puppet, the foreman and everything

Opensource Tools für das Datacenter Management

Nils Domrose

Köln, 24. Juni-2014

Aufgabenstellung im Datacenter

inovex

- Konsistente, nachvollziehbare Erstellung von physikalischen Servern und virtuellen Instanzen
- Flexible Anbindung von „Virtual Infrastructure“ und Cloud- Lösungen
- Deployment von Cloud-Lösungen und „Virtual Infrastructure“
- Configuration Management
- Patch Management
- ggf. Mandantenfähigkeit
- Application Deployment
- Monitoring und User Management
- Kurzum: Infrastructure as Code

Was ist Foreman?

inovex

Lifecycle Management Tool für virtuelle und physikalische Server- und Compute-Instanzen

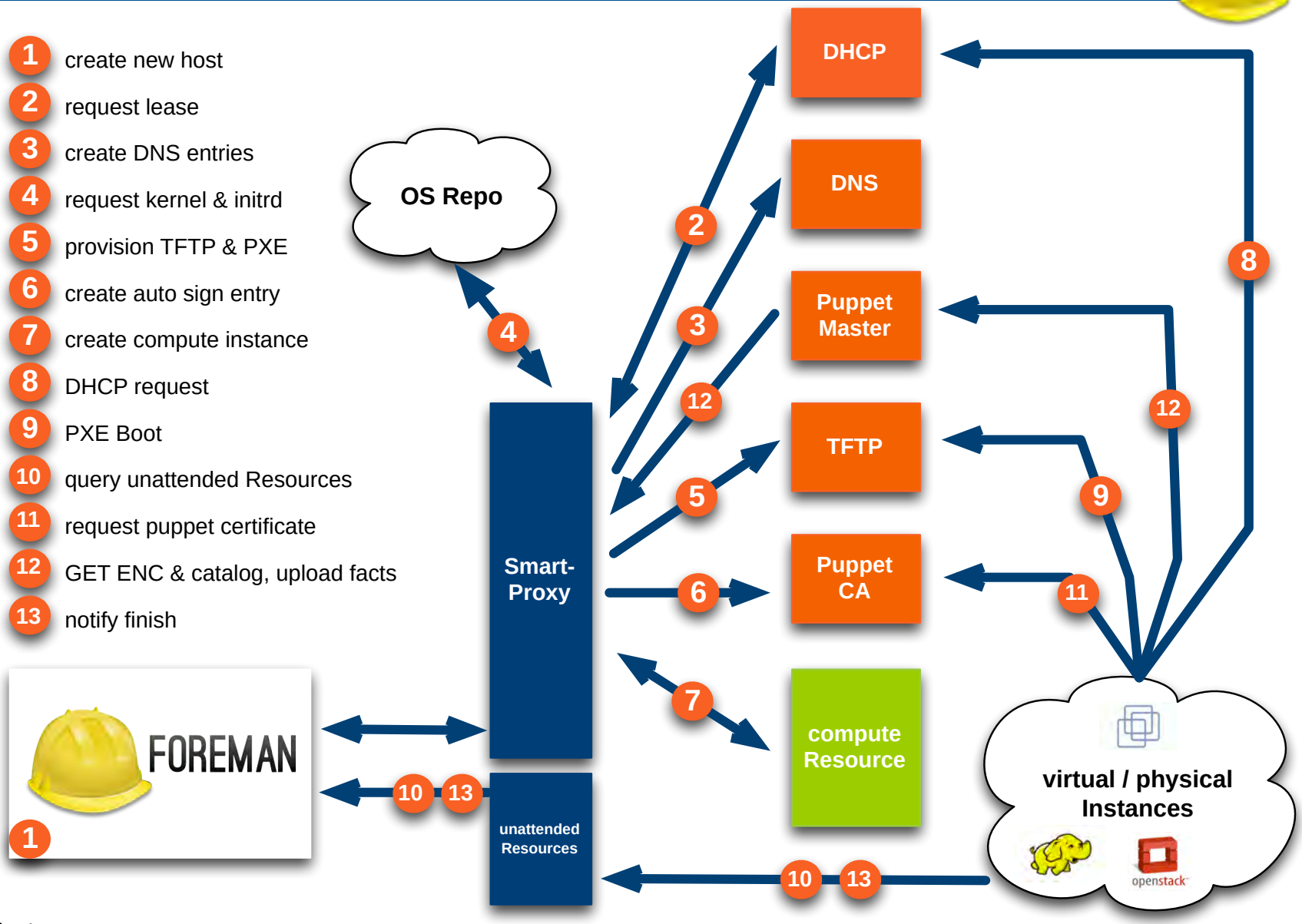
- RedHat-Projekt
- Image-basierte Deployments
- PXE- und ISO*-basierte Deployments
- Integration und Provisionierung von Basis-Infrastruktur-Komponenten wie TFTP, DHCP und DNS
- External Node Classifier (ENC) für Puppet und andere cfgmgmt-Systeme
- Dashboard für Puppet und Chef
- Integration von Katello für Repository Management
- Gute Erweiterbarkeit durch Plugins (GUI und Features)
- Autodiscovery via Discovery-Plugin

** verbesserungsfähig durch Automatisierung*

Foreman Deployment Workflow

inovex

1. create new host
2. request lease
3. create DNS entries
4. request kernel & initrd
5. provision TFTP & PXE
6. create auto sign entry
7. create compute instance
8. DHCP request
9. PXE Boot
10. query unattended Resources
11. request puppet certificate
12. GET ENC & catalog, upload facts
13. notify finish



Quelle: inovex

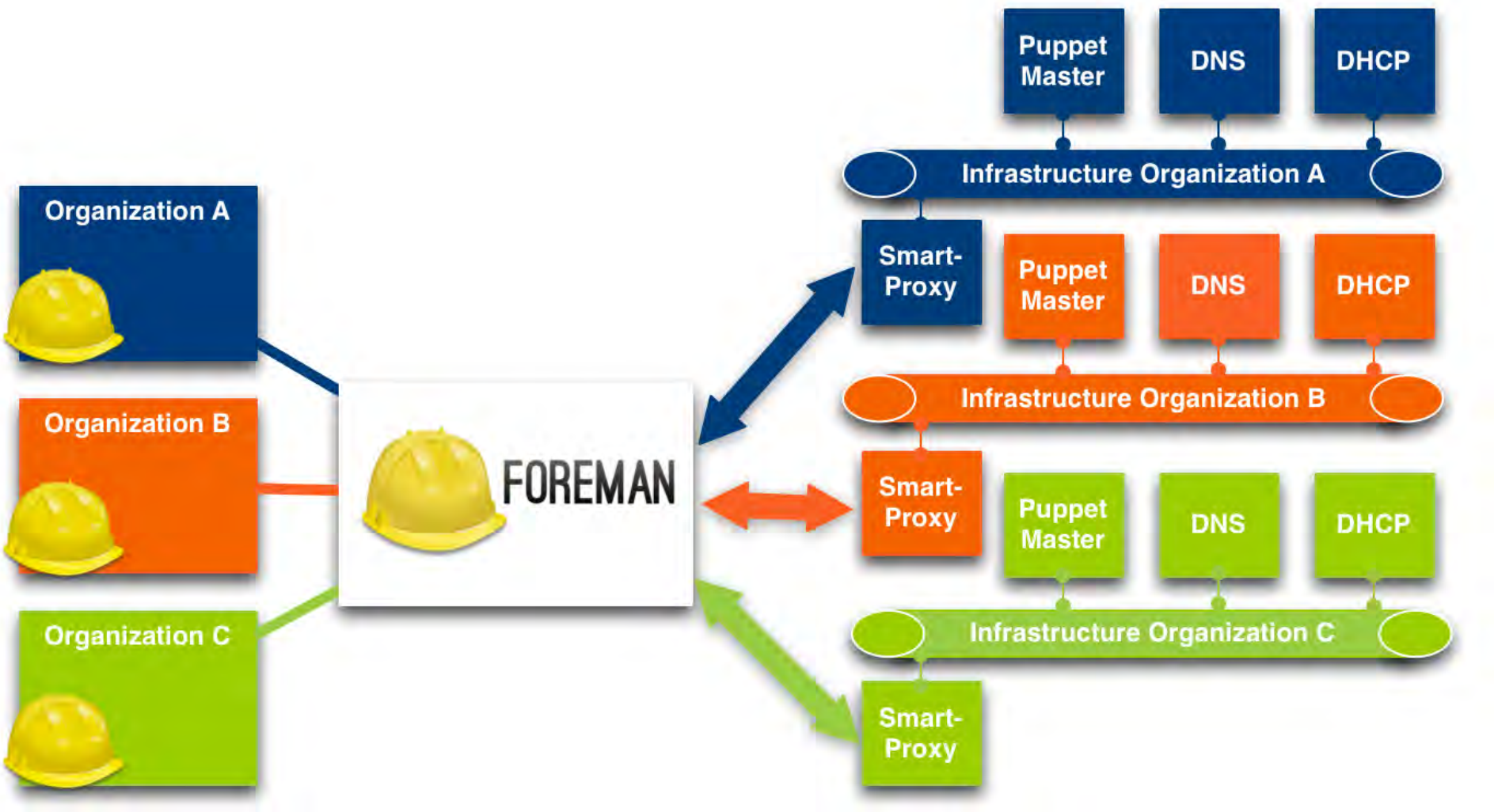
Was deckt Foreman ab?

inovex

- ✓ Konsistente, nachvollziehbare Erstellung von physikalischen Servern und virtuellen Instanzen
- ✓ Flexible Anbindung von „Virtual Infrastructure“ und Cloud- Lösungen
- ✓ Deployment von Cloud-Lösungen und „Virtual Infrastructure“
- ✓ Patch Management via Katello oder standalone via Pulp
- ✓ Mandantenfähigkeit
- ✗ Configuration Management
- ✗ Application Deployment
- ✗ Monitoring und User Management
- ✗ Infrastructure as Code

Mandantenfähigkeit

inovex



Quelle: theforeman.org / inovex GmbH

Foreman Roadmap

inovex

- Docker.io Support
- Neues Permission Management
- Service OS für Discovery Plugin
  (Firmware Updates, Inventory)
- Automatisierte, ISO-Image-basierte Installation für virtuelle Systeme mittels API und physikalischer Server mittels IPMI

Configuration Management

inovex

Puppet als Basis

- Gitlab als Quell-VCS für Puppet-Klassen
- Foreman als ENC zur Parametrisierung von Puppet Rollen
- Hiera Daten Parametrisierung von Modulen
- Gitlab Commit Hooks zur Synchronisierung auf die Puppet-Master inkl. Syntax-Validierung via Jenkins Job
- Mittels Crypt-Modul können Strings im ENC oder Hiera verschlüsselt abgelegt werden. Die Entschlüsselung erfolgt zentral durch den Puppetmaster.

Application Deployment != Configuration Management

inovex

- Das Configuration Management beschreibt die Laufzeit-Umgebung einer Applikation auf Basis von Host-spezifischen Informationen (z.B. puppet facts).

- Das Deployment-Tool provisioniert die Laufzeitumgebung mit den Applikationen.

  - Wir nutzen (R)?ex (www.rexify.org) und Jenkins. Alternativen: mcollective, rundeck, saltstack

- Das Lifecycle Management (Foreman) dient als CMDB.
  (Welche Systeme gibt es, wie sind diese parametrisiert?)

- Das Deployment-Tool fragt die Informationen ab und deployed auf Basis der Informationen die Applikation in der Laufzeitumgebung.

Puppet

inovex

- entwickelt Luke Kanies
- in ruby geschrieben
- maintained by puppetlabs
- Seit Version 2.7.0 unter Apache 2.0 Lizenz
- enterprise version verfügbar

*“Puppet Open Source is a flexible, customizable framework available under the Apache 2.0 license designed to help system administrators automate the many repetitive tasks they regularly perform. As a declarative, model-based approach to IT automation, it lets you define the desired state - or the “what” - of your infrastructure using the Puppet configuration language.”**

puppetlabs.com

Warum Puppet

inovex

Pros:

- Viele User
- Viele Module
- Skaliert gut
- Wir haben viel Erfahrung gesammelt
- ...

Cons:

- Manchmal Komplex
- Steile Lernkurve
- Teilweise inkonsistent
- ...

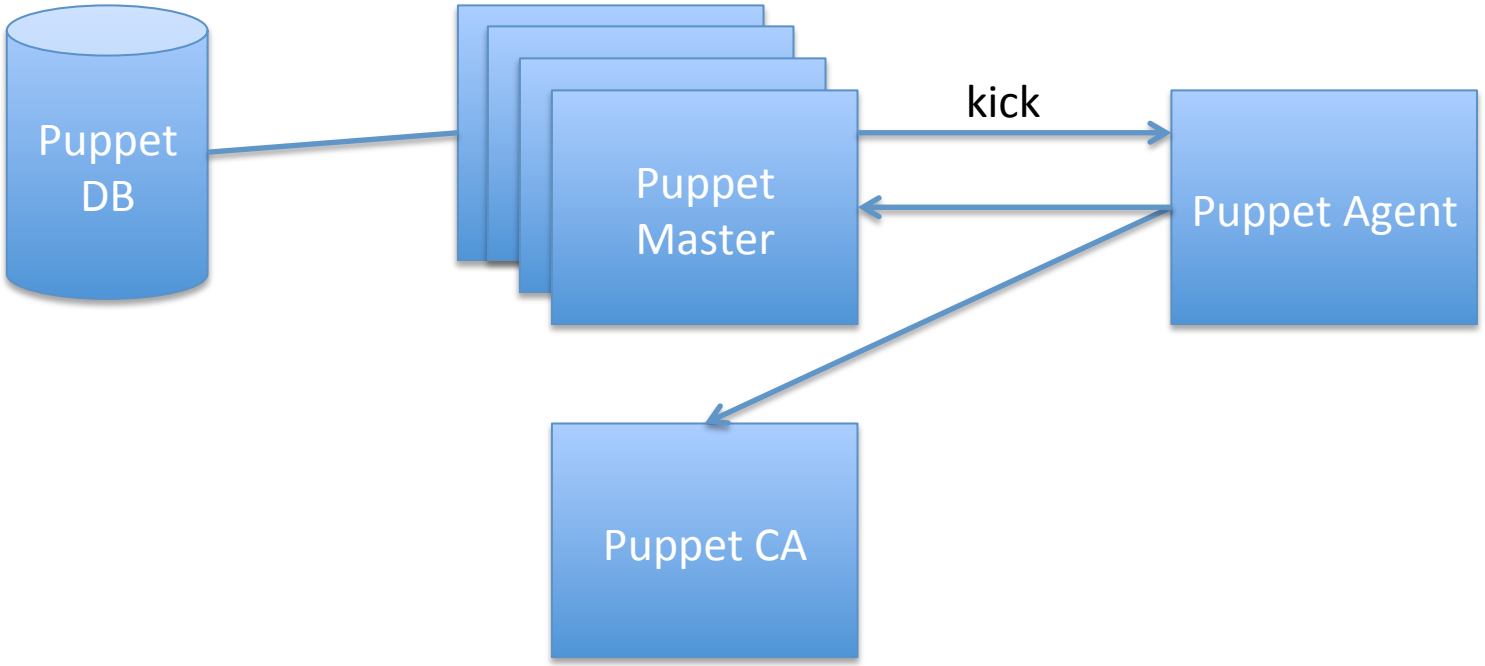
...es gibt Alternativen: Saltstack, Ansible, Chef, Cfengine...

Letzten Endes ist es eine Frage des verfügbaren Know-Hows.

Puppet Architektur

inovex

- Lokal oder Master-Agent Setup, Agent fragt sporadisch beim Master an.
- Eine Puppet CA
- Master erzeugt eine Agent-spezifischen Ausführungsplan (catalog), Agent wendet den catalog an.
- Optional mehrere Masters (Skalierbarkeit)
- Optional PuppetDB zum Austausch von Daten wie Reports und exported resources



Puppet Module Basics

inovex

- Puppet DSL
- Optional Dateien und Templates
- Optional Facts
- Optional Tests

```
.
├── Modulefile          - Modul Kurzbeschreibung (mandatory)
├── README              - Modul Hilfe (optional)
├── files               - Dateien die ausgeliefert werden (optional)
├── lib                 - Puppet Erweiterungen (optional)
├── manifests
│   ├── config.pp       - mögliche Aufteilung des Moduls (optional)
│   ├── init.pp         - Puppet Einstiegspunkt für das Modul (mandatory)
│   ├── install.pp      - mögliche Aufteilung des Moduls (optional)
│   ├── params.pp       - OS spezfische Werte (deprecated)
│   ├── service.pp      - mögliche Aufteilung des Moduls (optional)
│   └── <...>.pp        - Weitere Untergleiderungen des Moduls (optional)
├── spec                - Test-Driven Puppet Development (optional)
├── templates           - Templates die ausgeliefert werden (optional)
└── tests               - Puppet Modultests (mandatory)
```

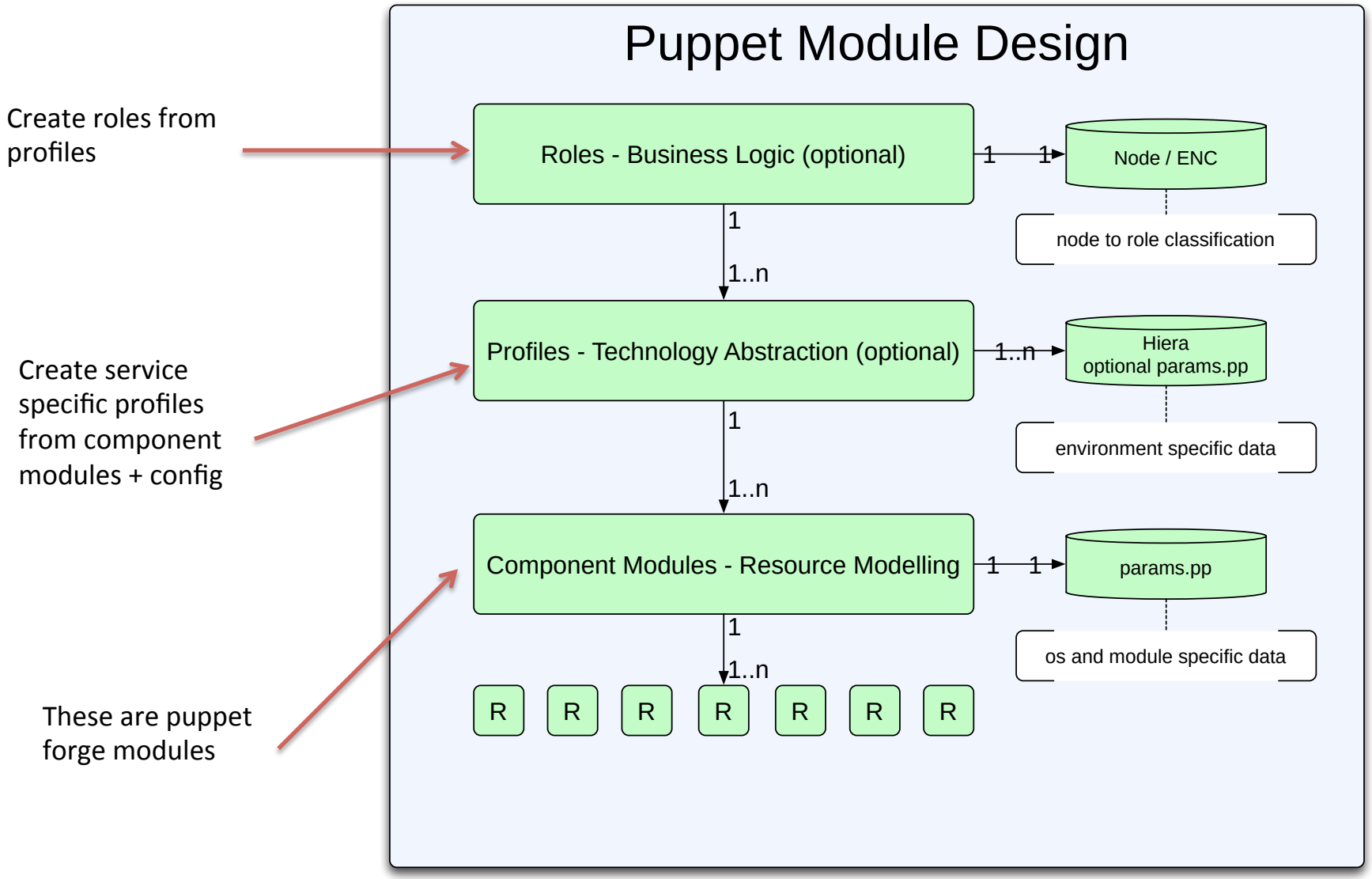
Puppet Module Basics

inovex

```
class openssh {

  package { 'openssh-server':
    ensure => installed,
    before => File['/etc/ssh/sshd_config'],
  }

  file { '/etc/ssh/sshd_config':
    ensure => file,
    owner  => 'root',
    group  => 'root',
    mode   => '0600',
    source => “puppet:///modules/openssh/${::operatingsystem}/
sshd_config“,
  }

  service { 'ssh':
    ensure    => running,
    enable    => true,
    subscribe => File['/etc/ssh/sshd_config'],
  }

}
```

Puppet Module Types

inovex



Puppet Profiles

inovex

```
class profile_one {
  class {'one':
    oned     => true
    sunstone => true
  }
}
```

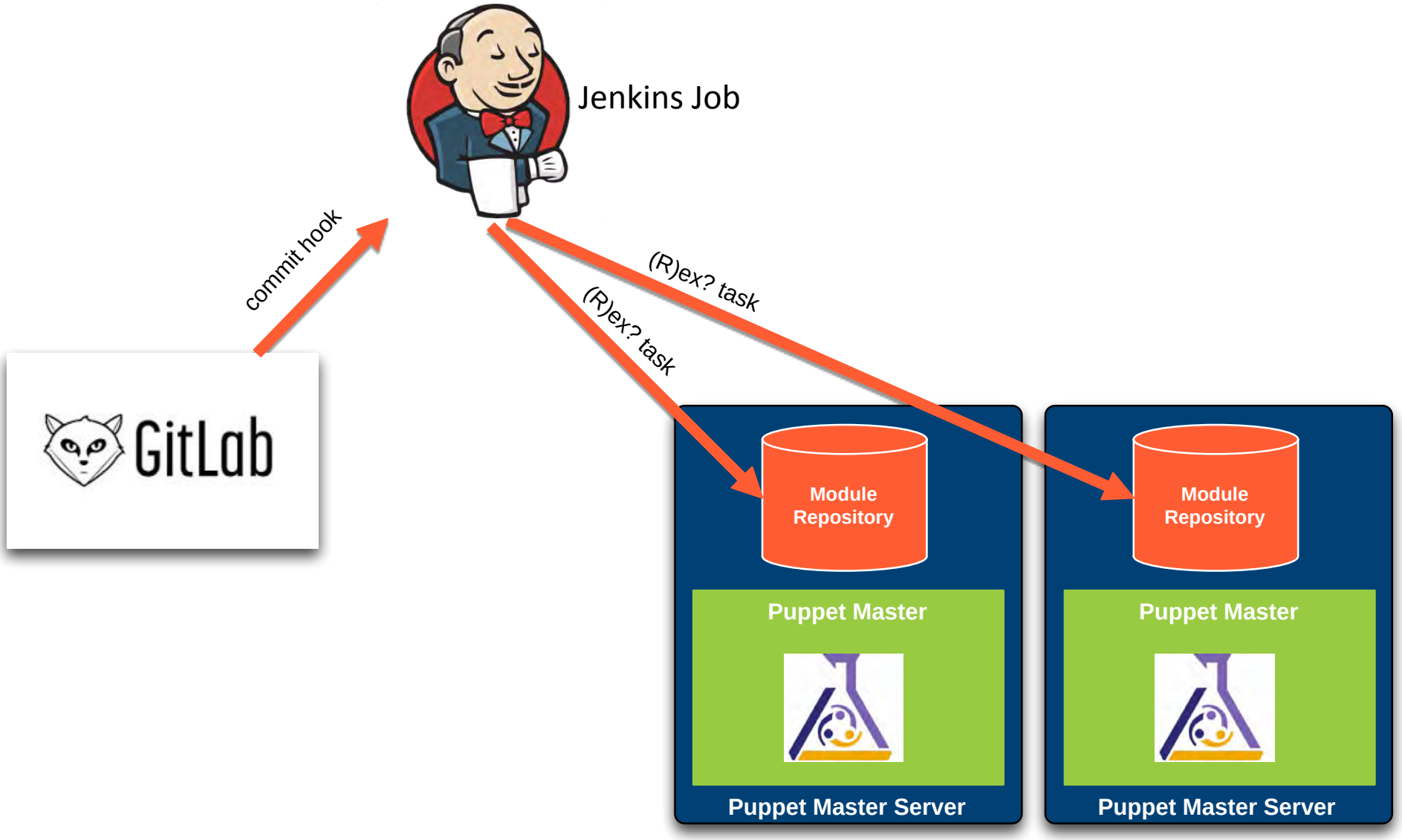
Puppet Roles

inovex

```
class role_one_node {
  class {‘profile_base‘:} ->

  class {‘profile_ceph ‘:} ->

  class {‘profile_one ‘:}
}
```

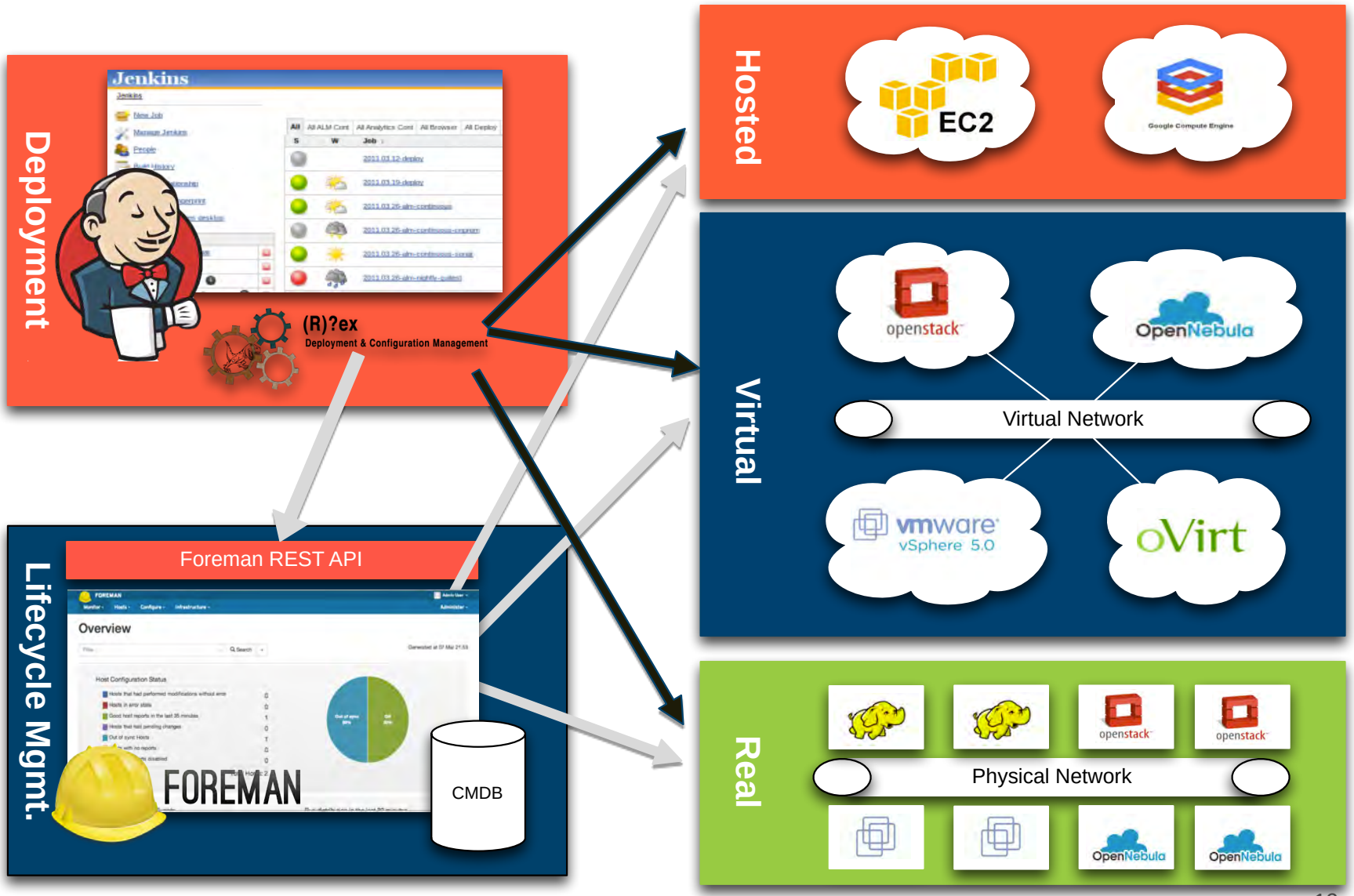
Puppet and Version Control

inovex



Deployment & Lifecycle Management

inovex



Was decken Foreman, Puppet, Rex & Jenkins ab?

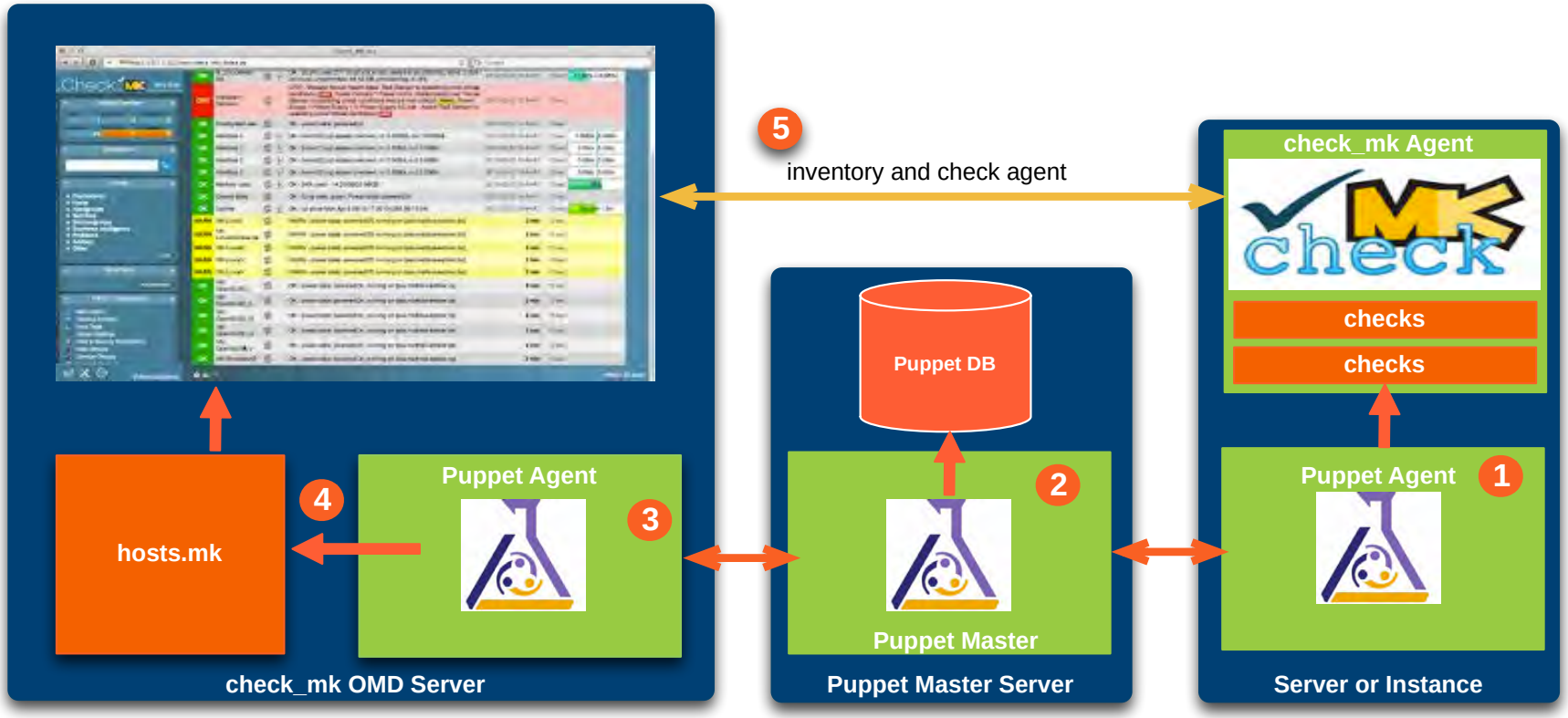
inovex

- ✓ Konsistente, nachvollziehbare Erstellung von virtuellen Instanzen
- ✓ Flexible Anbindung von „Virtual Infrastructure“ und Cloud- Lösungen
- ✓ Deployment von Cloud-Lösungen und „Virtual Infrastructure“
- ✓ Patch Management via Katello oder standalone via Pulp
- ✓ Mandantenfähigkeit
- ✓ Configuration Management
- ✓ Application Deployment
- ✗ Monitoring und User Management
- ✗ Infrastructure as Code

Monitoring mit puppet und check_mk

inovex

1. puppet agent deploys check_mk agent and checks
2. master stores exported resources in puppetDB
3. puppet agent on OMD server collects check_mk resources
4. host.mk is regenerated and inventory is triggered
5. inventory and agent checks of all deployed service checks



Usermanagement

inovex

- SSSD mit Anbindung an LDAP
- Dedizierter LDAP wird vom Konzern Identity Management System befüllt
- Rechte auf Systemen auf Basis von Gruppenzugehörigkeiten
- SSH Schlüssel im LDAP, Host-Keys via puppet exported resources
- Anbindung aller Subsysteme und Frontends an das AD oder den LDAP (GitLab, Foreman, usw.)

Infrastructure as Code

inovex

```
---
:hosts:
  - :name: test10.local.venv.de
    :hostgroup: 'generic kvm hosts'
    :compute_resource: 'kvm_local'
    :architecture: 'x86_64'
    :operatingsystem: 'ubuntu 12.04'
    :environment: 'production'
    :build: 1
    :compute_attributes:
      :cpus: 1
      :start: "1"
      :power_action: start
      :memory: 805306368

      :nics_attributes:
        0:
          :type: :bridge
          :bridge: virbr0
          :model: virtio
      :volumes_attributes:
        0:
          :pool_name: virtimages
          :capacity: 5G
          :allocation: 0G
```

Vielen Dank für Ihre Aufmerksamkeit

Kontakt

Nils Domrose
Senior Systems Engineer

inovex GmbH
Office cologne
Schanzenstr. 6-20
51063 Köln

nils.domrose@inovex.de