

Hálózatok építése és üzemeltetése

Linux

Rendszergazda jogosultságok

(Fehér Gábor slide-jai)

Root jogosultságok

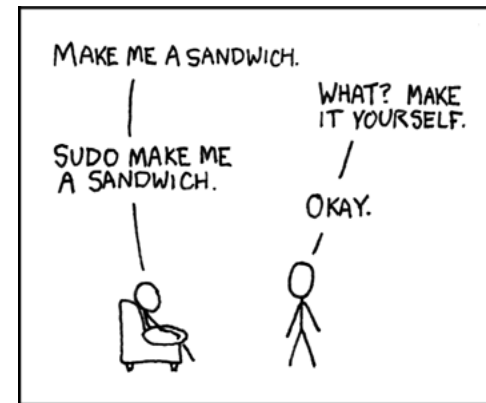
- ▶ **Többfelhasználós rendszerekben adminisztrációs teendők**
 - ▶ Felhasználók menedzselése
 - ▶ Külső erőforrások csatolása
 - ▶ Szolgáltatások futtatása
 - ▶ Szoftverek frissítése
- ▶ **Védelem a szándékos és nem szándékos rombolástól**

Felhasználók kezelése

Felhasználó menedzsment

▶ Jogosultság megszerzése

- ▶ **su, sudo, id** parancsok
- ▶ */etc/sudoers, /etc/sudoers.d*
- ▶ sudo csoport



▶ Felhasználó hozzáadása, csoportok hozzáadása

- ▶ **adduser, addgroup** parancsok
- ▶ A felhasználók csoportokba oszthatók
 - ▶ 1 felhasználó több csoportban is lehet

Jelszavak

- ▶ **passwd** parancs

- ▶ */etc/passwd*

- ▶ *root:x:0:0:root:/root:/bin/bash*

- ▶ */etc/shadow*

Jelszó

Adatok,
adatok

Kezdő folder

Parancs

UserID,
GroupID

- ▶ Jelszó tárolás titkosított formában + salt
 - ▶ + már nem publikus a jelszó mező
 - ▶ Alapból DES, de ma már többféle titkosítás (pl. SHA-512)

Partíció, fájlrendszer

Partíciók

- ▶ A lemezterület feldarabolás
 - ▶ Partíciók mérete
 - ▶ Partíciók típusa (+ boot)
 - ▶ swap partíció
- ▶ **fdisk**, **cfdisk** parancsok
 - ▶ */dev* rendszer
 - ▶ */dev/sd**, */dev/hd**, ...
- ▶ A partíció csak felosztás, nem fájlrendszer
- ▶ Különböző partíció leírás típusok (pl. DOS/MBR)



Ajánlott partíciók Linuxon

- ▶ Minimum 2 partíció
 - ▶ Rendszer adatok
 - ▶ swap partíció
- ▶ Opcionálisan további partíciók
 - ▶ home kötet
 - ▶ boot partíció

- ▶ Logical Volume Manager (LVM) - haladóknak
 - ▶ Több lemez együttes kezelése
 - ▶ Partíció méret nagyobb lehet a lemez méreténél
 - ▶ Dinamikus méret kezelés

FLASH partíciók

- ▶ MTD - Memory Technology Device
 - ▶ Absztrakciós réteg a különböző nyers FLASH memóriák kezeléséhez
 - ▶ Nem USB stick vagy memóriakártyák!
 - ▶ Hibás blokkok menedzselése
 - ▶ Használat kiegyenlítés (wear leveling)
 - ▶ /proc/mtd

```
dev:      size  erasesize  name
mtd0: 00040000 00010000  "cfe"
mtd1: 00fb0000 00010000  "linux"
mtd2: 0096e000 00010000  "rootfs"
mtd3: 00010000 00010000  "nvram"
mtd4: 004e0000 00010000  "ddwrt"
```

Fájlszisztemek

- ▶ Fájlok tárolása a lemezterületen (memóriaterületen)
 - ▶ ext2/3/4 – Linux fájlrendszer inode alapon
 - ▶ vfat, ntfs – Windows (DOS) fájlrendszer
 - ▶ iso9660, udf – CDROM, DVD, Bluray
 - ▶ jffs2, ubifs – Fájlrendszer Flash memóriához
 - ▶ ramfs, tmpfs – Memóriában tárolt fájlrendszer
 - ▶ nfs, cifs, smbfs, *davfs2* – Távoli fájlrendszer

Különleges fájlrendszerek

- ▶ Rendszer működéséhez köthető
 - ▶ */dev, /proc, /tmp, /sys*
- ▶ Fuse – Filesystem in Userspace
 - ▶ Userspace-ben futtatott fájlrendszerek
 - ▶ Egyszerűbb fejlesztés, felhasználói elérés
- ▶ Overlay fájlrendszerek
 - ▶ Pl.: Titkosítás, tükrözések, külön írható/olvasható részek
 - ▶ squashfs, unionfs, aufs (pl: Docker!)

Különleges fájlrendszerek

- ▶ squashfs, unionfs, aufs (pl: Docker!)

Container Layer

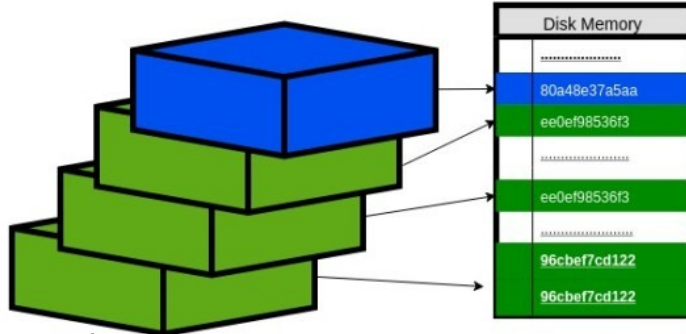
CMD java -jar Service.jar

Image Layers

RUN apt install -y gzip \ openjdk

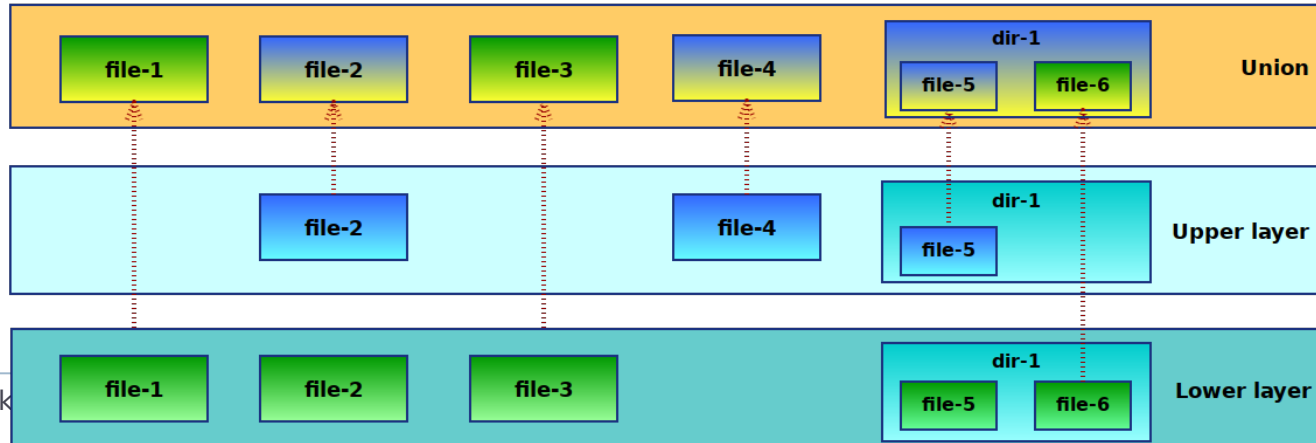
RUN apt install -y git

From ubuntu:latest



Unioning filesystem

<https://blog.knoldus.com/unionfs-a-file-system-of-a-container/>



Fájlrendszerek kezelése

- ▶ Fájlrendszer készítése
 - ▶ Partíció, kijelölt fájlban
 - ▶ **mkfs, mount, umount, fsck, df, dd, sync** parancsok
 - ▶ */etc/fstab, /etc/mtab* fájlok

- ▶ Swap fájlrendszer
 - ▶ **mkswap, swapon, swapoff** parancsok

Boot folyamat

Bootloader - PC

- ▶ Rendszer induláskor BIOS vagy UEFI boot
 - ▶ Basic Input/Output System (rég)
 - ▶ Unified Extensible Firmware Interface (új)
- ▶ Bootloader helye
 - ▶ Master Boot Record - MBR (lemez boot sector)
 - ▶ Volume Boot Record - VBR (Partíció boot sector)
 - ▶ Ma már MBR helyett: GPT (GUID Partition Table)
- ▶ MBR bootloader kötelező, mert a BIOS ezt indítja
- ▶ VBR bootloader indítható MBR-ből
 - ▶ first stage boot loader / chainload

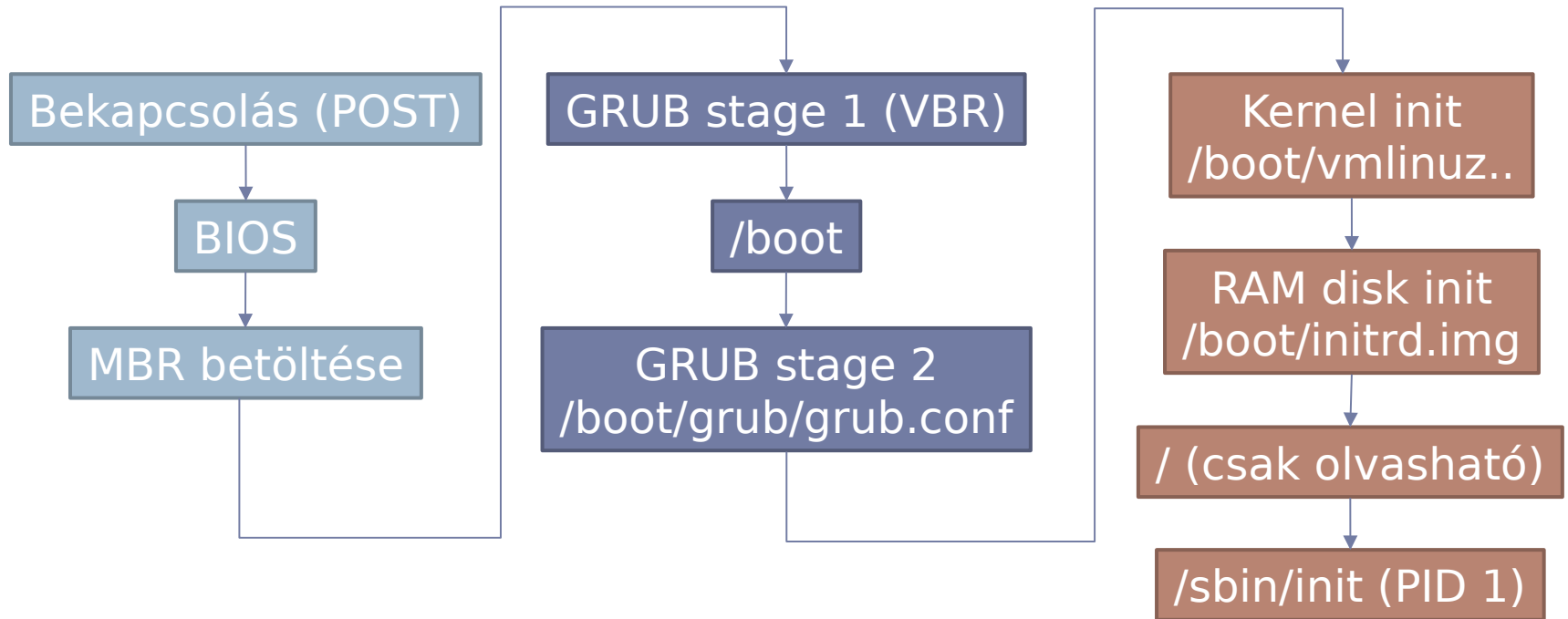
Bootloader – Beágyazott rendszerek

- ▶ Das U-Boot
 - ▶ Universal Boot Loader
- ▶ Common Firmware Environment – CFE

- ▶ Lehetőségek
 - ▶ Indítás FLASH területről
 - ▶ Feltöltés (+lementés) FLASH területre
 - ▶ Hálózati műveletek (Főként TFTP)
 - ▶ Partíció kezelés
- ▶ Device tree
 - ▶ A HW egységes leírása

Linux Boot Process - PC

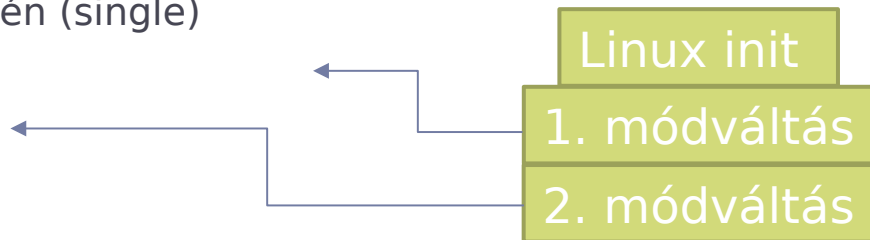
► Boot folyamat



sysvinit séma (System V, ~1980)

▶ Futási szintek (runlevel)

- ▶ S: egyedüli felhasználó boot esetén (single)
- ▶ 1: egyedüli felhasználóra váltás
- ▶ 2....5: Többfelhasználós mód
- ▶ 6: Újraindítás



▶ Futási szintek szerint külön scriptek az indulásnál

- ▶ /etc/rc<futási szint>.d
- ▶ Minden fájl csak softlink a /etc/init.d scriptekre
- ▶ S<XX><script név> és K<XX><script név> a futási szint váltásnál induláshoz és leállításhoz
 - ▶ XX: Indítási, leállítási sorrend meghatározása
 - ▶ szekvenciális - lassú!
- ▶ rc folderek kezelése
 - ▶ Pl.: **update-rc.d** parancs

Szoftverek, szolgáltatások

Szolgáltatások - Services

- ▶ Démonok (daemon)
 - ▶ Szolgáltatások indítása automatikusan pl. **sysvinit** szerint
 - ▶ **service** parancs használata
 - ▶ start, stop, restart
 - ▶ */etc/init.d* scriptek közvetlen hívása is lehetséges
 - ▶ Naplózások
 - ▶ */var/log/...*
- ▶ sysvinit rendszer: elavult
 - ▶ átmeneti megoldás volt pl.: Ubuntu upstart (2006-2014)
- ▶ modern megoldás: systemd
 - ▶ pl: `systemctl start/stop/status openvpn`
 - ▶ párhuzamos indítás, automatikus függőség feloldás, auto-recovery...

Szoftver telepítés és frissítés

- ▶ Csomagkezelő (package management)
 - ▶ Debian, Ubuntu: **dpkg (.deb)**
 - ▶ Speciális szerkezet (archívum)
 - Összefüggések, kompatibilitás
 - Integritás ellenőrzés
 - Telepítendő fájlok, scriptek a telepítéshez
 - ▶ packages.debian.org
 - ▶ Advanced Packaging Tool: **apt, aptitude**
 - ▶ Függőségek automatikus kezelése, feloldása
 - ▶ `/etc/apt/...`
 - ▶ `sudo apt-get update`
 - ▶ `sudo apt-get install ...`
 - ▶ újabb verzió: `sudo apt install`

Frissítések

- ▶ Csomagkezelőből vezérelve
 - ▶ Automatikus frissítés függőségek megtartásával

- ▶ Backport
 - ▶ Új verzió implementálása a régi rendszerre

Instabil disztribúciók

- ▶ Stabil és teszt (sid) verzió
 - ▶ unstable -> testing -> stable

Kis előrettekintés: SDN

Szoftver-alapú hálózatok

Cloud Native: új ökoszisztéma

▶ Szoftver

- ▶ új, extrém alkalmazások
- ▶ új szoftverfejlesztési, -üzemeltetési technológiák

▶ Felhő platformok

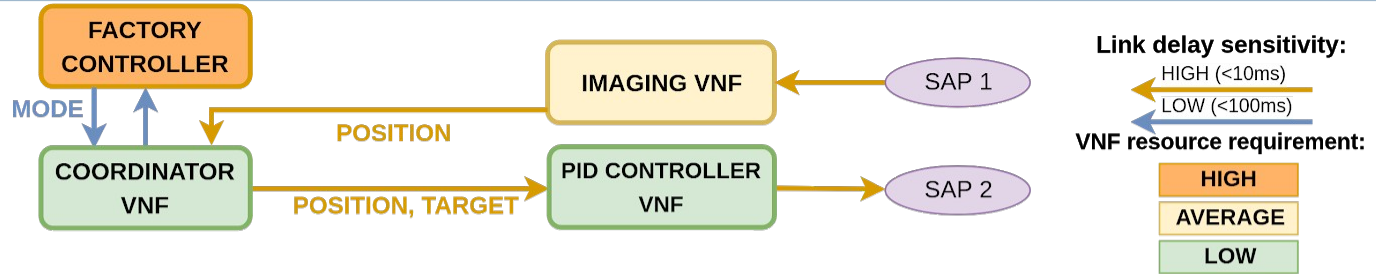
- ▶ publikus és privát felhők: konténerek, VM-ek
- ▶ újabb és újabb platform szolgáltatások

▶ Hálózat

- ▶ virtuális hálózatok “mozgó” konténerek és VM-ek között
- ▶ fizikailag elosztott felhők között



Drone control from private cloud



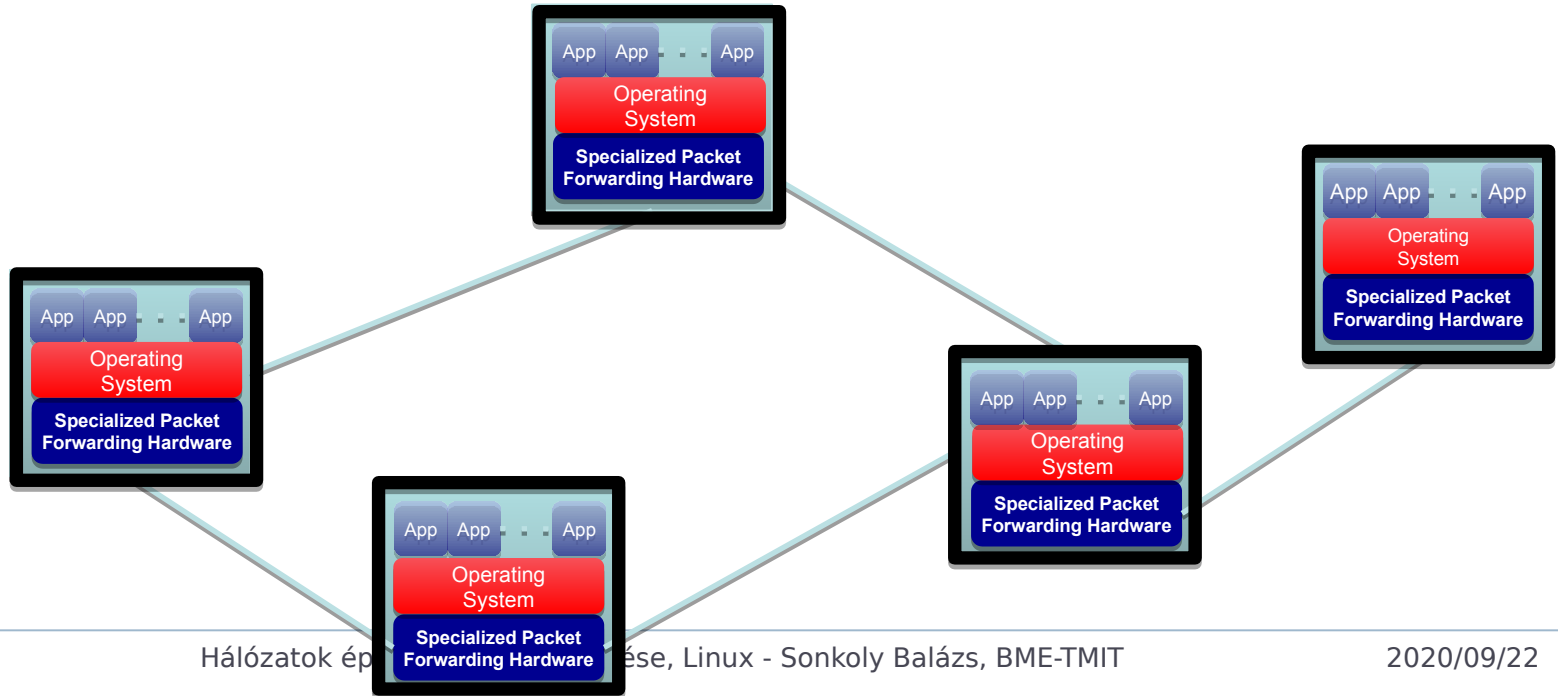
Milyen hálózat kell?

- ▶ Pl.: számítógép-hálózatok minél “jobb” programozhatósága
- ▶ programozhatóság
 - ▶ hálózat mint egész működésének meghatározása
 - ▶ több mint az egyes elemek működésének befolyásolása
 - ▶ konfigurálás ↔ programozás
 - ▶ időskála!
- ▶ “jobb”
 - ▶ könnyebb, gyorsabb
 - ▶ flexibilisebb
 - ▶ szélesebb körű
 - ▶ kevesebb hiba(lehetőség)
 - ▶ gyorsabb javíthatóság

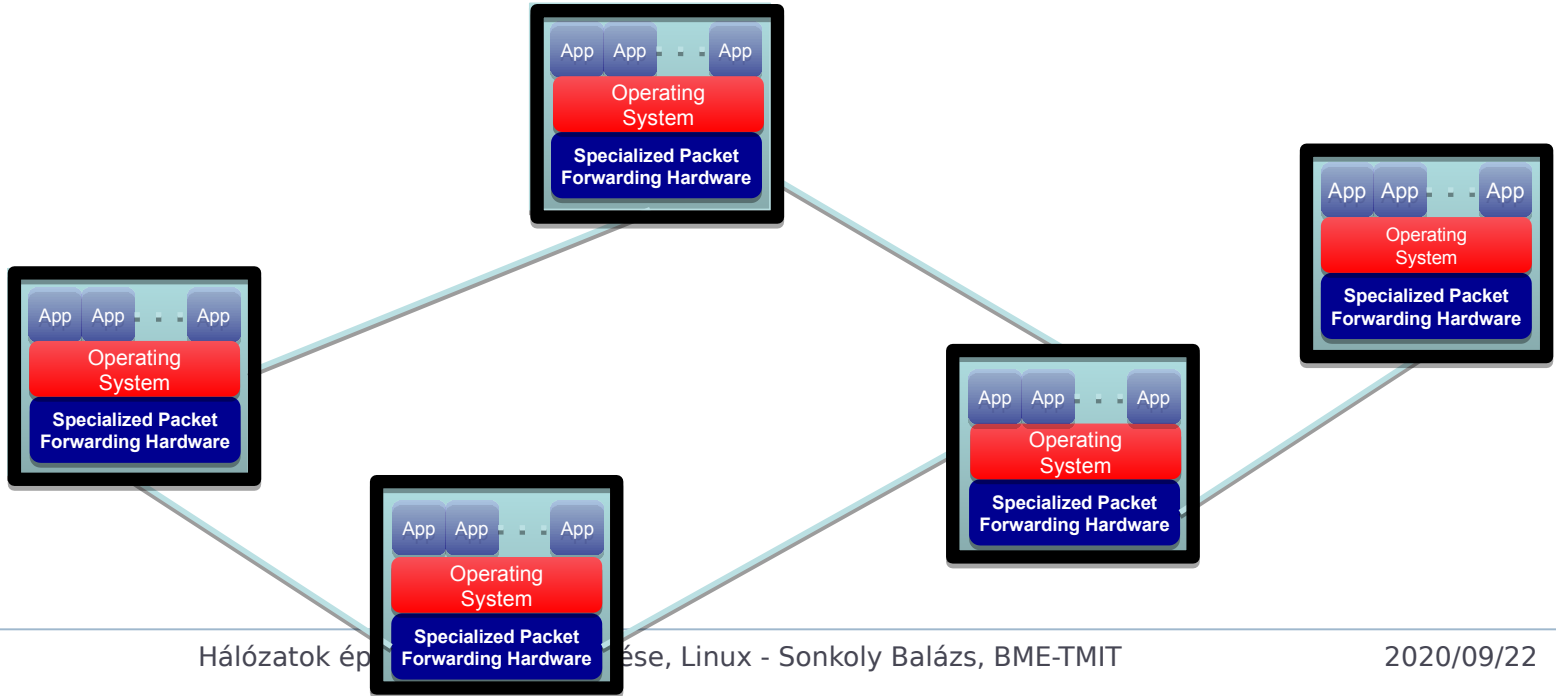
(Egy) Megoldás(i irány)

- ▶ **Software Defined Networking**
 - ▶ kontrollsík szoftverizálása
 - ▶ kontrollsík ↔ adatsík szeparálása
 - ▶ kontroll: döntés hogy mi történjen az adott forgalommal
 - ▶ adatsík: csomagok továbbítása
 - ▶ kontrollsík centralizálása / konszolidálása / egységesítése
 - ▶ közöttük: nyílt interfész(ek)
 - ▶ korábban: elosztott rendszer, „sok-sok” kapcsolat
 - ▶ most: elosztott rendszer, „egy-sok” kapcsolat!
- ▶ **Egy népszerű realizáció: OpenFlow**
 - ▶ kontroll-adat szeparálás +
 - ▶ hálózati eszköz általánosítása (absztrakció)
 - ▶ műveletek általánosítása (bizonyos mértékben)
 - ▶ új koncepció: hálózati operációs rendszer

Internet ma: zárt infrastruktúra



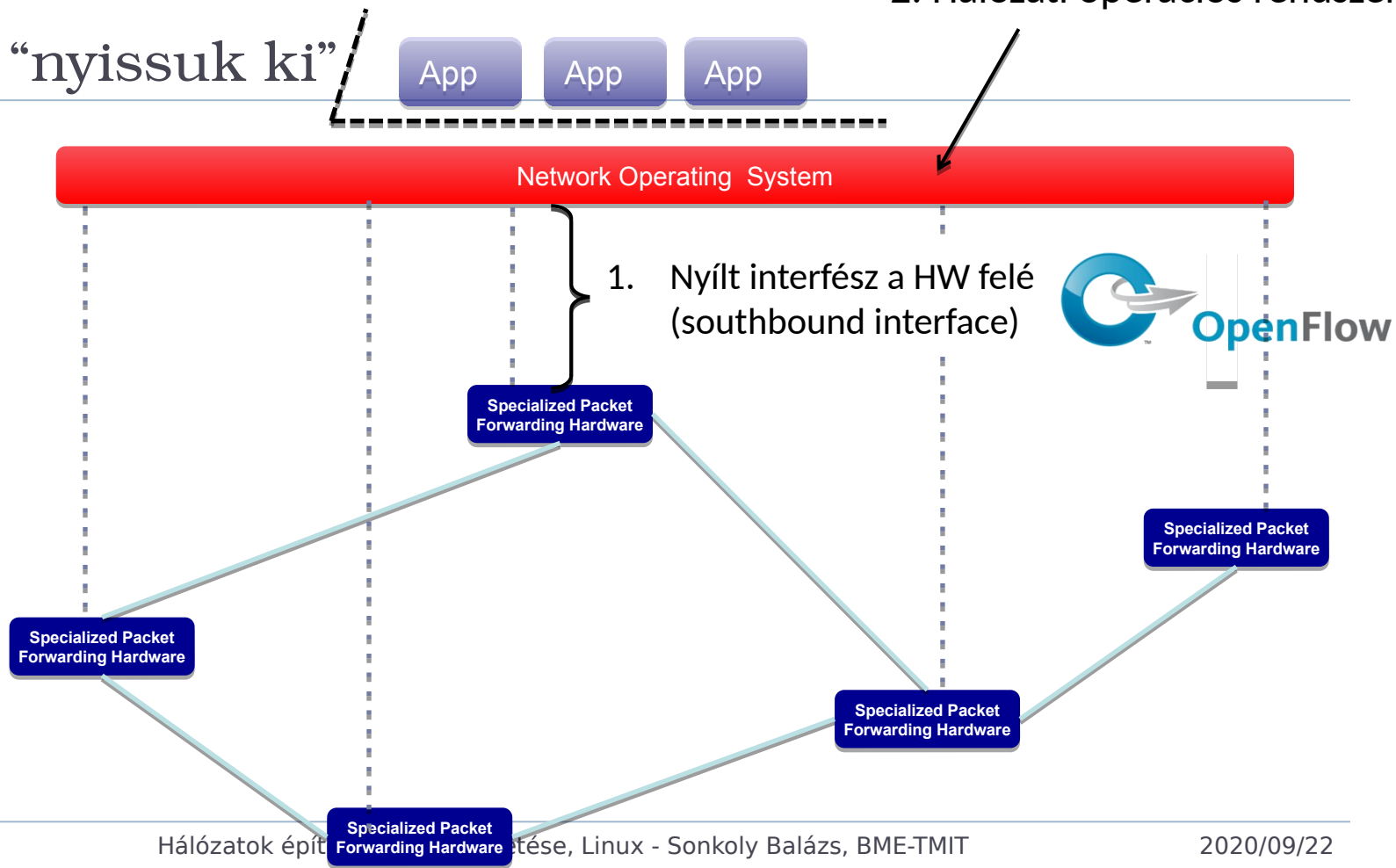
SDN: “nyissuk ki”



3. Jól definiált API (northbound interface)

2. Hálózati operációs rendszer

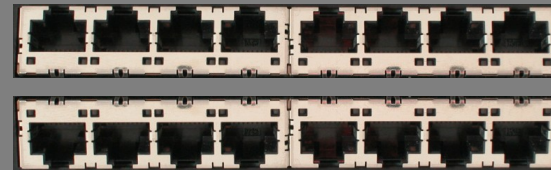
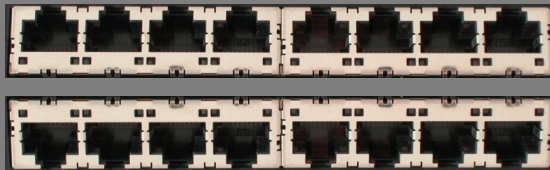
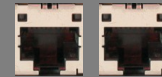
SDN: “nyissuk ki”



Mi is az az OpenFlow?

- ▶ OpenFlow egy API, interfész
- ▶ Ezen keresztül kontrollálható a csomag-továbbítás (forwarding)
- ▶ olcsó HW-en is implementálható
- ▶ Üzemeltetett hálózat programozható lesz
 - ▶ nem csak konfigurálható!
- ▶ Egyszerűbb innováció
- ▶ (egyszerűbb üzemeltetés, új szolgáltatások bevezetése)
- ▶ **Fő célok**
 - ▶ Ne kelljenek speciális testbedek
 - ▶ Kísérleti megoldások **valós hálózaton, valós forgalom** mellett, **vonali sebességen**

Ethernet Switch



Control Path (Software)

Data Path (Hardware)

OpenFlow Controller

OpenFlow Protocol (SSL/TCP)



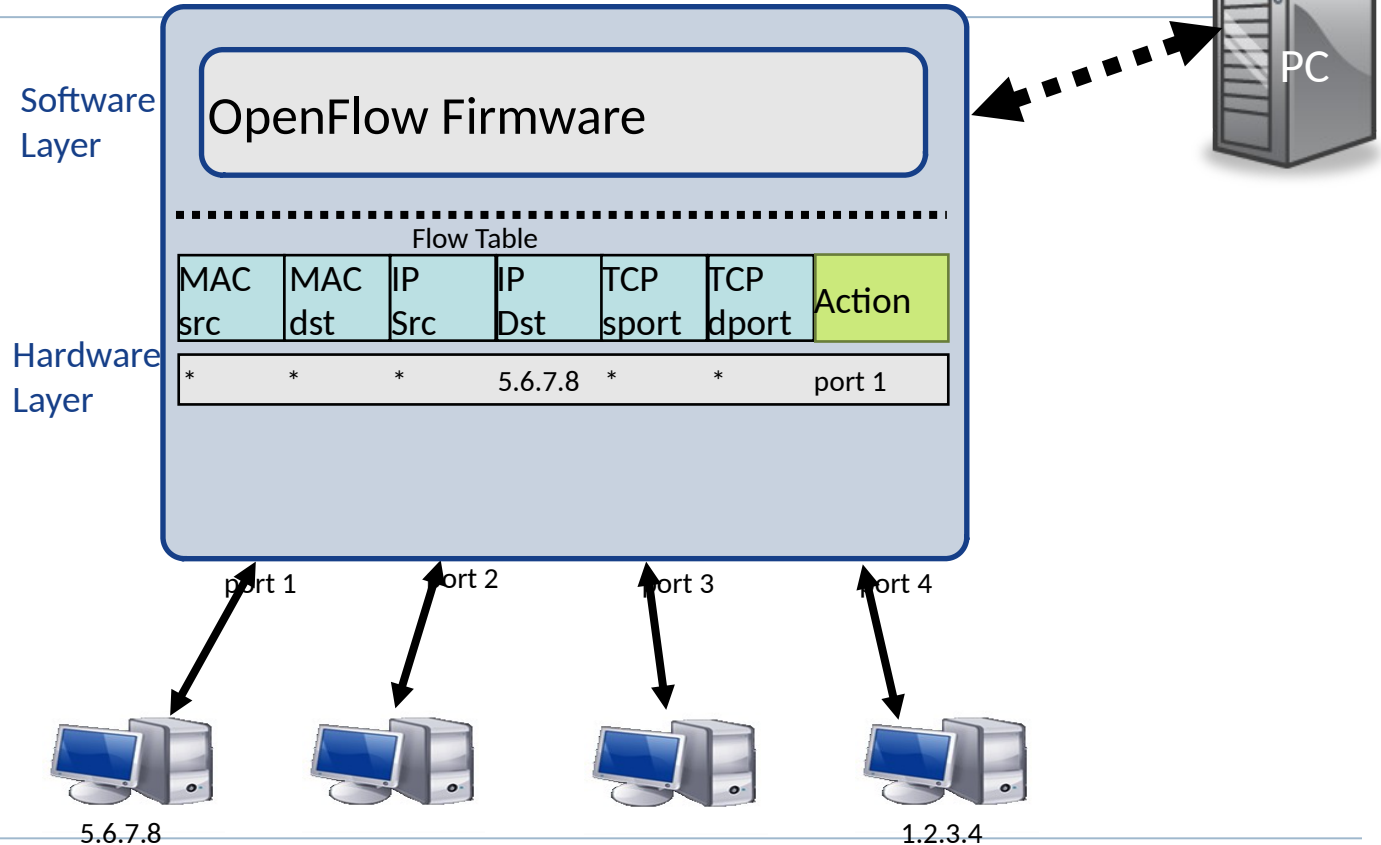
Control Path

OpenFlow

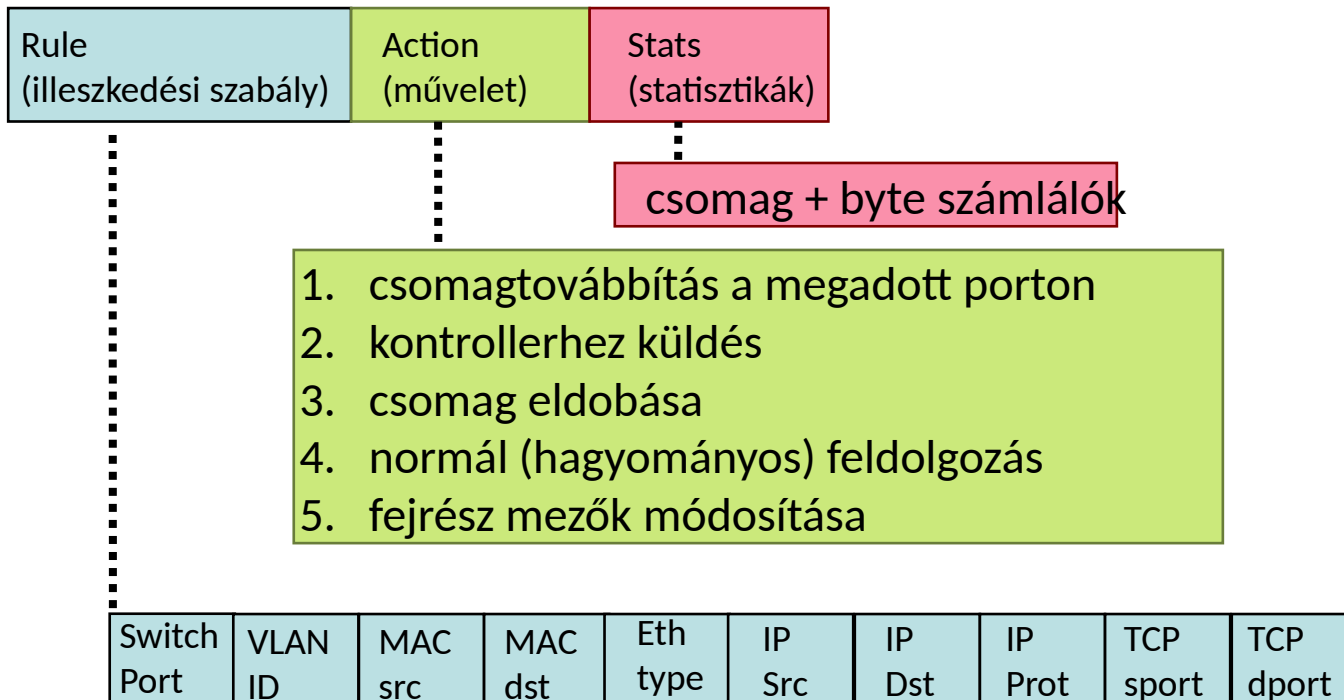
Data Path (Hardware)

OpenFlow flow tábla absztrakció

Controller



Flow tábla bejegyzések



+ a nem szükséges mezők maszkolhatók (wildcard)

Flow tábla bejegyzések: példák

Switching (L2 kapcsolás)

Switch Port	MAC src	MAC dst	Eth type	VLAN ID	IP Src	IP Dst	IP Prot	TCP sport	TCP dport	Action
*	*	00:1f:..	*	*	*	*	*	*	*	port6

Routing (L3 útvonalválasztás)

Switch Port	MAC src	MAC dst	Eth type	VLAN ID	IP Src	IP Dst	IP Prot	TCP sport	TCP dport	Action
*	*	*	*	*	*	5.6.7.8	*	*	*	port6

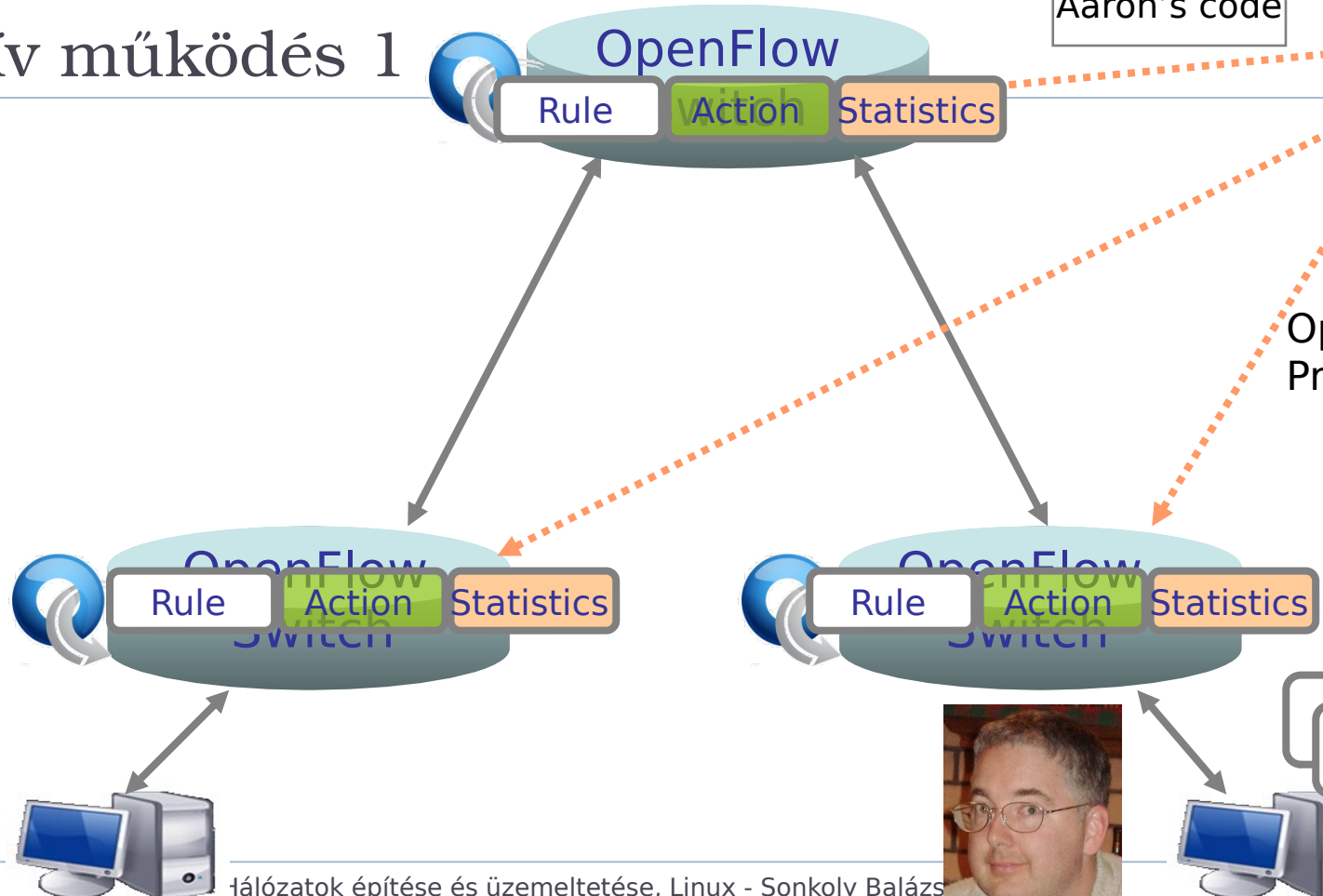
VLAN Switching

Switch Port	MAC src	MAC dst	Eth type	VLAN ID	IP Src	IP Dst	IP Prot	TCP sport	TCP dport	Action
*	*	00:1f..	*	vlan1	*	*	*	*	*	port6, port7, port9

Aaron's code



Reaktív működés 1



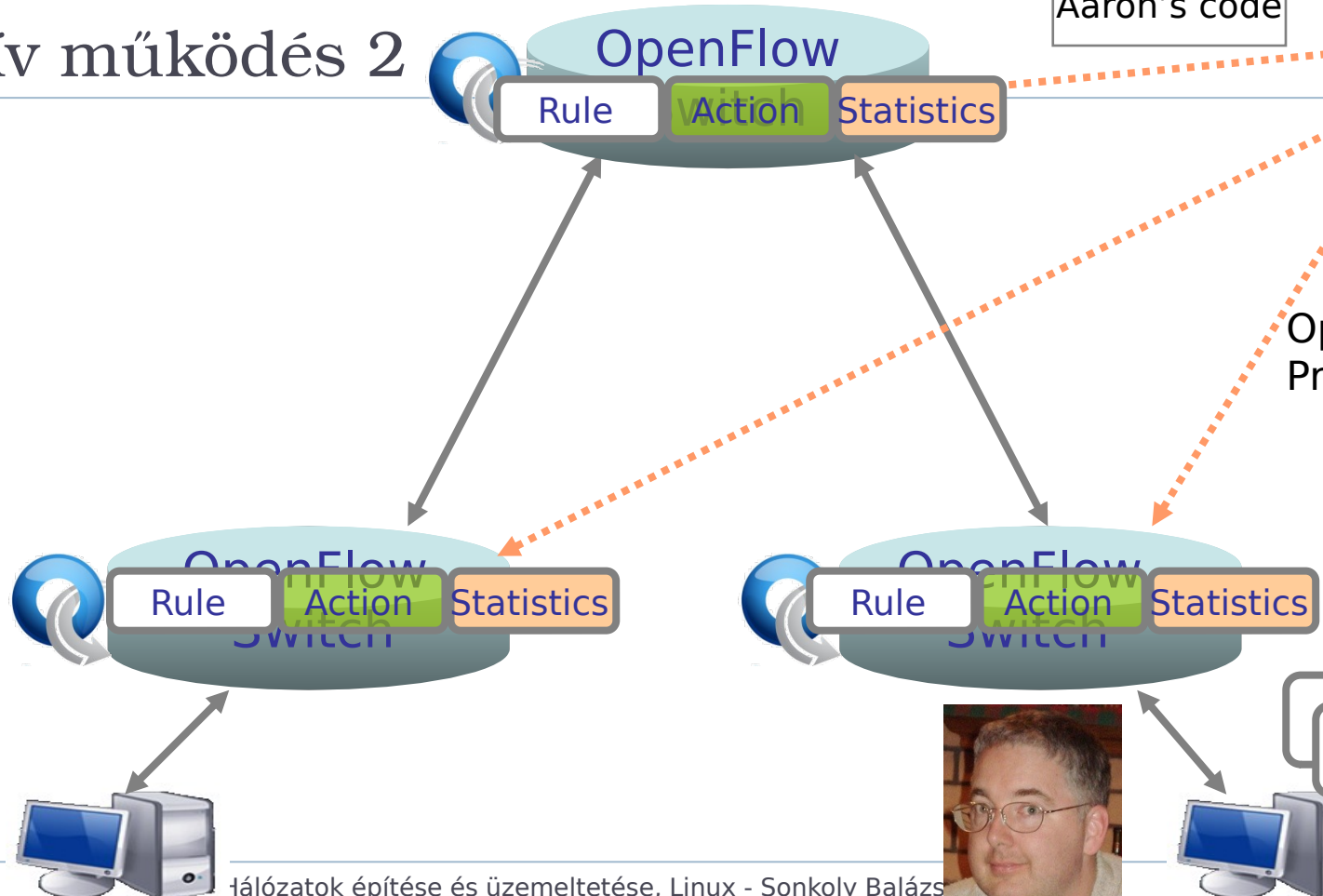
OpenFlow Protocol



Aaron's code



Reaktív működés 2



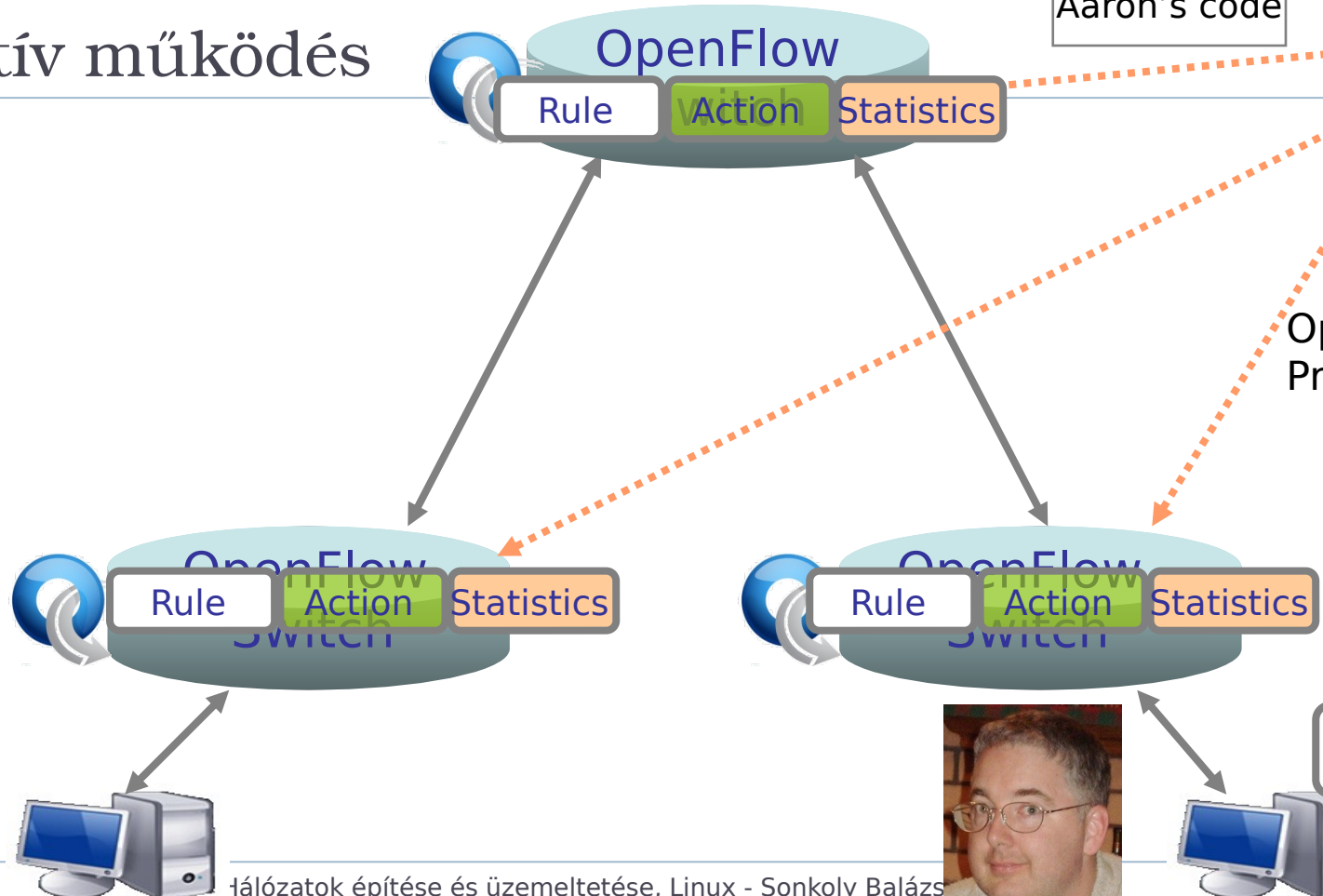
OpenFlow Protocol



Aaron's code



Proaktív működés



SFC

- ▶ Service Function Chaining
- ▶ Nem új koncepció
- ▶ SDN előretörésével fókuszba került
- ▶ Tipikus szolgáltatás
 - ▶ hálózati funkciók végrehajtása
 - ▶ adott sorrendben
 - ▶ adott forgalomra
 - ▶ csomagok irányítása a funkciót megvalósító blokkok között
- ▶ Absztrakció
 - ▶ service chain vagy service graph
 - ▶ magas szintű szolgáltatások generikus leírására
 - ▶ milyen típusú forgalom/felhasználó
 - ▶ milyen elemi szolgáltatások (vagy hálózati funkciók)
 - ▶ milyen sorrendben

