How Docker didn't invent containers

Docker Meetup Prague

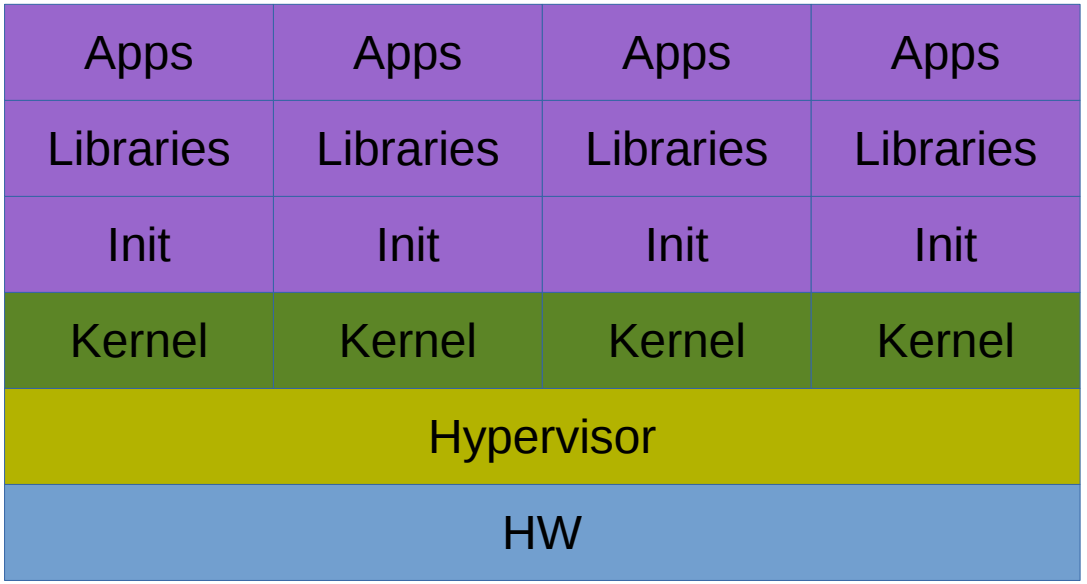
Pavel Šnajdr
vpsFree.cz

Contents

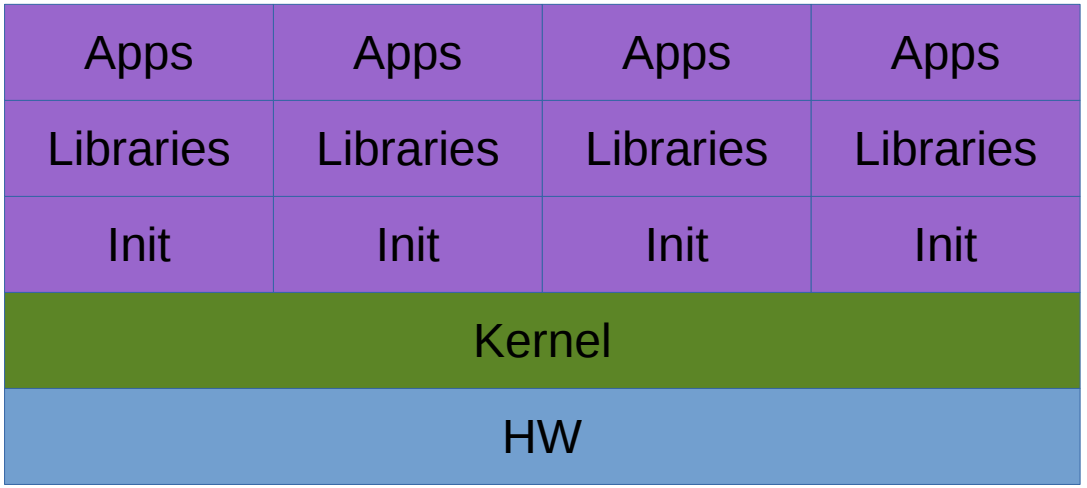
whoami

- vpsFree.cz
  - Container based VPS community/provider
  - Founder, admin, base OS dev/QA
  - Now full-time
- Formerly Relbit CTO
- Working with containers in prod since 2009

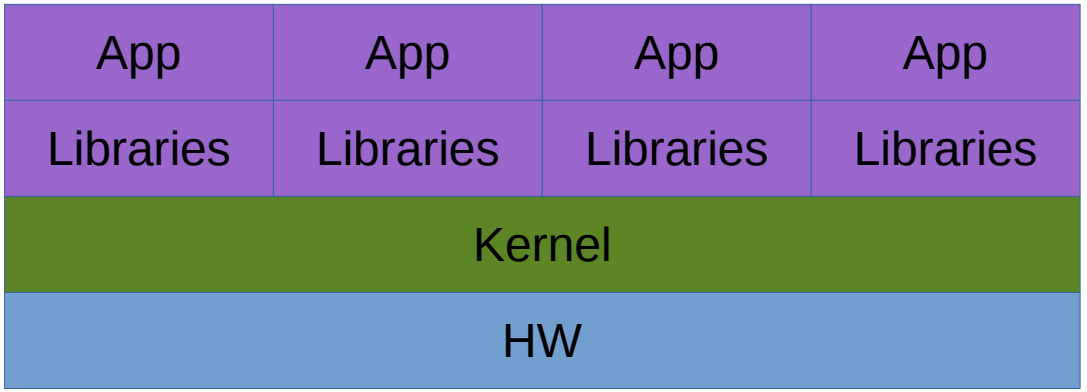
Hypervisors



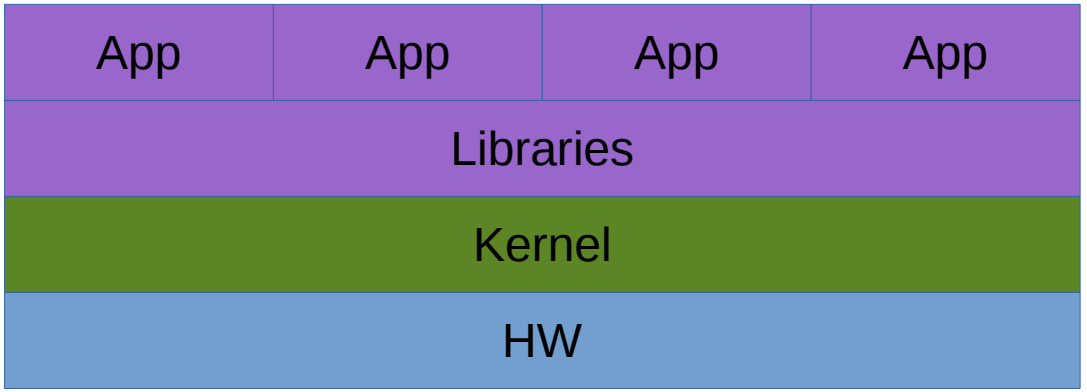
OS Containers



Application containers



Application containers



vpsFree.cz

1998: FreeBSD Jails

- FreeBSD 4.0
- Resource management very limited
- Origins: a small webhosting company

2001: SWSoft Virtuozzo 2.0

- Started in 1999
  - Groups of processes with namespace isolation
  - FS to share code and save RAM
  - Resources management
- SWSoft -> Parallels
- Also in 2001 linux-vserver
  - Mostly one-man show

2004

- Solaris Containers released (“Zones”)
- Virtuozzo for Windows
  - Live kernel patching

2005

- OpenVZ project
  - Namespaces (PID, mnt, UTS, net, IPC)
  - UBC
  - vzquota
  - vzctl

- Goal set to upstream containers
  ("better late than never")

OpenVZ

2006

- OpenVZ live migration
- Rebase to RHEL4 kernel

2007

- IBM AIX WPARs, HP-UX SRP containers
- OpenVZ rebase to RHEL5
  - Also 2.6.20 port
- cgroups upstreamed (Google & IBM)

2008

- OpenVZ namespaces upstreaming
  - PID, net, IPC, UTS, mnt
- LXC

2010

- OpenVZ
  - Rebase to RHEL6 kernel
  - VSwap (simplified UBC)
  - ploop (CT-in-a-file)
    - on-demand allocation
    - instant snapshots
    - online resize, compact, merge
    - write tracker

2011

- CRIU proposed
  - Checkpoint: get stuff from /proc debug fac.
  - Restore: read dump & recreate environment

- LinuxCon 2011 Prague
  - “There can be only one”
    ... container tech in vanilla
  - Avoid Xen vs. KVM mess

2012

- CRIU 0.1 released
- vzctl 4.0 with support for upstream kernel

vpsFree.cz

2013

- Docker
- lmctfy
- CoreOS
- vzctl adds IO limits
- user namespace in vanilla

2014

- vzctl 4.8, faster live migration
- Parallels announce PCS and OpenVZ to merge into common open-source code-base

2015

- OpenVZ RHEL7 kernel beta
  - CRIU for migration
  - cgroups replacing UBC
  - vzctl not compatible yet
  - public Git repo
    http://src.openvz.org/

OpenVZ and Docker

- Docker inside
  https://openvz.org/Docker_inside_CT
- Docker outside
  https://github.com/docker/libcontainer/pull/434
- Docker and CRIU
  - Merged in libcontainer

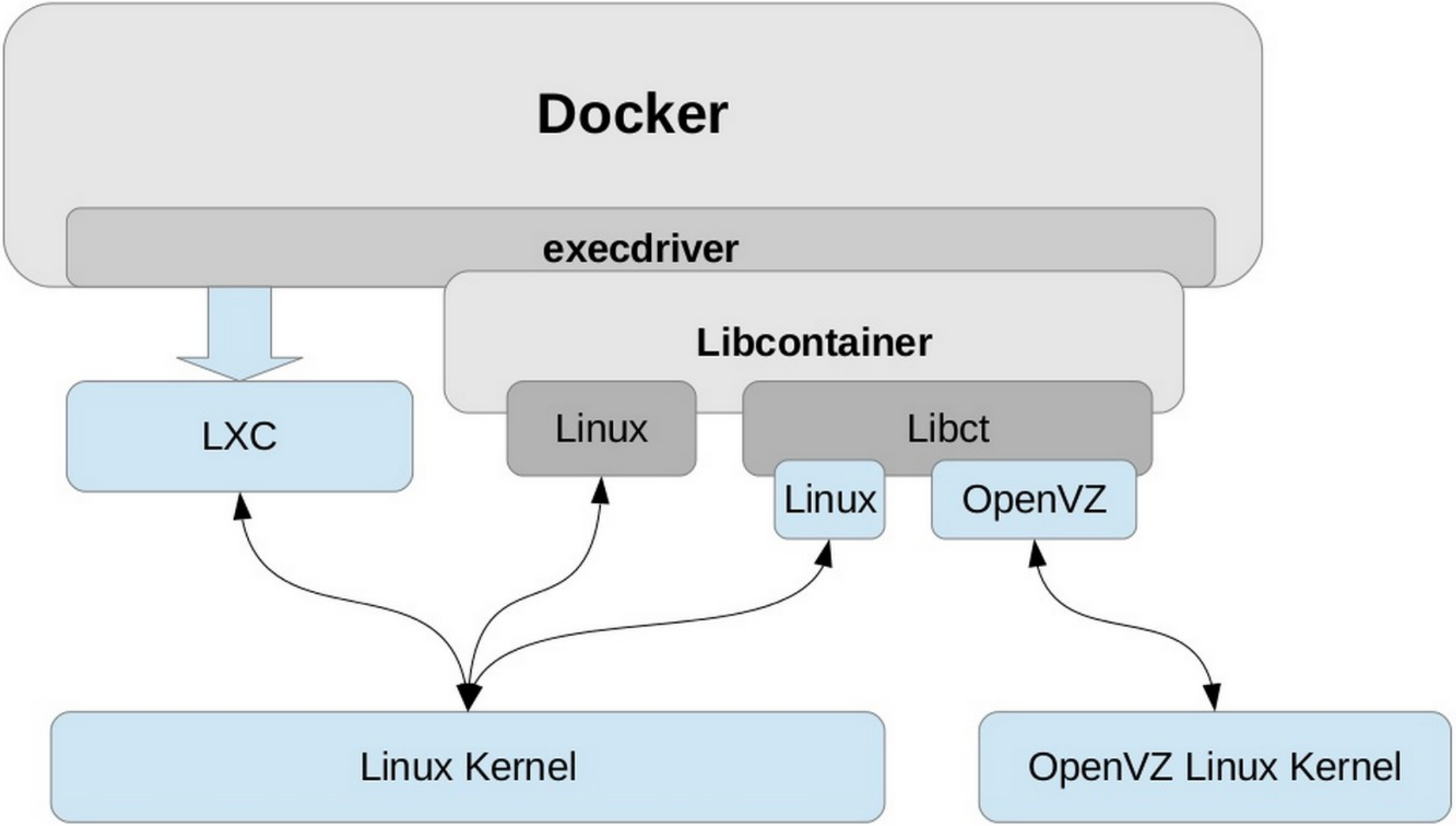
Containers in vanilla kernel

- Any of (cgroups + namespaces) = container
- Cgroups (blkio, cpu, cpuacct, cpuset, devices, freezer, ns)
- Namespaces (user, net, PID, UTS, mnt, IPC)

Container management tools

- LXC
- LXD
- Docker
- libvirt-lxc
- systemd-nspawn
- vzctl
- lmctfy
- libct
  https://github.com/xemul/libct
  - “Libvirt for containers”

Docker and containers



Conclusions

- Containers != Docker
- Docker = single (very) limited way of using container tech for apps deployment
  - Reinvent the wheel approach... (PID #1, logs...)

- Most mature container tech = OpenVZ
  - Best isolation (eg. Kmem)
  - Most features

- Do you actually want Docker or containers?

Q/A

- Questions?

  Anytime -> snajpa@snajpa.net