

2014-06-20 17:30 Room: Degerloch

Webservices deklarativ beschreiben mit NixOS (teil 2)

Joachim Schiele

Blog:

blog.lastlog.de

Wiki:

lastlog.de

eMail:

js@lastlog.de

irc:

[@nixos#irc.freenode.org](https://freenode.net/chat/qknight)



definition:

declarative programming

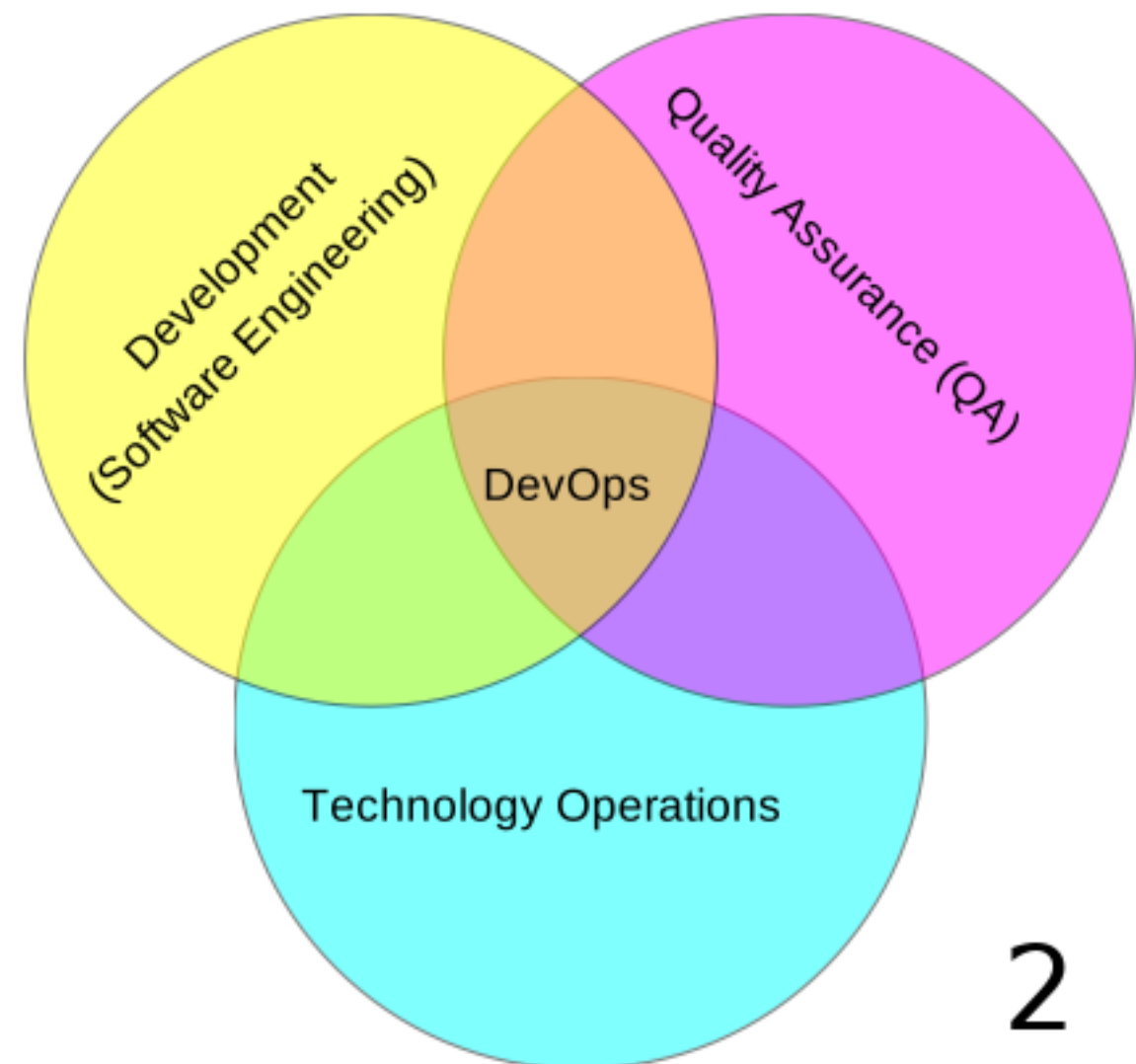
In computer science, **declarative programming** is a **programming paradigm**, a style of building the structure and elements of computer programs, that expresses the logic of a computation without describing its control flow.[1]

...

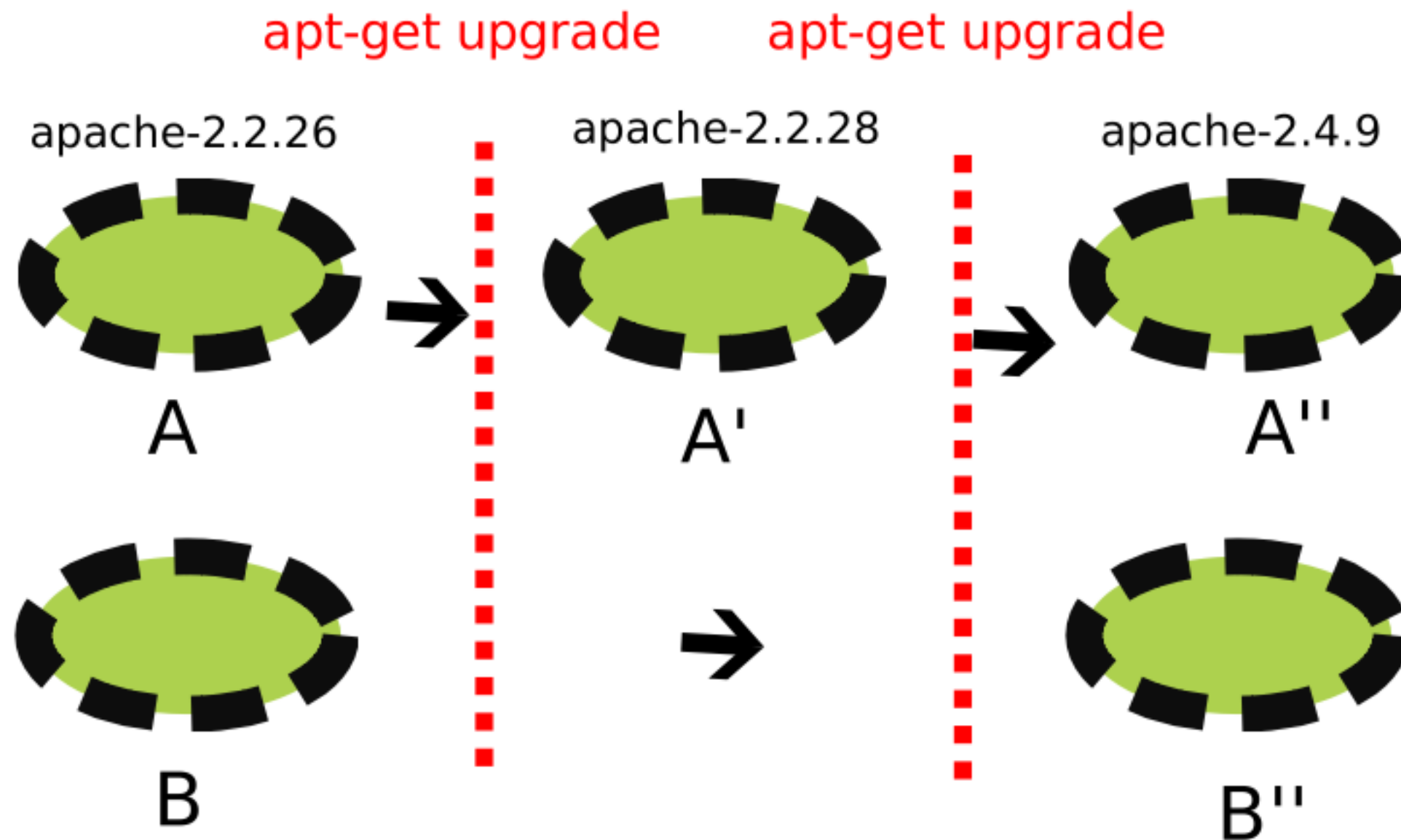
Common declarative languages include those of database query languages (e.g., **SQL**, **XQuery**), **regular expressions**, **logic programming**, **functional programming**, and **configuration management systems**.

definition: devOps

DevOps (a portmanteau of **development** and **operations**) is a software development method that stresses communication, collaboration and integration between software developers and information technology (IT) operations professionals.



was bedeutet **stateful** in bezug auf PM?

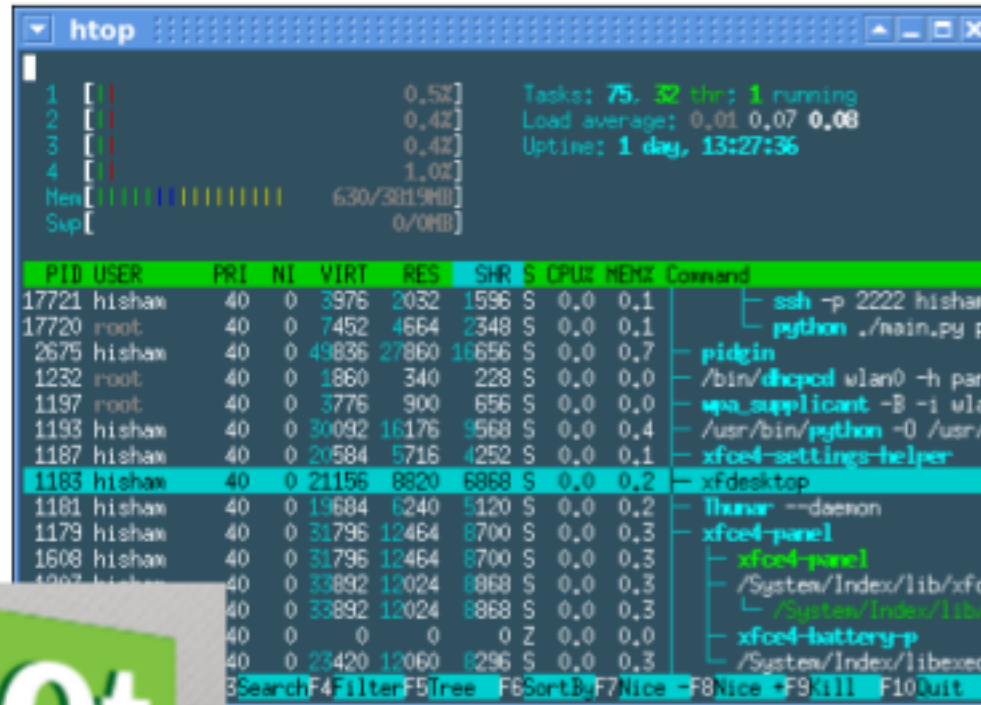


upgrades sind stateful, wenn gilt $A'' \neq B''$
beispiel: leftovers in /etc/

PM-usage

systemprogrammierung, scripting

- tools
- GUI



htop

Tasks: 75. 32 ttr: 1 running
Load average: 0.01 0.07 0.08
Uptime: 1 day, 13:27:36

Mem[|||||] 630/3819MB
Swp[|] 0/0MB

PID	USER	PRI	NI	VIRT	RES	SHR	S	CPU%	MEM%	Command
17721	hisham	40	0	3976	2032	1596	S	0.0	0.1	ssh -p 2222 hisham
17720	root	40	0	7452	4664	2348	S	0.0	0.1	python ./main.py p
2675	hisham	40	0	49836	27860	16656	S	0.0	0.7	pidgin
1232	root	40	0	1860	340	228	S	0.0	0.0	/bin/dhclient wlan0 -h par
1197	root	40	0	3776	900	656	S	0.0	0.0	wpasupplicant -B -i ula
1193	hisham	40	0	30092	16176	9568	S	0.0	0.4	/usr/bin/python -0 /usr/
1187	hisham	40	0	20584	5716	4252	S	0.0	0.1	xfce4-settings-helper
1183	hisham	40	0	21156	8820	6868	S	0.0	0.2	xfdesktop
1181	hisham	40	0	19684	6240	5120	S	0.0	0.2	thunar --daemon
1179	hisham	40	0	31796	12464	8700	S	0.0	0.3	xfce4-panel
1608	hisham	40	0	31796	12464	8700	S	0.0	0.3	xfce4-panel
4903	hisham	40	0	33892	12024	8868	S	0.0	0.3	/System/Index/lib/xfc
		40	0	33892	12024	8868	S	0.0	0.3	/System/Index/lib/
		40	0	0	0	0	Z	0.0	0.0	xfce4-battery-p
		40	0	23420	12060	8296	S	0.0	0.3	/System/Index/libexec

Search F4 Filter F5 Tree F6 Sort By F7 Nice F8 Nice F9 Kill F10 Quit



NGINX

webservices
benötigt **webserver** wie:
- apache/lighttpd/nginx
oder

language spezifischen webserver

diskussion PMs

probleme:

- DL(distribution-level)-PM **reden nicht mit** LL(language-level)-PM
- es wird viel von hand (stateful) gemacht
 - > u.a. gibt es deshalb **docker**
- sicherheitsprobleme: **bundled software!**
- deployment issues: **dependencies!**
- **verwaltungsalbtraum,**
z.b. bei upgrades

lösungsansätze um DL und LL PM wieder zu vereinen

lösungsansätze

configuration
management



NixOS/Disnix



lösungsansätze

containerization

LXC - Linux Containers



virtualisierung

configuration management



SALTSTACK



Puppet



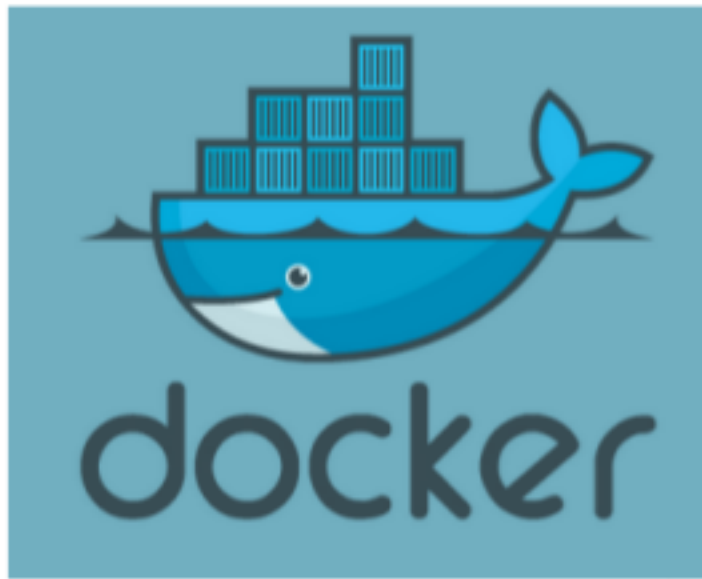
NixOS/Disnix



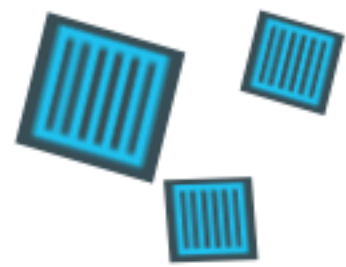
NixOps

containerization

LXC - Linux Containers



virtualisieru8g



docker vs. NixOS

stateful

good language
support

NO INIT!!



docker

size!



stateless

language PM
is a problem

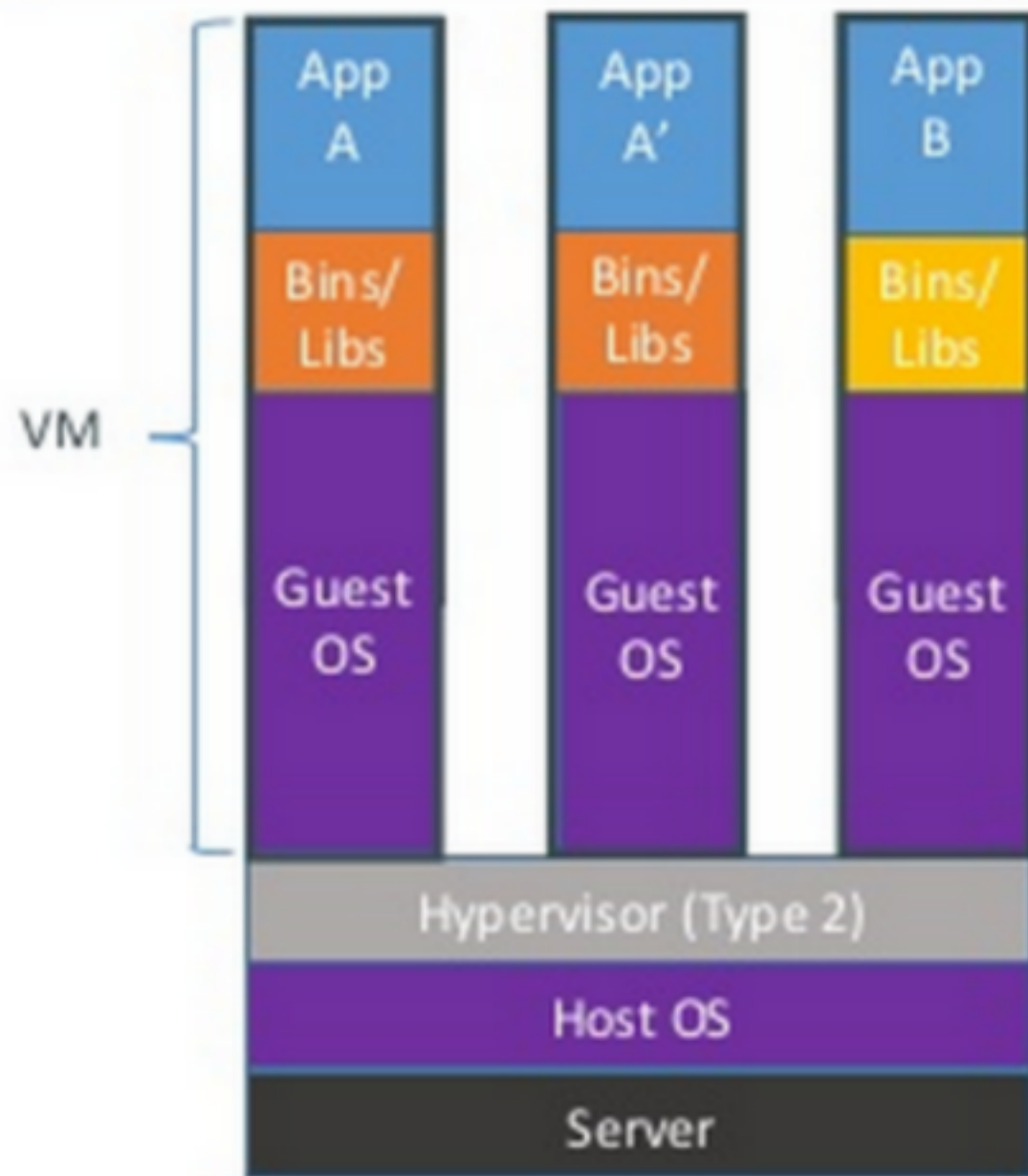
write once,
deploy everywhere

shared store!

NixOS



Containers vs. VMs



Containers are isolated, but share OS and, where appropriate, bins/libraries



source (trifork blog):

<http://blog.trifork.com/2013/08/08/next-step-in-virtualization-docker-lightweight-containers/>

umweltverschmutzung



(cc) Pascal Willuhn - Creative Commons Attribution 2.0 Generic I

NixOps



NixOps is a tool for **deploying NixOS machines in a network or cloud**. It takes as input a declarative specification of a set of “logical” machines and then performs any necessary steps/actions to realise that specification: instantiate cloud machines, build and download dependencies, stop and start services, and so on.

<http://nixos.org/nixops/manual/>

workshop

NixOps example

Example 3.1. trivial.nix: logical network specification

```
{
  network.description = "Web server";

  webserver =
    { config, pkgs, ... }:
    { services.httpd.enable = true;
      services.httpd.adminAddr = "alice@example.org";
      services.httpd.documentRoot = "${pkgs.valgrind}/share/doc/valgrind/html"
    };
}
```

Example 3.2. trivial-vbox.nix: VirtualBox physical network specification

```
{
  webserver =
    { config, pkgs, ... }:
    { deployment.targetEnv = "virtualbox";
      deployment.virtualbox.memorySize = 1024; # megabytes
    };
}
```

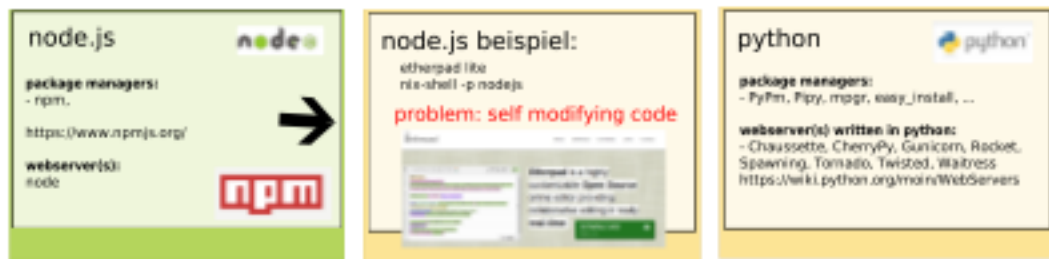
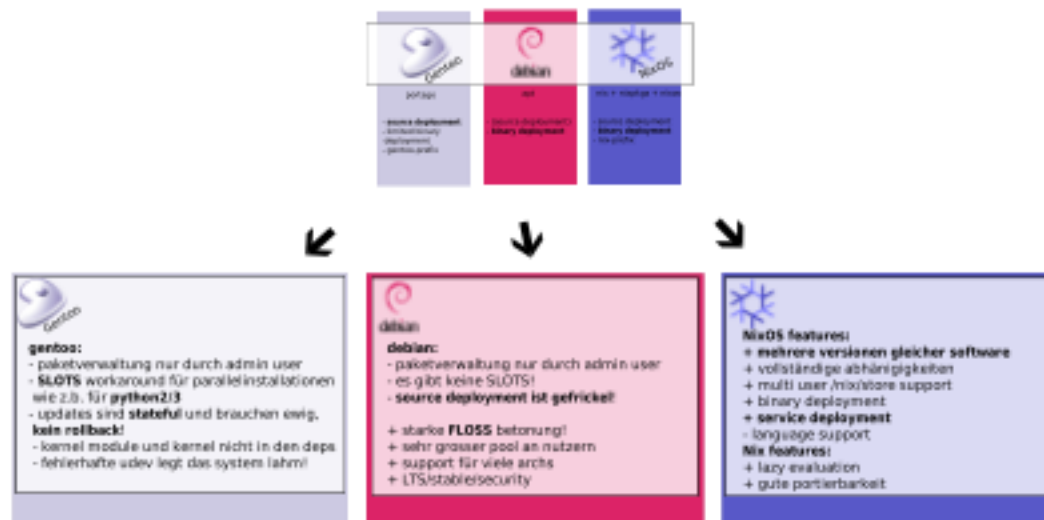
nixops deploy -d trivial



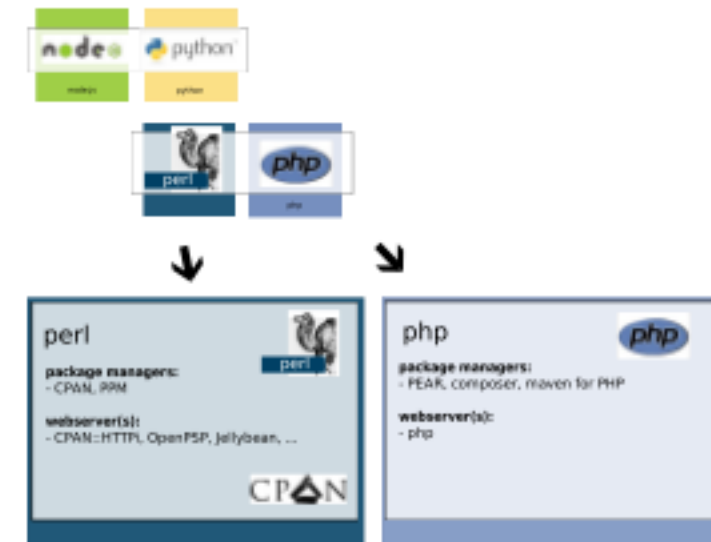
destinations₁₃

distribution-level PM VS language-level PM

distribution-level PM im vergleich



language-level PM[↑] im vergleich



distribution-level PM im vergleich

 Gentoo	 debian	 NixOS
portage	apt	nix + nixpkgs + nixos
- source deployment - limited binary deployment - gentoo-prefix	- (source deployment) - binary deployment	- source deployment - binary deployment - nix-prefix




Gentoo

gentoo:

- paketverwaltung nur durch admin user
- **SLOTS** workaround für parallelinstallationen wie z.b. für **python2/3**
- updates sind **stateful** und brauchen ewig, **kein rollback!**
- kernel module und kernel nicht in den deps
- fehlerhafte udev legt das system lahm!


debian

debian:

- paketverwaltung nur durch admin user
- es gibt keine SLOTS!
- **source deployment ist gefrickel!**

+ starke **FLOSS** betonung!
+ sehr grosser pool an nutzern
+ support für viele archs
+ LTS/stable/security



NixOS features:

- + **mehrere versionen gleicher software**
- + vollständige abhängigkeiten
- + multi user /nix/store support
- + binary deployment
- + **service deployment**
- language support

Nix features:

- + lazy evaluation
- + gute portierbarkeit



portage

- **source deployment**
- limited binary deployment
- gentoo-prefix



apt

- (source deployment)
- **binary deployment**



nix + nixpkgs + nixos

- source deployment
- **binary deployment**
- nix-prefix



gentoo:

- paketverwaltung nur durch admin user
- **SLOTS** workaround für parallelinstallationen wie z.b. für **python2/3**
- updates sind **stateful** und brauchen ewig, **kein rollback!**
- kernel module und kernel nicht in den deps
- fehlerhafte udev legt das system lahm!



debian

debian:

- paketverwaltung nur durch admin user
 - es gibt keine SLOTS!
 - **source deployment ist gefrickel!**
-
- + starke **FLOSS** betonung!
 - + sehr grosser pool an nutzern
 - + support für viele archs
 - + LTS/stable/security



NixOS features:

- + mehrere versionen gleicher software**
- + vollständige abhangigkeiten
- + multi user /nix/store support
- + binary deployment
- + service deployment**
- language support

Nix features:

- + lazy evaluation
- + gute portierbarkeit

node.js 

package managers:
- npm,

<https://www.npmjs.org/>

webserver(s):
node



node.js beispiel:

etherpad lite
nix-shell -p nodejs

problem: self modifying code



python 

package managers:
- PyPm, Pipy, mpgr, easy_install, ...

webserver(s) written in python:
- Chaussette, CherryPy, Gunicorn, Rocket, Spawning, Tornado, Twisted, Waitress
<https://wiki.python.org/moin/WebServers>

language-level PM im vergleich


node.js


python


perl


php

perl 

package managers:
- CPAN, PPM

webserver(s):
- CPAN::HTTPi, OpenPSP, Jellybean, ...



php 

package managers:
- PEAR, composer, maven for PHP

webserver(s):
- php



node.js



python



perl



php

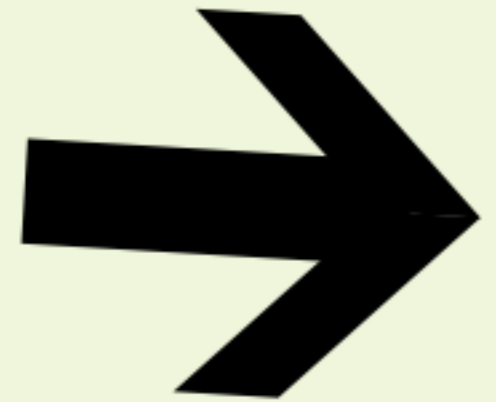
node.js



package managers:

- npm,

<https://www.npmjs.org/>



webserver(s):

node

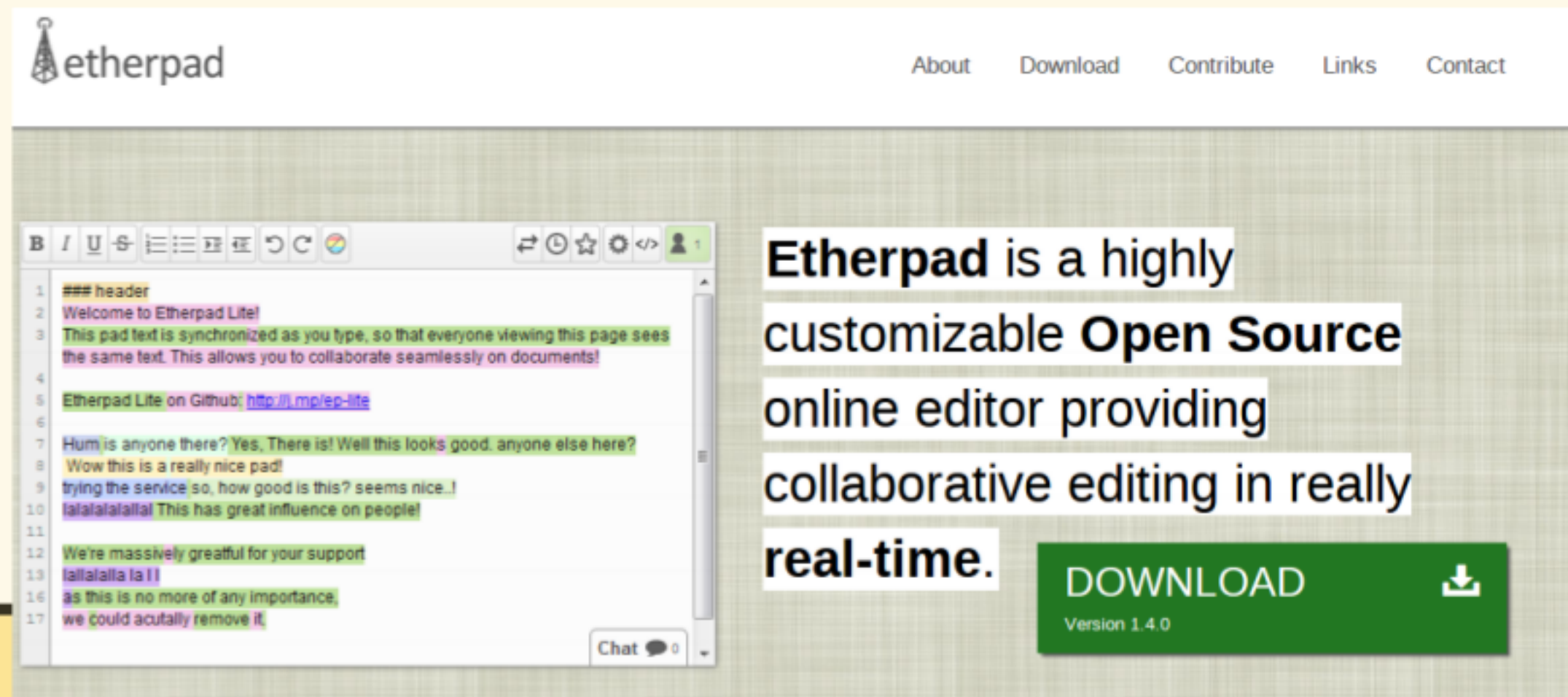


node.js beispiel:

etherpad lite

nix-shell -p nodejs

problem: self modifying code



The screenshot displays the Etherpad Lite web application. At the top, the 'etherpad' logo is on the left, and navigation links for 'About', 'Download', 'Contribute', 'Links', and 'Contact' are on the right. The main area features a text editor with a toolbar containing icons for bold, italic, underline, and other formatting options. The editor's content includes a header, a welcome message, a description of the collaborative editing feature, a link to the GitHub repository, and a series of chat messages. A 'Chat' button is visible at the bottom right of the editor. To the right of the editor, a text box describes Etherpad as a highly customizable Open Source online editor providing collaborative editing in real-time. Below this text is a green 'DOWNLOAD' button with a download icon and the text 'Version 1.4.0'.

etherpad

About Download Contribute Links Contact

Etherpad is a highly customizable **Open Source** online editor providing collaborative editing in really **real-time.**

DOWNLOAD Version 1.4.0

python



package managers:

- PyPm, Pipy, mpgr, easy_install, ...

webserver(s) written in python:

- Chaussette, CherryPy, Gunicorn, Rocket, Spawning, Tornado, Twisted, Waitress

<https://wiki.python.org/moin/WebServers>

perl



perl

package managers:

- CPAN, PPM

webserver(s):

- CPAN::HTTPi, OpenPSP, Jellybean, ...



php



package managers:

- PEAR, composer, maven for PHP

webserver(s):

- php

weiterführende dokus

NixOS container documentation:

<http://hydra.nixos.org/build/10339703/download/2/nixos/manual.html>

motivation zu diesem vortrag:

http://blog.lastlog.de/posts/advanced_webservices_on_nixos_revisited/

http://blog.lastlog.de/posts/advanced_webservices_on_nixos/

allg. zu NixOS/NixOPs:

<https://www.domenkozar.com/2014/03/11/why-puppet-chef-ansible-arent-good-enough-and-we-can-do-better/>

[http://www.admin-magazin.de/Das-Heft/2014/04/NixOS-und-NixOps/\(offset\)/8](http://www.admin-magazin.de/Das-Heft/2014/04/NixOS-und-NixOps/(offset)/8)

nix-docker:

<http://zef.me/6079/deploying-wordpress-using-nix-docker>

<http://zef.me/6049/nix-docker>

<http://zef.me/5981/deploying-a-simple-node-js-application-with-nixops>

mediawiki als service:

https://nixos.org/wiki/Setting_up_MediaWiki_%26_vhosts

service deployment

NixOS beschreibt auch services deklarativ

ein service benötigt:

- eine **nix expression (in nixpkgs)**
- eine **nix service expression (nixos)**



http://blog.lastlog.de/posts/advanced_webservices_on_nixos/

https://nixos.org/wiki/Setting_up_MediaWiki_%26_vhosts

http://blog.lastlog.de/posts/system_services_on_nixos/

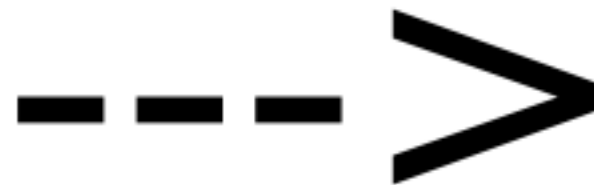


rechtlicher hinweis:

teile dieser präsentation wurden aus dem internet
ausgeliehen, werden aber nach beendigung zurückgegeben!



fragen



shutdown