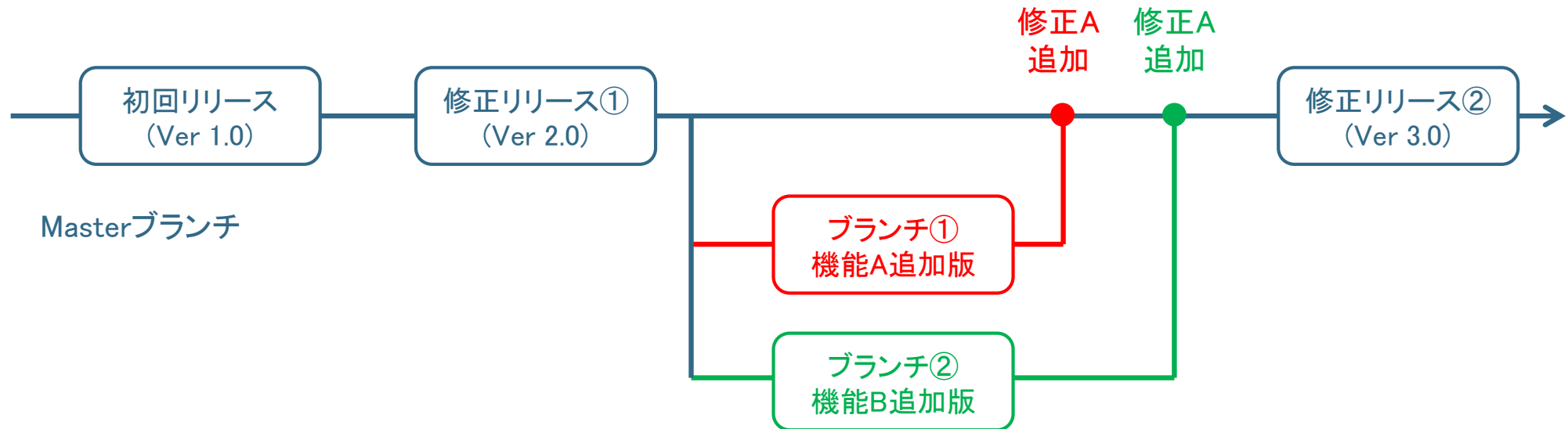
9. 分散管理方式



ユーザーが各自のパソコン内にリポジトリのコピー(ローカルリポジトリ)を持ち、これにファイルの変更を記録する方式。ユーザーがファイルの変更を自身のパソコン内に一旦記録し、これをリモートリポジトリに反映する。

(例: Git、Mercurial)

10. ブランチ (branch)



1つのプロジェクトから分岐させることにより、
プロジェクト本体に影響を与えずに開発を行える機能のことを『**ブランチ**』と呼ぶ。

【ブランチを利用するメリット】

- ・プロジェクトの本体に影響を与えずに作業が可能
- ・ブランチを切ることで目的別に同時並行で作業が可能

④Gitをインストール

「Git-2.45.0-64-bit.zip」をダウンロード後、
PDFファイル「Gitインストール手順_Windows.pdf」に従い実施