

# Hogyan legyél DevOps

Összefoglaló, iránymutatás

Mátó Péter

<mato.peter@andrews.hu>

# Bemutakozás

Húsz éve fejleszték és adminisztráltak változó arányban kisebb és nagyobb cégeknek

Szakterületek: IT biztonság (főleg web biztonság, tűzfalak, központi jogosultság kezelés)

Jelenleg egy központi menedzsment szoftvert fejleszték az Andrews IT Engineeringnél

# Sysadmin ismeretek

Jól ismerni valamilyen Unix/Linux rendszert (hálózat, diszk alrendszer, csomagkezelés, időzítés, finomhangolás)

Egyedi hálózati szolgáltatások: LAMP stack és mutánsai, Apache, NginX, MySQL, MariaDB, Postgres, PHP, Python, Django, Ruby, RoR, SMTP, IMAP, POP...

Központi hálózati szolgáltatások: NTP, DHCP, DNS, LDAP, Kerberos...

Monitoring és metrikák, hibajegy kezelés, HA rendszerek

Háttérismeretek: Crash – recovery, TCP/IP, PKI, RAID, VPN, SQL és NOSQL és még hosszan sorolhatnánk...

# A klasszikus admin munkakörnyezete

Elvárás: beton stabil futtató környezet

Régi libek, szoftverek, futtató környezet /**akár több éves** (!)/, alkalmazások, lassú reakció, tervezett konfigurációs ablakok, leállítás, karbantartás stb.

ITIL – ultra konzervatív üzemeltetési módszertan

Jó gyakorlatokon alapul. Jó lassú gyakorlatokon.

Alapos tervezés → fejlesztés → tesztelés van; kísérletezésnek **helye nincs**

Release és deploy menedzsment

Deploy előtt többszörös tesztelés, egy éles deploy (akár egy hibajavítás)  
**akár hónapokig is eltarthat**

# Béna rendszergazda (v0.1)

Fehér zokni, szandál; vicces feliratos póló

Titokzatos szavakat használ, nemigen lehet érteni amit mond

Csak akkor veszi észre, hogy baj van, ha felhívják vagy e-mailben megírják neki

Akkor is csak lassan reagál, mert túl sok a dolga

Probléma esetén képes hívni a központi support-ot

A problémák évekig megmaradnak



# Kezdő rendszergazda (v0.5)

A rendszereket még nem ismeri eléggé,  
probléma esetén rákeres a megoldásra

Sokat tanul a tapasztaltabb rendszergazdáktól

A hibákat a már létező monitoring rendszer  
riasztásaiból észleli, kis segítséggel elhárítja

Lassan, de képes felépíteni egy új rendszert

# Rendszergazda (v1)

A rendszereket jól ismeri, a legtöbb problémát rutinból ismeri és elhárítja

Napok alatt képes felépíteni egy teljes komplex rendszert

A rendszeresen előforduló teendőket villámgyorsan képes elvégezni

# A sysadmin megfelel, ha:

A cég tud fizetni átlagosan **egy**  
**rendszergazdát** szerverenként

Ha a cég nem fejleszt saját szoftvereket,  
vagy **nem számít a** fejlesztési **költség**

Ha a cégnek nem számít a meghibásodás  
utáni **helyreállási idő**



A fejlesztő – admin  
együttműködés általában  
rendkívül nehézkes

**A fejlesztői környezetben  
működik**



**Innentől az  
adminok problémája**

# SaaS – Software as a Service

# IaaS – Infrastructure as a Service

## jelentősége



# Fejlesztő cégek, SaaS

Az adminok általában **nem ismerik eléggé**  
a szoftvert, annak igényeit

A fejlesztőknek általában csak halvány  
fogalmuk van a futtató **környezet**  
**lehetőségeiről, korlátairól**

Valahogy közelebb kellene hozni őket

# A fejlesztés gyökeresen megváltozott

A vízesés modell és barátai: nem ad választ  
a változó igényekre, általában túl lassú

Az új paradigmák:

**agilis fejlesztés** – gyors reakció, újratervezés, pár hetes fejlesztési sprintek, melyek használható állapothoz vezetnek

**lean optimalizáció** – semmi felesleges, sok próbálkozás, a felhasználói tesztek után a rossz elgondolásokat azonnal dobják



# Lean – fejlesztési folyamat optimalizáció

El kell hagyni a felesleges dolgokat

felesleges folyamatok és funkciók (tipikusan 60-70%!!)

taszk váltások, várakozás, mozgatás, hibás-, félkész munka

A valós igényekre kell koncentrálni

Az eredményeket gyorsan kell tesztelni

Csak a jó mehet tovább

# Teljesen új elvárások

Sebesség. Sebesség! **Nagy sebesség!!!**

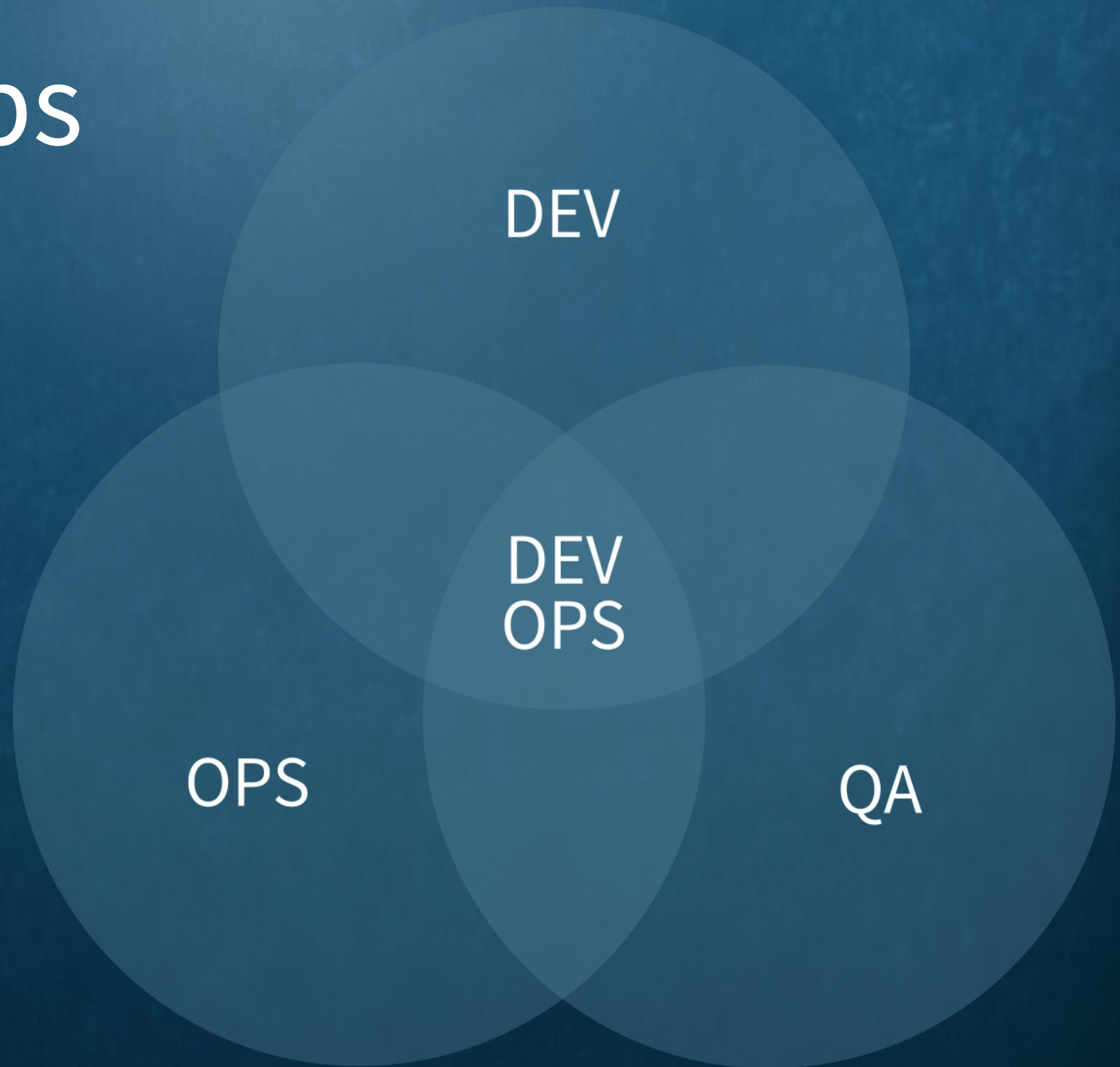
A fejlesztői hibák derüljenek ki **azonnal**,  
ne legyen regresszió

Gyorsan buildelhető, tesztelhető majd használható  
legyen a szoftver

Gyors deployment, akár több ezer szerverre

Automatizálás, virtualizáció támogatása

# DevOps



# DevOps

A DevOps nem egyszerűen Sysadmin 2.0

A DevOps nem pusztán technológia

A DevOps **hozzaállítás**

A DevOps igény arra, hogy a dolgok gyorsan,  
hibátlanul és automatikusan történjenek  
meg

# Rendszergazda (v1)

A rendszereket jól ismeri, a legtöbb problémát **rutinból** ismeri és **elhárítja**

**Napok alatt képes** felépíteni egy teljes komplex rendszert

A rendszeresen előforduló teendőket **villámgyorsan képes elvégezni**



# DevOps ( Sysadm v2.5 )

A rendszereket jól ismeri, a legtöbb probléma  
**nem ismétlődik meg** kétszer

**Percek alatt képes** felépíteni egy teljes  
komplex rendszert

A rendszeresen előforduló teendőkre  
**programot ír, nem csinálja meg többször  
kézzel**

# DevOps feladatok

Segít a fejlesztőknek olyan környezetet előállítani, amilyen az éles rendszer

A CI segít a hibák azonnali felfedezésében, a regressziók elkerülésében

A gyors deployment segít, hogy a javítások és új funkciók azonnal kipróbálhatók legyenek, élesedjenek a felhasználóknak is

# A DevOps eszközkészlet

**PERIODIC TABLE OF DEVOPS TOOLS (V1)**

Xebia Labs  
Deliver Faster

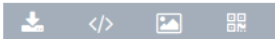
Os	Open Source	Database	SCM	Build
Fr	Free	CI	Repo Mgmt	Testing
Fm	Freemium	Deployment	Config / Provisioning	Containerization
Pd	Paid	Cloud / Iaas / Paas	Release Mgmt	Collaboration
En	Enterprise	BI / Monitoring	Logging	Security

The periodic table of DevOps tools is organized into groups based on their licensing and features. The tools are arranged in a grid, with some tools having multiple versions or variants. The table is titled 'PERIODIC TABLE OF DEVOPS TOOLS (V1)' and includes the Xebia Labs logo.

Share



Embed



Become Excellent!

 [Subscribe here!](#)

91 En XIr XL Release	92 En Ur UrbanCode Release	93 En Ls CA Service Virtualization	94 En Bm BMC Release Process	95 En Hp HP Codar	96 Pd Ex Excel	97 En Pl Plutora Release	98 En Sr Serena Release	99 Fm Tr Trello	100 Pd Jr Jira	101 Fm Rf HipChat	102 Fm Sl Slack	103 Fm Fd Flowdock	104 Pd Pv Pivotal Tracker	105 En Sn ServiceNow
106 Os Ki Kibana	107 Fm Nr New Relic	108 Os Ni Nagios	109 Os Gg Ganglia	110 Os Ct Cacti	111 Os Gr Graphite	112 Fm Le Logentries	113 En Sp Splunk	114 Fm Sl Sumo Logic	115 Os Ls Logstash	116 Fm Lg Loggly	117 Os Gr Graylog	118 Os Sn Snort	119 Os Tr Tripwire	120 En Cy CyberArk

# SCM – Software

## Configuration Management

Git, Gitlab, SVN, CVS

## Build

make, Maven, ANT, Buildr...

## CI – Continuous Integration

Gitlab CI, Jenkins, Travis CI...

# Deployment

ssh, rsync, csomagkészítők

# Provisioning

Puppet, Chef, Salt, Ansible,  
Vagrant, CFEngine...

# Virtualizáció, konténerek

KVM, Xen, LXC, Docker, rkt, Kubernetes...



# Javaslatok az induláshoz I.

Virtualizáció kísérletezésre, tesztekre

Ismerd meg a disztribúciód telepítőjének  
automatikus metódusát

Készíts egy teljes helyreállító megoldást,  
mely csak a csomagokat és a konfigokat  
teszi vissza, nem teljes mentésből

Tanulj meg Python-ban programozni

# Javaslatok az induláshoz II.

Használd Git-et a konfigurációs állományok tárolására (pl. etckeeper vagy saját)

Készíts rendszert a konfigok automatikus mentésére, terítésére

Ismerj meg egy jegykezelő rendszert (pl. Gitlab Issue Tracker)

Köszönöm a figyelmet.