Table of Contents

Preface vii

Chapter 1: Introduction to DevOps and Continuous Delivery 1

Introducing DevOps 1
How fast is fast? 3
The Agile wheel of wheels 5
Beware the cargo cult Agile fallacy 6
DevOps and ITIL 7
Summary 8

Chapter 2: A View from Orbit 9

The DevOps process and Continuous Delivery – an overview 9
The developers 11
The revision control system 12
The build server 13
The artifact repository 14
Package managers 14
Test environments 15
Staging/production 16
Release management 17
Scrum, Kanban, and the delivery pipeline 17
Wrapping up – a complete example 18
Identifying bottlenecks 19
Summary 20

Chapter 3: How DevOps Affects Architecture 21

Introducing software architecture 21
The monolithic scenario 22
Architecture rules of thumb 23
The separation of concerns 23
The principle of cohesion 24

Coupling 24
Back to the monolithic scenario 24
A practical example 25
Three-tier systems 25
The presentation tier 26
The logic tier 26
The data tier 26
Handling database migrations 27
Rolling upgrades 28
Hello world in Liquibase 29
The changelog file 29
The pom.xml file 30
Manual installation 31
Microservices 32
Interlude – Conway's Law 33
How to keep service interfaces forward compatible 34
Microservices and the data tier 35
DevOps, architecture, and resilience 35
Summary 37
Chapter 4: Everything is Code 39
The need for source code control 40
The history of source code management 40
Roles and code 41
Which source code management system? 42
A word about source code management system migrations 43
Choosing a branching strategy 43
Branching problem areas 45
Artifact version naming 46
Choosing a client 47
Setting up a basic Git server 48
Shared authentication 50
Hosted Git servers 50
Large binary files 51
Trying out different Git server implementations 52
Docker intermission 52
Gerrit 53
Installing the git-review package 54
The value of history revisionism 55

The pull request model 57
GitLab 57
Summary 59
Chapter 5: Building the Code 61
Why do we build code? 61
The many faces of build systems 62
The Jenkins build server 63
Managing build dependencies 66
The final artifact 67
Cheating with FPM 68
Continuous Integration 69
Continuous Delivery 70
Jenkins plugins 70
The host server 72
Build slaves 72
Software on the host 73
Triggers 74
Job chaining and build pipelines 74
A look at the Jenkins filesystem layout 75
Build servers and infrastructure as code 76
Building by dependency order 76
Build phases 77
Alternative build servers 78
Collating quality measures 78
About build status visualization 79
Taking build errors seriously 80
Robustness 80
Summary 81
Chapter 6: Testing the Code 83
Manual testing 83
Pros and cons with test automation 84
Unit testing 86
JUnit in general and JUnit in particular 87
A JUnit example 89
Mocking 89
Test Coverage 89
Automated integration testing 90
Docker in automated testing 91
Arquillian 92

Performance testing 92
Automated acceptance testing 93
Automated GUI testing 95
Integrating Selenium tests in Jenkins 96
JavaScript testing 97
Testing backend integration points 98
Test-driven development 100
REPL-driven development 100
A complete test automation scenario 101
Manually testing our web application 102
Running the automated test 105
Finding a bug 105
Test walkthrough 105
Handling tricky dependencies with Docker 109
Summary 110

Chapter 7: Deploying the Code 111

Why are there so many deployment systems? 111
Configuring the base OS 112
Describing clusters 113
Delivering packages to a system 114
Virtualization stacks 116
Executing code on the client 118
A note about the exercises 118
The Puppet master and Puppet agents 119
Ansible 120
PalletOps 123
Deploying with Chef 124
Deploying with SaltStack 125
Salt versus Ansible versus Puppet versus PalletOps execution models 127
Vagrant 128
Deploying with Docker 130
Comparison tables 131
Cloud solutions 131
AWS 132
Azure 132
Summary 133

Chapter 8: Monitoring the Code **135**
Nagios **135**
Munin **141**
Ganglia **145**
Graphite **150**
Log handling **152**
Client-side logging libraries 153
The ELK stack 155
Summary **158**
Chapter 9: Issue Tracking **159**
What are issue trackers used for? **159**
Some examples of workflows and issues **161**
What do we need from an issue tracker? **163**
Problems with issue tracker proliferation **166**
All the trackers **167**
Bugzilla 168
Trac 174
Redmine 182
The GitLab issue tracker 188
Jira 192
Summary **194**
Chapter 10: The Internet of Things and DevOps **195**
Introducing the IoT and DevOps **195**
The future of the IoT according to the market **199**
Machine-to-machine communication **201**
IoT deployment affects software architecture **202**
IoT deployment security **202**
Okay, but what about DevOps and the IoT again? **203**
A hands-on lab with an IoT device for DevOps **205**
Summary **211**
Index **213**