Introduction to Git Version Control

Department of Computing Science
University of Alberta

CMPUT 301 – Introduction to Software Engineering
Slides adapted from Dr. Abram Hindle

Version Control

- Your Project changes over time:
  - Today: it runs
  - Yesterday: it mostly works
  - Last week: it did not fully work
  - Last year: it may have been something totally different

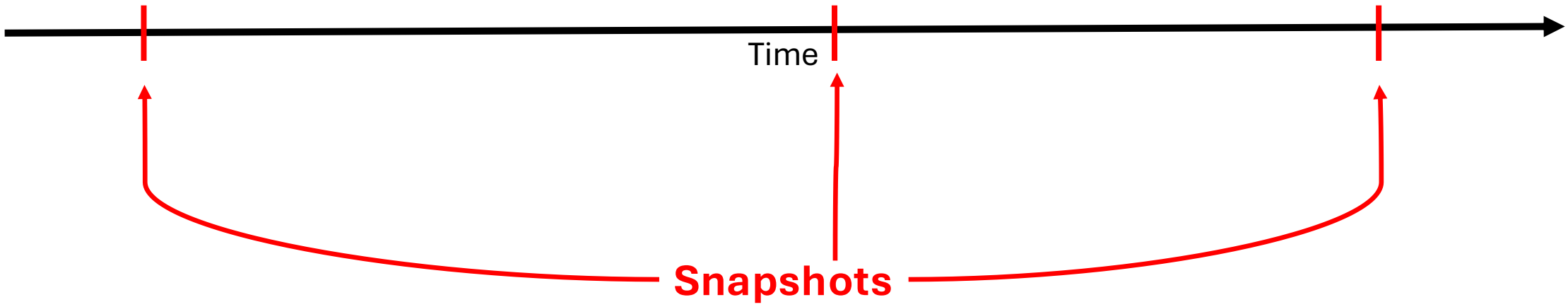
Version Control

- Version control tracks:
  - Changes
  - Files
  - Authors
  - Time

Version Control

- Version control snapshots a state of the project at a point in time

Your Project
(Last year) → Your Project
(Last week) → Your Project
(Now)

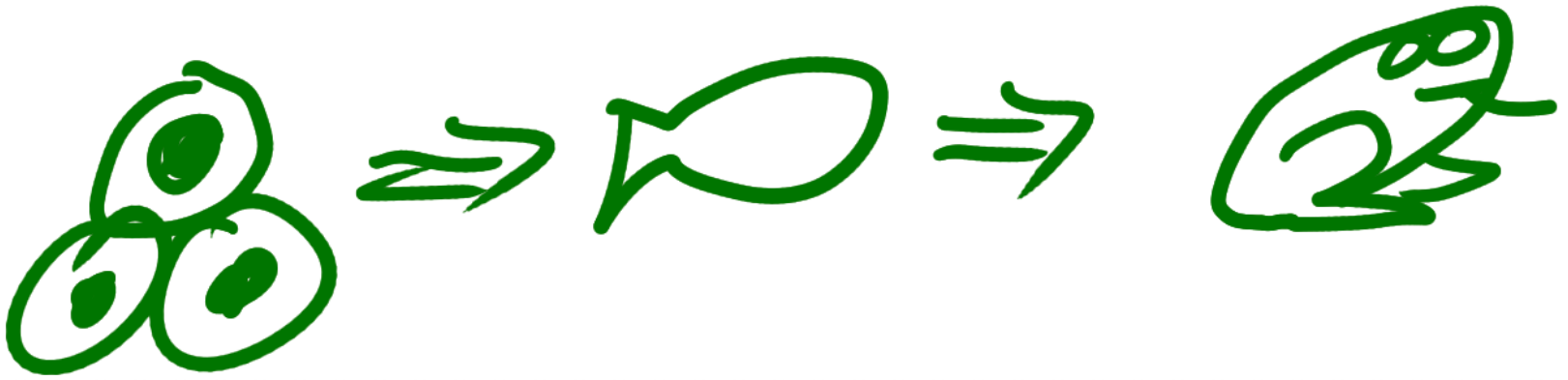


Software Has Context

- A project is not just code
- A project has:
  - Current time
  - Current requirements
  - Current functionality
  - Current issues
  - Current team

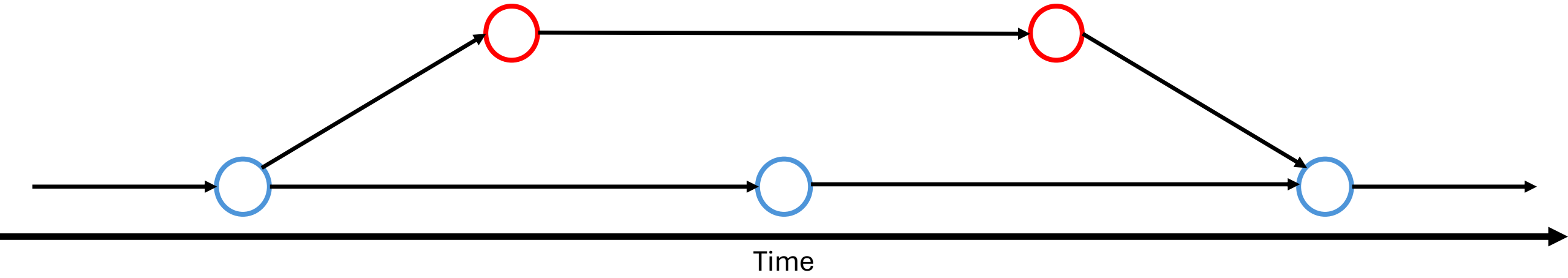
Software Evolves

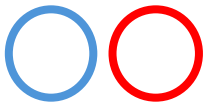
- Version control helps track evolution



Commits Record Change

- Each commit records a meaningful change to the project



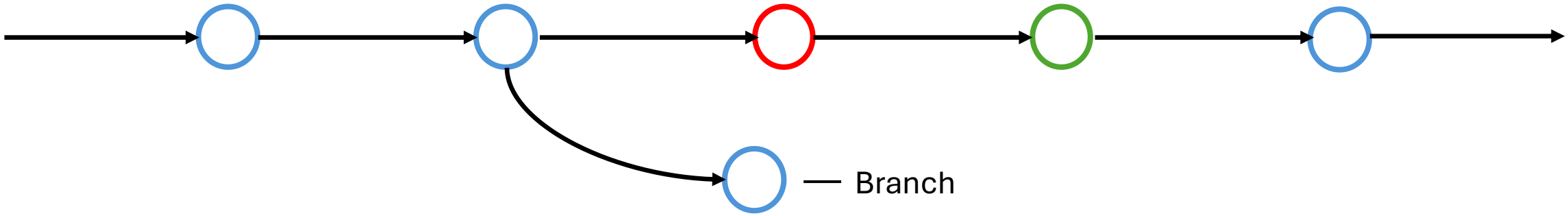


Commit

Code Flow

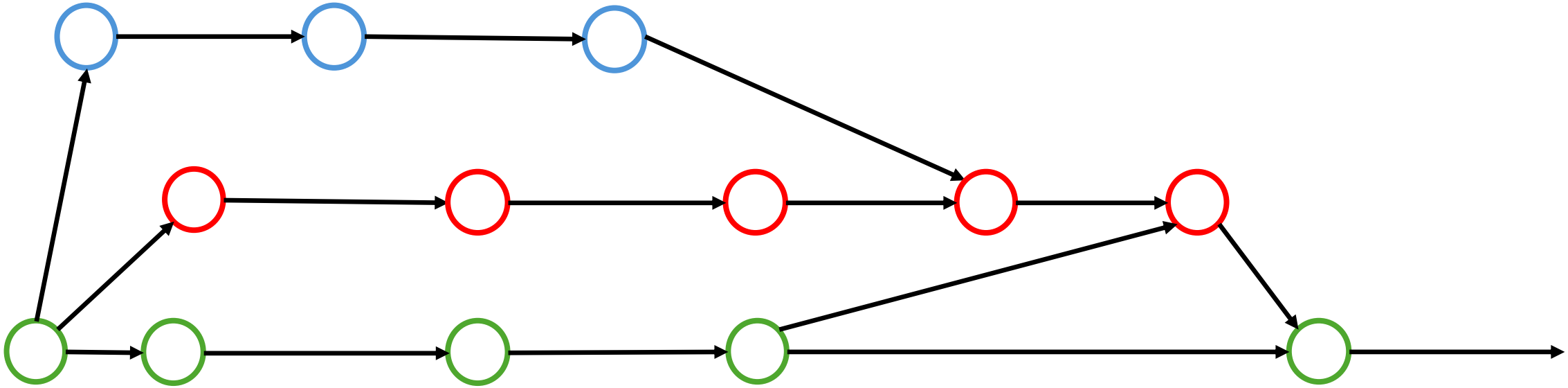
Centralized Version Control

- A centralized system has one main repository that users interact with



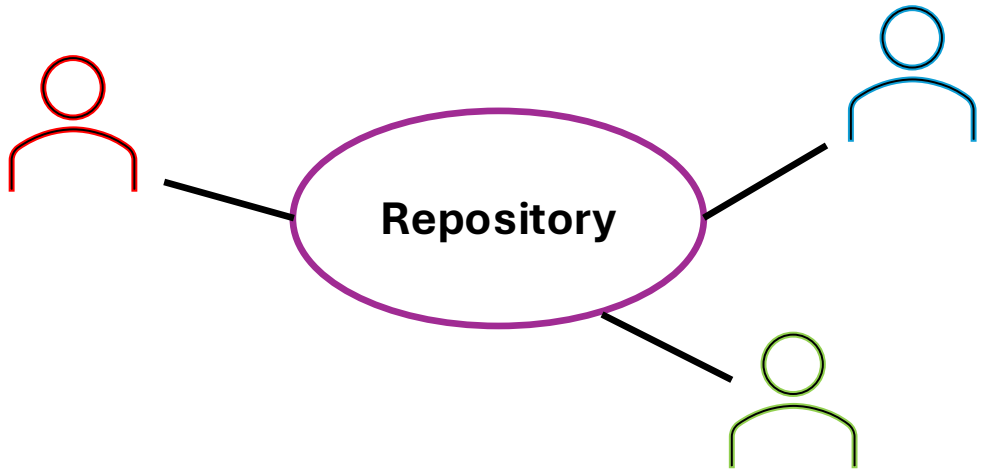
Distributed Version Control

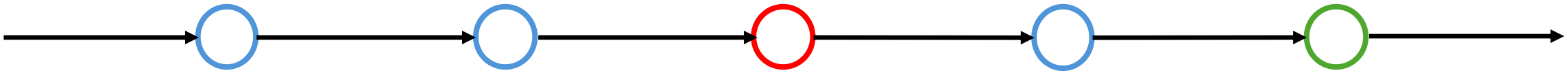
- Each repository is like a branch
- (Repo, Branch) tuple is a unique branch
- Each developer has a full repository with its own history



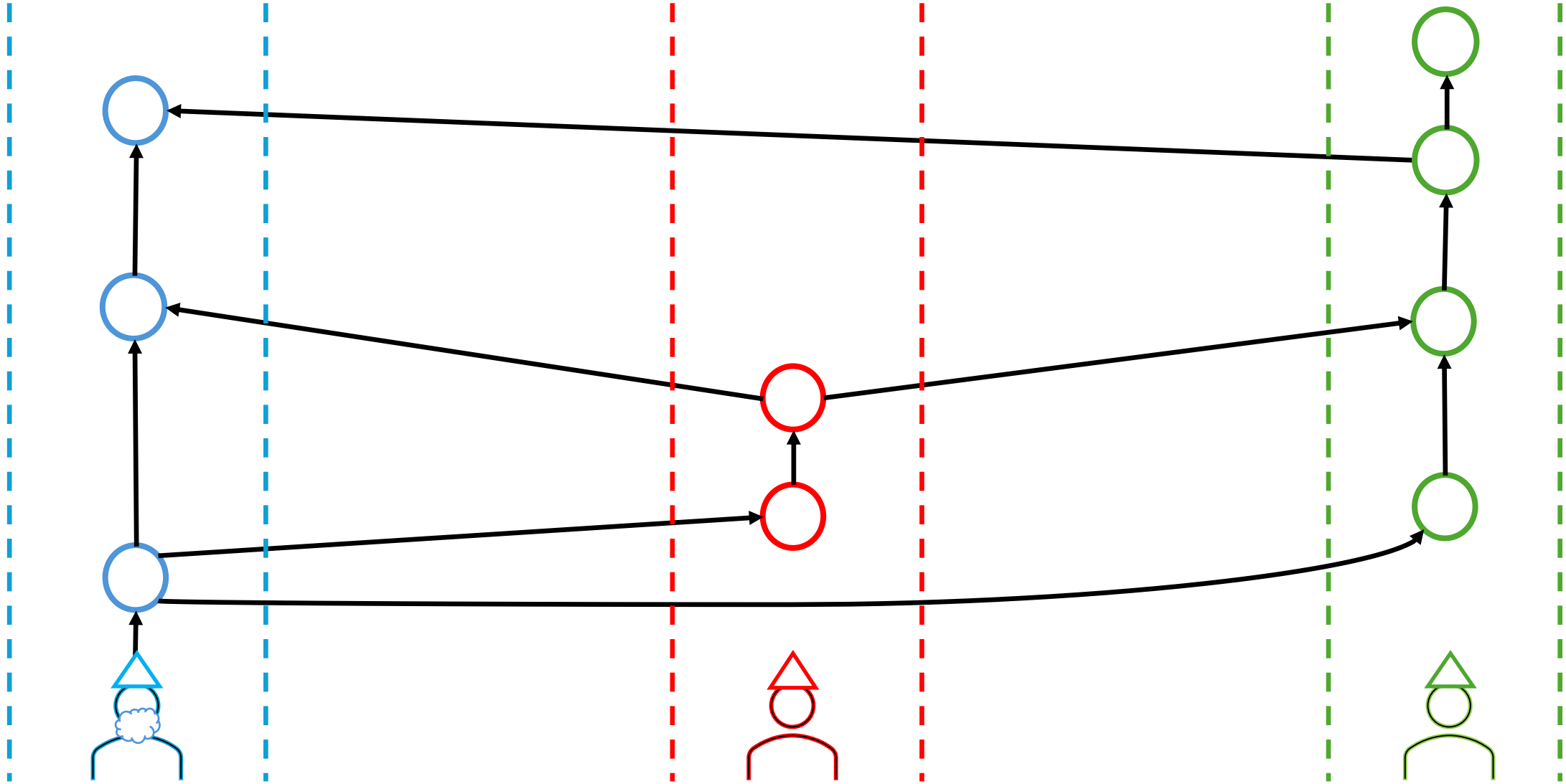
Repository Style: Central Repository

- Shared place for updates, checkouts, modifies, and commits





Repository Style: Separate Towers



Repository Style: Separate Towers

- Only need to give READ access
- No need to give write access
- Provide isolation from change
- Each repository is a personal branch