

CONTINUOUS DEPLOYMENT MIT OPSI UND PIPELINES OSAD 2018

AGENDA

- Einleitung und Vorstellung
- Was ist opsi?
- Motivation
- Continuous Deployment
- Neue Testumgebung
- QA

VORSTELLUNG

EROL UELUEKMEN

- @uib since 2003
- Geschäftsführer der uib gmbh
- Core-Member in der opsi Entwicklung
- Support, Training und Services Rund um opsi
- Twitter: @eroluel

UIB GMBH

- Gegründet 1995
- Zusammenschluss von Systemverwaltern
- Von Anfang an Schwerpunkt in der Systemtechnik
- Creator und Maintainer von opsi

UIB GMBH

- Aktuell 20 Mitarbeiter
- Sitz in Mainz
- Mitglied in der OSBA
- Twitter: @uibDE

OPSI

*Enterprise OpenSource Clientmanagement und
Configuration Management für Windows- und Linux-
Clients.*

- Twitter: @opsi_org

OPSI

opsi ist Modular einsetzbar.

OPSI-MODULE

- OS-Deployment
- Software-Deployment
- Inventarisierung
- Configurationmanagement
- Patchmanagement

OPSI

ERWEITERUNGSMODULE

- Directory-Connector
- WAN/VPN-Modul (z.B.: Homeoffice)
- Lizenzmanagement
- Local-Image / VHD-Reset
- UEFI-Support
- WIM-capture
- ...

WARUM OPSI?

- Lange History (Startschuss war 1994)
- Bei Bedarf Herstellersupport
- Große opsi-Community
- Freie opsi-Repos

WINDOWS UND LINUX- UNTERSTÜTZUNG

- Windows- und Linux-Unterstützung aus einer Codebase
- Automatisierung der Hersteller-Installer (kein chroot-Gebastel)
- Möglichkeit jegliche Skripte zu verwenden, wenn der Interpreter vorhanden ist

MOTIVATION

- Clientmanagement != Servermanagement
- Einblick in die Entwicklung/Releasemanagement von opsi
- Erfahrungsaustausch

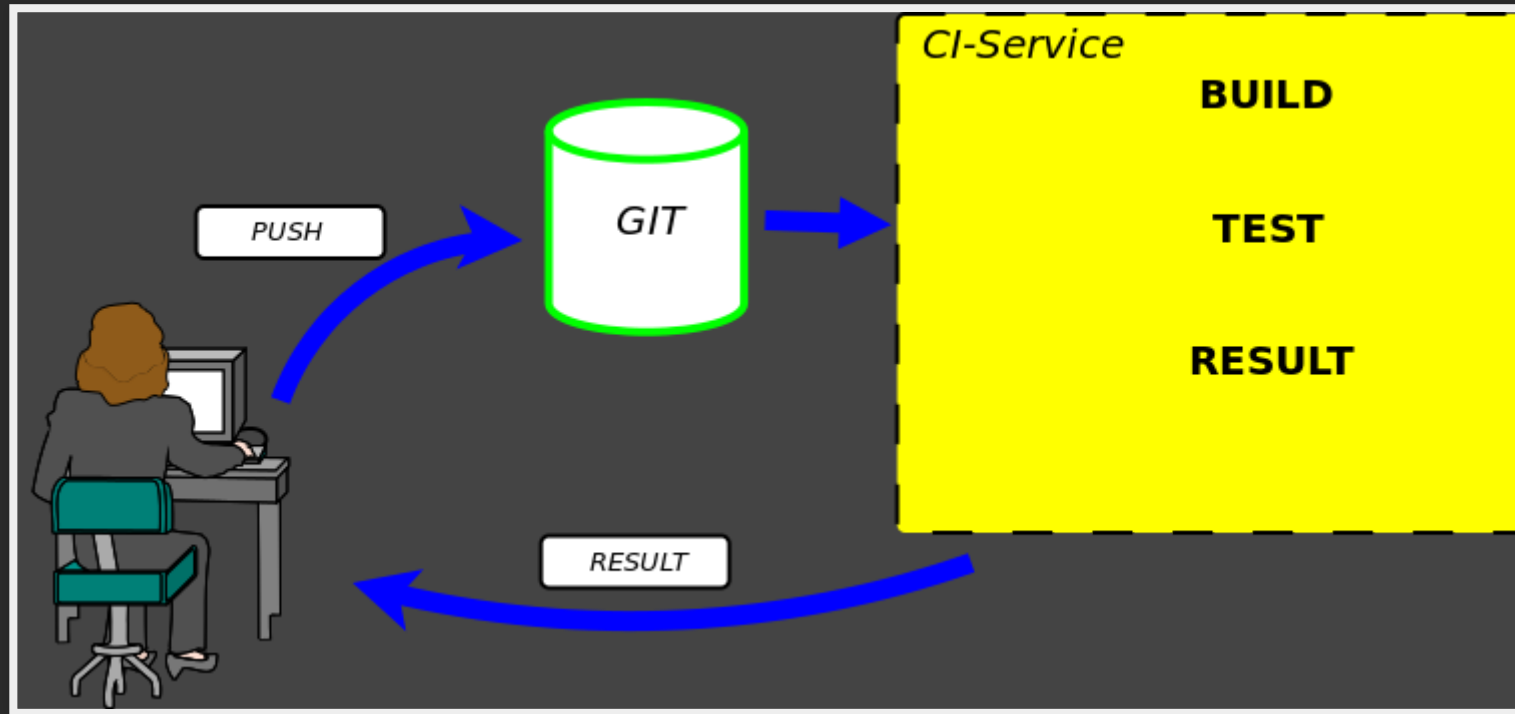
CONTINUOUS...

...hölle

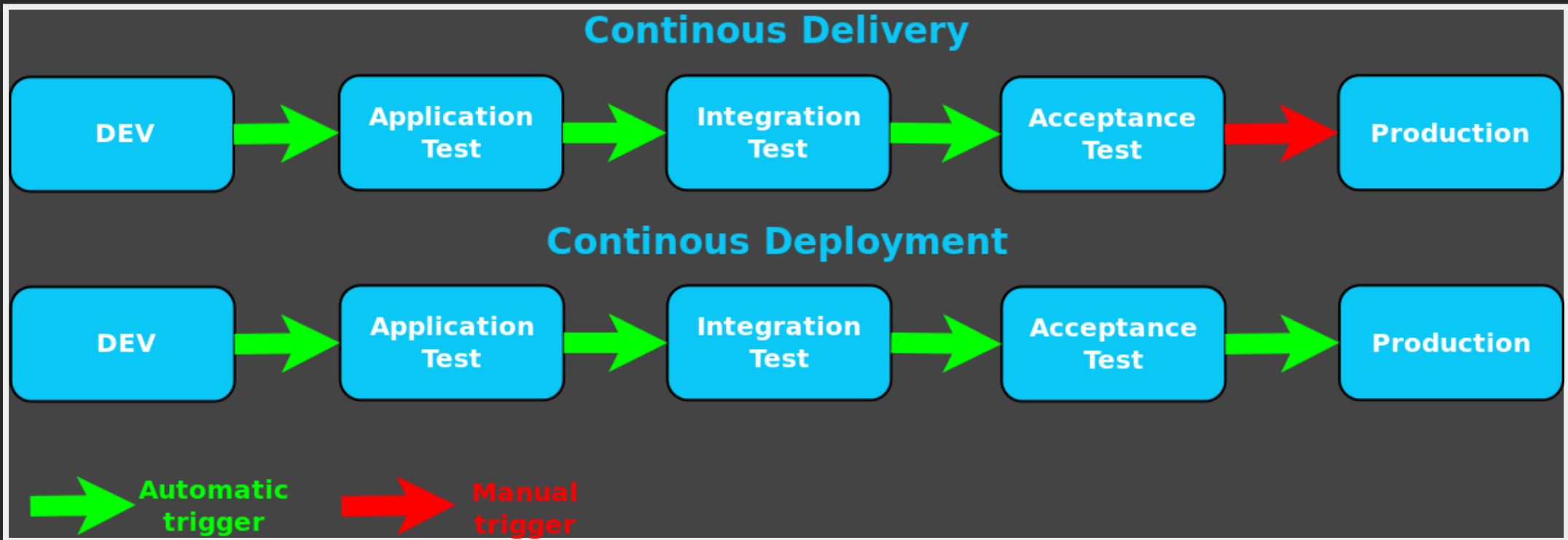
DISCLAIMER

Folgende Ausführungen sind Interpretation und gelebte Prozesse. Das folgende hat nicht immer die selbe Funktion, wie in Fachliteraturen beschrieben wird.

CONTINUOUS INTEGRATION



CONTINUOUS DELIVERY/DEPLOYMENT



DEVOPS

FAZIT IN BEZUG AUF OPSI

Ist doch alles nur Entwicklerzeugs! Oder doch nicht?

ENTWICKLUNG VON OPSI

Die Entwicklung von opsi ist sehr stark von diesen Ideen getrieben.

PAKETIERUNG VON OPSI- PAKETEN

Bei opsi-Paketen werden diese Ideen auch für die QA verwendet.

OPSI-RELEASES

- Rolling Release
- Stable, Testing und Experimental
- Bald zwei supportete Zweige: 4.0.x und 4.1.x

SUPPORTMATRIX

- Explosion der Supportmatrix seit Linuxclient-Support
- Das manuelle Testen der Supportmatrix ist nicht mehr möglich

WINDOWS

WINDOWS-CLIENTS

- Windows 7, Windows 8.1 und Windows 10
- Windows XP wird noch unterstützt, ist aber in opsi freezed

WINDOWS-SERVER

- Windows Server 2008r2, Windows 2012, Windows 2012r2 und Windows 2016
- Windows 2003 wird noch unterstützt, ist aber in opsi freezed

SUPPORT-WINDOWS 10

- 1511 (kennt sich intern als Technical Preview)
- 1607 (AniversaryUpdate)
- 1703 (CreatorsUpdate)
- 1709
- 1803
- 1809

LINUX

OPSI-BETRIEB

Distributionen die für den Betrieb als opsi-Server unterstützt werden.

DEB-BASIIERT

- Debian 8 und 9
- Ubuntu 14.04, 16.04 und 18.04 (LTSe)
- Univention UCS 4.1, 4.2 und 4.3

RPM-BASIERT

- OpenSuseLeap 42.3
- SLES 11Sp4,12,12SP1, 12SP2, 12SP3
- CentOS 6 und 7
- RedHat 6 und 7

OPSI-CLIENT BETRIEB

Clients/Server die mit opsi gemanaged werden können.

DEB-BASIIERT

- Debian 8 und 9
- Ubuntu 16.04 und 18.04 (mehr über ubuntu-Paket installierbar)
- Univention UCC (nicht offiziell, steht aber auf der Roadmap)

RPM-BASIERT

- OpenSuseLeap 42.1 und 42.2
- SLES 11sp4, 12, 12sp1
- CentOS 7
- Redhat 7

OPSI-PAKETE

OPSI-LOCALBOOTPAKETE (STANDARD)

- opsi-client-agent
- opsi-linux-client-agent
- opsi-script/opsi-winst
- hwaudit, hwinvent und swaudit
- und einiges mehr....

OPSI-NETBOOTPAKETE (STANDARD)

- Windows Netbootpakete
- Linux Netbootpakete
- opsi-clonezilla, memtest
- etc

OPSI-LOCAL-IMAGE

- opsi-local-image-prepare
- opsi-local-image-backup
- opsi-local-image-win*
- opsi-local-image-capture

OPSI-MODULPAKETE

- opsi-wim-capture
- etc

ANFORDERUNGEN AN DIE TESTUMGEBUNG

- Es muss verdammt hoch skalieren
- Es darf sich nicht selber testen
- Es muss einfach zu bedienen und zu Erweitern sein
- Es sollte möglichst auf Standards setzen

ALTE TESTUMGEBUNG

- Skalierung: 0
- Selbst gebaut und eigene Standards umgesetzt
- Testimplementierung war als Nichtdev fast unmöglich
- Hat sich ständig selbst getestet

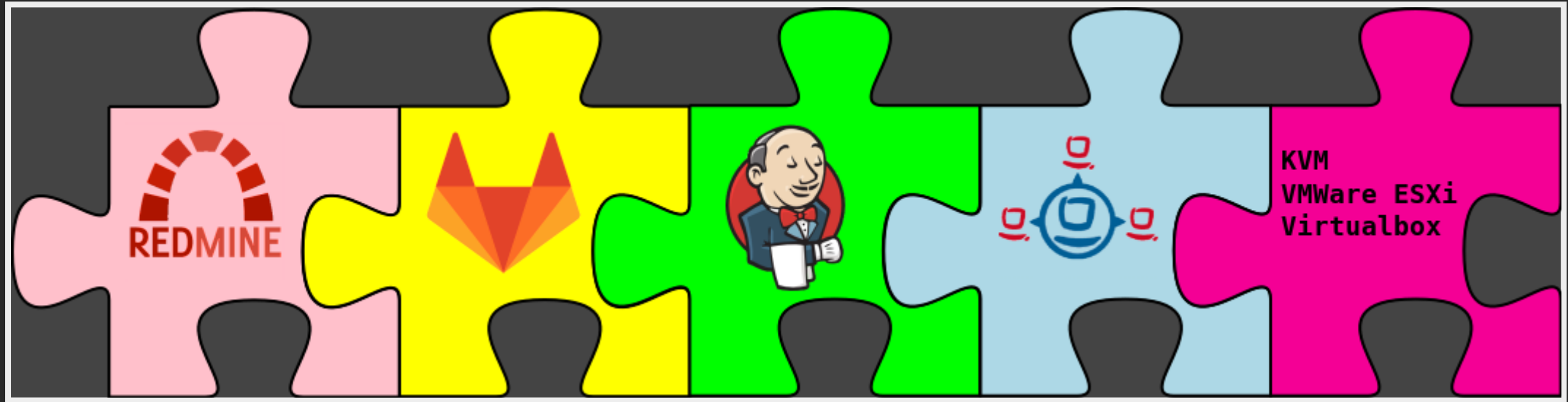
NEUE TESTUMGEBUNG

- Was wir brauchen:
- 1. Neues Konzept
- 2. Neue Toolchain

TESTKONZEPT

Alles muss mit allem in allen Zweigen getestet werden.

TOOLCHAIN



TOOLCHAIN FUTURE

- Folgende Dinge fehlen in der Toolchain:
- OBS (OpenBuildservice)
- Icinga2
- Kix
- Docker/Moby oder LinuxContainer

INTEGRATIONEN

Eine Toolchain bringt nichts, wenn die Tools nicht wirklich zusammenarbeiten. Nur wie macht man das?

FUNKTIONEN DER TOOLS

- gitlab: Versionierungstool
- jenkins: Scheduler
- opsi: Automatisierungsframework
- Virtualisierungen: Knechte für die Arbeitsbewältigung

EINBLICK IN DIE INTEGRATION

- gitlab und Jenkins arbeiten relativ nativ zusammen. (meistens Konfiguration)

JENKINS UND OPSI

- opsi-plugin für jenkins bauen? (Java... aargh)

JENKINS UND OPSI

- Dann lieber Anbindung über opsi-admin

JENKINS ARBEITET MIT SLAVES

OPSI WIRD ERST MAL EIN
BUILDSLAVE VON JENKINS

JENKINS KANN NUN "BUILDS"
AUF DEM OPSI-SERVER
AUSFÜHREN

DAFÜR BAUT MAN JOBS IN
JENKINS

VIRTUALISIERUNGEN

- Einbindung über fabric (python)
- Nutzung der Virtualisierungsccli's über ssh

JENKINS DEMO

Reset target	install xenial	Server preconfig	Install opsi	Test server...	Win7-x64	Win81-x64	Win10-x64
10s	45min 21s	2min 2s	8min 4s	3min 54s	54min 18s	32min 40s	33min 15s
10s	20min 2s	2min 2s	10min 46s	4min 4s	59min 15s	33min 27s	33min 6s

VIELEN DANK

Fragen?

WERDE TEIL DER COMMUNITY

- <https://www.opsi.org>
- <https://forum.opsi.org>
- <https://github.com/opsi-org/>
- Translation: <https://www.transifex.com/opsi-org/opsiorg/>
- Kaufe Professionellen Hersteller-Support und Enterprise-Erweiterungen bei <https://www.uib.de> ;)