Outline of *Professional Git*

Brent Laster

Written by Brent Laster
Outlined by WLHarvey4

Table of Contents

Introduction 1

How This Book is Unique 1

Target Audience 1

Structure and Content 2

Reader Value 2

UNDERSTANDING GIT CONCEPTS

1 What Is Git 5

2 Key Concepts 6

3 The Git Promotion Model 7

USING GIT

4 Configuration and Setup 11

5 Getting Productive 12

6 Tracking Changes 13

7 Working With Changes Over Time and Using Tags 14

8 Working With Local Branches 15

9 Merging Content 16

10 Supporting Files in Git 17

11 Doing More With Git 18

12 Understanding Remotes—Branches andc Operations 19

13 Understanding Remotes—Workflows for Changes 20

14 Working With Trees and Modules in Git ... 21

15 Extending Git Functionality With Git Hooks .. 22

Index 23

Introduction

Professional Git is intended to help you understand and use Git to get your job done. In the process, it will also make Git part of your professional comfort zone.

This section will explain how this book is unique from other books about Git, the intended audience, the book's overall structure and content, and some of the value it offers you.

How This Book is Unique

Git in terms of Known Concepts

Most books are aimed at providing the technical usage as their major and singular goal. This book provides that, but it will aso provide you with an understanding of Git in terms of *concepts* that you probably already know.

Connected Labs for Actual Examples

Most books do not provide practical ways to integrate the concepts they describe. Learning is most effective when you have *actual examples* to work through so you can internalize the concepts and gain proficiency. This book provides **Connected Labs** that you can work through to absorb what you've read.

Illustrations for Key Ideas and Workflows

This book also provides clear *illustrations* to help you visualize key ideas and workflows. These sections provide additional explanations of how to use some lesser-known features of Git as well as how to go beyond the standard Git features to gain extra valua.

Git Model and Workflow

To be most effective, you need to comprehend the Git *model* and *workflow*. You should also know what to watch out for as you make the transition and why it's important to consider not only the commands and workflow, but also the structure and scope of its underlying repositories.

Target Audience

There is only one assumption in this book: that the reader has experience with at least one source management system. It doesn't matter which one: CVS, Subversion, Mercury—any will do. The reader is assumed to have a basic awareness of what a source management system does as well as fundamental concepts such as checking in and checking out code and branching. Even if you have significant experience with Git or another system, the reader will find something of benefit here.

Git requires a mind shift. In fact, requires a series of mind shifts. However, each shift is easy to understand once you can relate it to something you already know. Understanding each of these shifts will, in turn, allow you to be more productive and to harness the features of this powerful tool—that's what this book is about.

Structure and Content

This book is organized as a series of chapters that present Git from the ground up, teaching what is needed to know and build on to become proficient before adding new concepts.

Part One: Foundational Git

The first three chapters cover the foundational concepts of Git: how it is different from other systems, the ecosystem that's been built around it, its advantages and challenges, and the model that allows you to understand its workflow and manage content effectively with it. This section will provide you with a basic understanding of the ideas, goals and essential terminology of Git.

Part Two: Usage and Features of Git

The remaining chapters of the book cover the usage and features of Git, from performing basic operations to create repositories and commit changes into them, to creating branches, doing merges, and working with content in public repositories.

Reader Value

You will find:

- examples and guidance on the commands and workflows needed to be productive with Git;
- ways to relate concepts to what you already know and understand;
- many illustrations to help you understand concepts visually;
- a feature that allows you to get hands-on experience with Git, via **Connected Labs**, insterspersed throught the chapters. These labs are designed to reinforce the concepts presented in the text of the prceding chapters and to get you actively involved in the learning process.

To get the most out of the book, you should take the time to complete each lab. As well, talk a look at the Advanced Topics sections.

For the later labs, custom Git repositories with example content are provided for the user at `http://hithub.com/professional-git`. In addition, downlable copies of the code for the hooks from the last chapter are availabe in `http://github.com/professional-git/hooks`.

UNDERSTANDING GIT CONCEPTS

1 What Is Git

Chapter Goals

- A brief introduction to Git and its history
- The different ways to find and access Git
- Types of applications that incorporate Git
- The advantages of using Git
- The challenges of using Git

2 Key Concepts

3 The Git Promotion Model

USING GIT

4 Configuration and Setup

5 Getting Productive

6 Tracking Changes

7 Working With Changes Over Time and Using Tags

8 Working With Local Branches

9 Merging Content

10 Supporting Files in Git

11 Doing More With Git

12 Understanding Remotes—Branches andc Operations

13 Understanding Remotes—Workflows for Changes

14 Working With Trees and Modules in Git

15 Extending Git Functionality With Git Hooks

Index

(Index is nonexistent)