

# Cloud Computing – Mythen, Fakten, Empfehlungen



Thomas Eisenbarth

12. Augsburger Linux-Infotag 2013

23. März 2013

# Agenda

---

\$ whoami

Cloud Computing: Das kleine Einmaleins

Worum gehts heute?

Wolkenrechner: Vor-, Nachteile, Mythen & Sagen

Empfehlungen

Wolkenrechnen für Einsteiger

... und für Fortgeschrittene

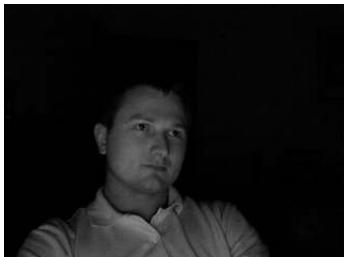
\$ whoami



whoami?

# \$ whoami

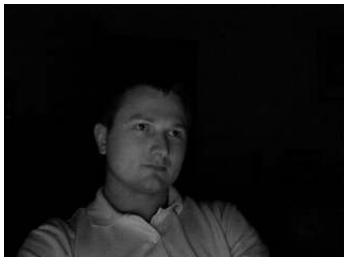
---



- Thomas Eisenbarth
- @fnordonaut
- thomas.eisenbarth@makandra.de
- makandra GmbH
  - Web App-Entwicklung
  - Ruby on Rails
  - Linux, FreeBSD
  - Apache, nginx, Passenger
  - MySQL, Solr, memcached
  - Amazon AWS & physikalische Hardware

## \$ whoami

---



- Thomas Eisenbarth
- @fnordonaut
- thomas.eisenbarth@makandra.de
- makandra GmbH
  - Web App-Entwicklung
  - Ruby on Rails
  - Linux, FreeBSD
  - Apache, nginx, Passenger
  - MySQL, Solr, memcached
  - Amazon AWS & physikalische Hardware

# Cloud Computing: 1×1



# Was ist diese „Cloud“ ?

---



- **Software as a Service (SaaS)**, bis 2010:  
„mandantenfähige Webanwendung“
  - Vollständig von Dienstleister betriebene Software
  - Dropbox
  - Salesforce.com
- **Platform as a Service (PaaS)**
  - Vorkonfigurierte Laufzeitumgebung für Anwendungen
  - Google App Engine
  - Salesforce force.com
  - EngineYard
  - Heroku
- *Neu*: Desktop as a Service ( $\approx$  Terminal Services)
- *Neu*: Backend as a Service  
 $\approx$  „Mobile backend as a Service“ ← **WTF!**

# Was ist diese „Cloud“ ?

---



- **Software as a Service (SaaS)**, bis 2010:  
„mandantenfähige Webanwendung“
  - Vollständig von Dienstleister betriebene Software
  - Dropbox
  - Salesforce.com
- **Platform as a Service (PaaS)**
  - Vorkonfigurierte Laufzeitumgebung für Anwendungen
  - Google App Engine
  - Salesforce force.com
  - EngineYard
  - Heroku
- *Neu:* Desktop as a Service ( $\approx$  Terminal Services)
- *Neu:* Backend as a Service  
 $\approx$  „Mobile backend as a Service“ ← **WTF!**



# Was ist diese „Cloud“ ?

---



- **Software as a Service (SaaS)**, bis 2010:  
„mandantenfähige Webanwendung“
  - Vollständig von Dienstleister betriebene Software
  - Dropbox
  - Salesforce.com
- **Platform as a Service (PaaS)**
  - Vorkonfigurierte Laufzeitumgebung für Anwendungen
  - Google App Engine
  - Salesforce force.com
  - EngineYard
  - Heroku
- *Neu*: Desktop as a Service ( $\approx$  Terminal Services)
- *Neu*: Backend as a Service  
 $\approx$  „Mobile backend as a Service“ ← **WTF!**

# Was ist diese „Cloud“ ?

---



- **Software as a Service (SaaS)**, bis 2010:  
„mandantenfähige Webanwendung“
  - Vollständig von Dienstleister betriebene Software
  - Dropbox
  - Salesforce.com
- **Platform as a Service (PaaS)**
  - Vorkonfigurierte Laufzeitumgebung für Anwendungen
  - Google App Engine
  - Salesforce force.com
  - EngineYard
  - Heroku
- *Neu:* Desktop as a Service ( $\approx$  Terminal Services)
- *Neu:* Backend as a Service  
 $\approx$  „Mobile backend as a Service“ ← **WTF!**

# Was ist diese „Cloud“ ?

---



- **Software as a Service (SaaS)**, bis 2010:  
„mandantenfähige Webanwendung“
  - Vollständig von Dienstleister betriebene Software
  - Dropbox
  - Salesforce.com
- **Platform as a Service (PaaS)**
  - Vorkonfigurierte Laufzeitumgebung für Anwendungen
  - Google App Engine
  - Salesforce force.com
  - EngineYard
  - Heroku
- *Neu*: Desktop as a Service ( $\approx$  Terminal Services)
- *Neu*: Backend as a Service  
 $\approx$  „Mobile backend as a Service“ ← WTF!

# Was ist diese „Cloud“ ?

---



- **Software as a Service (SaaS)**, bis 2010:  
„mandantenfähige Webanwendung“
  - Vollständig von Dienstleister betriebene Software
  - Dropbox
  - Salesforce.com
- **Platform as a Service (PaaS)**
  - Vorkonfigurierte Laufzeitumgebung für Anwendungen
  - Google App Engine
  - Salesforce force.com
  - EngineYard
  - Heroku
- *Neu*: Desktop as a Service ( $\approx$  Terminal Services)
- *Neu*: Backend as a Service  
 $\approx$  „Mobile backend as a Service“ ← **WTF!**

# Infrastructure as a service (IaaS)

---

## Definition (IaaS)

Automatisierbare, virtuelle  
Bereitstellung von  
**IT-Infrastruktur** u.a. per API

- virtuelle Maschinen (VM)
  - Storage
  - Netzwerk
- **Public Cloud**
    - Amazon AWS
    - Rackspace
    - Microsoft Azure
    - Rightscale
  - **Private Cloud** („im Keller“)
    - OwnCloud
    - *OpenStack*
    - Eucalyptus
    - OpenNebula

# Infrastructure as a service (IaaS)

---

## Definition (IaaS)

Automatisierbare, virtuelle  
Bereitstellung von  
**IT-Infrastruktur** u.a. per API

- virtuelle Maschinen (VM)
- Storage
- Netzwerk

- **Public Cloud**
  - Amazon AWS
  - Rackspace
  - Microsoft Azure
  - Rightscale
- **Private Cloud** („im Keller“)
  - OwnCloud
  - *OpenStack*
  - Eucalyptus
  - OpenNebula

# Infrastructure as a service (IaaS)

---

## Definition (IaaS)

Automatisierbare, virtuelle  
Bereitstellung von  
**IT-Infrastruktur** u.a. per API

- virtuelle Maschinen (VM)
- Storage
- Netzwerk

- **Public** Cloud
  - Amazon AWS
  - Rackspace
  - Microsoft Azure
  - Rightscale
- **Private** Cloud („im Keller“)
  - OwnCloud
  - *OpenStack*
  - Eucalyptus
  - OpenNebula

# Infrastructure as a service (IaaS)

---

## Definition (IaaS)

Automatisierbare, virtuelle  
Bereitstellung von  
**IT-Infrastruktur** u.a. per API

- virtuelle Maschinen (VM)
- Storage
- Netzwerk

- **Public** Cloud
  - Amazon AWS
  - Rackspace
  - Microsoft Azure
  - Rightscale
- **Private** Cloud („im Keller“)
  - OwnCloud
  - *OpenStack*
  - Eucalyptus
  - OpenNebula



# Infrastructure as a service (IaaS)

---

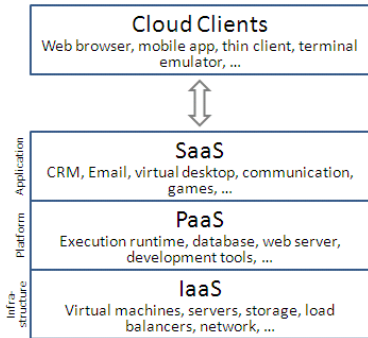
## Definition (IaaS)

Automatisierbare, virtuelle  
Bereitstellung von  
**IT-Infrastruktur** u.a. per API

- virtuelle Maschinen (VM)
  - Storage
  - Netzwerk
- **Public** Cloud
    - Amazon AWS
    - Rackspace
    - Microsoft Azure
    - Rightscale
  - **Private** Cloud („im Keller“)
    - OwnCloud
    - *OpenStack*
    - Eucalyptus
    - OpenNebula

# Die Wolke im Überblick

---



## Worum gehts heute?

---

# IaaS

- Mythen & Sagen
- Vorteile (gibt es!)
- Eigene IaaS-Wolke mit OpenStack
- Volles Potential ausschöpfen mit Konfigurationsmanagement (hier: Puppet)

# Mythen & Sagen

---

Es war einmal...

## „Cloud Computing“



# Mythen & Sagen

---

## „Cloud Computing“ ist...

*...automatisch sicher...  
...billig...  
...selbstheilend und funktioniert ohne Zutun...  
...besser...* } **...nicht.**

⇒ Never trust the marketing.

# Mythen & Sagen

---

## „Cloud Computing“ ist...

*...automatisch sicher...  
...billig...  
...selbsteilend und funktioniert ohne Zutun...  
...besser...* } *...nicht.*

⇒ Never trust the marketing.

# Mythen & Sagen

---

## „Cloud Computing“ ist...

*...automatisch sicher...  
...billig...  
...selbsteilend und funktioniert ohne Zutun...  
...besser...* } *...nicht.*

⇒ Never trust the marketing.

# Mythen & Sagen

---

## „Cloud Computing“ ist...

*...automatisch sicher...  
...billig...  
...selbsteilend und funktioniert ohne Zutun...  
...besser...* } **...nicht.**

⇒ Never trust the marketing.



# Mythen & Sagen

---

## „Cloud Computing“ ist...

*...automatisch sicher...  
...billig...  
...selbsteilend und funktioniert ohne Zutun...  
...besser...* } **...nicht.**

⇒ Never trust the marketing.

# Nachteile

---

- Vendor Lock-in
- Kosten: AWS EC2 Small Instance, Irland  
1 × CPU  $\approx$  1,2 GHz; 1,7 GB RAM; 160 GB HDD  $\rightarrow$  mtl. 47\$
- VMs fallen aus, verschwinden.
- Support (sigh!)
- Datenschutz...  
Wo (geografisch) bootet eine VM?
- VM-Layer + VM-Nachbarn  $\Rightarrow$  IO oft beeinträchtigt

# Nachteile

---

- Vendor Lock-in
- Kosten: AWS EC2 Small Instance, Irland  
1 × CPU  $\approx$  1,2 GHz; 1,7 GB RAM; 160 GB HDD  $\rightarrow$  mtl. 47\$
- VMs fallen aus, verschwinden.
- Support (sigh!)
- Datenschutz...  
Wo (geografisch) bootet eine VM?
- VM-Layer + VM-Nachbarn  $\Rightarrow$  IO oft beeinträchtigt

# Nachteile

---

- Vendor Lock-in
- Kosten: AWS EC2 Small Instance, Irland  
1 × CPU  $\approx$  1,2 GHz; 1,7 GB RAM; 160 GB HDD  $\rightarrow$  mtl. 47\$
- VMs ~~fallen aus~~, verschwinden.
- Support (sigh!)
- Datenschutz...  
Wo (geografisch) bootet eine VM?
- VM-Layer + VM-Nachbarn  $\Rightarrow$  IO oft beeinträchtigt

# Nachteile

---

- Vendor Lock-in
- Kosten: AWS EC2 Small Instance, Irland  
1 × CPU  $\approx$  1,2 GHz; 1,7 GB RAM; 160 GB HDD  $\rightarrow$  mtl. 47\$
- VMs ~~fallen aus~~, verschwinden.
- Support (sigh!)
- Datenschutz...  
Wo (geografisch) bootet eine VM?
- VM-Layer + VM-Nachbarn  $\Rightarrow$  IO oft beeinträchtigt

# Nachteile

---

- Vendor Lock-in
- Kosten: AWS EC2 Small Instance, Irland  
1 × CPU  $\approx$  1,2 GHz; 1,7 GB RAM; 160 GB HDD  $\rightarrow$  mtl. 47\$
- VMs ~~fallen aus~~, verschwinden.
- Support (sigh!)
- Datenschutz...  
Wo (geografisch) bootet eine VM?
- VM-Layer + VM-Nachbarn  $\Rightarrow$  IO oft beeinträchtigt

# Nachteile

---

- Vendor Lock-in
- Kosten: AWS EC2 Small Instance, Irland  
1 × CPU  $\approx$  1,2 GHz; 1,7 GB RAM; 160 GB HDD  $\rightarrow$  mtl. 47\$
- VMs ~~fallen aus~~, verschwinden.
- Support (sigh!)
- Datenschutz...  
Wo (geografisch) bootet eine VM?
- VM-Layer + VM-Nachbarn  $\Rightarrow$  IO oft beeinträchtigt

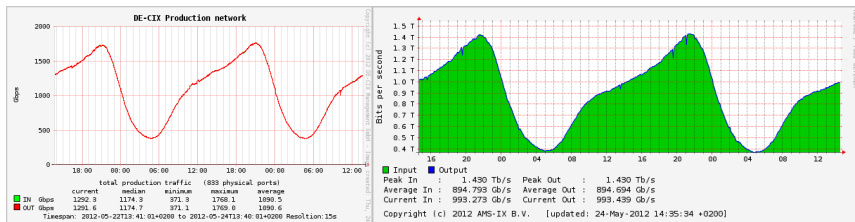


Alles doof?



# Vorteile

- Kostenvorteile durch Virtualisierung (Private)
- Bereitstellung in Echtzeit
- Up- & Downscaling



# IaaS: Für wen? Wann? Wofür?

---

- Public
  - Komplexe, verteilbare Berechnungen/Simulationen mit temporärem Bedarf an Ressourcen
  - Lastspitzen abfangen
  - Bedarf nach schneller Bereitstellung
  - Geografisch globale Verteilung
- Private
  - IaaS ist Virtualisierung 2.0
    - ⇒ Wer Virtualisierung intensiv nutzt, kann mit Inhouse-IaaS weitere Vorteile ausnutzen.

Go!



# Wie kann ich in die Wolke starten?

---

- Public IaaS: Kreditkarte besorgen, Anbieter heraussuchen, Account anlegen, starten.  
Tipp: AWS Free Usage Tier: z.B. 750hrs Linux Micro Instance
- Private IaaS: OpenStack  
„Open source software for building private and public clouds.“
  - Compute
  - Networking
  - Storage
  - Dashboard
  - APIs
- [www.openstack.org](http://www.openstack.org)

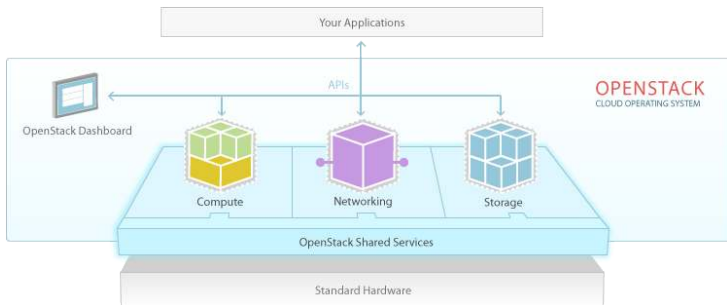
# Wie kann ich in die Wolke starten?

---

- Public IaaS: Kreditkarte besorgen, Anbieter heraussuchen, Account anlegen, starten.  
Tipp: AWS Free Usage Tier: z.B. 750hrs Linux Micro Instance
- Private IaaS: OpenStack
  - „Open source software for building private and public clouds.“
  - Compute
  - Networking
  - Storage
  - Dashboard
  - APIs
- [www.openstack.org](http://www.openstack.org)

# OpenStack: Open Source cloud software

---



# OpenStack

---

- Komponente zur Steuerung der Compute-Knoten und Instanzen darauf  
→ CPU, RAM, HDD, Netzwerk
- Image-Verwaltung für Instanzen
- Netzwerk-Management (LAN zwischen VMs & öffentliches Netz)
- Frontend für Hypervisoren (KVM, Xen, Hyper-V) und Management der VMs
- Security Groups
- REST API
- ...

## Instances

1 Instance Instance

<input type="checkbox"/>	Tenant	Host	Instance Name	IP Address	Size	Status	Task	Power State	Actions
<input type="checkbox"/>	makandra	knorr	██████████.makandra.de	10.10.13.6 80.237.241.137	4GB RAM   4 VCPU   50.0GB Disk	Active	None	Running	<a href="#">Edit Instance</a>
<input type="checkbox"/>	makandra	ungerer	██████████.makandra.de	10.10.13.7 80.237.241.138	4GB RAM   4 VCPU   50.0GB Disk	Active	None	Running	<a href="#">Edit Instance</a>
<input type="checkbox"/>	makandra	ungerer	██████████.makandra.de	10.10.13.9 80.237.241.133	4GB RAM   4 VCPU   50.0GB Disk	Active	None	Running	<a href="#">Edit Instance</a>
<input type="checkbox"/>	makandra	knorr	██████████.makandra.de	10.10.13.8 80.237.241.132	4GB RAM   4 VCPU   50.0GB Disk	Active	None	Running	<a href="#">Edit Instance</a>
<input type="checkbox"/>	makandra	ungerer	postgress1.makandra.de	10.10.13.5	1GB RAM   1 VCPU   10.0GB Disk	Active	None	Running	<a href="#">Edit Instance</a>
<input type="checkbox"/>	makandra	ungerer	dbwiki1.makandra.de	10.10.13.4 80.237.241.134	2GB RAM   2 VCPU   200.0GB Disk	Active	None	Running	<a href="#">Edit Instance</a>
<input type="checkbox"/>	makandra	ungerer	log.makandra.de	10.10.13.3 80.237.241.136	4GB RAM   4 VCPU   10.0GB Disk	Active	None	Running	<a href="#">Edit Instance</a>
<input type="checkbox"/>	makandra	knorr	dbwiki2.makandra.de	10.10.13.0 80.237.241.135	2GB RAM   2 VCPU   200.0GB Disk	Active	None	Running	<a href="#">Edit Instance</a>

Displaying 9 items

# Demo!

# Fragen?



## All Instances

### Instances

Terminate Instance

<input type="checkbox"/>	Tenant	Host	Instance Name	IP Address	Size	Status	Task	Power State	Actions
<input type="checkbox"/>	makandra	knorr	██████████.makandra.de	10.10.13.6 80.237.241.137	4GB RAM   4 VCPU   50.0GB Disk	Active	None	Running	<a href="#">Edit Instance</a>
<input type="checkbox"/>	makandra	ungerer	██████████.makandra.de	10.10.13.7 80.237.241.138	4GB RAM   4 VCPU   50.0GB Disk	Active	None	Running	<a href="#">Edit Instance</a>
<input type="checkbox"/>	makandra	ungerer	██████████.makandra.de	10.10.13.9 80.237.241.133	4GB RAM   4 VCPU   50.0GB Disk	Active	None	Running	<a href="#">Edit Instance</a>
<input type="checkbox"/>	makandra	knorr	██████████.makandra.de	10.10.13.8 80.237.241.132	4GB RAM   4 VCPU   50.0GB Disk	Active	None	Running	<a href="#">Edit Instance</a>
<input type="checkbox"/>	makandra	ungerer	postgress1.makandra.de	10.10.13.5	1GB RAM   1 VCPU   10.0GB Disk	Active	None	Running	<a href="#">Edit Instance</a>
<input type="checkbox"/>	makandra	ungerer	dbwiki1.makandra.de	10.10.13.4 80.237.241.134	2GB RAM   2 VCPU   200.0GB Disk	Active	None	Running	<a href="#">Edit Instance</a>
<input type="checkbox"/>	makandra	ungerer	log.makandra.de	10.10.13.3 80.237.241.136	4GB RAM   4 VCPU   10.0GB Disk	Active	None	Running	<a href="#">Edit Instance</a>
<input type="checkbox"/>	makandra	knorr	dbwiki2.makandra.de	10.10.13.0 80.237.241.135	2GB RAM   2 VCPU   200.0GB Disk	Active	None	Running	<a href="#">Edit Instance</a>

Displaying 9 items

# Demo!

# Fragen?

## IaaS 2.0: Fortgeschrittene Themen

---

- Vorteile bisher: VM-Layer, Echtzeit-Provisioning mit API  
„Instanz starten“ [2 Minuten] vs.  
„Hardware bestellen + Rack-Einbau“ [2 Wochen]
- Danach folgt...
  - OS-Setup
  - Userland-Setup
    - Benutzer anlegen
    - OpenSSH installieren, PubKey-Auth anschalten, Root-Login verbieten
    - LAMP-Stack o.ä. installieren
    - ...

## IaaS 2.0: Fortgeschrittene Themen

---

- Vorteile bisher: VM-Layer, Echtzeit-Provisioning mit API  
„Instanz starten“ [2 Minuten] vs.  
„Hardware bestellen + Rack-Einbau“ [2 Wochen]
- Danach folgt...
  - OS-Setup
  - Userland-Setup
    - Benutzer anlegen
    - OpenSSH installieren, PubKey-Auth anschalten, Root-Login verbieten
    - LAMP-Stack o.ä. installieren
    - ...

## IaaS 2.0: Fortgeschrittene Themen

---

- Vorteile bisher: VM-Layer, Echtzeit-Provisioning mit API  
„Instanz starten“ [2 Minuten] vs.  
„Hardware bestellen + Rack-Einbau“ [2 Wochen]
- Danach folgt...
  - OS-Setup
  - Userland-Setup
    - Benutzer anlegen
    - OpenSSH installieren, PubKey-Auth anschalten, Root-Login verbieten
    - LAMP-Stack o.ä. installieren
    - ...

## „Benutzer anlegen und Zugriff geben“

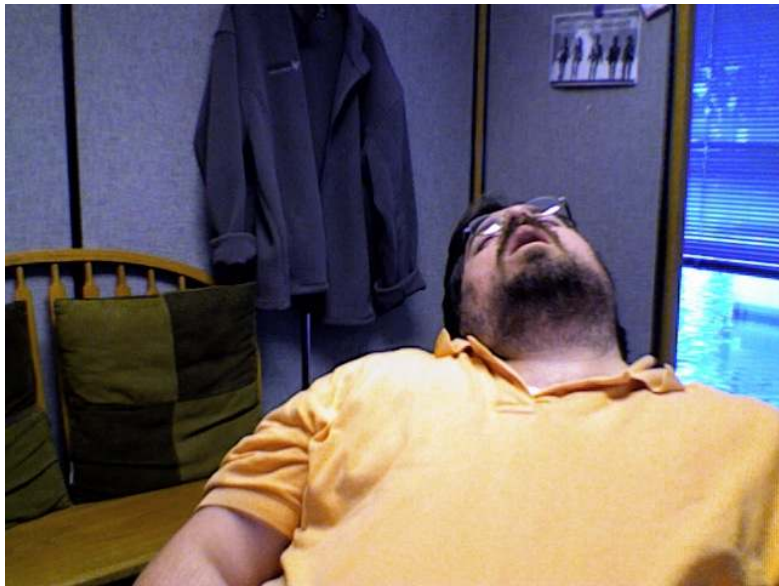
---

- Installiere OpenSSH
  - Abhängig vom Zielsystem `pkg_add`, `apt-get`, `yum`, Sourcen, ...
  - Dienst starten
  - root-Login soll nicht möglich sein (`sshd_config`)
  - Nur PubKey-Authentifizierung soll erlaubt sein, Passwort nicht!
- Lege Benutzer an: Alice, Bob, Charlie
  - Setze default-shell auf bash, kopiere `.bashrc`
  - Kopiere `.authorized_keys` zum Login

## „Benutzer anlegen und Zugriff geben“

---

- Installiere OpenSSH
  - Abhängig vom Zielsystem `pkg_add`, `apt-get`, `yum`, Sourcen, ...
  - Dienst starten
  - root-Login soll nicht möglich sein (`sshd_config`)
  - Nur PubKey-Authentifizierung soll erlaubt sein, Passwort nicht!
- Lege Benutzer an: Alice, Bob, Charlie
  - Setze default-shell auf bash, kopiere `.bashrc`
  - Kopiere `.authorized_keys` zum Login



# Manuelles Ausführen Kommandos auf vielen Rechnern...

---

- ... ist langweilig und macht keinen Spass
- ... skaliert nicht
- ... ist langsam und fehleranfällig
- ... hinterlässt oft "Test-Konfiguration" (→ Einzelkunstwerk!)

→ Was tun?

- **Puppet**
- Chef
- CFengine
- ...



## Manuelles Ausführen Kommandos auf vielen Rechnern...

---

- ... ist langweilig und macht keinen Spass
- ... skaliert nicht
- ... ist langsam und fehleranfällig
- ... hinterlässt oft "Test-Konfiguration" (→ Einzelkunstwerk!)

→ Was tun?

- **Puppet**
- Chef
- CFengine
- ...

# Was ist Puppet?

---



- „Data Center Automation“-Tool
- Konfigurationsmanagement
- zentrale Konfiguration für verteilte Systeme
- Deklarative, modellhafte Sprache

# Was kann Puppet?

---



Verwaltung von ...

- Benutzer- & Gruppen
- Software- / Paket- & Servicemanagement
- Berechtigungen, Dateien, Ordner, Cronjobs

# Was kann Puppet?

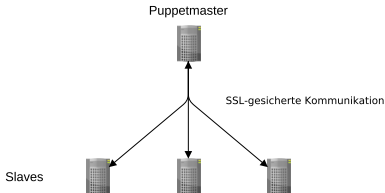
---

## Support für...

- Red Hat Enterprise Linux, version 4 and higher
- CentOS, version 4 and higher
- Scientific Linux, version 4 and higher
- Oracle Linux, version 4 and higher
- Debian, version 5 (Lenny) and higher
- Ubuntu, version 8.04 LTS and higher
- Fedora, version 15 and higher
- SUSE Linux Enterprise Server, version 11 and higher
- Gentoo Linux
- Mandriva Corporate Server 4
- ArchLinux
- FreeBSD 4.7 and later
- OpenBSD 4.1 and later
- Mac OS X, version 10.4 (Tiger) and higher
- Oracle Solaris, version 10 and higher
- AIX, version 5.3 and higher
- HP-UX
- Windows Server 2003 and 2008 (Puppet version 2.7.6 and higher)
- Windows 7 (Puppet version 2.7.6 and higher)

# Puppet-Kommunikation: Client-Server-Setup

---



- Slaves wissen um ihre Rollen, z.B.
  - Web
  - App
  - DB
- Slaves melden sich periodisch beim Puppetmaster
- Puppetmaster übergibt Slaves den gewünschten Zustand ihrer Konfiguration

# Puppet: Funktionsweise

---

Puppet-DSL ist

- Deklarativ  
→ Beschreibt Zielzustand, keine Reihenfolge von Anweisungen
- Modellhaft, überschreibt lokale Änderungen

## Puppet: Package & File-Handling

---

```
node 'test1.example.com' {
  package { 'openssh-server':
    ensure => installed,
  }

  file { '/etc/ssh/sshd_config':
    source  => 'puppet:///modules/sshd/sshd_config',
    owner   => 'root',
    group   => 'root',
    mode    => '640',
    notify  => Service['sshd'],
    require => Package['openssh-server'],
  }
}
```

## Puppet: Node-Konfiguration

---

```
node 'www.testing.com', 'www2.testing.com' {  
    include common  
    include apache, squid  
}
```

```
node 'www2.testing.com' inherits 'www.testing.com' {  
    include loadbalancer  
}
```

```
node 'db.testing.com' {  
    include common  
    include mysql  
}
```



## Puppet: Benutzerverwaltung

---

```
user { "dude":  
  ensure    => present,  
  uid       => '507',  
  gid       => 'admin',  
  shell     => '/bin/zsh',  
  home      => '/home/dude',  
  managehome => true,  
}
```

# Vorteile

---



- Einmalige, zentrale, **gültige** Definition
- Weitestgehend unabhängig vom Zielsystem
- Versionskontrolle
- Automatisierung
- Skalierung

# PuppetForge

---

- <http://forge.puppetlabs.com>
- „Puppet Forge is a repository of modules written by our community for Puppet Open Source and Puppet Enterprise IT automation software.“
- Module für viel Software vorhanden
- unterschiedliche Qualität, aber guter Startpunkt

# Puppet + OpenStack

---

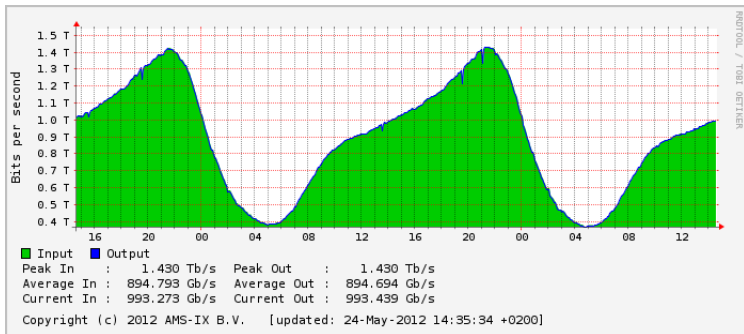
- Bereits festgestellt: Cloud Computing kann Zeit für Provisioning drastisch reduzieren
- *Neu:* Puppet kann die Zeit für reproduzierbares Installieren und Konfigurieren drastisch reduzieren
- Instanz starten: 2 Minuten
- Konfiguration per Puppet ausrollen: 2 Minuten

# Puppet + OpenStack

---

- Bereits festgestellt: Cloud Computing kann Zeit für Provisioning drastisch reduzieren
- *Neu:* Puppet kann die Zeit für reproduzierbares Installieren und Konfigurieren drastisch reduzieren
- Instanz starten: 2 Minuten
- Konfiguration per Puppet ausrollen: 2 Minuten

# Auto scaling: Traffic am AMS-IX



# Fragen?

---



# Credits

---

- "Puppets"  
<http://www.flickr.com/photos/facing-my-life/2907531865/>
- "Bored"  
<http://www.flickr.com/photos/stovak/2378145902/>
- "Clouds"  
<http://www.flickr.com/photos/johnmueller/52621490/>
- "Puppetrocker"  
<http://www.flickr.com/photos/facing-my-life/2907587409/in/photostream>
- "Dream Fairy"  
<http://www.flickr.com/photos/alexandrialanier/2966580560/>
- "Go!"  
<http://www.flickr.com/photos/jronaldlee/7662637844/>
- "lucas unhappy"  
<http://www.flickr.com/photos/ukalipt/2372562917/>



Danke!