What to do with ARM devel board?
ARM board examples

- there is plenty of boards
- all depends on your use-case
- prices from 15 USD to thousands of USD
- some are really easy to get
- some are harder - have to be shipped
ARM board examples

Raspberry Pi 1/0

- armv6
- 1 core @ 700 MHz
- 256M / 512M RAM
- USB connected 100M Ethernet
- ~ 940 Kč
ARM board examples

RaspberryPi 2

- armv7
- 4 cores @ 900 MHz
- 1G RAM
- USB connected 100M Ethernet
- ~ 1 350 Kč
ARM board examples

BananaPi

- armv7
- 2 cores @ 1 GHz
- 1G RAM
- native 1Gbit Ethernet
- SATA
- ~ 1 090 Kč
ARM board examples

RaspberryPi 3

- armv8
- 4 cores @ 1.2 GHz
- 1G RAM
- USB connected 100M Ethernet
- currently running only 32bit
- WiFi and BT
- ~ 35 USD
ARM board examples

**Pine64**

- armv8
- 4 cores @ 1.2 Ghz
- 512M - 2G RAM
- 100Mbit/1Gbit Ethernet
- optionally with WiFi and BT
- currently on the way
- 15 - 29 USD
ARM board examples

Turris Omnia

- armv7
- 2 cores @ 1.6 GHz
- 1G - 2G RAM
- 3 native 1Gbit Ethernets
- VLAN configurable switch
- 2 USB 3.0
- mSATA
- 4G flash
- 3 miniPCI
- 119 - 278 USD
Compilation cluster

- some boards are weak (RaspberryPi 1/0)
- some boards are stronger

Icecream to the rescue

- way to distribute build job
- you can use all your ARM boards
- you can even mix architectures
  - `icecc --build-native`
- in theory you can throw in x86 as well
NAS use-case

NFS, sshfs, samba, FTP

- plain stupid network drive
- every NAS can do that

rsync

- much more useful for backups

rsync + cron

- nightly backups to different datacenter
- not so easy with off-the-shelf NAS
NAS use-case

RAID

- decide RAID level by yourself
- encrypt your harddrives

Naive network cluster storage

- there is something called nbd
- you don’t want to do this
- but you can ;-)
NAS use-case

**gluster**

- real distributed network storage
- easy to configure
- can do "RAID 0" and "RAID 1" and combinations
- works on top of existing filesystem
  - if it breaks, you don’t lose data
  - you can have your RAID1 sharing space with RAID0
- can be used as storage for KVM
NAS use-case

ownCloud

- BFU friendly NAS
- can be your calendar
- can be your contacts list
- can be your webmail client
- can be your multimedia center
Multimedia center

- most boards do have HW video decoders
- usually requires ancient kernel
- quite often requires use of board specific API
- video - various bad options
  - Kodi
  - VLC
  - ownCloud (OCTV) 😞

- audio
  - mpd
  - pulseaudio
GPIO, I2C & friends

- GPIO = General-purpose input/output
- I2C - simple bus
- you can attach other HW
  - T&H sensor
  - motion sensor
  - LED
  - remote switches
  - ...
Simple security system

- attach motion sensor to GPIO ~ 60 Kč
- attach WebCam to USB ~ 300 Kč
- take a picture whenever motion sensors triggers
- upload it to remote location
- can have multiple sensors and webcams

Easy DIY project :-)
"Router" use-case

- don’t need to be the real router
- it is better if it is a router
- basically network "server"
- much more CPU power and RAM than typical router
  - you can run moooore
  - and it will run reasonably fast

**PXE**

- boot recovery/installation over the network
"Router" use-case

OpenVPN server
- if you have a public IP, easy to setup
- can be running on 443 port

DNS resolver
- you can validate DNSSEC
- you can forward subdomains
- you can add your local names
- you can do DNS64
- examples
  - knot-dns-resolver
  - unbound
"Router" use-case

**IPv6 connectivity**
- can be endpoint of tunnel
- can allocate IPs for your network
- can connect you to your other networks

**NAT64 gateway**
- allows access to IPv4 via IPv6
- you can build yourself IPv6 only network

**MIPv6**
- get your favorite IPv6 everywhere
Virtualization

KVM

- Cortex-A15, armv8
- basically you have to pick the right HW

LXC

- everywhere
- friendlier to your RAM
- easy to get started
Questions and your ideas?