CONSULTING.TECHNOLOGY.OUTSOURCING

# GitHub Training

March 2016

People matter, results count.

# Ground Rules for Face-to-face Classrooms

Everyone participates

Respect individual opinions and diversities

Be open and honest

Give headlines, be concise

One speaker at a time

Make language a non-issue

Stick to time contracts

Seek first to understand, and then to be understood

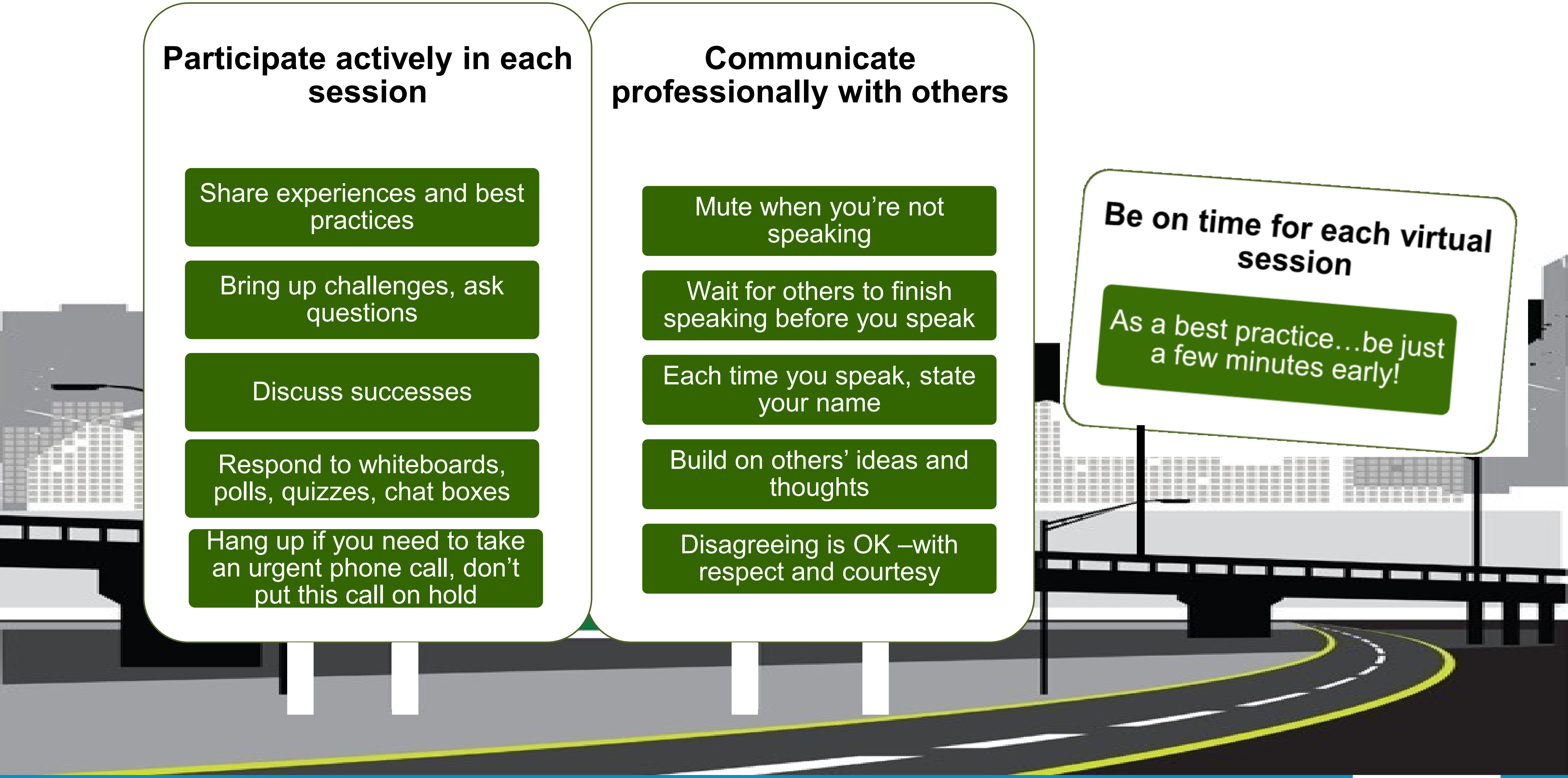
Clean desk / room policy

No mobile phone,
No computer (except for surveys, polls etc)

Clients & Leadership are often around, please remember that

Maintain a spirit of fun and enthusiasm

# Ground Rules for Virtual Classrooms



## Participate actively in each session

- Share experiences and best practices
- Bring up challenges, ask questions
- Discuss successes
- Respond to whiteboards, polls, quizzes, chat boxes
- Hang up if you need to take an urgent phone call, don't put this call on hold

## Communicate professionally with others

- Mute when you're not speaking
- Wait for others to finish speaking before you speak
- Each time you speak, state your name
- Build on others' ideas and thoughts
- Disagreeing is OK –with respect and courtesy

## Be on time for each virtual session

- As a best practice…be just a few minutes early!

# Module at a Glance

SME to provide the details required in the table.

| | |
|---|---|
| **Target Audience:** | |
| **Course Level:** | *Basic* |
| **Duration (in hours):** | 30 mins |
| **Pre-requisites, if any:** | NA |
| **Post-requisites, if any:** | *Submit Session Feedback* |
| **Relevant Certifications:** | None |

The information contained in this document is proprietary and confidential.

# Introductions (for Virtual Classrooms)

SME to provide the photos and names of the facilitators.

Business Photo

Business Photo

*Facilitator*
**Name**
**Role**

*Moderator*
**Name**
**Role**

# Agenda

[redacted]

1 GitHub for Windows Developers

2 Using the GitHub Website

3 GitHub for Windows Basics

4 Teamwork with GitHub for Windows

# Module Objectives

Note to the SME : Please provide the module Objectives or validate the partially updated content

## What you will learn

At the end of this module, you will learn:

- What is GitHub

## What you will be able to do

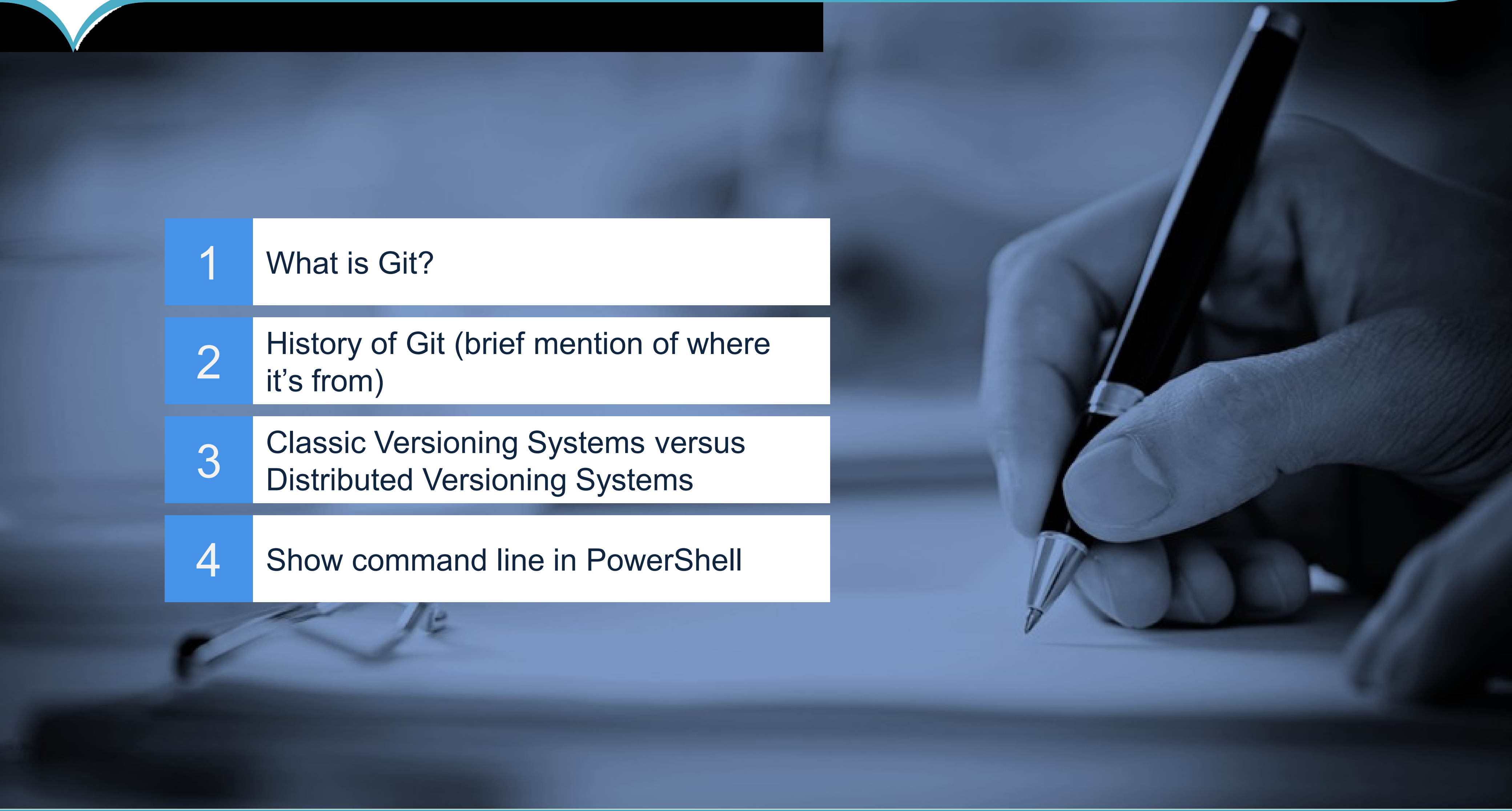
At the end of this module, you be able to:

- Explain how GitHub can be used by Windows Developers
- Describe how to usr the GitHub Website
- Understand GitHub for Windows Basics
- State how  GitHub and Windows work as a team

# GitHub for Windows Developers

# Outline

[redacted]



1. What is Git?
2. History of Git (brief mention of where it’s from)
3. Classic Versioning Systems versus Distributed Versioning Systems
4. Show command line in PowerShell

# What is Git?

Version Control System ("Source Control")

Distributed Version Control System ("DVCS")

Very Fast

Highly Scalable

# What is Git? (contd.)

Distributed Version Control System ("DVCS")

Branch

Merge

Commit

Push

Pull

Very Fast

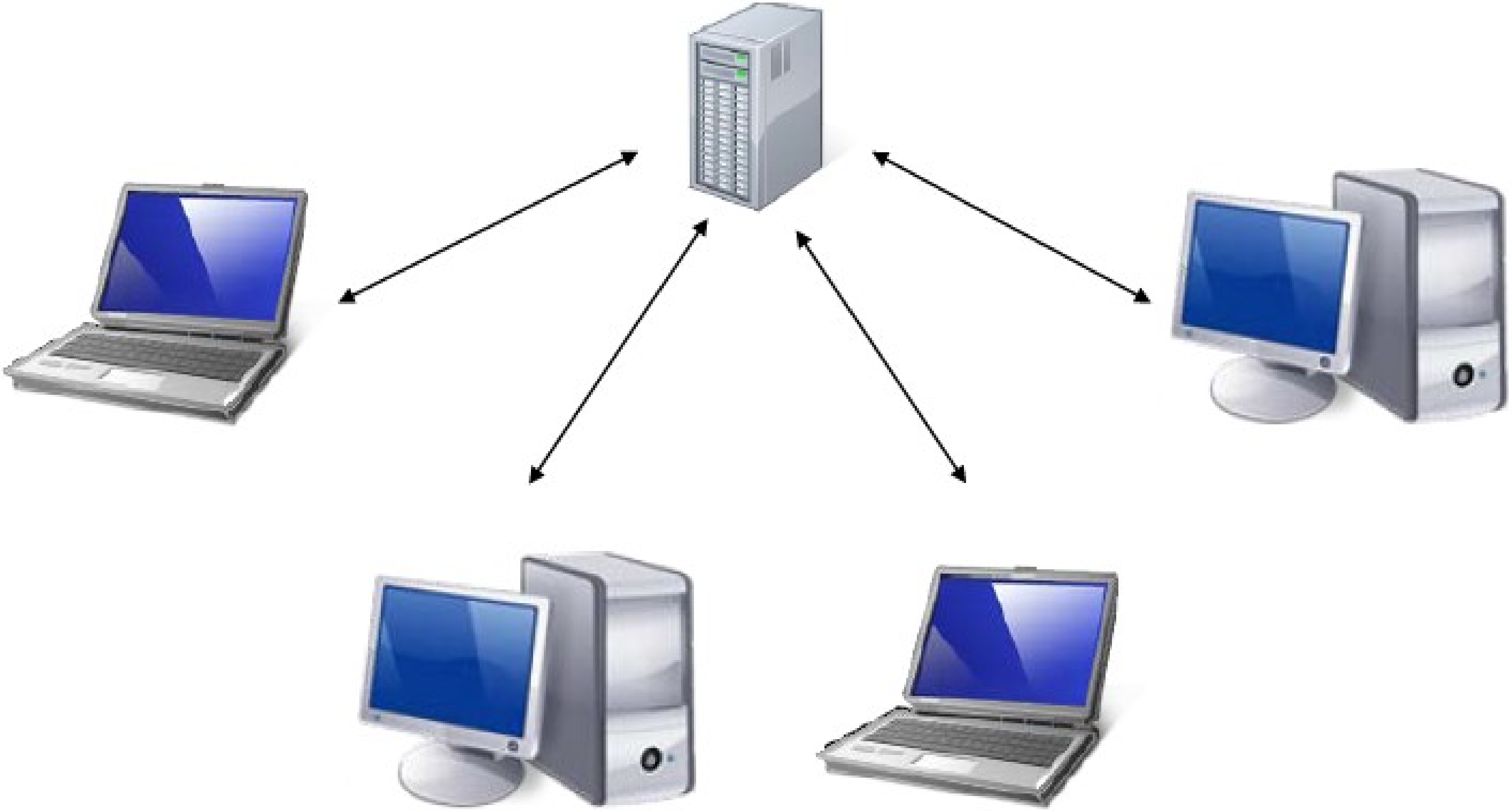
Scalable

Distributed

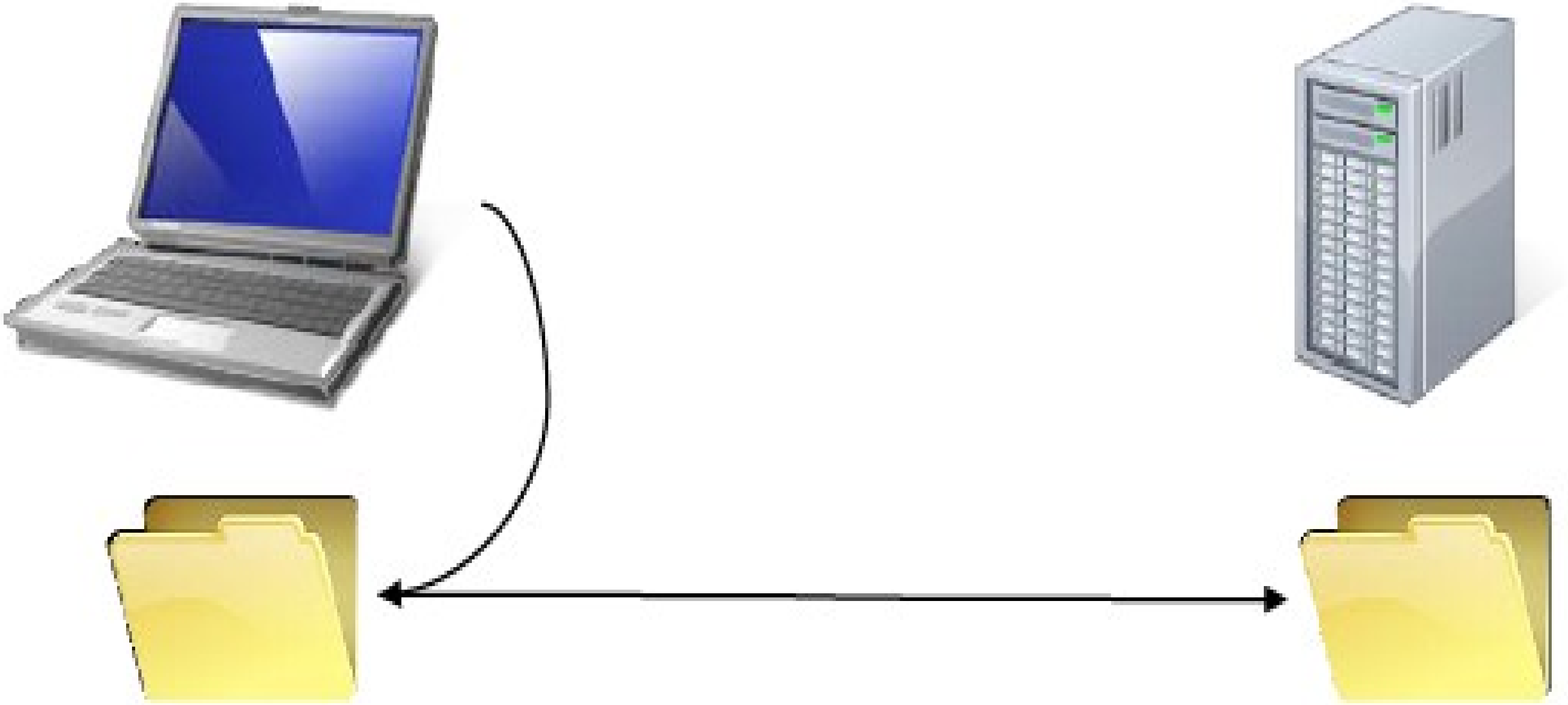
Parallel

# History of Git

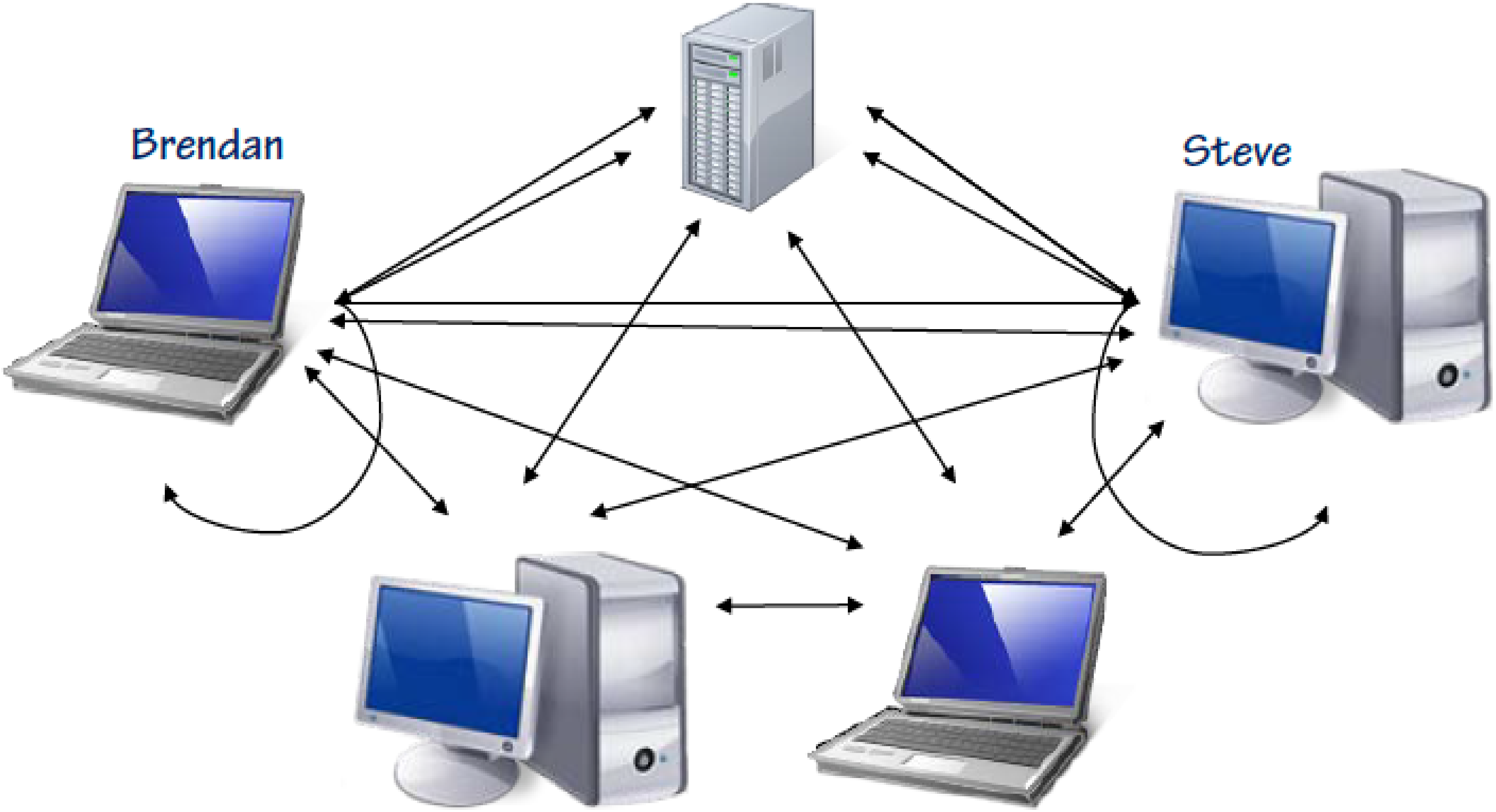
# Centralized Version Control



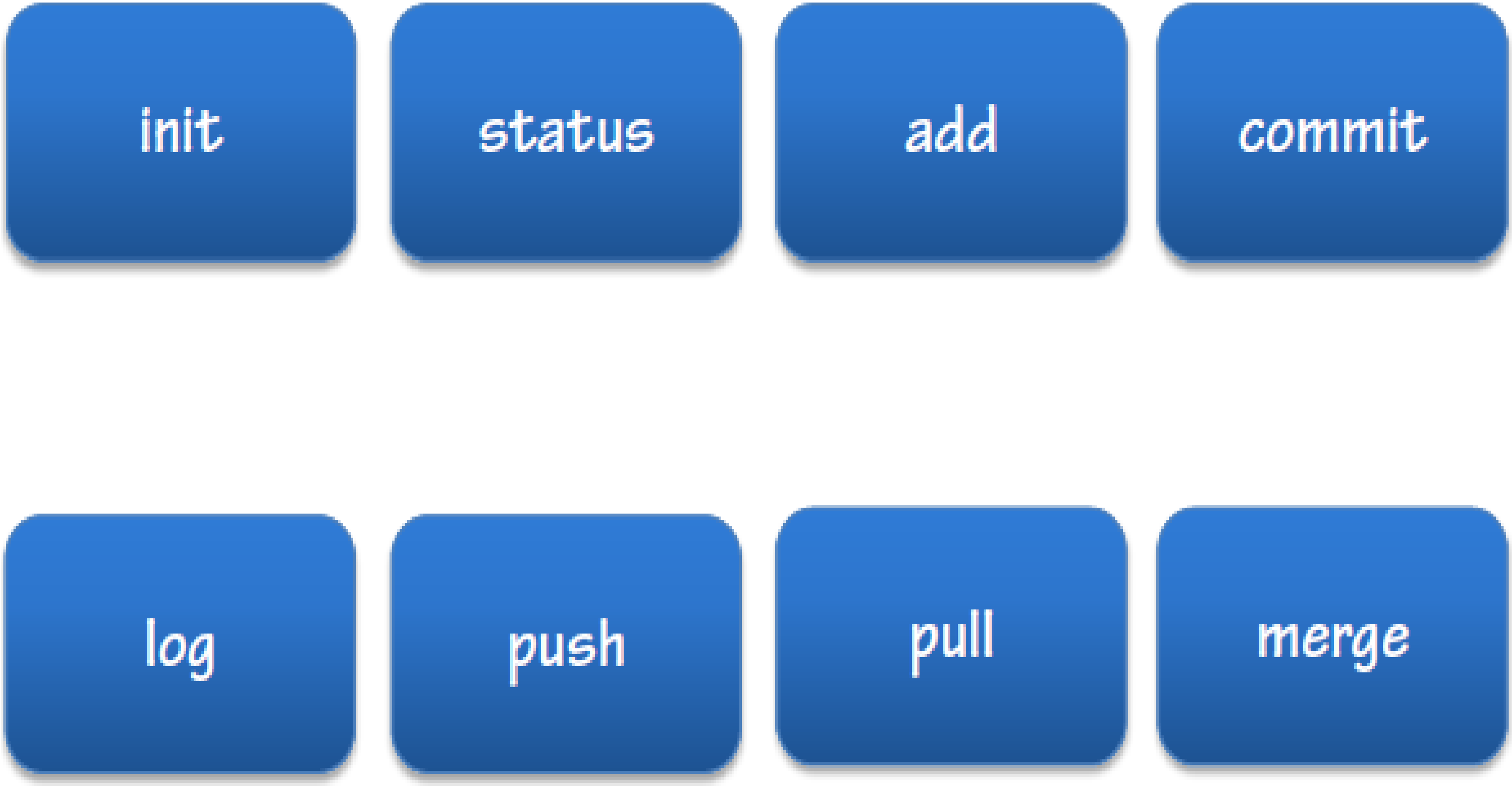
# Distributed Version Control



# Distributed Version Control (contd.)



# Basic Git Commands



init

status

add

commit

log

push

pull

merge

# Summary

Git is a distributed version control system.

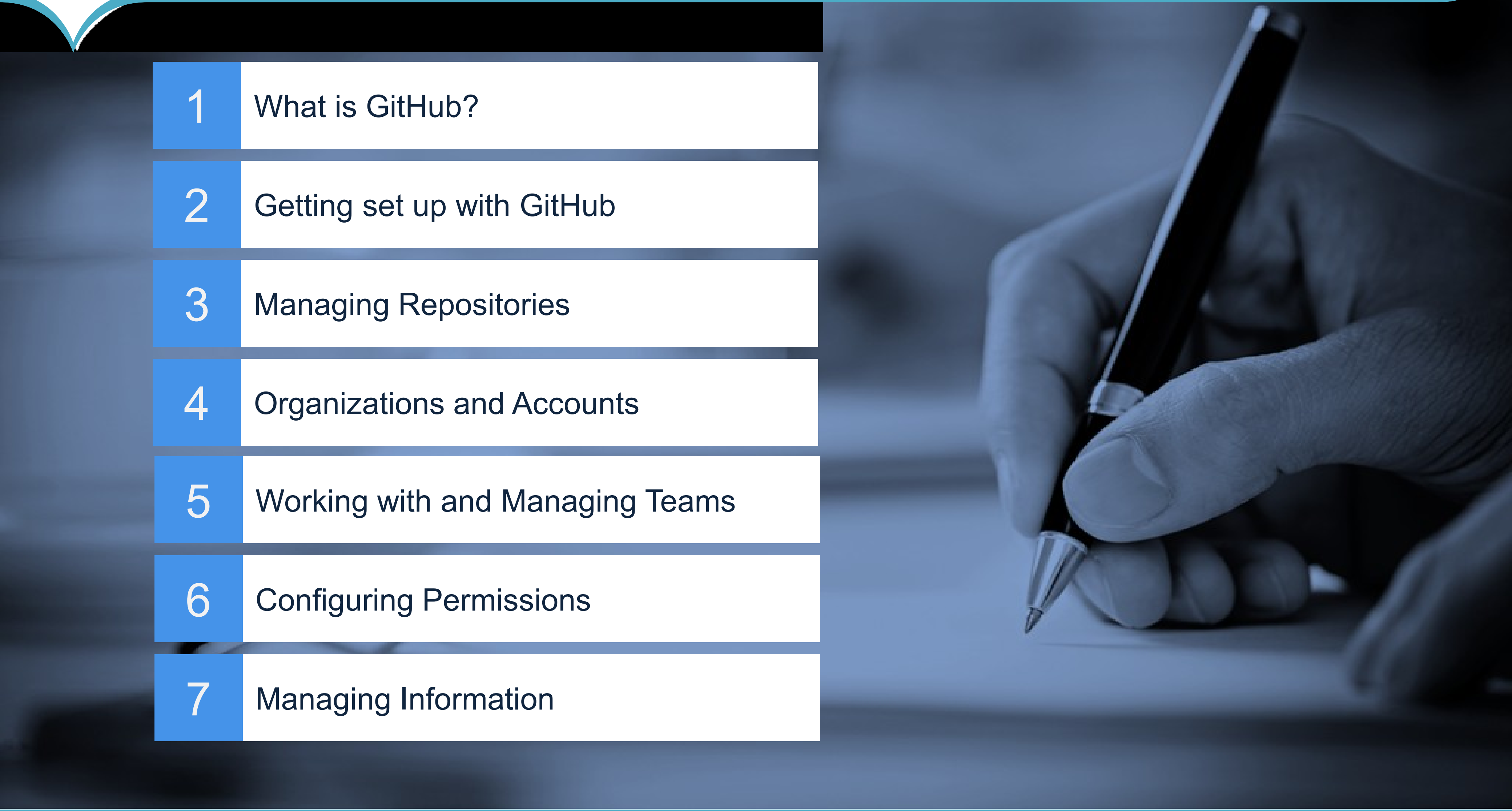
Commits happen locally.

It makes branching and merging easier.

Basic git commands and workflow.

# Using the GitHub Website

# Outline



1. What is GitHub?
2. Getting set up with GitHub
3. Managing Repositories
4. Organizations and Accounts
5. Working with and Managing Teams
6. Configuring Permissions
7. Managing Information

# What is GitHub?

**Hosted Git repositories**

**Project Collaboration**

**Social Coding**

**Octodex(http://octodex.github.com/)**

# Summary

GitHub is a hosting place for projects using Git.

How to set up repositories on accounts and organizations.

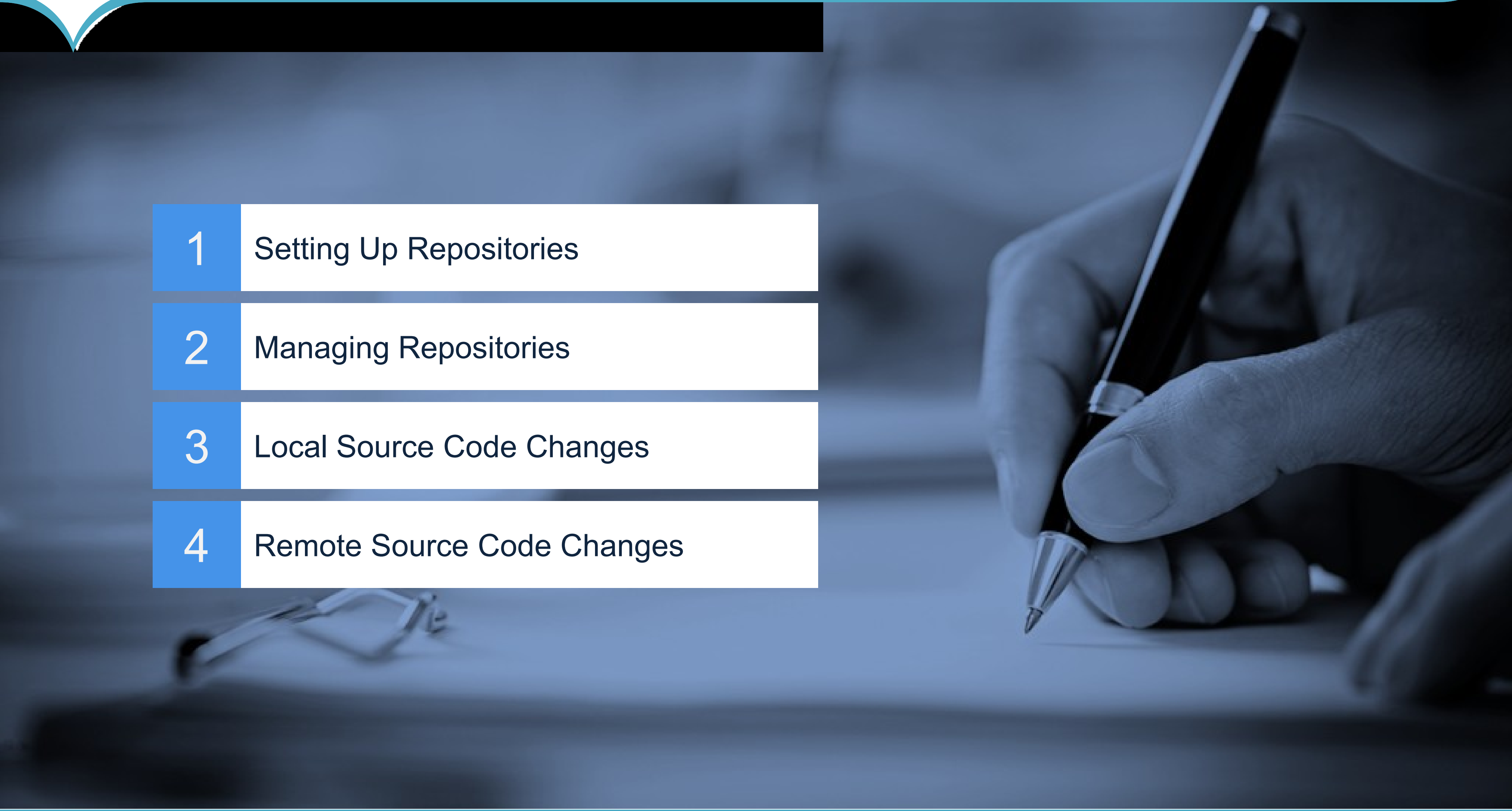
Accounts and Organizations.

How to collaborate and manage some permissions.

Managing information using issues, wiki, and pages.

# GitHub for Windows Basics

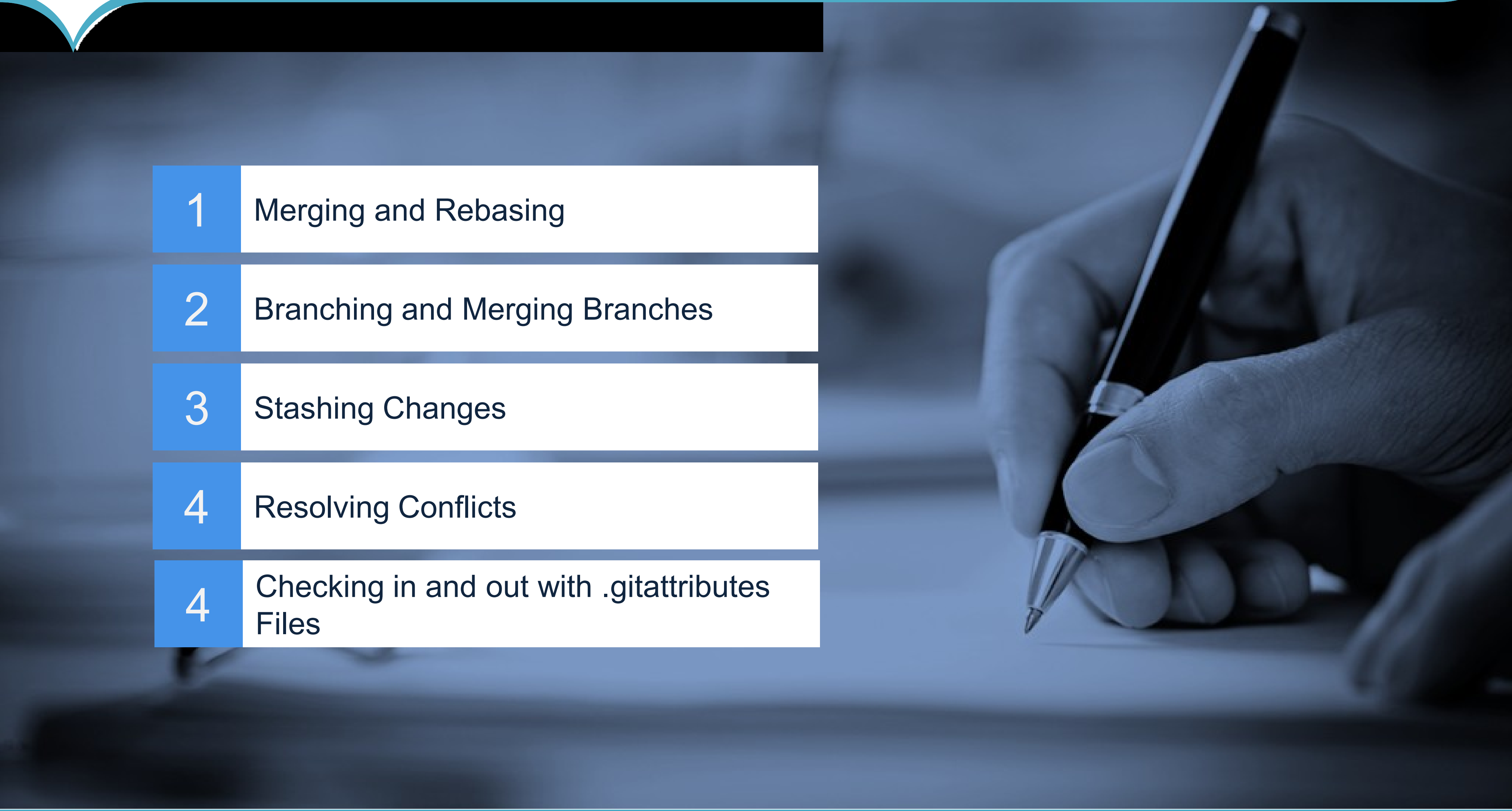
# Outline



1. Setting Up Repositories
2. Managing Repositories
3. Local Source Code Changes
4. Remote Source Code Changes

# Teamwork with GitHub for Windows

# Outline



1 Merging and Rebasing

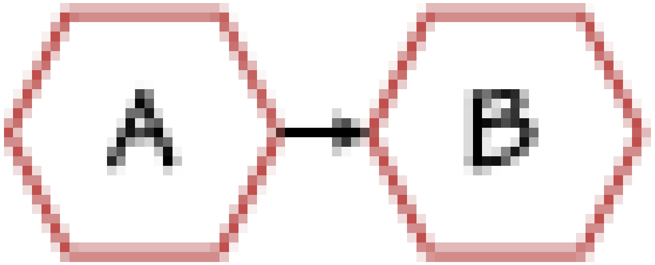
2 Branching and Merging Branches

3 Stashing Changes

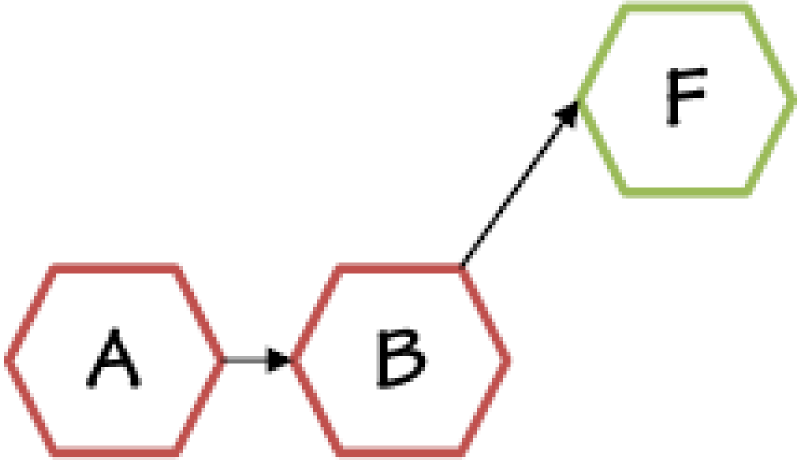
4 Resolving Conflicts

4 Checking in and out with .gitattributes Files

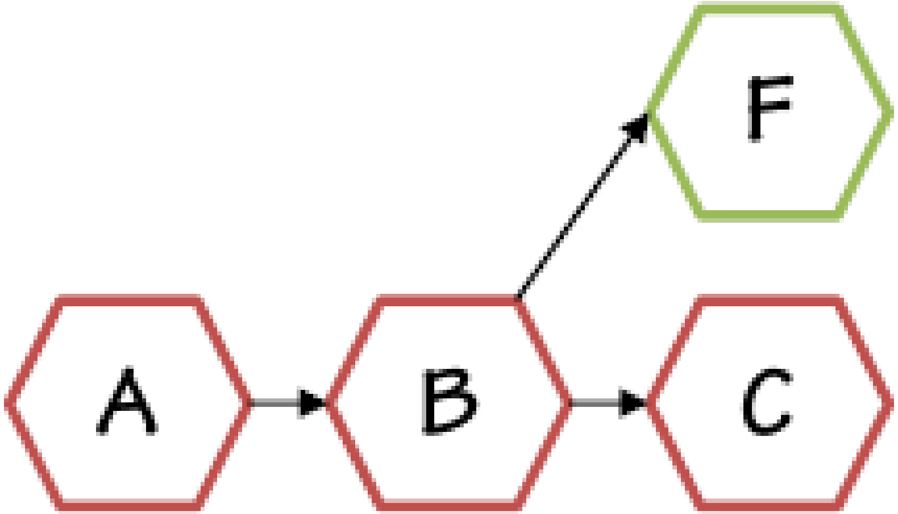
# Merging Example



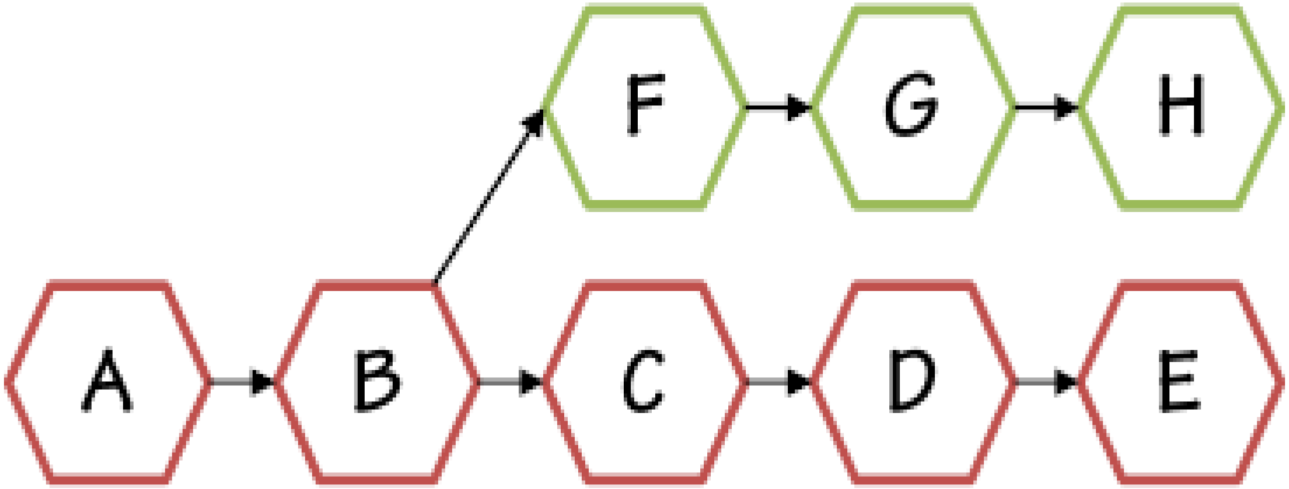
# Merging Example (contd.)



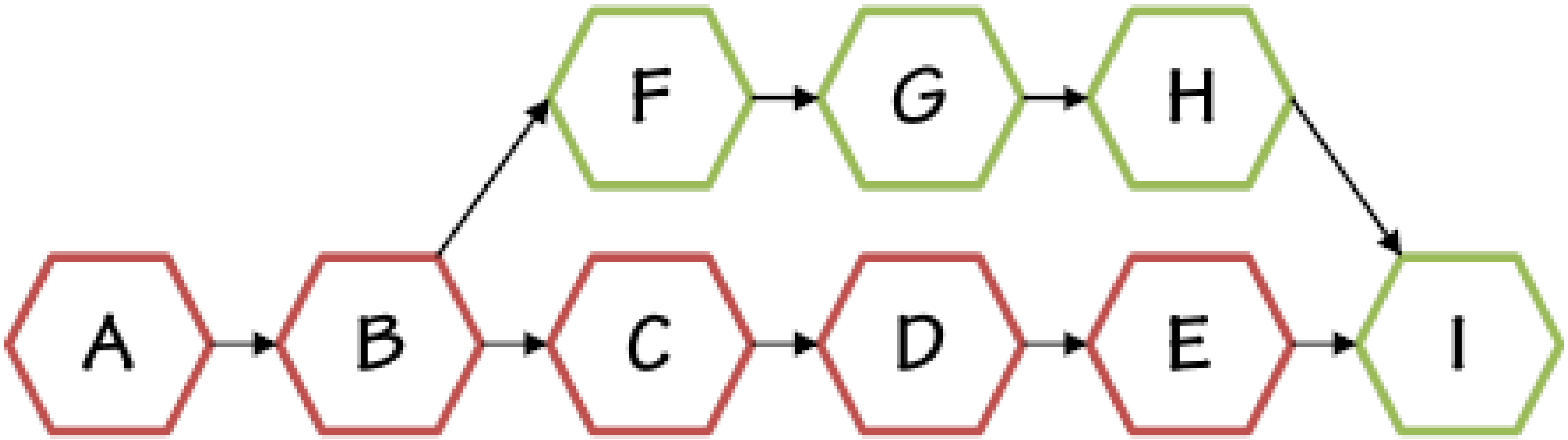
# Merging Example (contd.)



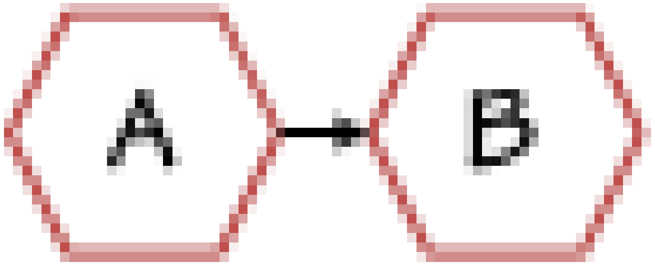
# Merging Example (contd.)



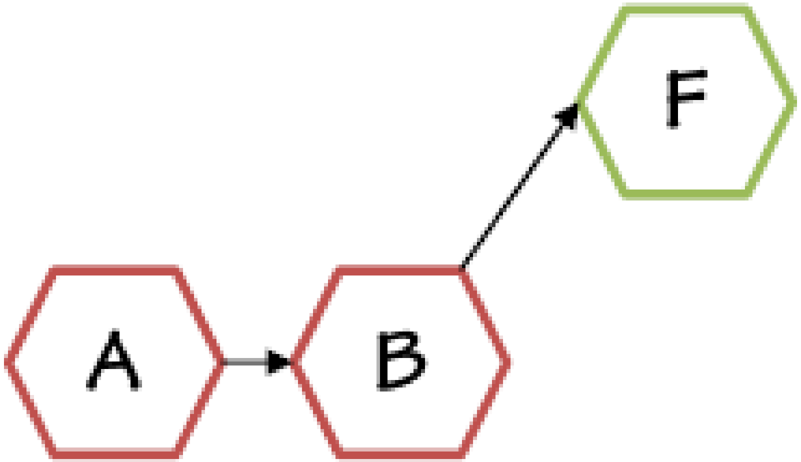
# Merging Example (contd.)



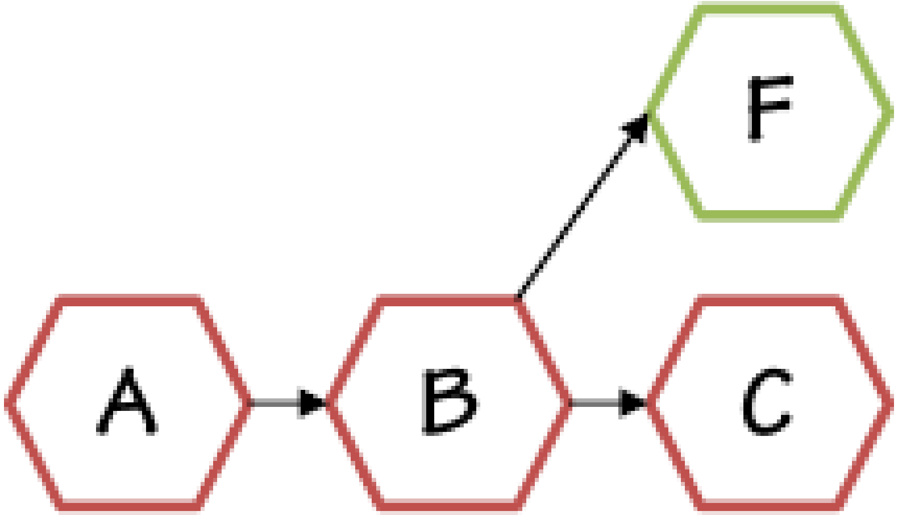
# Rebasing Example 1



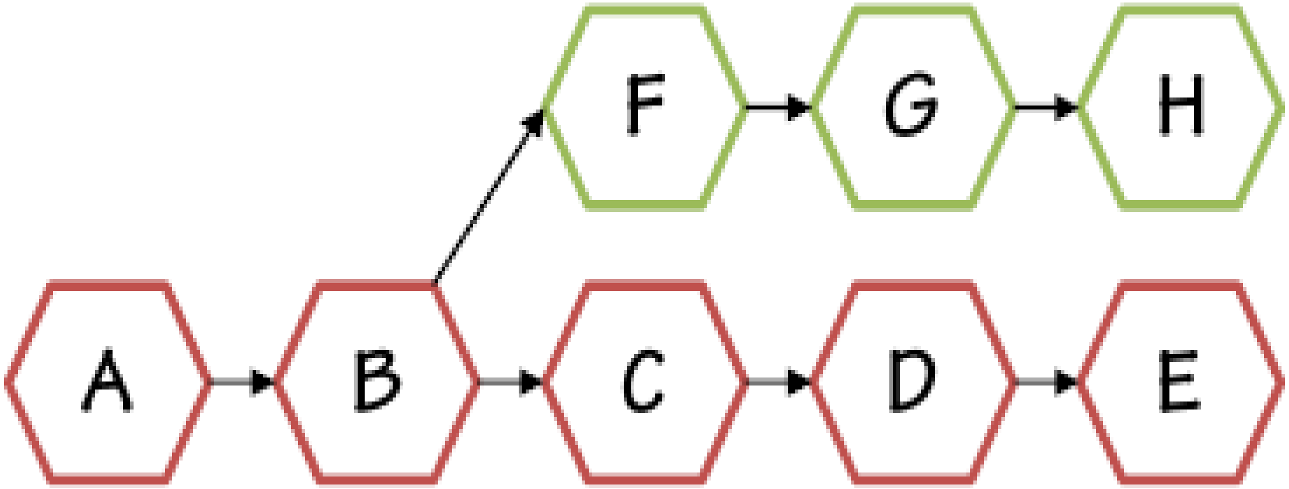
# Rebasing Example 1 (contd.)



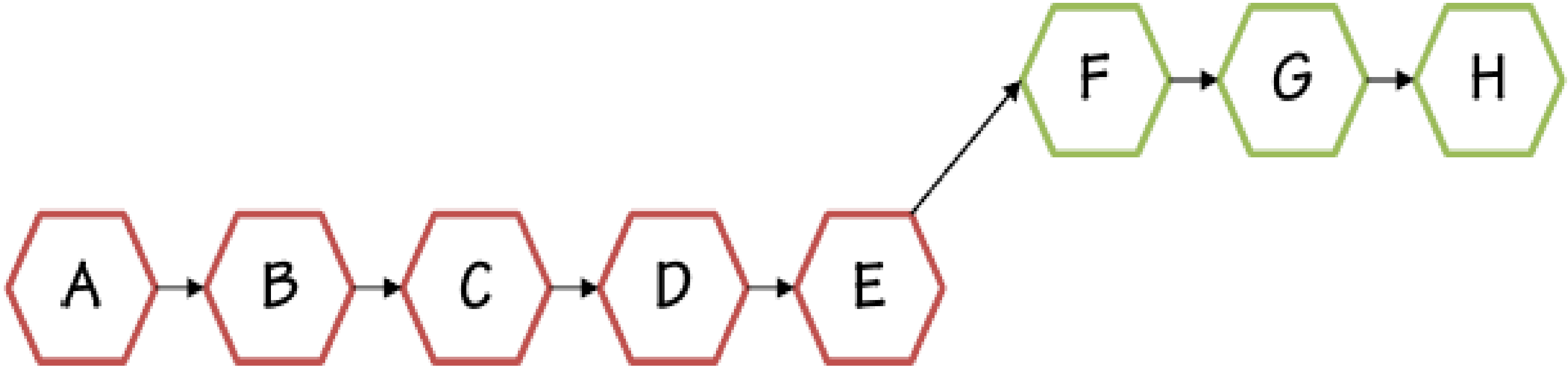
# Rebasing Example 1 (contd.)



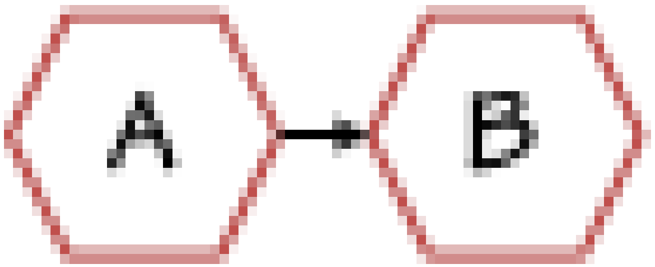
# Rebasing Example 1 (contd.)



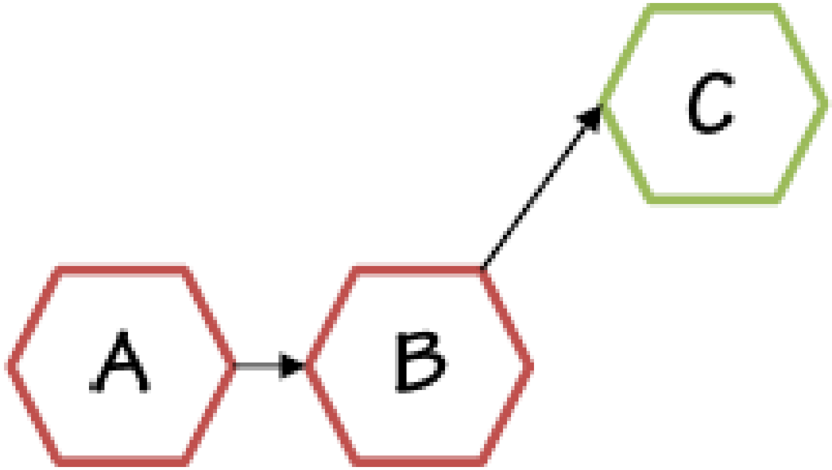
# Rebasing Example 1 (contd.)



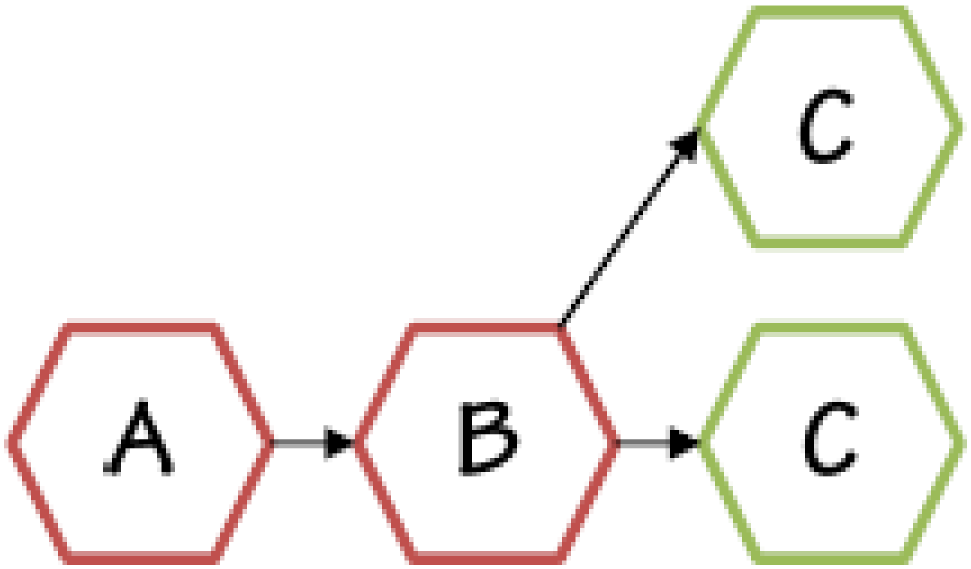
# Rebasing Example 2



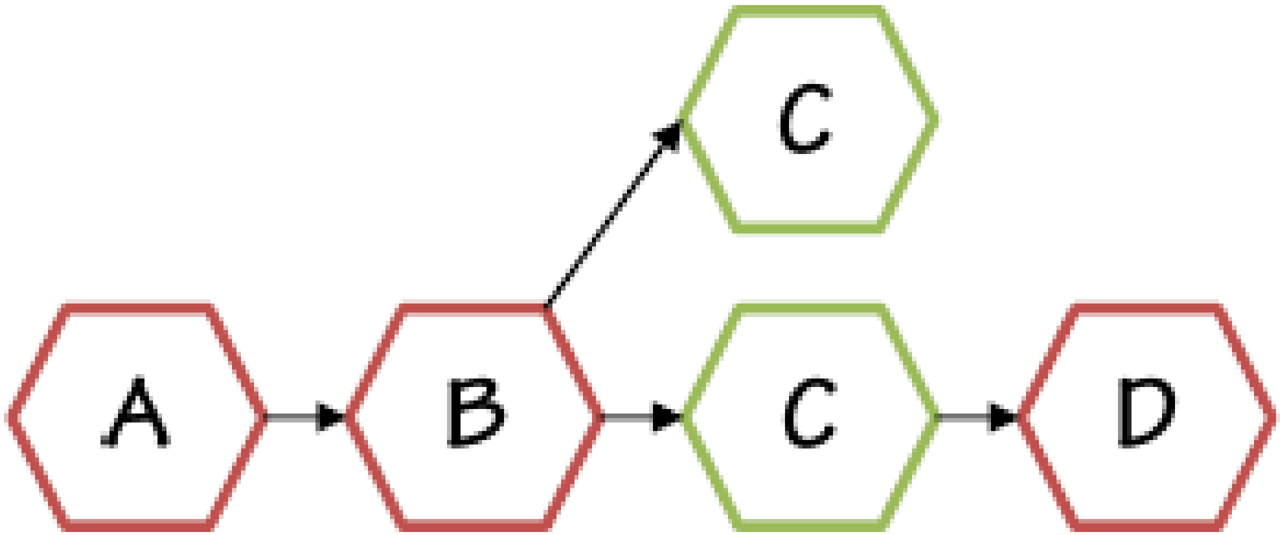
# Rebasing Example 2 (contd.)



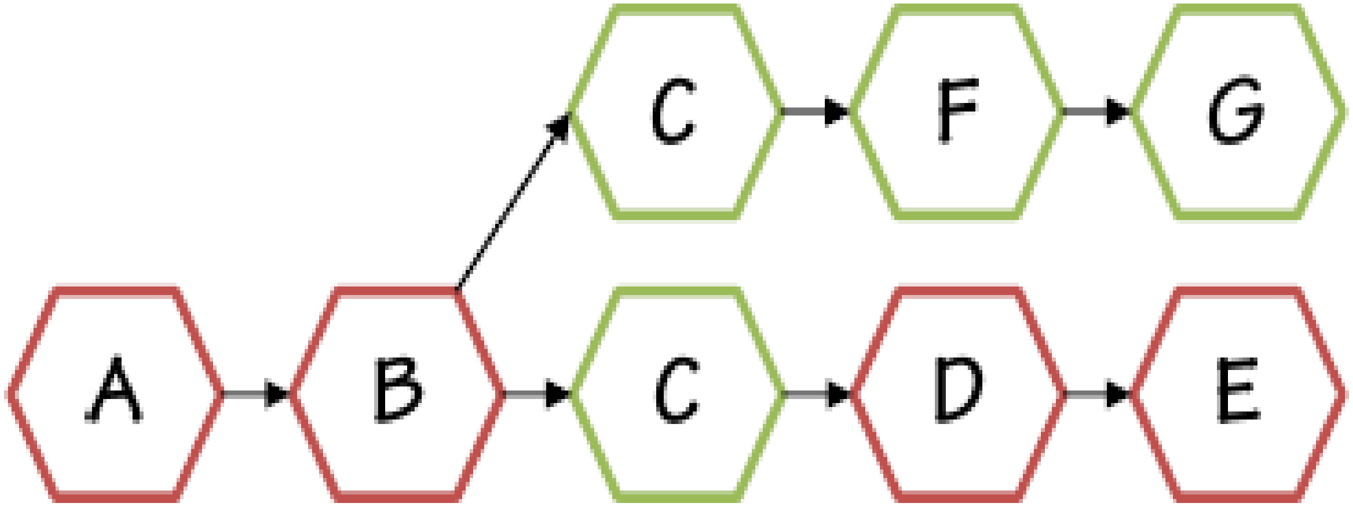
# Rebasing Example 2 (contd.)



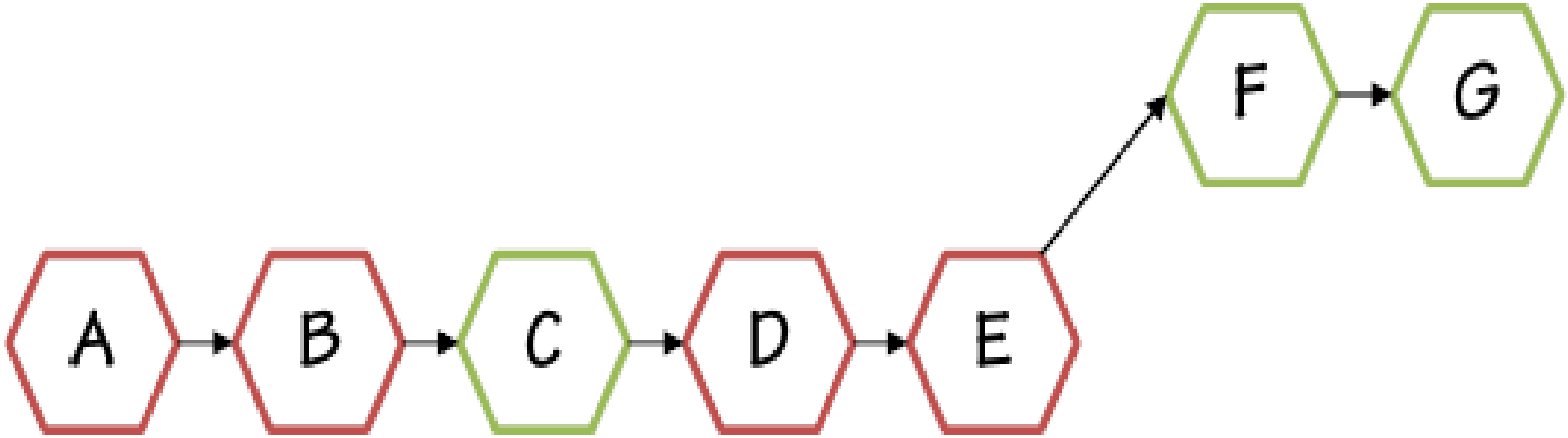
# Rebasing Example 2 (contd.)



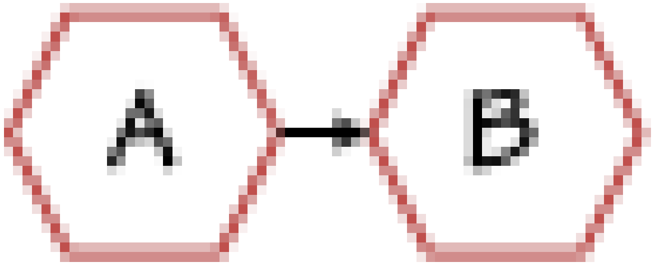
# Rebasing Example 2 (contd.)



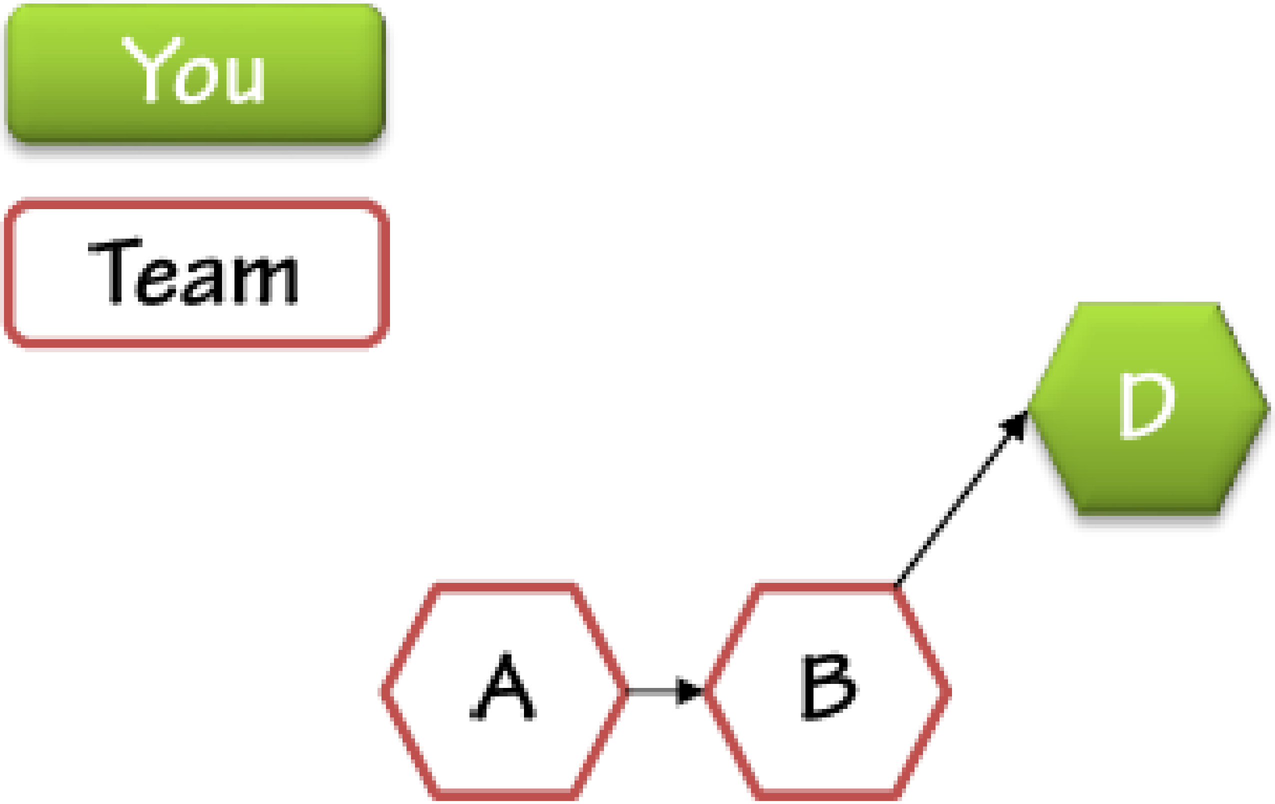
# Rebasing Example 2 (contd.)



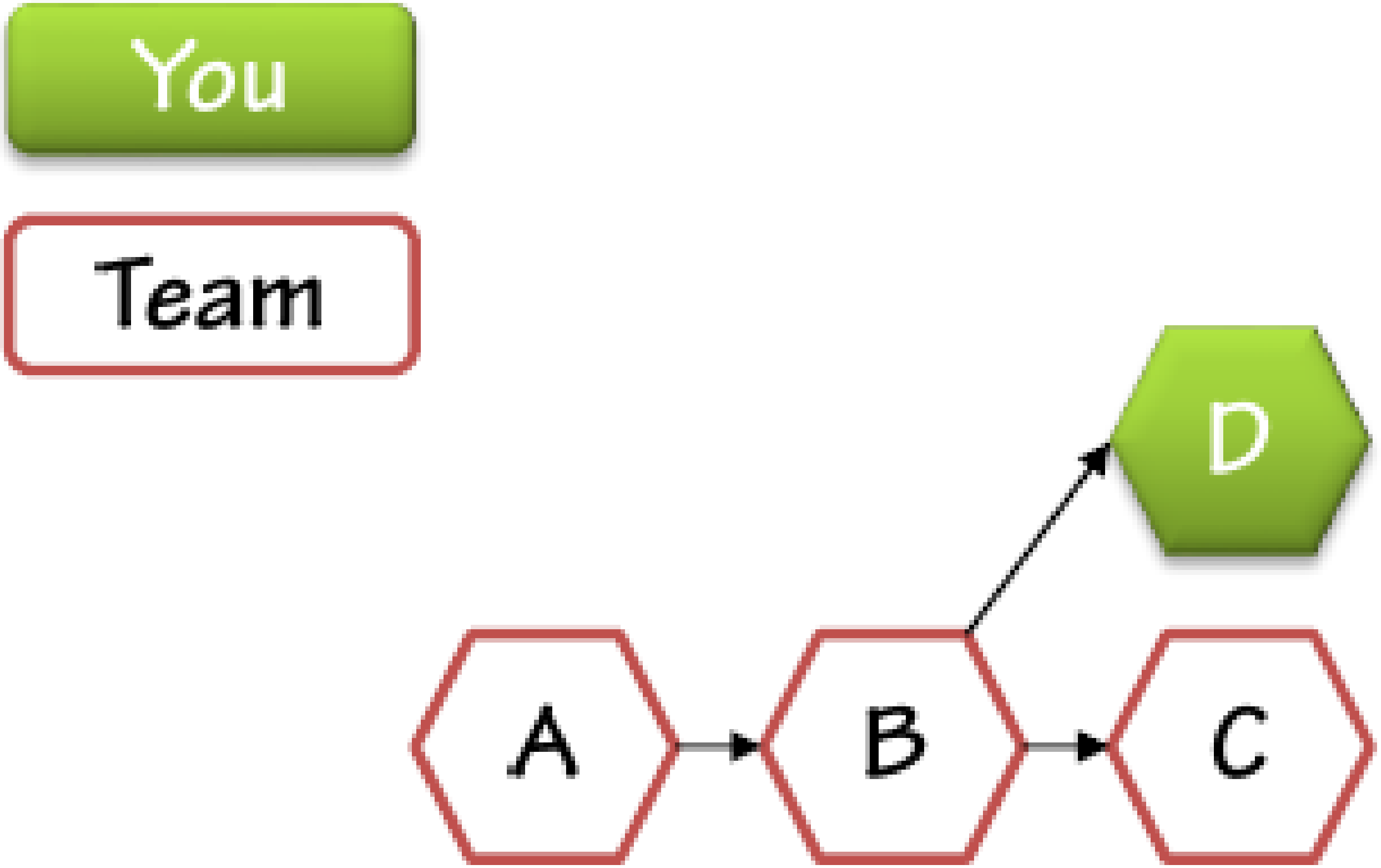
# Dangerous Rebase Example



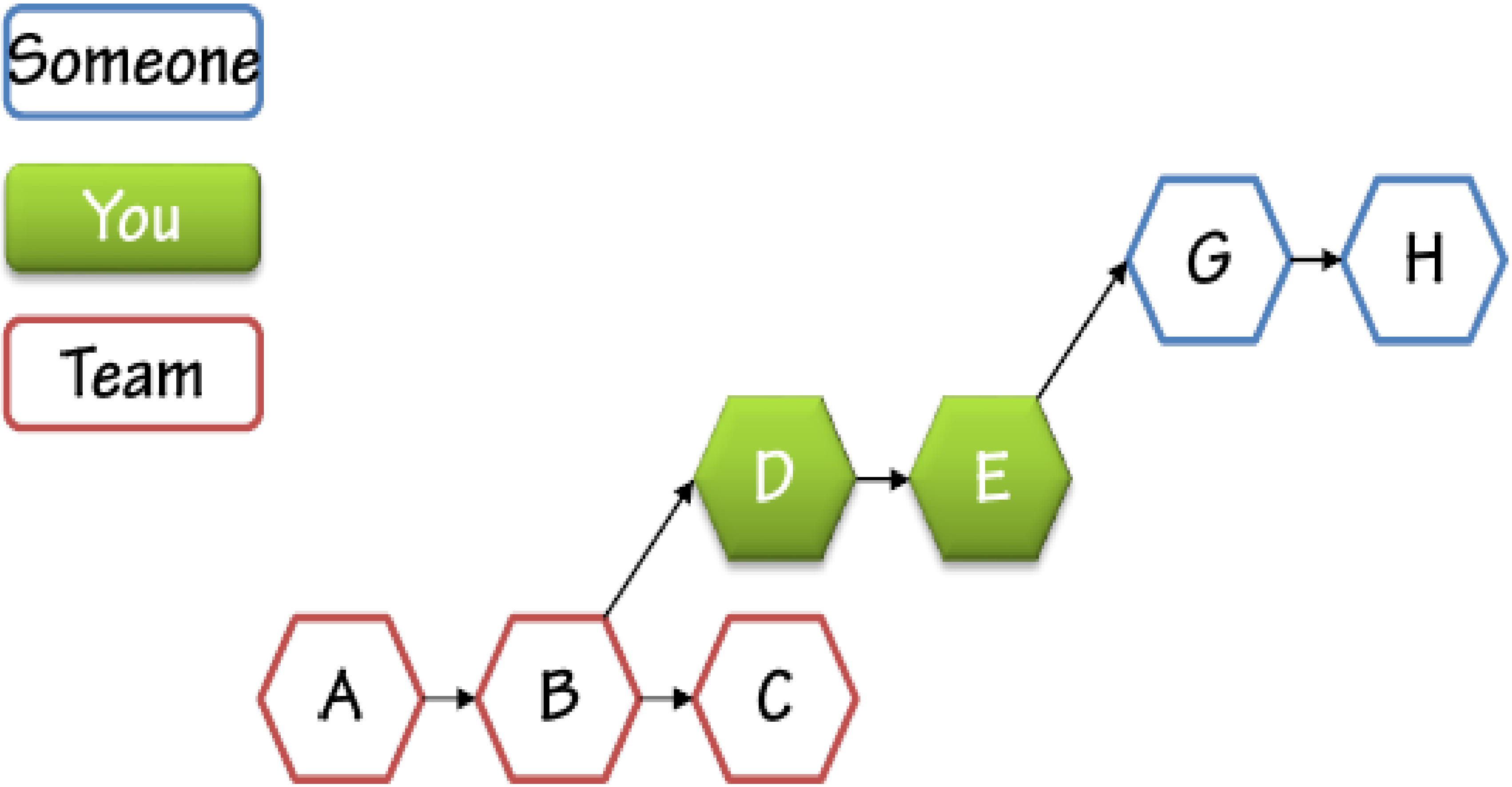
# Dangerous Rebase Example 2



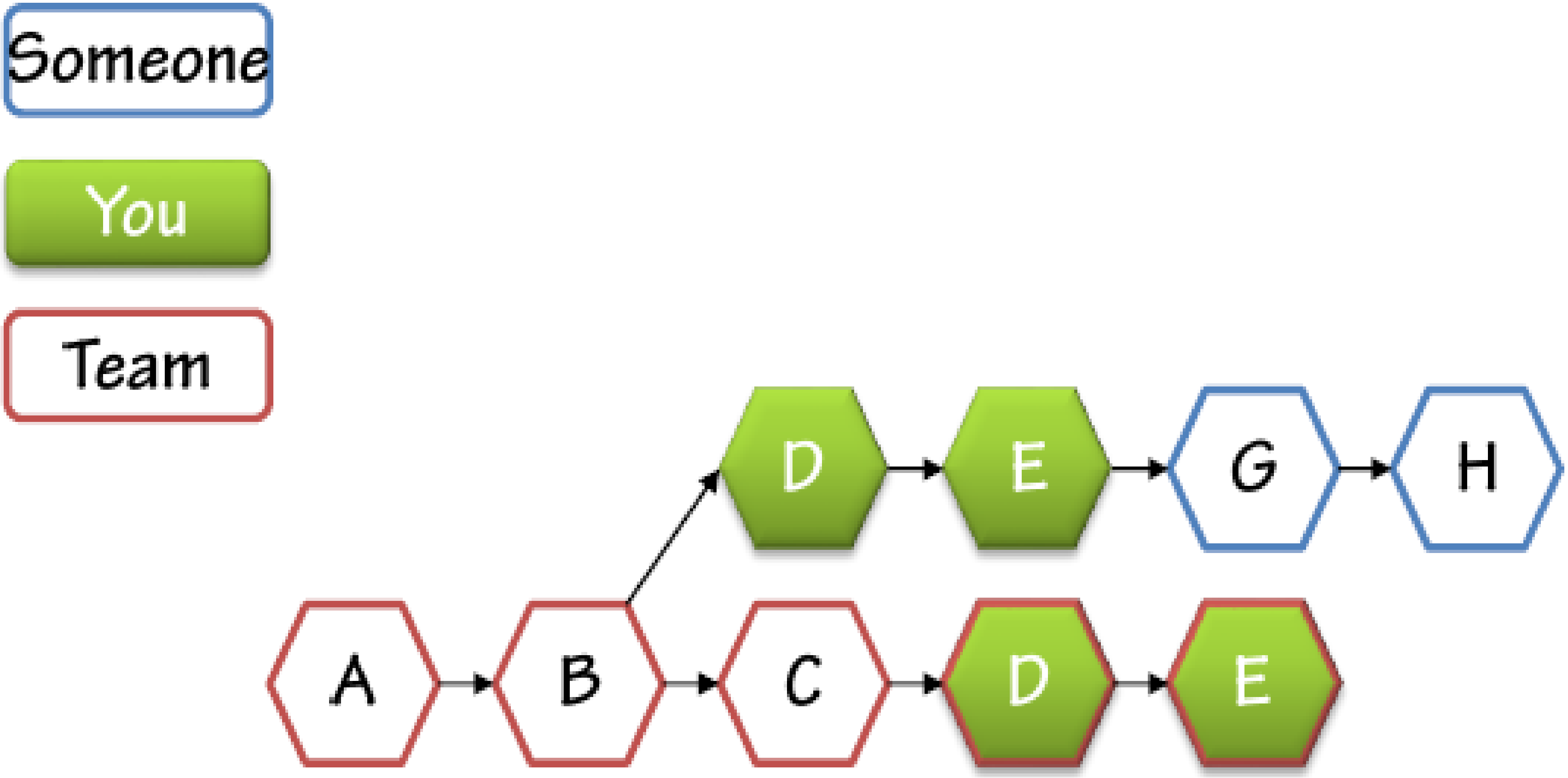
# Dangerous Rebase Example 2 (contd.)



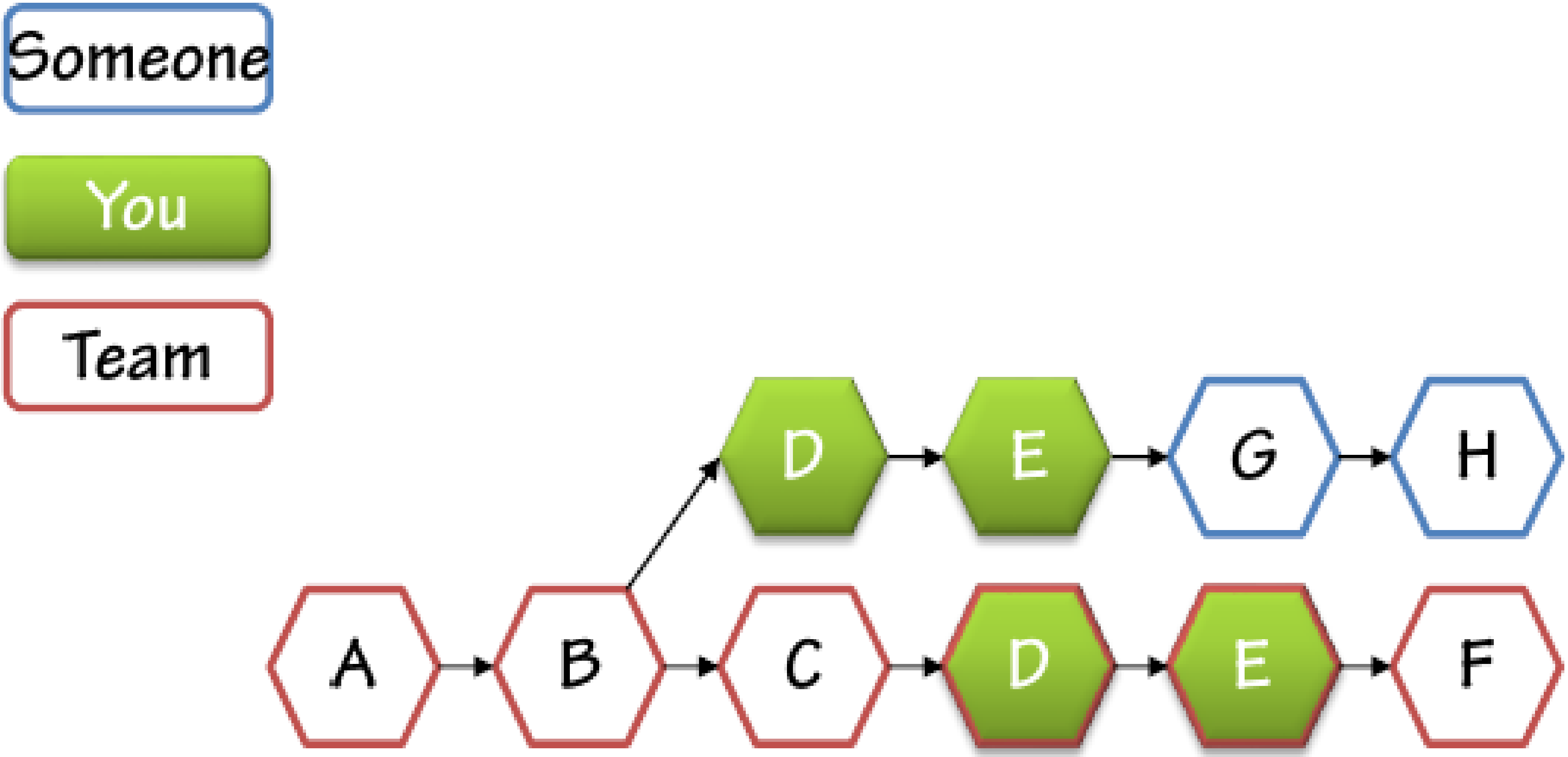
# Dangerous Rebase Example 2 (contd.)



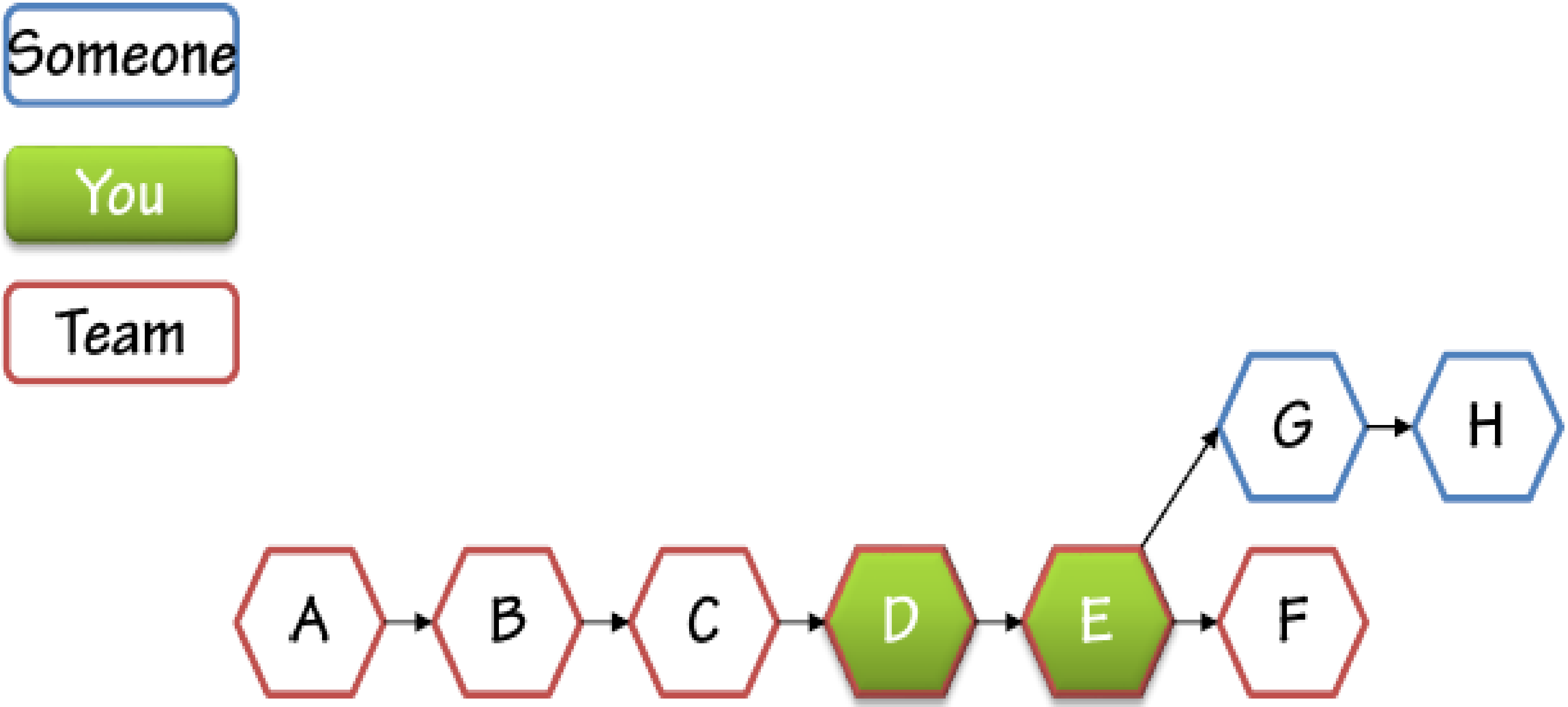
# Dangerous Rebase Example 2 (contd.)



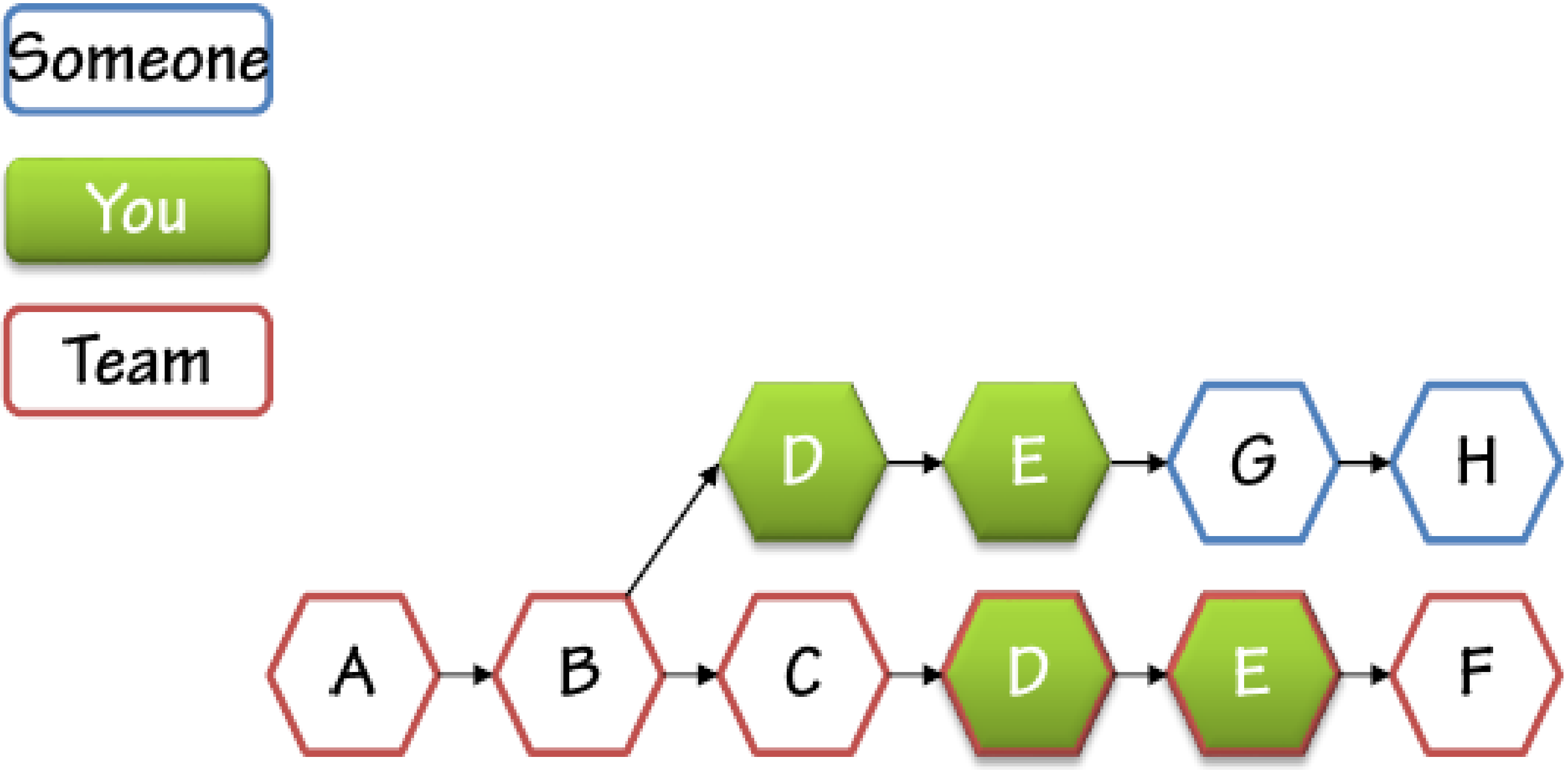
# Dangerous Rebase Example 2 (contd.)



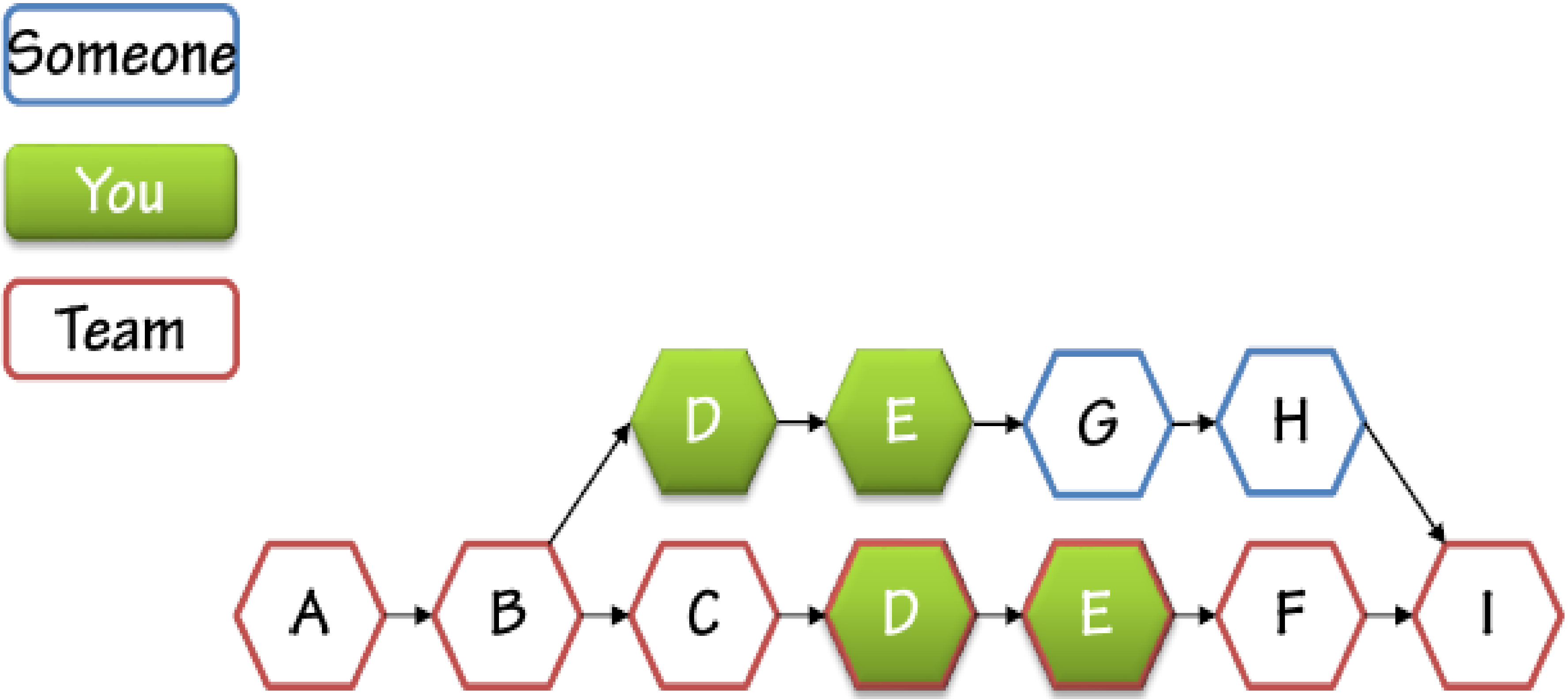
# Dangerous Rebase Example 2 (contd.)



# Dangerous Rebase Example 2 (contd.)



# Dangerous Rebase Example 2 (contd.)



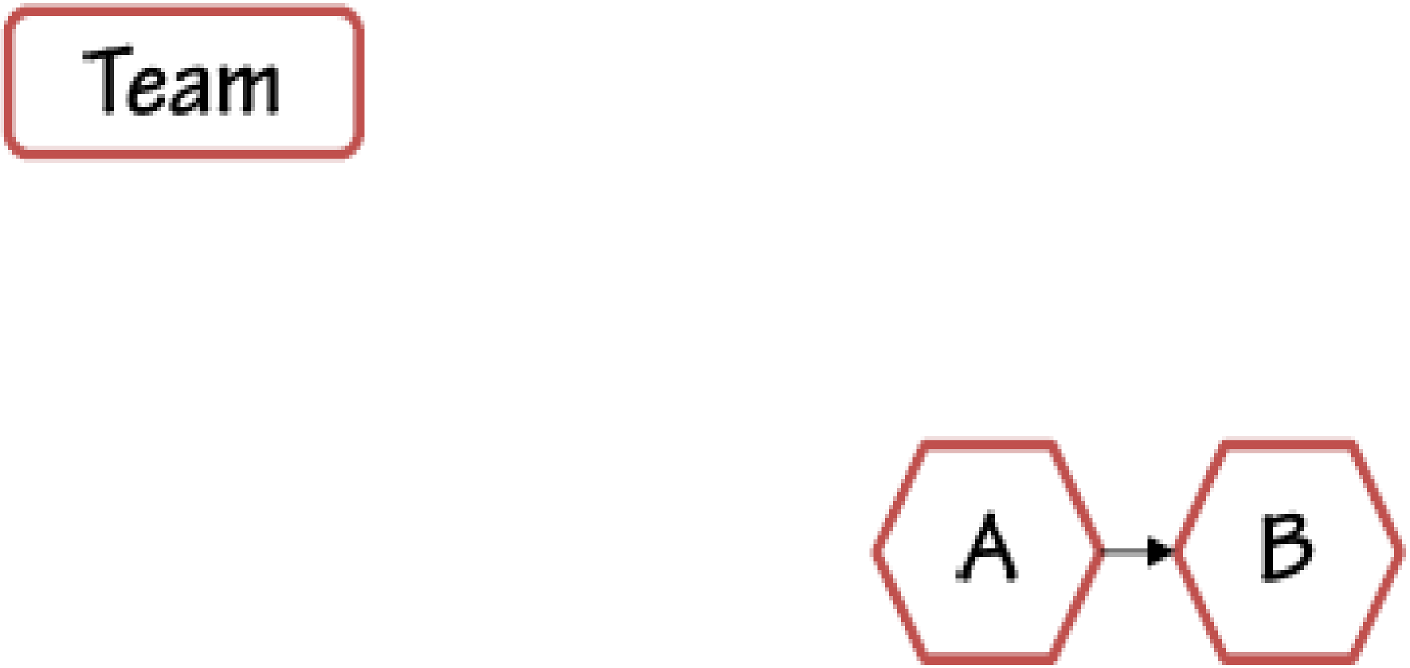
# 3 Common Rules for Rebasing

Never Rebase after Pushing a Branch.

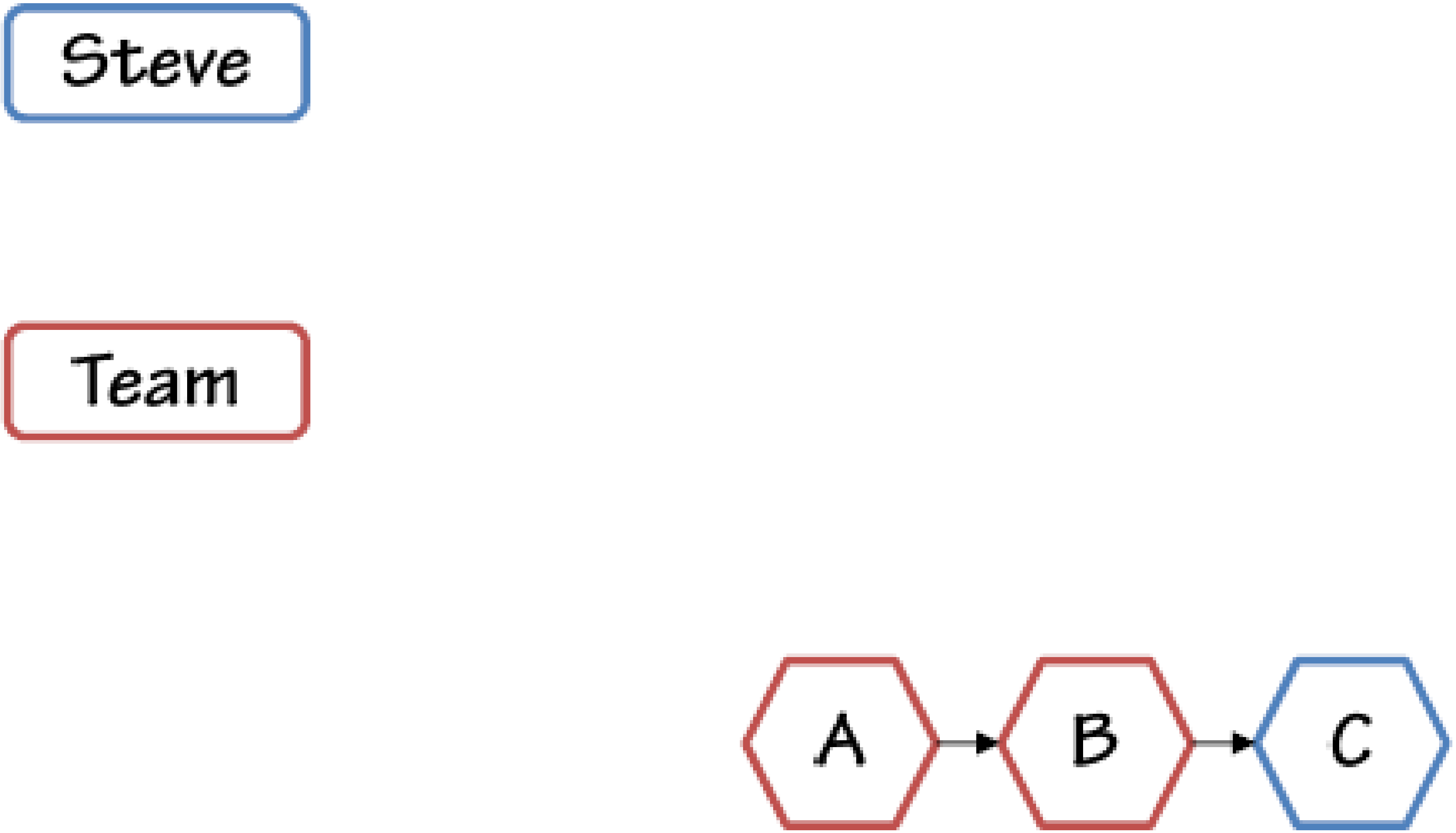
Never Rebase after commits use your Branch.

Never Rebase after Pulling someone else’ Branch.

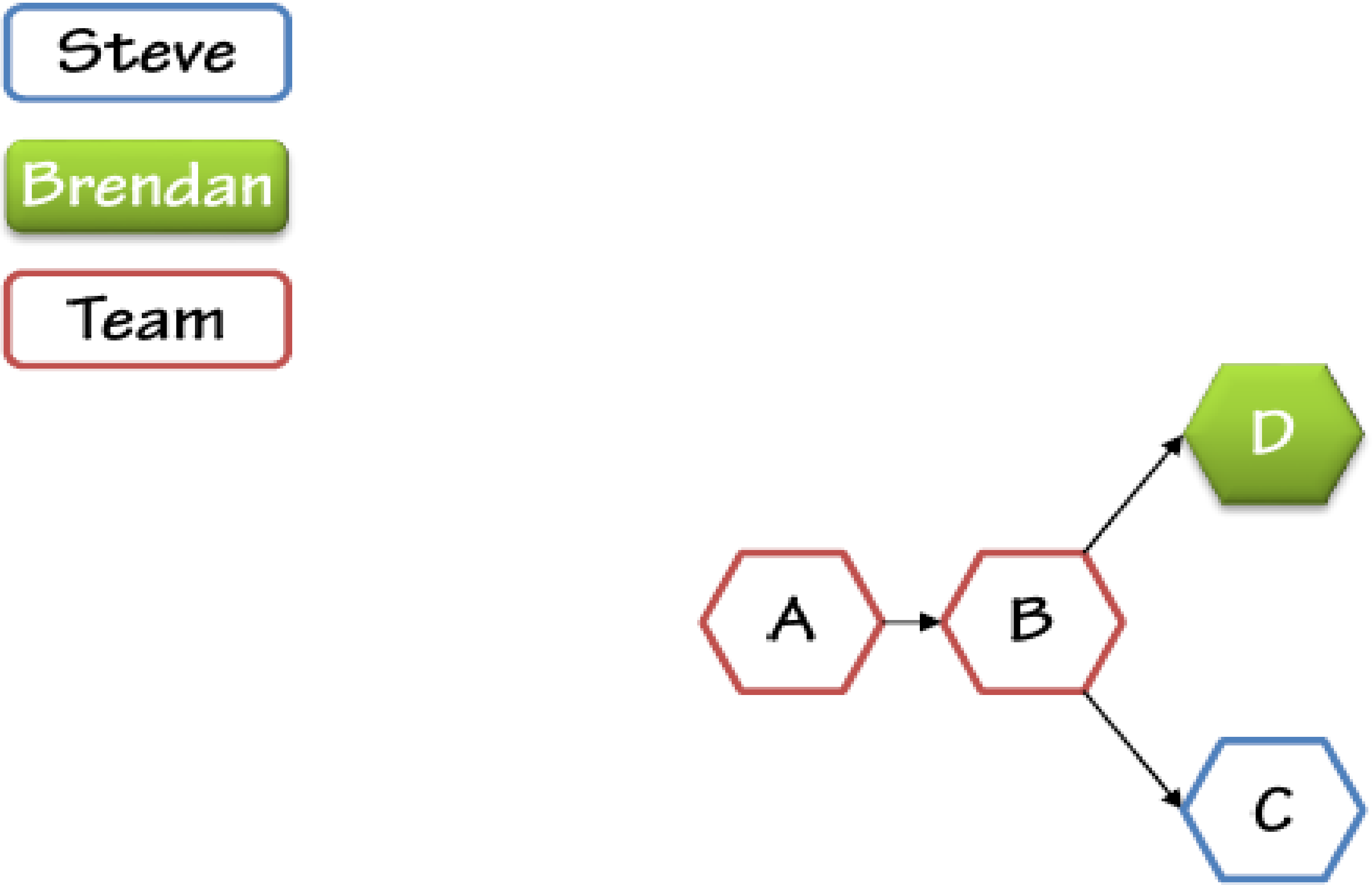
# Good Rebase Example



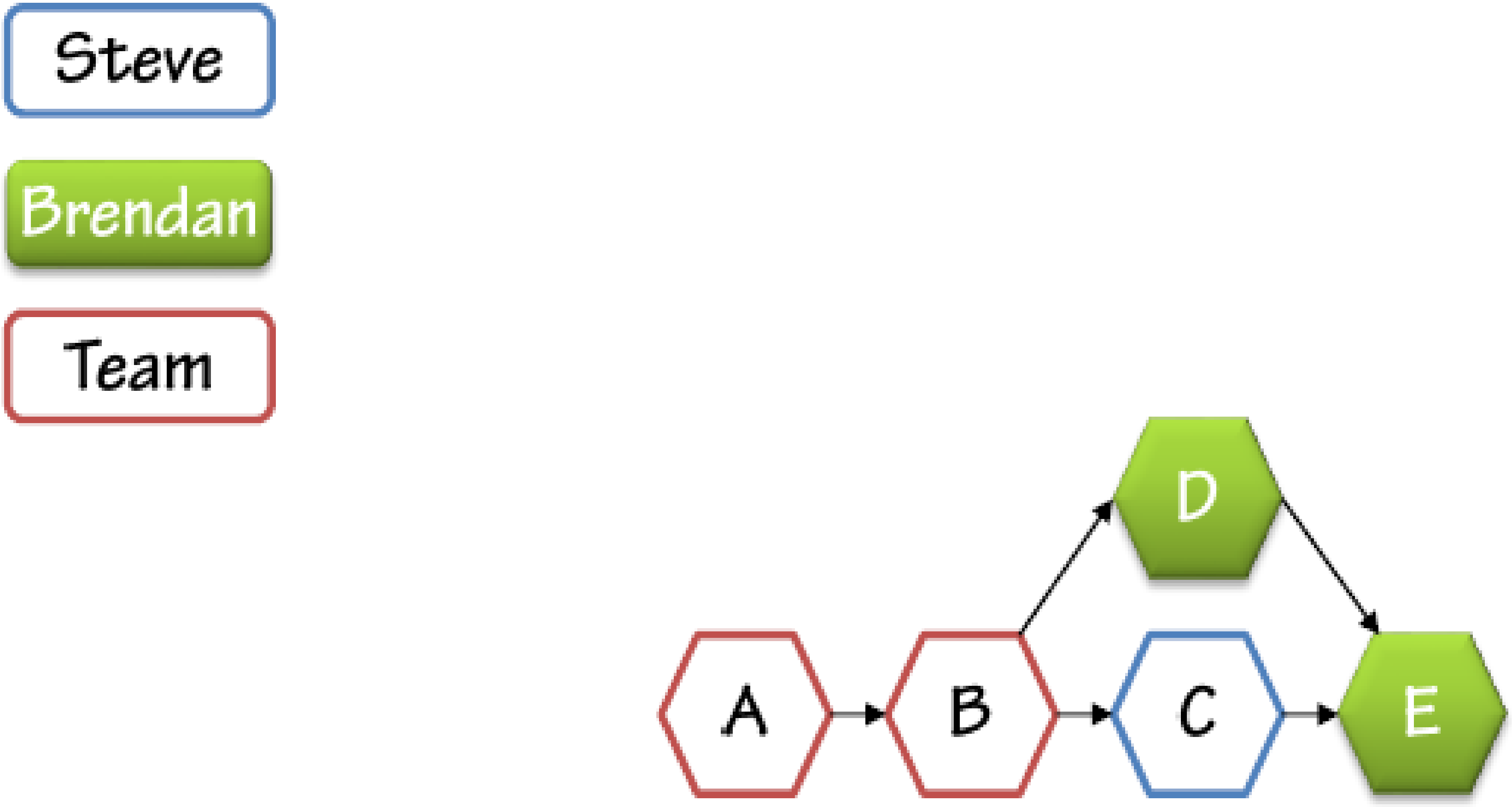
# Good Rebase Example (contd.)



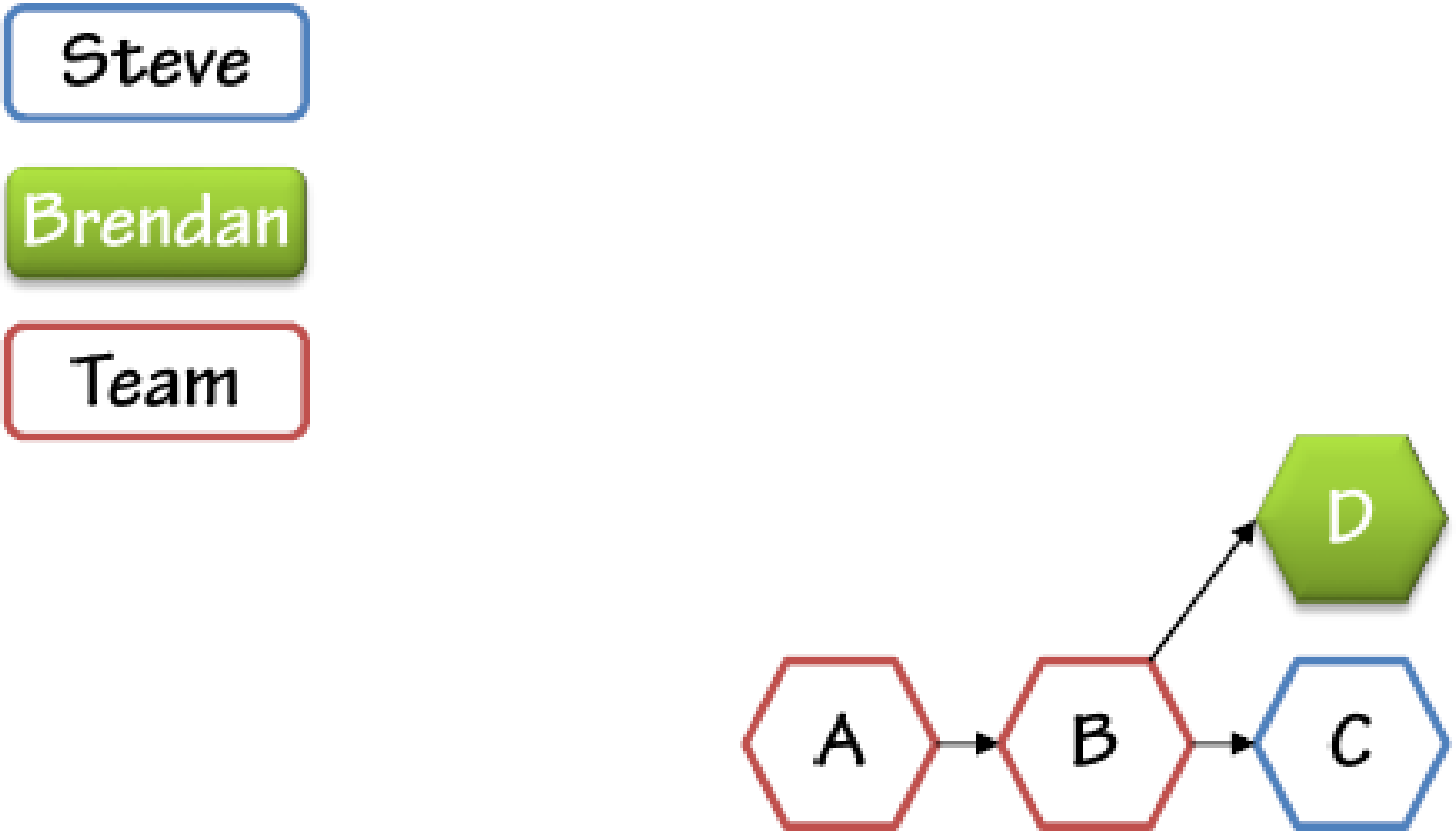
# Good Rebase Example (contd.)



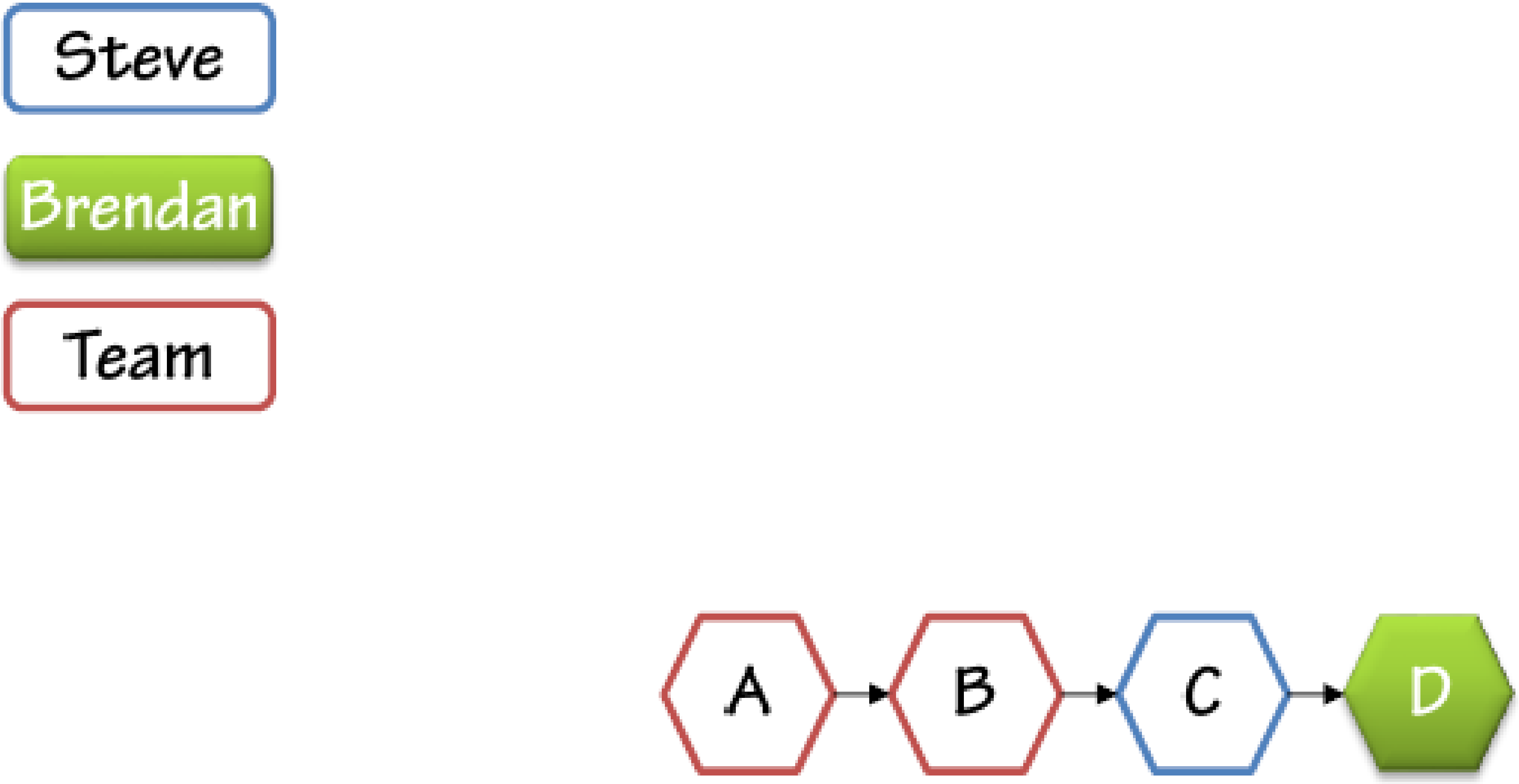
# Good Rebase Example (contd.)



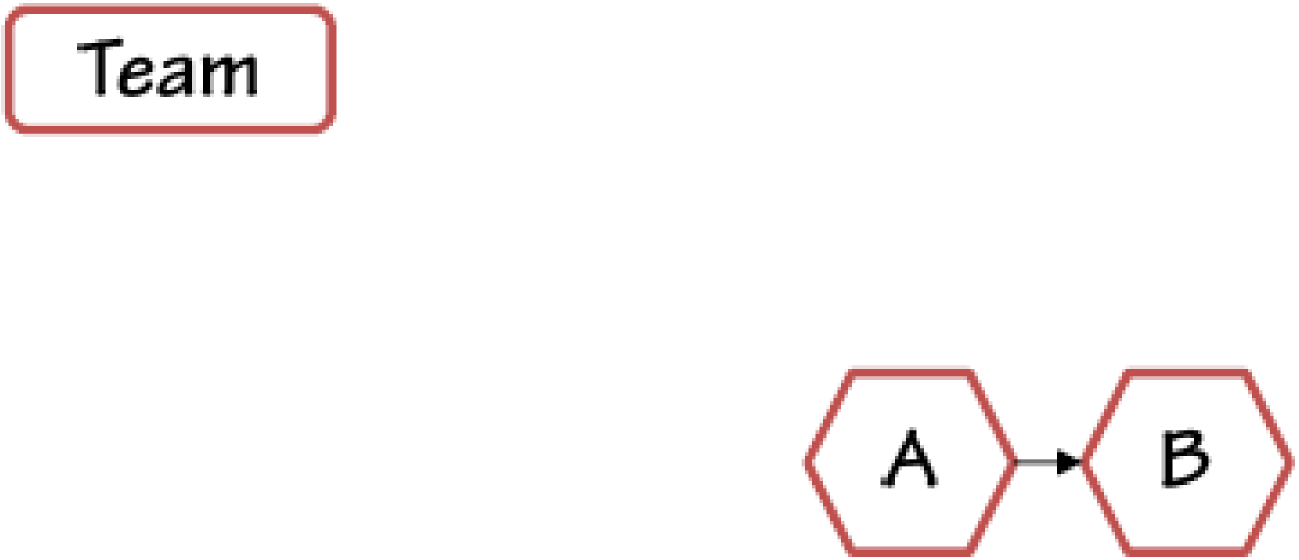
# Good Rebase Example (contd.)



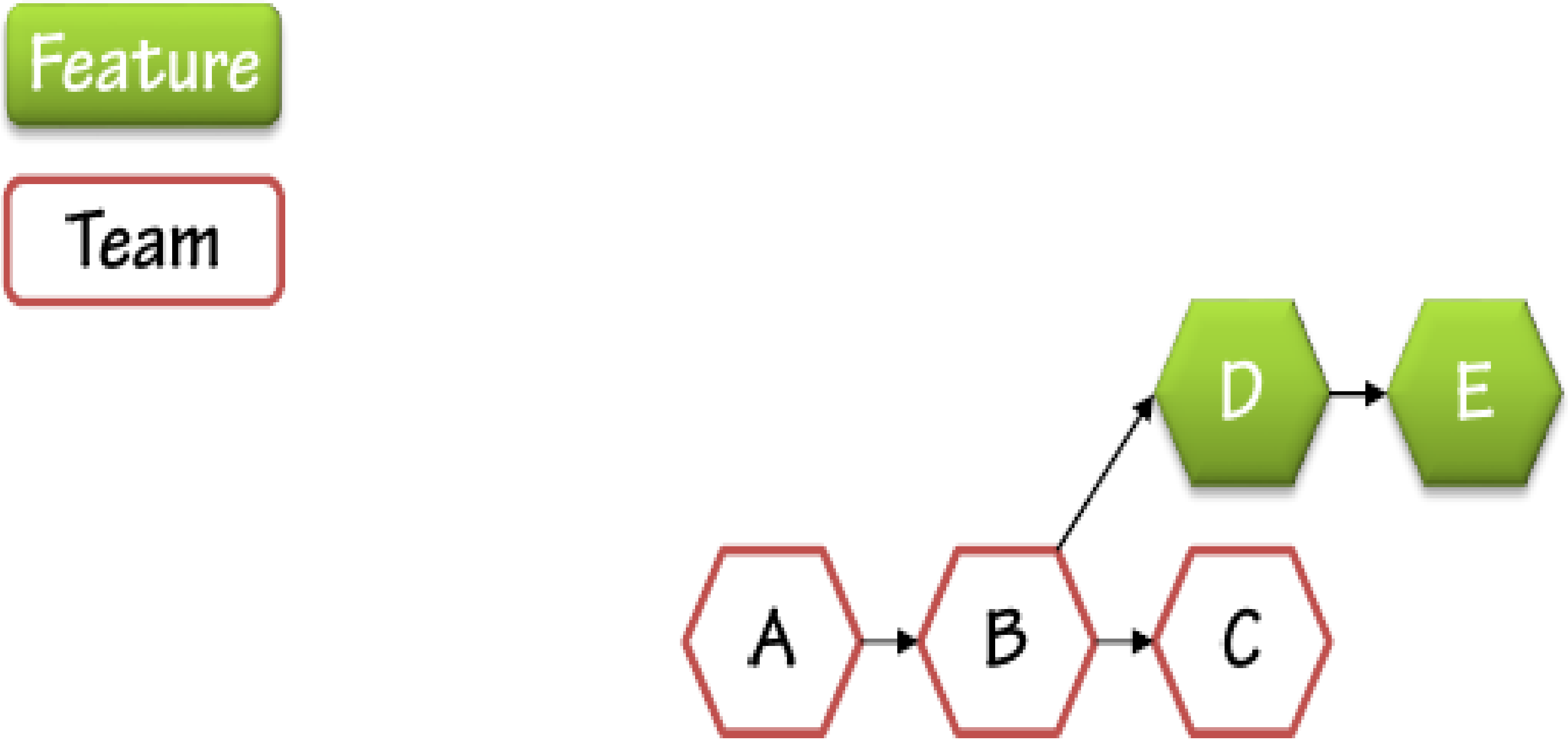
# Good Rebase Example (contd.)



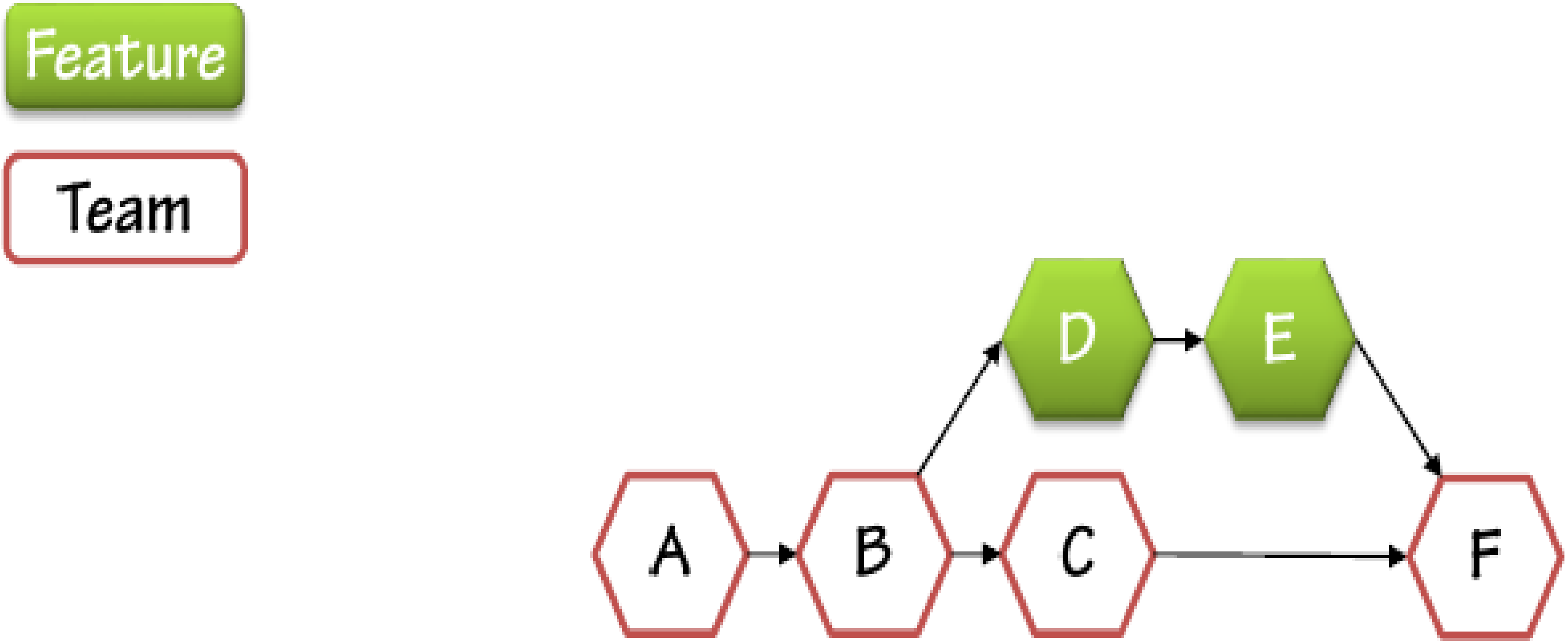
# Good Merge Example



# Good Merge Example (contd.)



# Good Merge Example (contd.)



# Summary

Merging and Rebasing

Stashing Changes

Resolving Conflicts

.gitattributes Files

Capgemini

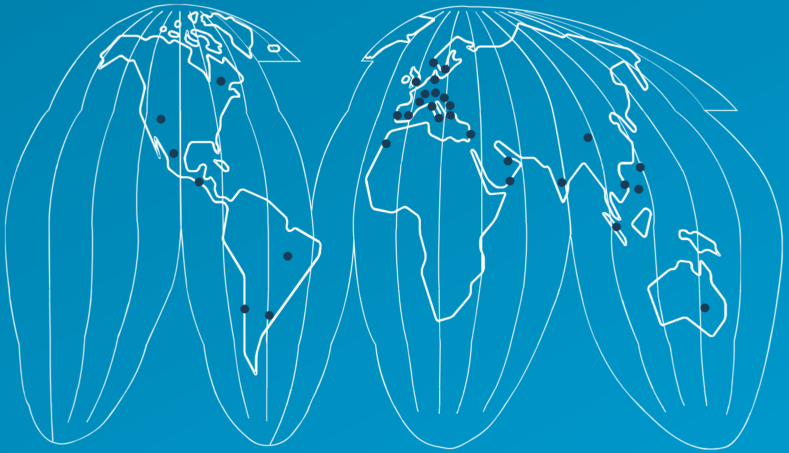
CONSULTING.TECHNOLOGY.OUTSOURCING

People matter, results count.

## About Capgemini

With more than 145,000 people in 40 countries, Capgemini is one of the world's foremost providers of consulting, technology and outsourcing services. The Group reported 2014 global revenues of EUR 10.5 billion.
Together with its clients, Capgemini creates and delivers business and technology solutions that fit their needs and drive the results they want. A deeply multicultural organization, Capgemini has developed its own way of working, the Collaborative Business Experience™, and draws on Rightshore®, its worldwide delivery model.

*Rightshore® is a trademark belonging to Capgemini*



www.capgemini.com