INTRODUCTION TO GIT

https://git-scm.com/book/en/v2/

Iwan Njoto Sandjaja

August 25, 2016

Petra Christian University

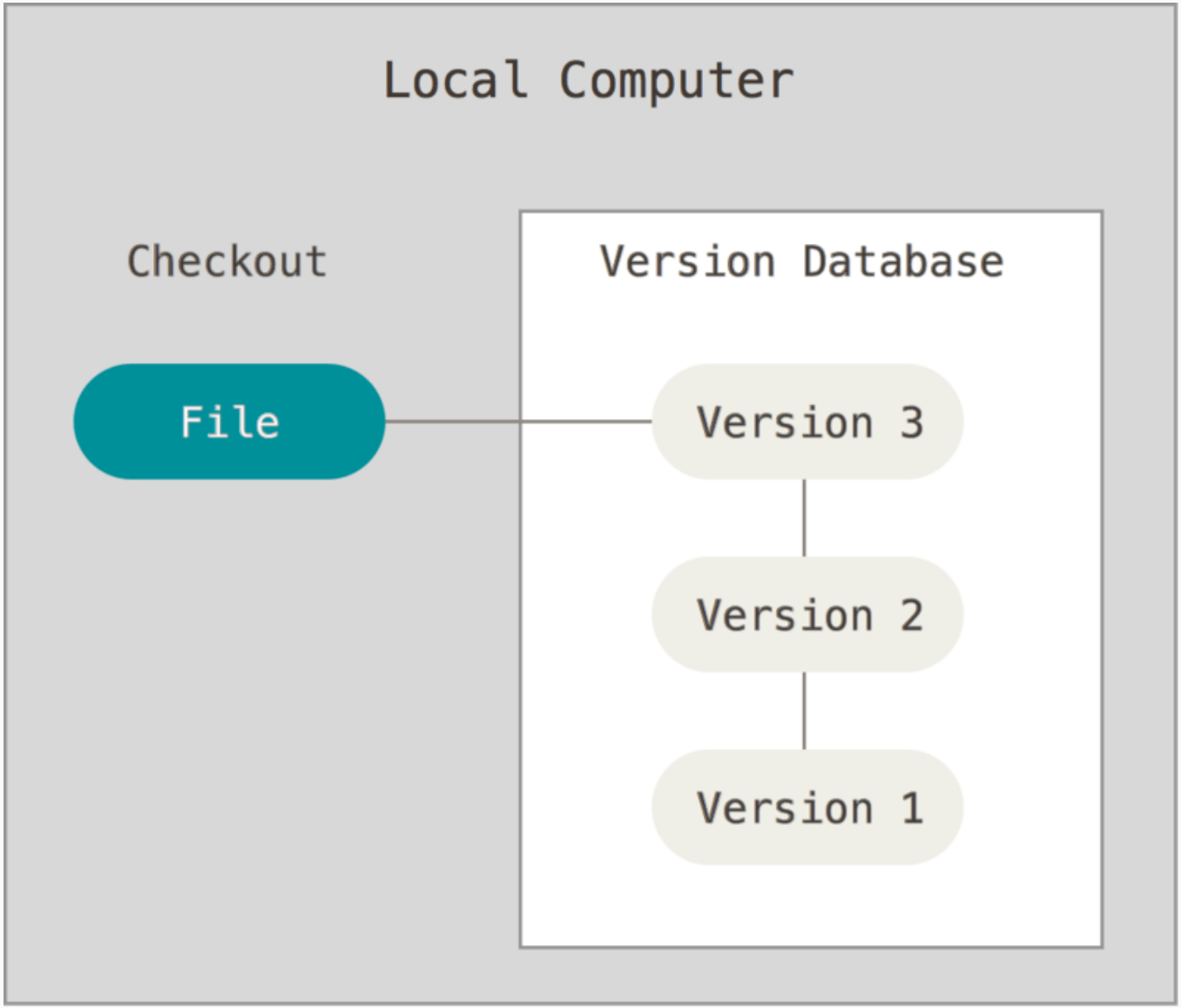
VERSION CONTROL - HEADACHE!?

Final_2_lastversion
THE FINALEST_done
_FIX_copy3_edited

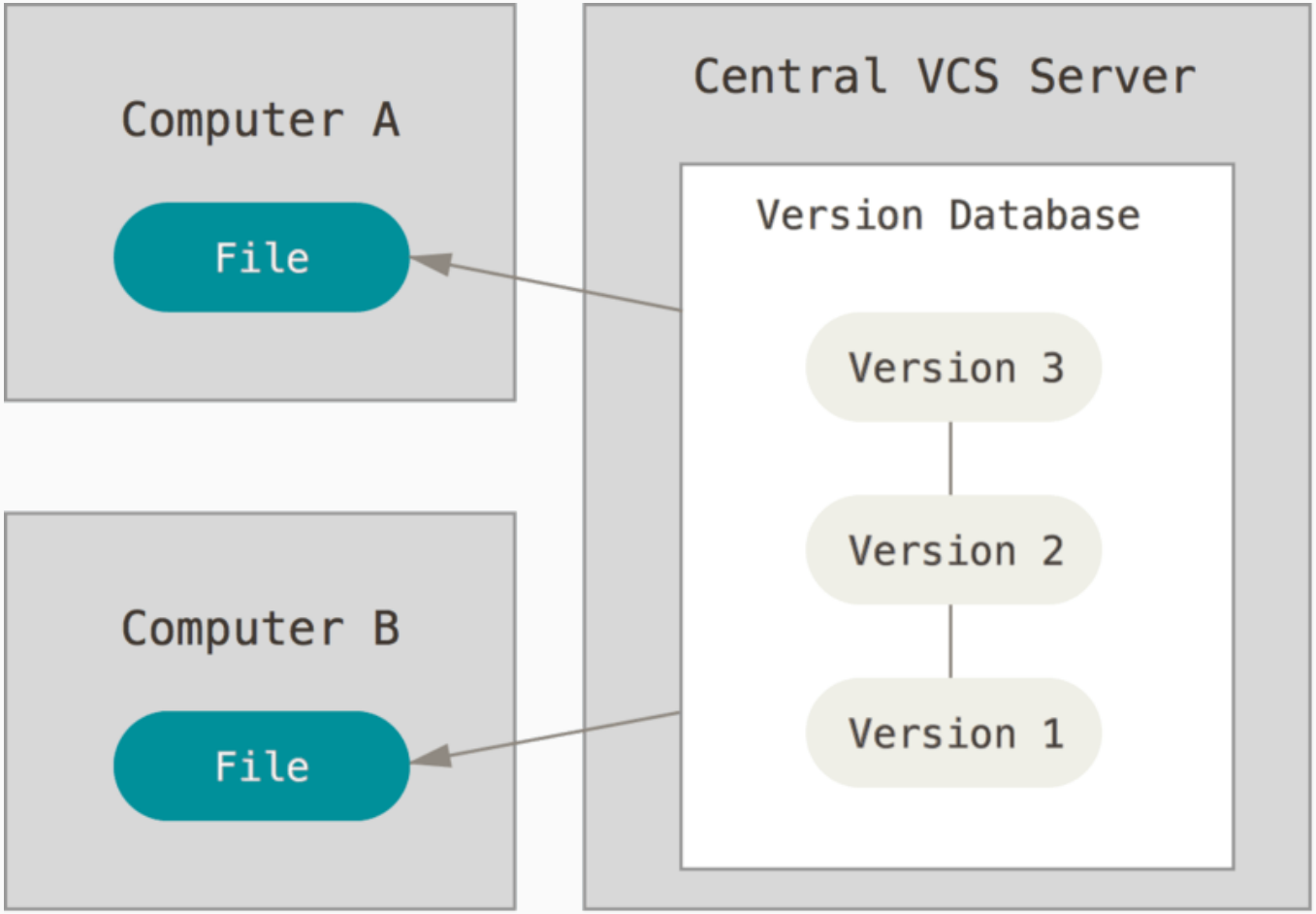
VERSION CONTROL - SOLUTION

1. Local Version Control Systems
2. Centralized Version Control Systems
3. Distributed Version Control Systems

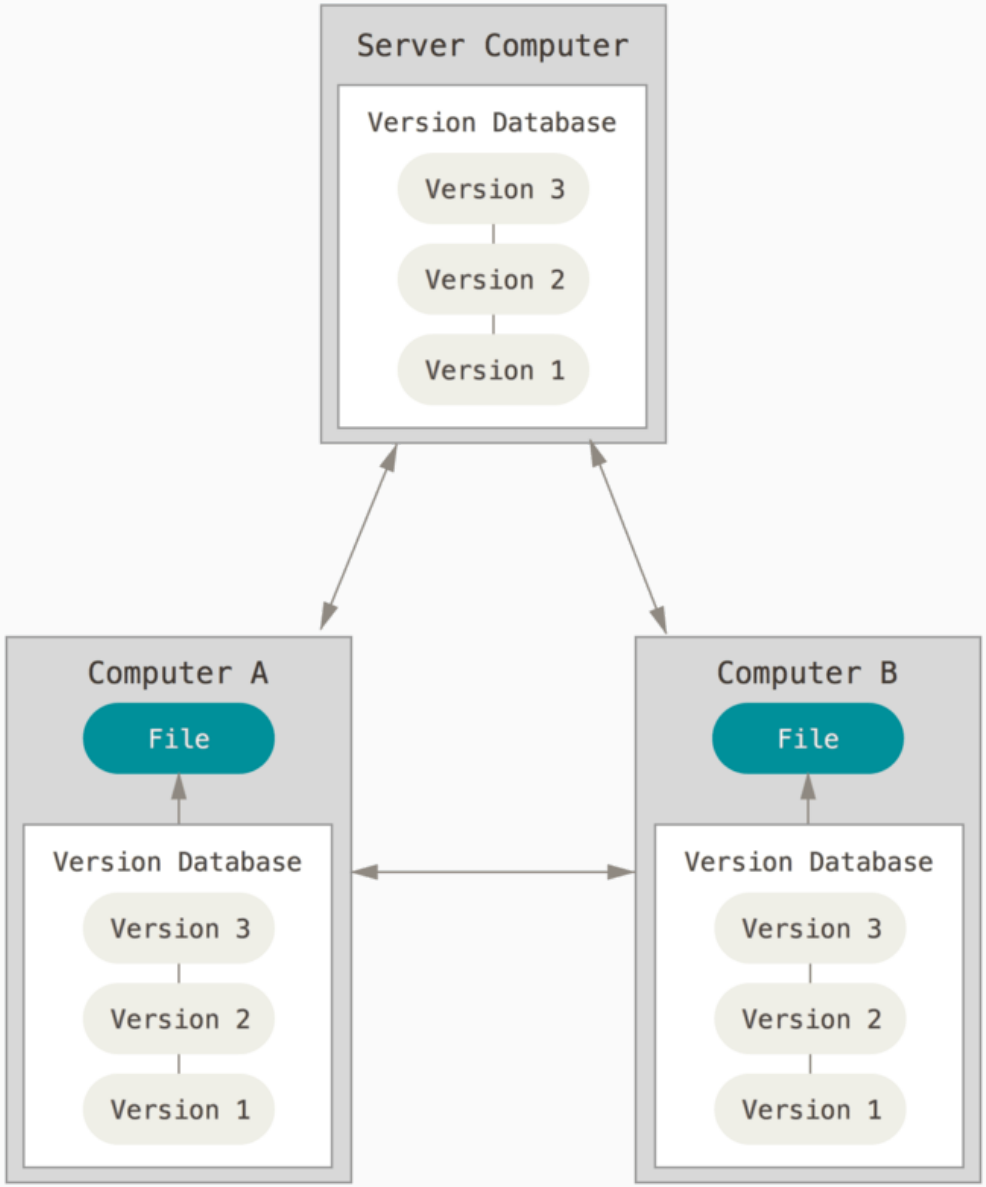
LOCAL VERSION CONTROL SYSTEMS



CENTRALIZED VERSION CONTROL SYSTEMS



DISTRIBUTED VERSION CONTROL SYSTEMS



A SHORT HISTORY OF GIT

In 2002, the Linux kernel project began using a proprietary DVCS called BitKeeper.

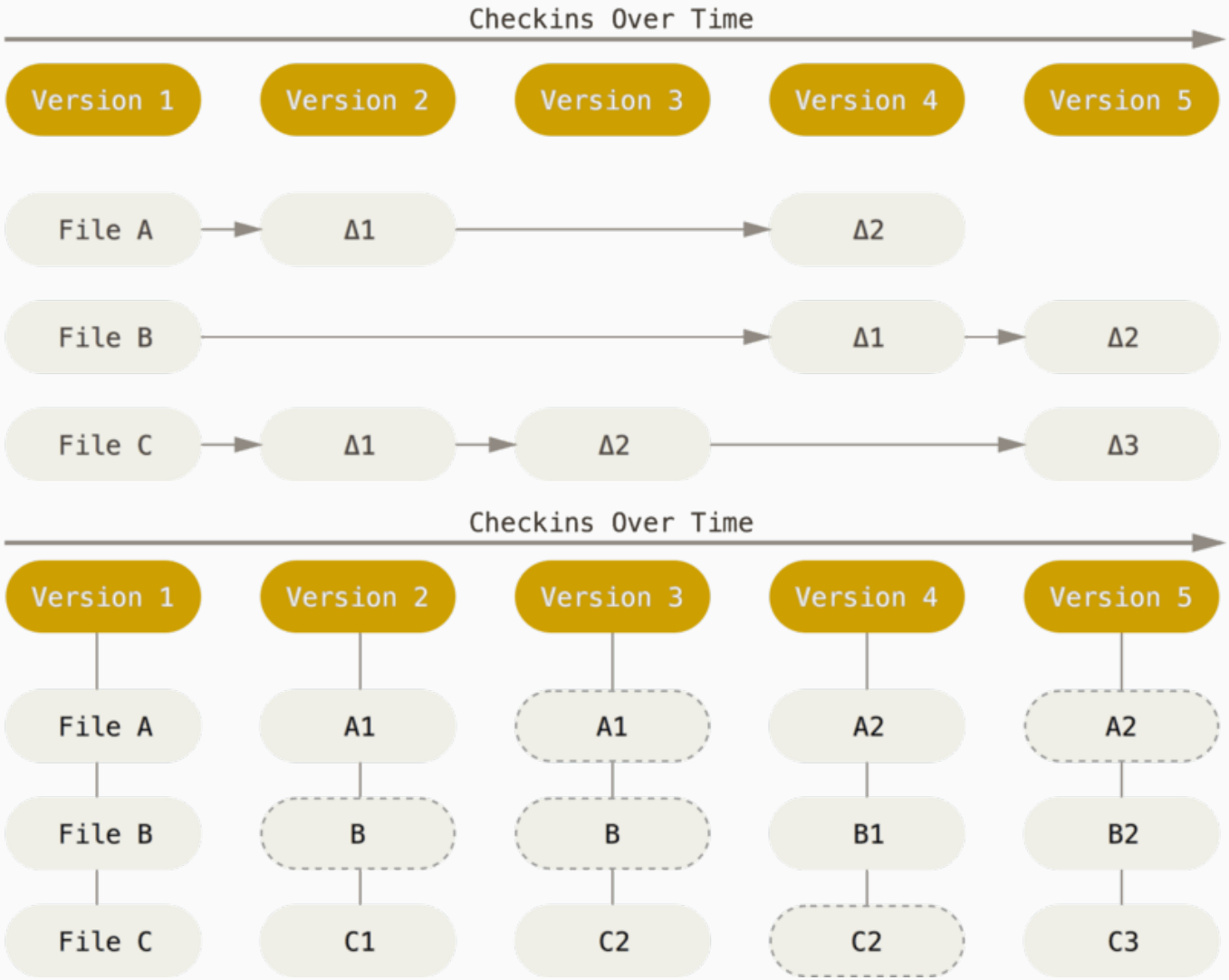
In 2005, BitKeeper is no longer supporting Linux and Linus Torvalds have to develop a new system with these goals:

- Speed
- Simple design
- Strong support for non-linear development (thousands of parallel branches)
- Fully distributed
- Able to handle large projects like the Linux kernel efficiently (speed and data size)

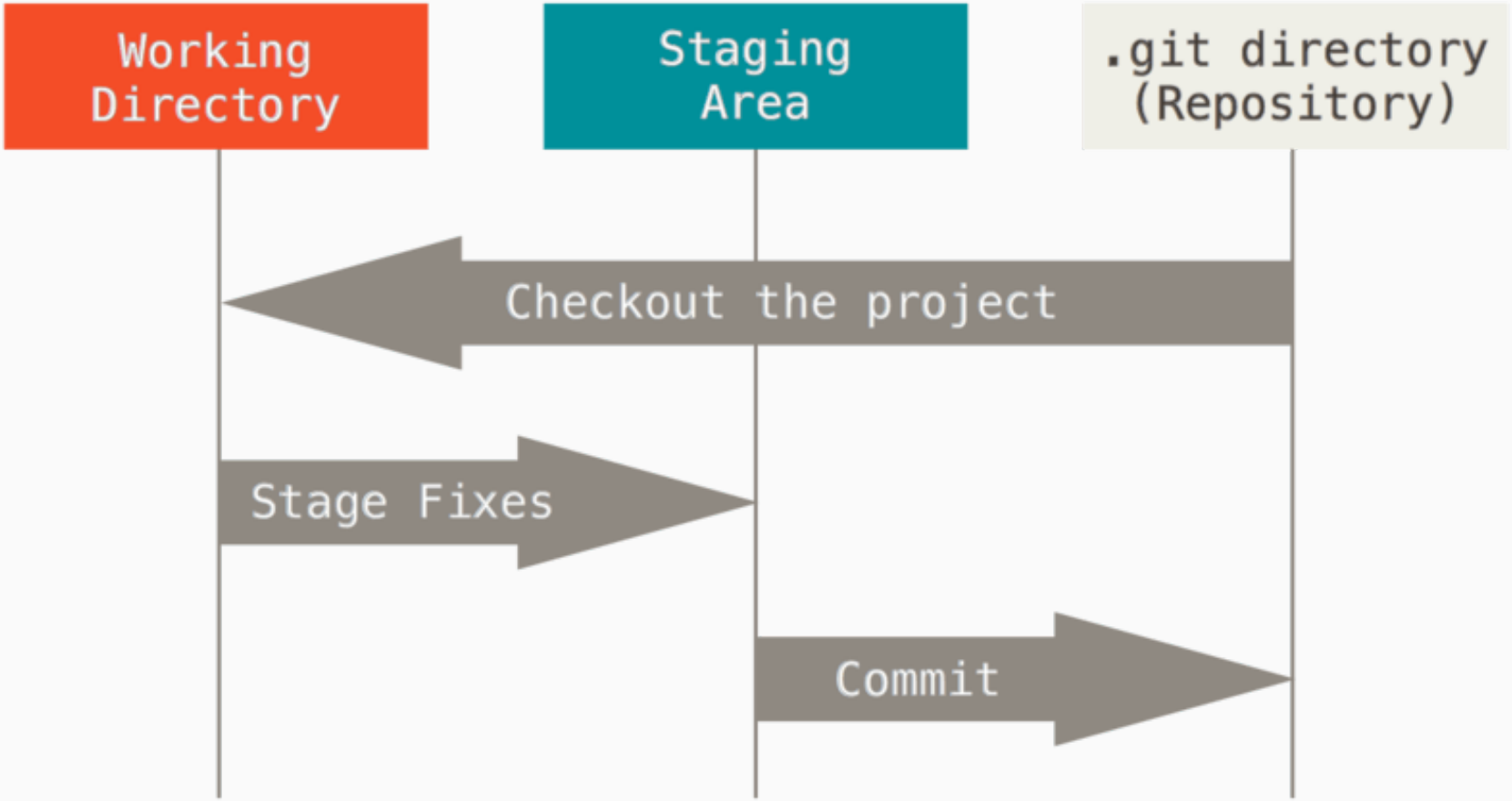
GIT BASICS

- Snapshots, Not Differences
- Nearly Every Operation Is Local
- Git Has Integrity
- Git Generally Only Adds Data
- The Three States

DELTAS VS. SNAPSHOTS



MODIFIED, COMMITED, AND STAGED



GIT WORKFLOW

1. You modify files in your working directory.
2. You stage the files, adding snapshots of them to your staging area.
3. You do a commit, which takes the files as they are in the staging area and stores that snapshot permanently to your Git directory.