

CONTENTS

1	Introduction	1
1.1	Post Release Testing supports developers to efficiently do operations work	1
1.2	How to read this masterthesis	2
2	State of the art technologies and practices are the foundations for NPRT	3
2.1	The Continuous Delivery Pipeline consists of commit, automated testing and deployment	3
2.2	Docker packages applications	3
2.3	Kubernetes is a cluster operating system	3
2.4	Monitoring a highly dynamic infrastructure is role centric	3
3	Continuous Delivery only covers practices until release	5
3.1	Continuous Delivery disregards security and operations topics	5
3.2	Fast time to market is crucial	5
3.3	Continuous monitoring is hard	5
3.4	Simple day to day work must be automated	5
4	Nonfunctional Production Regression Testing extends the Continuous Delivery Pipeline	7
4.1	NPRT evolves from continuous deployment and the CD Pipeline	7
4.2	A new version is validated with production traffic	7
		i

4.3	Nonfunctional tests in NPRT focus on latency, utilization and errorrate	7
4.4	NPRT tests a new version for regression in its characteristics	7
4.5	NPRT enables unattended continuous updates	7
5	Deployer implements and automates NPRT	9
5.1	Version centric testing via commit hashes	9
5.2	A canary and its testing metrics know about themselves	9
5.3	Controlling deploys in the pipeline and manually	9
5.4	Comparison of versions triggers webhooks for further actions	9
6	Evaluation	11
6.1	How NPRT changes the behaviour of development teams	11
6.1.1	Deploys	11
6.1.2	Cycletime	11
6.1.3	Change	11
6.1.4	True/False Positives/Negatives	11
6.2	NPRT compared to other in production testing strategies	11
6.2.1	Netflix Simian Army to intensify NPRT	11
6.2.2	Synthetic Monitoring is functional post release testing	11
7	Conclusion	13
7.1	Resume	13
7.2	Outlook and future work	13

CHAPTER 1

INTRODUCTION

1.1 Post Release Testing supports developers to efficiently do operations work

Many people talk about DevOps as well as there are multiple definitions and interpretations of the term DevOps. DevOps is referred as a philosophy, a culture, practices and specific tools. For my research, I will focus on two different aspects of the term DevOps:

The first one is the perspective of operation teams. Operation teams traditionally modeled infrastructure by installing physical hardware and by manually installing software components. With the rise of virtual machines and the cloud, it became possible to model infrastructure in software¹. Modelling via software enables operation teams to use tools and practices² as seen in software engineering. Infrastructure code is version controlled, tested and can be automatically deployed.

The other aspect of DevOps³ is the perspective of developer teams. Previously developer teams were only responsible for developing new features. Software engineering practices got established and proven. One of those practices is the continuous delivery pipeline⁴. The last step of the continuous delivery pipeline is the deployment. Formerly operation teams were responsible for deploying new features. The deployment as last step of the continuous delivery pipeline shifts a responsibility from operation to de-

¹“Infrastructure as Code” describes different dynamic infrastructure types [3, p. 30] and how to model those by code [3, p. 42].

²In the chapter “Software Engineering Practices for Infrastructure” [3, p. 179-194] practices like version controlling, continuous integration are described.

³The book “DevOps” [1] is written in the view of a developer running a system.

⁴For theoretical details on the continuous delivery pipeline read Part II of “Continuous Delivery” [2, p. 103-140] or a more practical approach by Wolff [4].

velopment. This shows that developer teams are becoming more and more responsible for running the software, they built.

1.2 How to read this masterthesis

CHAPTER 2

STATE OF THE ART TECHNOLOGIES AND PRACTICES ARE THE FOUNDATIONS FOR NPRT

- 2.1 The Continuous Delivery Pipeline consists of commit, automated testing and deployment**
- 2.2 Docker packages applications**
- 2.3 Kubernetes is a cluster operating system**
- 2.4 Monitoring a highly dynamic infrastructure is role centric**

CHAPTER 3

CONTINUOUS DELIVERY ONLY COVERS PRACTICES UNTIL RELEASE

3.1 Continuous Delivery disregards security and operations topics

3.2 Fast time to market is crucial

3.3 Continuous monitoring is hard

monitoring change and trying to predict the future from data

3.4 Simple day to day work must be automated

CHAPTER 4

NONFUNCTIONAL PRODUCTION REGRESSION TESTING EXTENDS THE CONTINUOUS DELIVERY PIPELINE

- 4.1 NPRT evolves from continuous deployment and the CD Pipeline**
- 4.2 A new version is validated with production traffic**
- 4.3 Nonfunctional tests in NPRT focus on latency, utilization and errorrate**
- 4.4 NPRT tests a new version for regression in its characteristics**
- 4.5 NPRT enables unattended continuous updates**

CHAPTER 5

DEPLOYER IMPLEMENTS AND AUTOMATES NPRT

5.1 Version centric testing via commit hashes

build, test, deploy, only 2 versions in production. undeploy a canary.

5.2 A canary and its testing metrics know about themselves

5.3 Controlling deploys in the pipeline and manually

5.4 Comparison of versions triggers webhooks for further actions

monitoring validation, fail

CHAPTER 6

EVALUATION

6.1 How NPRT changes the behaviour of development teams

6.1.1 Deploys

6.1.2 Cycletime

6.1.3 Change

6.1.4 True/False Positives/Negatives

6.2 NPRT compared to other in production testing strategies

6.2.1 Netflix Simian Army to intensify NPRT

6.2.2 Synthetic Monitoring is functional post release testing

CHAPTER 7

CONCLUSION

7.1 Resume

7.2 Outlook and future work

Bibliography

- [1] Len Bass, Ingo Weber, and Liming Zhu. *DevOps: a software architect's perspective*. Addison-Wesley, 2015.
- [2] Jez Humble and David Farley. *Continuous Delivery: reliable software releases through build, test, and deployment automation*. Addison-Wesley, 2010.
- [3] Kief Morris. *Infrastructure as code : managing servers in the cloud*. O'Reilly, 2016.
- [4] Eberhard Wolff. *Continuous Delivery: der pragmatische Einstieg*. 2. dpunkt.verlag, 2016.