Addison Babcock

Submitted August 12, 2016

Comparison of Source Code Control Systems

Contents

[Introduction 2](#_Toc455405705)

[Subversion 2](#_Toc455405706)

[Architecture 2](#_Toc455405707)

[Capabilities 2](#_Toc455405708)

[Typical Uses 2](#_Toc455405709)

[Team Foundation Server 2](#_Toc455405710)

[Architecture 2](#_Toc455405711)

[Capabilities 2](#_Toc455405712)

[Typical Uses 2](#_Toc455405713)

[Git 2](#_Toc455405714)

[Architecture 2](#_Toc455405715)

[Capabilities 2](#_Toc455405716)

[Typical Uses 2](#_Toc455405717)

[Conclusion 2](#_Toc455405718)

[Guidance and Recommendations 2](#_Toc455405719)

# Introduction

Developers are currently spoiled for choice when it comes to selecting which source control system their team will use. But with so much choice how are we supposed to make an informed decision? The goal of my research is to lay out some of the pros and cons of some of the more popular source control systems. I will be exploring Subversion, Team Foundation Server and Git. For each I will describe the overall architecture, the unique capabilities and the typical use cases. I will conclude by attempting to give some guidance on which system would be best for a given team.

# Subversion

## Architecture

## Capabilities

## Typical Uses

# Team Foundation Server

## Architecture

## Capabilities

## Typical Uses

# Git

## Architecture

## Capabilities

## Typical Uses

# Conclusion

## Guidance and Recommendations