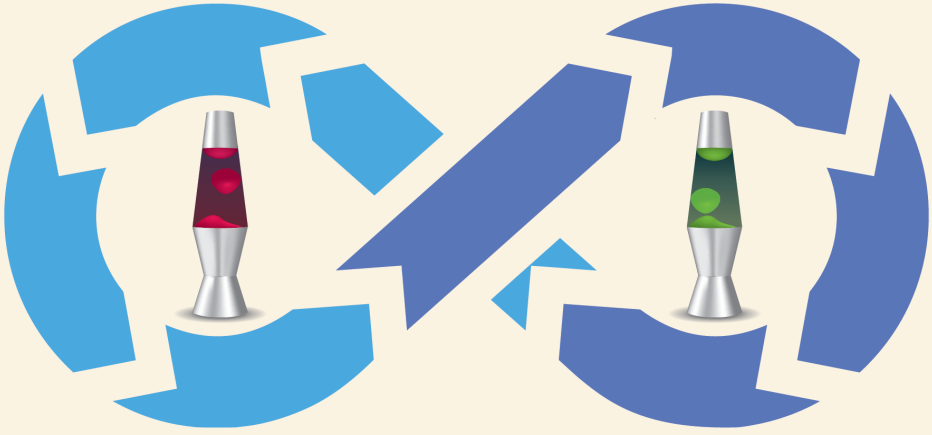
Practical Continuous Testing

make Agile/DevOps real



Zhimin Zhan

Practical Continuous Testing

make Agile/DevOps real

Zhimin Zhan

This book is for sale at http://leanpub.com/practical-continuous-testing

This version was published on 2021-03-01

This is a Leanpub book. Leanpub empowers authors and publishers with the Lean Publishing process. Lean Publishing is the act of publishing an in-progress ebook using lightweight tools and many iterations to get reader feedback, pivot until you have the right book and build traction once you do.

Contents

1. Preface

Continuous Testing, simply speaking, is to run all automated functional tests as regression testing multiple times a day, to help software teams deliver high-quality software to production frequently. Continuous Testing (CT in short) was lesser known as Continuous Integration (CI) or Continuous Delivery (CD). Since last year (2019), we hear people talking more about CT, as it is the key process of DevOps.

"*Too many jargon words!*", some may say. I don't like using jargon words either. Unfortunately, anyone who works in this area needs to understand these terms, because you are going to hear them a lot. In this book, I will use plain English to explain and illustrate techniques, with easy-to-follow hands-on exercises.

The prerequisite of CT is automated functional testing, obviously. If you are new to test automation, please read my other book "Practical Web Test Automation[1]". It puzzled me when I realized some self-proclaimed DevOps/CI specialists had never written a single automated functional test.

My interest in automated functional testing started with HtmlUnit (GUI-less browser testing) in 2005. Shortly after, I discovered Watir, a framework that supports real and visible functional testing in an Internet Explorer browser. Nowadays, I use Selenium WebDriver to test web apps, Appium + WinAppDriver to test Windows Desktop apps and standard Ruby for non-UI functional testing.

Back to my early days on test automation, I was quite happily developing new automated tests and running them individually. Not long after, a big challenge (so obvious but I had never thought of) came: how can I run all test scripts efficiently?

1.1 My Continuous Testing journey

My approaches of running all automated tests (as regression testing) have gone through the following stages:

1. **Run tests from IDE**

[1] https://leanpub.com/practical-web-test-automation

As a programmer, I naturally tried running functional UI tests in IDEs (including NetBeans and my own TestWise). However, it didn't take long for me to realize that it was wrong. Functional (UI) tests, compared to unit tests, takes much longer to execute. For example, a small suite of 20 tests, each test with an average of 30 seconds execution time, will take 10 minutes. This makes it impractical to run functional tests frequently in IDEs.

2. **Run tests from command line**

   To free my IDE (to develop code and tests), I started using build scripts to run tests from the command line. With build scripts, I could also easily add customisation to test executions, such as updating tests from Subversion and excluding certain tests.

   A major drawback of running tests from the command line is "No feedback until it completes". Therefore, I turned to a Continuous Integration Server.

3. **Run in CI Server**

   In 2006, there was only one CI server available: CruiseControl, developed by ThoughtWorks. I set up a CruiseControl server to run our Watir test suite. Initially, it worked well. The team could trigger a build easily, view changelog, and build results, all on CruiseControl's web interface. We could act quickly based on feedback.

   However, with a growing number of test cases, it was getting harder and harder to pass all tests (a green build). The nature of UI tests, comparing to unit tests, is fragile. A single test step failure, maybe due to an issue on the server or the build machine, failed the whole build. At the same time, the project had become dependent on passing all tests, as the gatekeeper, to release to the production server. Moreover, when a build was failed, developers were not allowed to check in new features, which would complicate the fixing process.

   The team embraced automated regression testing, as the benefits were obvious. However, we could not cope with the growing test suites. As a result, the development halted.

4. **Customize CruiseControl with parallel execution, dynamic ordering, ...**

   At that time, I could not find an existing solution to reliably run a set of automated UI tests (in Watir) daily. So I decided to extend CruiseControl (thankfully, it was open-source) with features that might improve execution stability and shorten execution time.

   The two most important features I had in mind were :

- distributing automated test scripts to multiple build machines to run them in parallel, which greatly reduced the execution time,
- auto-retry of a failed test script on another build machine, to reduce the fragility of overall test execution.

I came up with a design and customized CruiseControl to support parallel testing and auto-retry. The code was by no means of good quality, but it worked. The project was a great success (200,000+ test case executions over 14 months). The team was confident to push the latest green build to production. Looking back, we implemented DevOps more or less in 2009.

5. **Create my own CT server: BuildWise**

   CruiseControl was abandoned soon after ThoughtWorks started to work on their commercial CI product, which I did not like. By then, there were a number of CI server products on the market such as Hudson (later renamed to Jenkins) and Bamboo. However, the test executions in those CI servers were suitable for executing unit tests only, and lacked the features for long-running and brittle functional tests. Up to today, I haven't yet seen a single successful Continuous Testing implementation with those CI servers, either non-existing or fake. (My definition of 'Level 1 Success': run 75+ automated UI tests reliably daily).

   I decided to create my own Continuous Testing Server with built-in support for the features I added to CruiseControl, and more. Long story short, I started using BuildWise for my own software development in 2012. As of 2020-02-26, the total number of user story level test cases (in Selenium WebDriver) for ClinicWise (one of our web apps) is 608, with over 600,000 test executions over the last 7 years. This enabled us to respond to customers' requests promptly, 95% of customers' feature requests or reported defects were implemented overnight, and available on the production server the next day. By the way, at AgileWay, we never used a defect tracking system (*Note: I am not totally against DFS, just never had the needs in our case, as we are efficient with replicating issues into automated tests and solid regression testing with a good CT process*), and probably never will.

   Some of our test automation consulting clients have used BuildWise to replace their failed CI servers. In 2018, BuildWise won the 2nd prize of the prestigious Ruby International Award, judged by Matz, the software legend and the creator of Ruby.

1.2 What's unique about this book?

When I studied the "Operating Systems" course at university in 1996, I was deeply impressed by the textbook's author, Prof. Andrew S. Tanenbaum, who actually implemented an operating system: Minix, for teaching purposes. (Prof. Tanenbaum's book and MINIX were Linus Torvalds' inspiration for the Linux kernel [wikipedia[2]]). I gained a lot of insights into CT by designing BuildWise, as well as implementing CT with BuildWise for myself and the clients. I believe that if I do a good job of explaining, readers can benefit from my experiences.

Over the last decade, I have mentored a number of programmers/testers on different projects, and I learned a lot from their perspectives. I built these understandings into BuildWise server as well. For instance, to help new-to-CT professionals to gain confidence, setting up a BuildWise server (from scratch) to run a suite of Selenium tests can be done under 30 minutes.

It is important to note that readers shall focus on the techniques, rather than the actual uses of BuildWise. These techniques are generally applicable, may be implemented in other CT servers, just like what I did to CruiseControl 14 years ago. Treat BuildWise server as a reference implementation, to help you understand the whys and hows, but not limited to it. BuildWise server is free and open-source, you may enhance it or add new features as well.

The example test scripts used in this book are in Selenium WebDriver, the dominant test automation framework for web applications, with RSpec (Ruby binding). Most CT techniques are independent of test automation and syntax frameworks. Three other test syntax frameworks Mocha (JS), PyTest (Python) and Cucumber (Ruby) are covered in Chapter 16.

1.3 Who should read this book?

IT professionals who are involved in software development, from testers, programmers, software architects, agile coaches, managers and chief executives, who want to improve the quality of the software and their work life, can benefit from reading this book. It may sound like a bold statement, but it is the feedback I received from some projects whose team members were willing to make a change and embraced the techniques and practices presented in this book. Those projects delivered high-quality software releases frequently, stress-free. You can achieve this too.

[2] https://en.wikipedia.org/wiki/Andrew_S._Tanenbaum

People with a basic understanding of the software development cycle will find the texts are easy to follow. Prior experience with automated testing and continuous integration is not necessary. Basic scripting knowledge will help, but again, not necessary.

1.4 How to read this book?

I strongly recommend readers to read through chapters in order. More importantly, do the exercises and use new-learned techniques immediately at work. For example, after completing the exercise of setting up BuildWise server to run Selenium tests, try setting up one at work and run a couple of simple real tests for your job. I created video screencasts for most exercises, available on the book site. If you got stuck, watch those videos to see how it is done step by step.

1.5 Send me feedback

I would like to hear from you. Comments, suggestions, errors in the book and test/build scripts are all welcome. You can submit your feedback via the book website.

Zhimin Zhan
Brisbane, Australia

2. Introduction

If you are already familiar with the concept of CI/CD and cannot wait to set up your own Continuous Testing (CT) server to run automated functional tests (in a Chrome browser), please feel free to move onto Chapter 2: *setting up a Continuous Testing server* to run a set of Selenium WebDriver tests. After getting it done (in about 30 minutes), come back and read this introductory chapter. It will probably make more sense to you then.

2.1 What is Continuous Testing?

Continuous Testing, according to Wikipedia[1], is "*the process of executing automated tests as part of the software delivery pipeline to obtain immediate feedback on the business risks associated with a software release candidate." The keywords are "**automated tests**", "**delivery pipeline**", "**immediate feedback**" and "**software release candidate**". (*The reason I quote Wikipedia, a non-academic reference, for definitions: because Continuous Testing is new and there are a lot of confusions over it. So I reach for the common understanding at Wikipedia*)

Like many other formal technology definitions, the above sounds right, but not quite clear. Let me interpret:

- "**executing automated tests**", "**business risks**"

  The automated tests are at the user story level: testing business features. For example, for a web app, the CT process will run a set of automated test scripts to drive the app to verify business functions, in a browser.

  Comparatively, unit testing is conducted by programmers only, that's why unit tests are also called programmer tests.

- "**pipeline**", "**immediate feedback**"

  In this pipeline, Customers/Business Analysts and programmers are waiting for feedback, and more importantly, ready to act on the feedback. Modification follows feedback, i.e, if there are test failures, a new build (with potential fixes) will be triggered to run another execution of automated tests to ensure the quality.

[1] https://en.wikipedia.org/wiki/Continuous_testing

- "**software release candidate**"

  The software is in a ready-to-release state if it has passed all automated functional tests, with little or no human activities on pushing the latest release to production.

Here is my interpretation of Continuous Testing: "*Run automated end-to-end (UI) as regression testing, frequently on new builds. If all tests pass, the software is ready for a production release. If there are test failures, the team must act quickly on the feedback.*"

Continuous Testing is the Holy Grail of Software Development

I didn't invent this term (as much as I would like to take credit for, I heard it from a conversation years ago, but pitifully, could not remember the source), but I think it is a perfect metaphor for Continuous Testing. Yes, it is a big claim, and many of you probably are doubtful. It is my hope to convince you with this book.

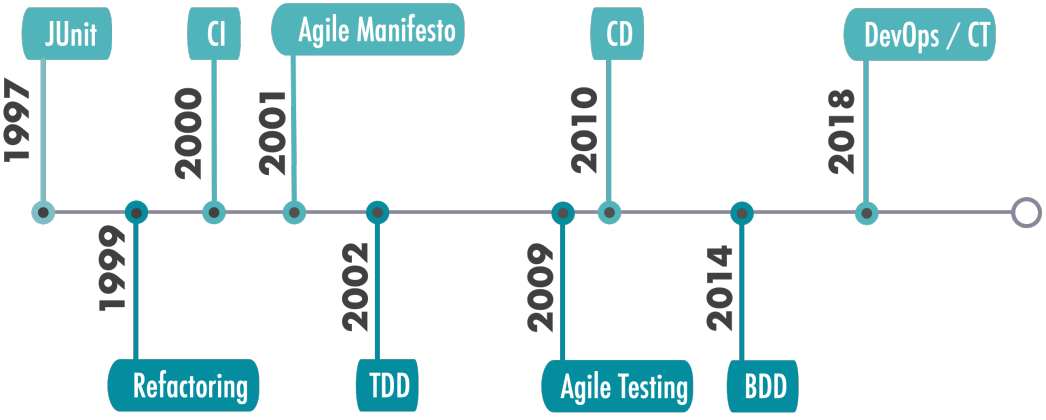
Here I share one customer's comments after I helped implement the continuous testing process, which enabled the project to push updates to the production server on a daily basis. This product owner agrees: "*continuous testing is **super valuable** and **super rare**, despite many heard of it, but very few saw it, just like the Holy Grail*".

2.2 Continuous Testing is the trend

Let's examine some hot IT terminologies (software teams use them every day) over the last two decades.

- 1997 - JUnit by Kent Beck and Erich Gamma
- 1999.07 - "Refactoring" book by Martin Fowler, a process of refining code design backed up comprehensive automated unit testing.
- 1999.10 - First Agile Book by Kent Beck: "Extreme Programming Explained"
- 2000.10 - "Continuous Integration" paper by Martin Fowler
- 2001.02 - "The Agile Manifesto" was written by Kent Beck and other 16 wise software development practitioners
- 2002.11 - "Test Driven Development: By Example" book by Kent Beck

- 2007.07 - “Continuous Integration” book by Paul Duvall, Steve Matyas and Andrew Glover
- 2009.01 - “Agile Testing” book Lisa Crispin and Janet Gregory (Author)
- 2010.08 - “Continuous Delivery” book by Jez Humble and David Farley
- ~2014 - Behaviour Driven Development (BDD)
- ~2018 - DevOps and Continuous Testing (CT)



Quite clearly, the trend of software development is moving towards **frequent releases backed up by automated functional testing**.

1. Unit tests ⇒ Functional tests
2. Adhoc test execution ⇒ Repeatable process to run all automated tests
3. Programmers only ⇒ The whole team
4. Testing for better code quality ⇒ Testing for overall quality and frequent releases

This movement shall not come as a surprise, as *being able to push out software releases frequently with high quality* is every software project pursuing. You might have heard of “Quality at speed”.

It is worth noting that the practices above (such as unit testing, refactoring, functional testing, and CI/CD) are complementary, rather than as replacements. For example, in a software team that embracing a whole-team-involved continuous testing process, programmers are still encouraged to do automated unit testing with TDD.

Can't wait to see running some UI tests in a CT server? you may skip to Chapter 2 (to set up one yourself).

2.3 Continuous Testing vs Continuous Integration

We cannot talk about Continuous Testing without comparing it to "Continuous Integration" (CI in short). Continuous Integration is "*a software development practice where members of a team integrate their work frequently*" [Fowler 00]. In this famous CI article (original version), Martin Flower used "*often talked about but seem to be rarely done*" in the first sentence. Based on my observation over the last two decades, this still remains true: the term CI is favoured by "*talkers*".

> **CI Reality**
>
> I remember at one CITCON (Continuous Integration and Testing Conference) in 2009, a delegate talked about why he attended the conference: "*I want to know how other projects are doing CI? The closest CI experience I ever encountered was that one machine was assigned to do it, then ticked the box. No one touched the machine again.*" Many agreed with him.
>
> A decade later, most software claimed "doing CI" is no more than building (code) and deployment (package), with little or no execution of automated tests …

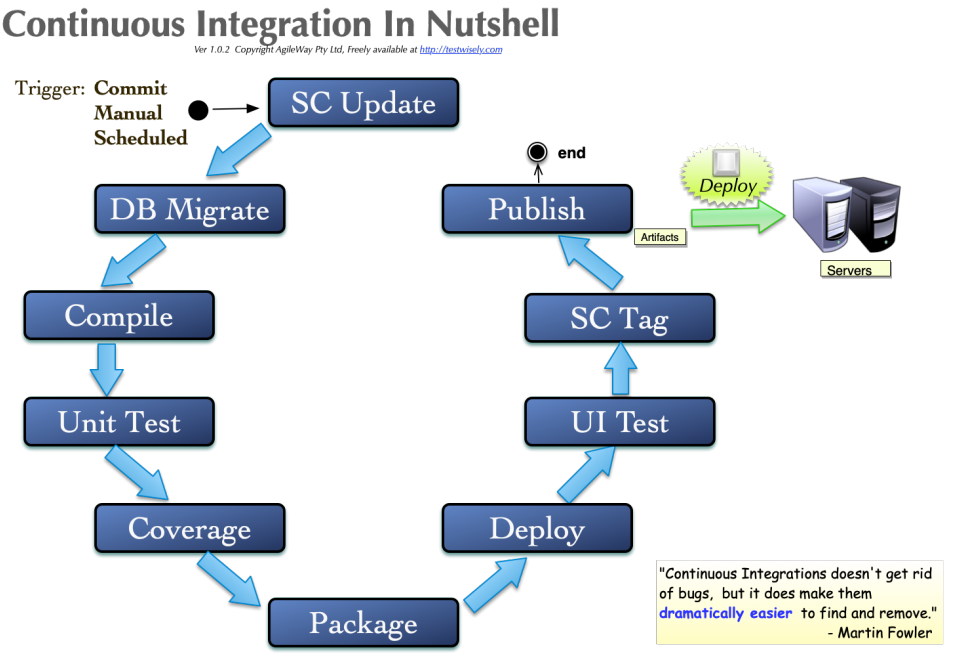
Demystify CI

"Continuous" in CI means doing integration frequently when the software is ready, NOT "doing it all the time", which is simply impossible and unnecessary. "Integration" in CI means packaging the team's work together (along with dependent third party products) into a release candidate and verify the software works as a whole, the keywords here are '**integrate** and **verify**'. What is the point of integrating a web app 100 times a day but keep showing 'Internal Error' on the home page?

"*Continuous Integration doesn't get rid of bugs, but it does make them dramatically easier to find and remove*" – Martin Fowler

As Martin Fowler pointed out, the main purpose of CI is to help the team to find and fix bugs quicker and easier (via executing automated test scripts), in other words. Automated testing is an essential part of a proper CI process. Now the question comes, why do we rarely see the execution of automated tests in CI?

If we examine the typical CI tasks in a real CI process, it becomes quite clear why incompetent CI specialists exclude the execution of automated tests: **it is hard**.



1. **Update source from a version control system** (Difficulty: *easy*)

   This is a built-in CI feature, and you don't usually need to do anything for this step, except installing the source control command line (or called client) tools.

2. **Database Migration** (Difficulty: *medium* or *hard*)

   During the life of software development, inevitably, your database schema needs changes along with the development, and in fact, quite frequently. For example, creating a new table or adding a new table column. This is needs to be done systematically.

3. **Compile** (Difficulty: *easy*)

This only applies to compiled languages such as Java or C#, You don't need this step if using a dynamic language such as Ruby.

4. **Unit Testing** (Difficulty: *hard*)

   Unit test, as the key concept of Test Driven Development, helps programmers to produce not only robust code, more importantly, but also better-designed code. If a piece of code is hard to write the unit test for it, its design is most likely not optimal.

   Programmers who claim 'refactoring code' without a suite of unit tests really are 'cowboys change with hope for good luck'. No unit tests, no code refactoring.

5. **Code Coverage** (Difficulty: *medium*)

   Code coverage is measuring the percentage of code (in terms of methods/lines) that is covered by unit tests. This helps to find out untested and redundant code, to keep source code tidy and lean. It is also a good incentive for Test-Driven Development.

   In reality, achieving 100% code coverage is often not practical, 80% is already a good figure. As a matter of fact, you may find many projects without code coverage data or at a single-digit percentage.

   By including code coverage into the build process, we can help new/junior (not by age) programmers to develop the habit of writing unit tests.

6. **Package** (Difficulty: *medium*)

   A software release package typically contains compiled source code, configurations, web pages (with CSS and JavaScript), file templates, ..., etc. This step is mostly concerned with how to package files using build scripts into a specific format. For example, a war file is a zipped file format used for web applications developed in Java.

7. **Deploy** (Difficulty: *easy* or *medium*)

   Depending on the nature of your application, deployment can be quite complex (if heavyweight frameworks are used) or very simple (such as Ruby on Rails). A typical deployment process consists of the following steps:

   1. Stop the server if it is currently running
   2. Unpack the new release
   3. Update the database schema (database migration).
   4. Update configuration
   5. Start the server

Deployment needs to be simple, reliable, and quick. With the popularity of cloud deployment, new deployment technology emerges such as Chef, Docker/Kubernetes containers. Unfortunately, many DevOps engineers (by the way, I think it is a wrong title for a person who solely does deployment, as deployment is only counted for about 5% of DevOps work from my experiences) don't use them well. More often than not, they make deployment over-complicated, and as a result, fragile and slow.

In 2019, all projects I witnessed using Docker/Kubernetes containers were not good. One was particularly bad, I have never seen a deployment process that was so fragile (and slow) in my over 20 years of IT career (even worse than the dark days using EJB containers). The test servers (a batch of containers) can barely function properly for one day. But I did learn one thing new: "run out of inodes" error (I learned from the Operating System course at Uni) can actually happen.

I am not against new deployment technologies, given they can indeed increase productivity and simplify the work. If the end results are completely opposite to your goal, stop and revert it back until you find the right person who can actually do it properly. There are plenty of costly lessons in the software industry of blindly following new hypes.

WhenWise deployment in 12 seconds

The below is a typical deployment log of WhenWise, one of my web applications written in Ruby on Rails. The deployment tool is **mina**[a], basically an old-school executing a series of batch scripts via SSH.

```
$ mina production deploy
-----> Fetching new git commits
-----> Using git branch 'master'
       Cloning into '.'...
       done.
-----> Using this git commit
       Zhimin Zhan (06405f79):
       > bump ver 0.8.30
-----> Symlinking shared paths
-----> Installing gem dependencies using Bundler
-----> DB migrations unchanged; skipping DB migration
-----> Skipping asset precompilation
-----> Cleaning up old releases (keeping 5)
       /var/www/rails/whenwise/tmp/build-157846690712699
-----> Deploy finished
-----> Building
```

```
-----> Moving build to /var/www/rails/whenwise/releases/561
-----> Build finished
-----> Launching
-----> Updating the /var/www/rails/whenwise/current symlink
       /var/www/rails/whenwise/current
-----> Loading rbenv
-----> Quiet sidekiq (stop accepting new work)
       /var/www/rails/whenwise/current
-----> Stop sidekiq
       Sidekiq shut down gracefully.
-----> Start sidekiq
-----> Update crontab for ap11.agileway.net_au
       [write] crontab file updated
-----> Done. Deployed version 561
       Connection to whenwise.com closed.
       Elapsed time: 12.24 seconds
```

The deployment script (under 160 lines) performs the following tasks:

1. Get new code (`git pull`),
2. Update/install dependent libraries, not necessary in this case.
3. Database migration, no need for this deployment as
4. Precompile JS/CSS assets, no need for this deployment.
5. Package for release
6. Stop queue process (Sidekiq)
7. Start the queue process
8. Verify/Update cron job
9. Reload the app

If there are database changes or new libraries to be installed, the scripts will apply installations as necessary.

For my every CI build, our deployment steps will deploy the updates to 9 test servers: ci1.whenwise, ci2.whenwise, ..., ci9.whenwise. Then automated tests will be run against these test servers. We will cover the set up in later chapters. Here, I just want to emphasise the importance of quick and reliable deployment, which is the prerequisite task of CT.

[a] https://github.com/mina-deploy/mina

8. **Functional Testing** (Difficulty: *very hard*)

   Executing automated functional tests against the test server(s) with a new version of software deployed (by the last step), essentially, Continuous Testing.

   Some might not agree with the difficulty (**very hard**) I rated there.

   I would say we might have different perspectives. Most UI testing, if present, in CI is no more than smoke and mirrors. For me, CT is the core of software development. If all automated functional tests pass in CI, this build will be released to the production. (For nearly all user stories and customer-found defects, I have automated tests for them).

   Let me illustrate it by an example. Let's say that we have 200 user-story-level functional tests written in Selenium WebDriver, on average, one test case has 30 test steps (each step represents a user operation, such as entering text, clicking a link and verifying certain text) and execution time of a single test is 30 seconds. In total, there are `30 x 200 = 6000` test steps and a full regression testing will take `6000` seconds. To get a green build (all tests pass), each of every 6000 test steps needs to pass within nearly 2 hours of test execution. A single failure results in a broken build.

   Now, do you agree this is a very, very hard task?

9. **Tag a build/release** (Difficulty: *easy*)

   Tagging refers to labelling the repository at a certain point of time so that it can be easily retrieved in the future to achieve repeatable builds.

   You don't have to label every build, probably only for the green builds (which have passed all tests). Be aware of the time difference between checking out from source control and actual tagging. A common approach is to introduce 'Code Freeze' or schedule full builds at night time.

10. **Publish** (Difficulty: *easy*)

    Once a build finishes, besides showing build status and other information on CI's web interface, there are numerous ways to publish the result, such as email, Slack, or even switching on a lava lamp.

Apart from the success/failure indicator, build results might also highlight failed test cases and artifacts (files generated out of the build process). Most CI servers keep the build history, which can be generated into pretty charts for reporting purposes.

As you can see, there are three hard tasks (in the order of difficulty): "**Database Migration**", "**Unit Testing**" and "**Functional Testing**". Few will deny the needs of these steps, though most probably have never seen them done properly.

This book is to help you to implement CT, the most challenging and rewarding task in CI: execution automated functional tests.

Tips for Database Migration and Unit Testing Tasks

This is a CT book, so I will be light on these.

- **Database Migration**

  Study the database migration approach in Ruby on Rails[a], you might be able to work out a feasible approach. For one Java project, I embedded Ruby on Rails's database migration scheme directly in the project code, then use JRuby to invoke database migration in CI.

- **Unit Testing**

  There are plenty of books and on-line resources on this topic. However, in practice, despite majority programmers acknowledging the benefits of unit testing, unit testing is mostly done adhocly. The fundamental reason, I think, is that programmers lack knowledge of good unit testing practices. Please read 'A Set of Unit Testing Rules[b] by Michael Feathers.

[a] https://edgeguides.rubyonrails.org/active_record_migrations.html
[b] https://www.artima.com/weblogs/viewpost.jsp?thread=126923

If CI is implemented properly, no need for CD or CT

CI has been so messed up in practice that it is becoming meaningless. That is why a new term comes up "Continuous Delivery" (CD in short, which later quickly lost its meaning as well), somehow people find it fancier to say these two terms together "CI/CD". In every project I visited over the last decade, agile coaches/architects talked a lot about CI/CD and did not do hands-on test automation, their continuous integration processes were all embarrassing failures.

Once I worked at a software company, they had a Bamboo CI server with a number of projects, which seldom ran and virtually no sign of executing automated tests. However, they claimed they were providing CI consultancy to one of the top four banks in Australia.

If CI's main purpose is to build a releasable software package, this has been achieved years ago with build scripts, like an Ant task generating a deployable war file (back to J2EE days). Triggering a build from a web interface and seeing build results (on CI server) is nice, but

don't you think there is not much to brag about? The purpose of CI is to ensure quality releases by running automated tests against release candidates. The testing is the main part.

Some people might say "Continuous Testing" could be the next ruined 'talker term'. Yes, that could well be true, and probably already is. At this moment, we have run out of terms, sadly. I will settle with the term "CT", because of its emphasis on testing.

So what is the relationship between CT and CI? In simple words, CT is a part of CI, the most important and difficult part. If a CI process is implemented well, there is no need for "CD" or "CT".

"Continuous Delivery is really about testing"

The origin of 'Continuous Delivery' is the book with the same title by Jez Humble and David Farley, published in 2009. Some will debate the differences among CI, CD and CT. Frankly, I don't think it is necessary. The core of all three is the same: executing automated functional tests as regression testing.

Don't just take my word for it, let's hear the views from the authors of the Continuous Delivery book and a highly claimed authority in this field. In an interview[2] in October 2019, Lisa Crispin, the co-author of Agile Testing book, said this: "They (Jez Humble and David Farley) asked me to be a technical reviewer for their manuscript (Continuous Delivery book) … I read it. It's a book about testing, you know, the whole book is really about testing. That's the heart of continuous delivery. Jez Humble is very supportive of my saying that."

2.4 Separate CT from existing CI/CD

If CI is real, it would run automated functional tests as a part of build tasks, However, this is a big 'if'. For software teams who are about to embark on the journey of CT, I recommend separating from your existing CI process, as your current CI/DevOps engineers most likely had no knowledge or experience of running automated functional tests in CI.

By separating the CT process out, you may avoid

- be forced to use a particular CI product.

  The techniques and hardware requirements (as you will see in later chapters) are quite different from executing unit tests. By following the existing CI process (with a

[2](https://www.infoq.com/articles/current-future-testing/)

particular server product), without a doubt, the perspective has been set in unit testing, which is wrong.

I have never seen a single successfully implemented CT using Jenkins, Bamboo, TeamCity, GitLab, …, etc, these so-called popular CI servers. This does not mean CT is impossible with them. Back in 2006, I implemented most of the techniques in this book as a Java plugin to CruiseControl (the first CI server), with a good outcome. So technically, it is possible with all CI servers. However, this requires quite a lot of work.

- be forced to use a certain test script syntax and framework.

  Different from unit testing, the functional test script syntax can be in a different programming language. For example, I have implemented CT with functional test scripts in Ruby for products developed in Java, C#, JavaScript, and Ruby.

- be asked to follow certain inappropriate practices

  This may include version control policies (a bad example: branching on each user story), directory structures, naming conventions, …, etc.

2.5 Continuous Testing vs DevOps

"By 2020, DevOps initiatives will cause 50% of enterprises to implement continuous testing using frameworks and open-source tools." – Predicts 2017: Gartner Report[3]

If I ask you the hottest term in the software development industry in the past 2 years, many will say "DevOps". Frankly, I think the term DevOps is quite vague (as opposing to '10-minute build' and 'pair programming'), therefore, is open to interpretation.

I have heard a few DevOps talks, however, they left no marks on my brain. For one project I witnessed in 2019, the executives got sold by the 'impressive talk' by a 'DevOps talking-expert', engaged the consulting company to implement DevOps. These consultants were busy talking, presenting, introducing new software …, for a few months. The result was a total disaster, in the end, the teams were told to revert back to the old way. The reason is simple: the foundation of DevOps is Continuous Testing. It is easy to understand, just imagine a pipeline producing poor quality products in a factory. As we know, the quality problems magnify in an order of magnitude.

[3] https://www.gartner.com/doc/3525622/predicts--application-development

Let's switch the focus to DevOps' objective (instead of its definition). I resolve to Wikipedia[4]: "*It (DevOps) aims at establishing a culture and environment where building, testing, and releasing software can happen rapidly, frequently, and more reliably*". This sounds quite like the objective of Continuous Integration, doesn't it? Except with an emphasized focus on quality releases and feedback to the team. For example, if you set up a Jenkins or TeamCity project to build software and run a few unit tests (i.e. programmer tests), you might call it CI, but it is incomplete in terms of DevOps, as it does not include regression testing (at the functional level) for releases.

I don't mind DevOps at all. As a matter of fact, I have been developing software this way (releasing high-quality software frequently with comprehensive automated testing) since 2007, and have shared my experience in numerous presentations. Only at that time, the term "DevOps" and "CT" did not exist yet.

2.6 Reality Check

Despite all the hype of CT (and previously CI/CD) and DevOps, the reality is 99%+ software teams at level 0 (*a term I borrowed from the movie: Kungfu Panda*) on CT. If you are not convinced, try to answer the two questions below:

- When was the last time your project pushed a release to the production?
- How often do you do that?

In the context of DevOps, the correct answers for the above are "Yesterday" and "At least once a day".

I have my reasons for using 99%+. Alan Page, the first author of "How We Test Software at Microsoft" book, said this at Test Talk PodCast #44, March 2015[5] "*95% of the time, 95% of test engineers will write bad GUI automation just because it's a very difficult thing to do correctly*". Alan's view remains unchanged since 2008, when he wrote on his blog[6] "*For 95%*

[4] https://en.wikipedia.org/wiki/DevOps
[5] https://testguild.com/podcast/automation/44-alan-page-testing-software-at-microsoft-lessons-learned/
[6] https://angryweasel.com/blog/gui-schmooey/

of all software applications, automating the GUI is a waste of time. For the record, I typed 99% above first, then chickened out. I may change my mind again."

Alan used '99%' there, I added '+' because most software companies won't match Microsoft on

- quality of software test engineers
- resources (both technically and financially)

Furthermore, Continuous Testing adds more challenges, by running the whole test suite as regression testing frequently.

2.7 Why will this book help?

I know some might think: "*if CT is almost mission impossible, how could I believe that you can do it?*". I understand no matter what I say, it probably won't be enough to convince you. Instead, I write this book (and produced videos on the book site) to guide you in implementing CT step by step.

In the next chapter, I will show you how to set up a CT server and run a set of Selenium WebDriver tests, under 30 minutes.

3. Set up a CT server to run Selenium tests in minutes

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

3.1 Objectives

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

3.2 Prerequisite

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

3.3 Install CT Server

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

3.4 Create a Build Project

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

3.5 Trigger test execution manually

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

3.6 Feedback while test execution in progress

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

3.7 Build finished

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

3.8 Common Errors

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

4. How Continuous Testing Works?

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

4.1 Terminology

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

4.2 CT Process in detail

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

4.3 Install BuildWise Server for production use

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

4.4 Understand build working directories

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

4.5 Exercise: Fix a failed build

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Prerequisite: Make sure parent repository is writable

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Identify the failed tests

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Fix failed tests locally

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Commit the changes and push up to the parent repository

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Trigger another build and verify

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

4.6 Review

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

5. Why is CT important?

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

5.1 Continuous Testing is the key to Agile

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

5.2 Detect regression errors quickly

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

5.3 Reduce/eliminate the needs for Defect Tracking

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

5.4 Try new ideas / upgrades

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

5.5 CT is vital for the maintenance

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

5.6 Training

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

5.7 Benefits for All Team Members

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Executives get financial rewards

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Managers can sleep well

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Programmers get more efficient and satisfied

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Testers have more fun

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Business Analysts

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Boost Customer's confidence

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

5.8 Review

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

6. Set up your own build project

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

6.1 Prepare your test scripts

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

6.2 Create a new build project

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

6.3 Trigger a build manually

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

6.4 View a build in progress

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

6.5 Cancel a build

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

6.6 View Change Log

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

6.7 Build history

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

6.8 View test failures and screenshot

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

6.9 View test script content

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

6.10 View test execution history

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

6.11 Build report

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

7. How to succeed in CT?

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

7.1 Definition of Success: AgileWay CT Grading

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

7.2 Functional test automation and continuous execution are interdependent

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

7.3 Success Factors (test automation)

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Solid Test Automation Framework

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Test script in a scripting language

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Reliable individual test execution

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Maintain test script with high efficiency

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Freedom

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Affordable

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

7.4 Success Factors (infrastructure)

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Reliable server infrastructure

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Rapid, reliable and repeatable deployment

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Dedicated test lab

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

7.5 Success Factors (continuous execution)

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Low false alarm rate

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Quick Feedback

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Scalable with parallel test execution

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Auto retry

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Same testing tool for the whole team

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

7.6 Success Factors (human)

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Realize on-going efforts (executives)

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Open mindset and judge objectively (tech leads)

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Access to a good test automation & CT mentor

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Deploy on green builds

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Developers stop and fix regression defects

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Whole Team involved

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

CT server is the heart of the team

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Reward CT engineers well

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

8. The Magic - Build Script

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

8.1 What is Build Script?

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

8.2 Use Build Script in BuildWise

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

8.3 Configure Build Tasks in BuildWise

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

8.4 Execute build tasks from the command line

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

8.5 Functional Testing (Sequential) Task

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

8.6 Functional Testing (Parallel) Task

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

9. Manage a project

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

9.1 Build Step Management

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Add a new build step

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Change the order of build steps

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Disable/Delete a build step

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

9.2 Common Pre-Functional-Testing Tasks

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Reset data

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Test Scripts Stats

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Common CI tasks

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Warm-up Test Servers

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

9.3 Clone an existing project

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

9.4 Build Artifacts

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

9.5 Enable the project's API key

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

9.6 Notifications

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

9.7 Project Statistics

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

9.8 Hide a project

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

10. Manage builds

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

10.1 Trigger a build via API

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Curl from the command line

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Ruby script

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

GUI tools such as Postman

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

10.2 Schedule a build

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Scheduled build on macOS/Linux

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Scheduled build on Windows

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

10.3 Add Build Summary

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

10.4 Invalidate a build

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

10.5 Delete a build

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

10.6 View and download build artifacts

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

10.7 View build logs

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Build Log

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Step Log

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

11. Sequential E2E Test Execution

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

11.1 Sequential Mode

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

11.2 Pre-requisite

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

11.3 Decide what tests to be included

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Why not just go for Parallel mode completely?

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

11.4 Select test script files in the build script

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Specific test scripts

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

All test scripts in a folder

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Exclude certain tests

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

11.5 Define a build target

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Verify build target from the command line

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

11.6 Task Configuration in BuildWise

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Prepare task

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Manage build tasks

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Cleanup task

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

11.7 Test Execution Order

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

11.8 Feedback during the build

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

11.9 Analyse test reports

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

11.10 Anti-Pattern: Split tests into multiple sub-builds

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

12. Sequential Test Execution Best Practices

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

12.1 Limit the test count

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

12.2 Show test results immediately

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

12.3 Capture the error stack trace

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

12.4 Capture the error screenshot

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

12.5 View test script content

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

12.6 View test output

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

12.7 Customize test executions with Environment Variables

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Server

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Browser

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Headless mode

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

12.8 Clone build project for different purposes

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

12.9 Intelligent Ordering

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

13. Parallel Test Execution

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

13.1 Build Agents

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Build Agent Machines

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Decide the number of agents

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Install BuildWise Agent

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Configure - General Settings

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Configure - Application

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Verify software on an Agent machine

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Verify Test Execution in an Agent

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Configure - Pre Execution (optional)

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

13.2 Prerequisite for Test Scripts

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

13.3 Multi-Agents against the Single Server

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Unable to reset the database

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

13.4 Multi-Agents against the Multi-Servers

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Server warm-up scripts

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

13.5 Set up a parallel build project

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

13.6 Monitor the build progress

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

13.7 Assess the benefits of parallelism

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

13.8 Common Issues

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

14. Parallel Test Execution Best Practices

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

14.1 Optimal agent count

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

14.2 Intelligent test execution ordering

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

14.3 Auto retry failed tests once more

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

14.4 Manually rerun a failed test

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

14.5 Cross-Platform Testing

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

14.6 Headless or not?

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

14.7 Distribution rules

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

14.8 Delay completion

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

15. Parallel Testing Lab

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

15.1 Hosting option: Cloud-based or On-premises?

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

15.2 Cost

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

15.3 Advice: starting small, grow gradually

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

15.4 My preferred setup for CT Lab

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Operating Systems

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

On-premise build agent machines

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

15.5 Set up BuildWise Server

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Run BuildWise Server on HTTPS

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

15.6 Set up Build Agents

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Name machine well

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Set up Git Authentication

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

SSH without password for macOS/Linux

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

SSH without password on Windows

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Set up a working folder on Build Agents

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

15.7 Best Practices

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Make your agent machine fast and reliable

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Uses logic server names defined in the hosts file

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Ideally make the agents the same specification

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Site License for Dynamic Scaling

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Prefer Linux/Mac over Windows

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Be aware of limitations

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Time-box the set up in days, not months

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

15.8 Wrap up

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

16. Why do most projects fail in CT?

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

16.1 Failure Factors - test automation

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Wrong choice of automation framework

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Wrong choice of test syntax framework

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Wrong choice of test automation tool

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Test script in a compiled language

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Unreliable individual test execution

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Test scripts are poorly designed and hard to maintain

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Inefficient to debug test failures

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

16.2 Failure Factors - infrastructure

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Lack of dedicated server infrastructure

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

No dedicated test lab for automation

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Immature cloud-based deployment

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

16.3 Failure Factors - continuous execution

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Long feedback

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

No Auto-retry and Manual-rerun

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Priorize test execution

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Use conventional CI servers for CT

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

16.4 Failure Factors - Human

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Executives: like the idea, but not actively involved

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Tech leads/managers: pretend to know CI/CT

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Tech leads: Fixated on a particular technology

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Developers: delay fixing regression errors

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Manual testers: fear of losing the job

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Agile coaches: Fake agile ceremonies

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Managers: The team spent too much time on JIRA, not CT

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Managers: Not allocating time for on-going maintenance

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Executives/managers: CT engineers' hard work often is underappreciated

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Executives/managers: wrong hirings

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

17. CT with web app testing in other frameworks

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

17.1 BuildWise supports multi-frameworks

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Prerequisite

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

17.2 PyTest (Python)

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Python test syntax frameworks

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

unittest

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

pytest

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Preparation

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Sequential Build

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

17.3 Mocha (JavaScript)

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Mocha test syntax framework

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Preparation

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Sequential Build

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

17.4 Cucumber (Ruby)

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Preparation

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Sequential Build

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

17.5 Parallel Build with multi frameworks

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

17.6 Review

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

18. CT with Native apps and Microservices

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

18.1 Will my tests run in this CT server?

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

18.2 Desktop App Testing with Appium

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

18.3 Non-UI Functional Testing

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Non-UI functional testing characterists

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

19. Other Uses

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

19.1 Load Testing

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

A load testing example

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

19.2 Cross-Browser Testing

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Sequential Build

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Parallel Build

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

19.3 Prepare Application Data

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

19.4 Requirement Traceability

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Understand Requirement Traceability first

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Enable requirement traceability in functional test scripts

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

Generate Requirement Traceability with ease in BuildWise

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

19.5 Execute specific tests on Server via Web Interface

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

19.6 Utilities

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

19.7 Wrap up

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

20. Appendix 1 CI Steps

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

21. Resources

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

21.1 Books

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

21.2 Tools

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.

22. References

This content is not available in the sample book. The book can be purchased on Leanpub at http://leanpub.com/practical-continuous-testing.