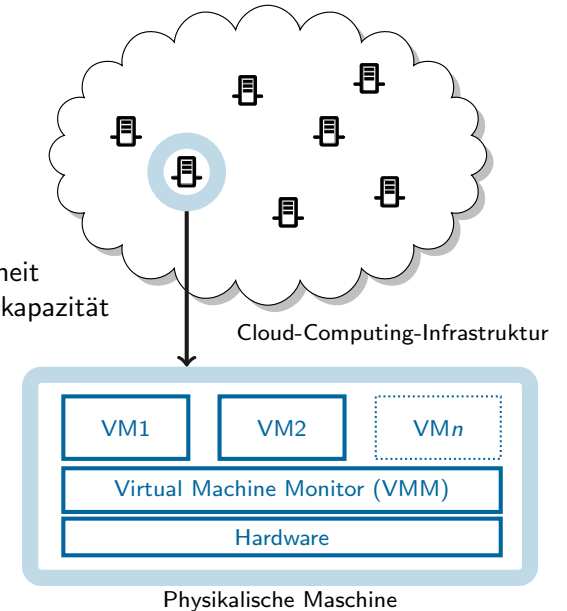


Cloud-Computing-Infrastruktur
Physikalische Infrastruktur
Software-Infrastruktur
Aufbereitung des Abbilds für Eucalyptus
Eucalyptus
Auslieferung des VM-Abbilds
Betrieb der virtuellen Maschine



- Standard-Hardware
 - „commodity hardware“
 - billig
 - **fehleranfällig**
- Charakteristik
 - hoher Grad der Verteiltheit
 - „unbegrenzte“ Speicherkapazität



Software-Infrastruktur am Beispiel von Eucalyptus

- **Cloud-Controller**
 - Zentrale Steuerinstanz, „verwaltet die Cloud“
 - Benutzer-Schnittstelle (Web-Service)
 - Abbildverwaltung
 - Benutzer- und Rechteverwaltung
 - API ähnelt der von Amazon EC2
- **Storage-Controller: Walrus** (→ Amazon S3)
- **Cluster-Controller**
 - Ressourcen-Verwaltung, Scheduling
 - Verteilung der virtuellen Maschinen
- **Node-Controller**
 - Ausführung und Überwachung der virtuellen Maschinen
- Ein privates Netz umspannt den Cluster-Controller mit den ihm zugeordneten Node-Controllern



Aufbereitung des Abbilds für Eucalyptus

- Automatisches Starten der Dienste
 - Beim Systemstart führt `init(8)` die Init-Skripte aus
 - Kopieren der Projektdateien nach `/opt/mwcc`
 - Modifikation von `/etc/rc.local`
- ```
JAVA="/usr/bin/java"

cd /opt/mwcc
$JAVA -cp <classpath> mw.cache.MWCache <parameter> &
$JAVA -cp <classpath> mw.path.MWPathServer <parameter> &
$JAVA -cp <classpath> mw.MWRegistryAccess register <parameter>
```
- Hilfestellung und Debugging
    - Hinzufügen von `echo`-Anweisungen
    - Ausführen von `/etc/rc.local` in der `chroot`-Umgebung
    - Starten von `/etc/rc.local` mit `bash -x`
    - Der Befehl `hostname(1)` gibt den aktuellen Host-Namen der Instanz aus



## Shell Quoting

- Doppelte Anführungszeichen:
  - Übergabe als einzelnes Argument
  - Expansion von Variablen möglich, „Escaping“ mittels Backslash (\)

```
$ argc() { echo "$@" has $# args"; }
$ argc Mein Home ist \"$HOME\"
Mein Home ist "/home/cloud" has 4 args
$ argc "Mein Home ist \"$HOME\""
Mein Home ist "/home/cloud" has 1 args
```

- Einfache Anführungszeichen (keine Expansion, kein Escaping):

```
$ argc 'Mein Home ist \"$HOME\"'
Mein Home ist "\"$HOME\"" has 1 args
```

- „Backticks“ (Gravis)
  - Führt Inhalt als Befehl aus
  - Ausgaben werden als Argument eingesetzt

```
$ argc Es ist `date`
Es ist Fri Nov 11 11:11:00 CET 2011 has 8 args
$ argc "Es ist `date`"
Es ist Fri Nov 11 11:11:00 CET 2011 has 1 args
```



## Eucalyptus Cloud am Lehrstuhl

- Web-Frontend (nur Account-Verwaltung)
  - Cloud-Controller: <http://i4cloud.informatik.uni-erlangen.de>
  - Zugangsdaten: SVN-Login
  - „Credentials“ laden (.zip) und nach ~/.euca extrahieren
  - Vor fremdem Zugriff schützen: `chmod 700 ~/.euca`
  - „Sourcen“ der Umgebungsvariablen: `source ~/.euca/eucarc`
- Kommandozeile
  - Zugriff über die Kommandozeile mittels „euca2ools“
  - Inhalt des Debian-Pakets anzeigen: `dpkg -L euca2ools`
  - Anleitungen: <http://open.eucalyptus.com/wiki/Euca2oolsGuide>
  - **Vor Verwendung:** `eucarc sourcen` (siehe oben)!



## Hybridfox

- Browser-Plugin für Firefox (bzw. Iceweasel)
- Installation über Datei → Öffnen...
  - Im CIP: `/proj/i4mw/pub/aufgabe2/hybridfox-1.7.000119.xpi`
  - Ansonsten erst Download <http://code.google.com/p/hybridfox/>
- Konfiguration der Region
  - Neue Region, Bezeichnung „i4cloud“
  - Typ Eucalyptus
  - Endpunkt:  
`http://i4cloud.informatik.uni-erlangen.de:8773/services/Eucalyptus`
- Konfiguration der „Credentials“
  - Benutzererkennung angeben
  - Access Key und Secret-Key aus eucarc-Datei:  
→ Werte aus `EC2_ACCESS_KEY` und `EC2_SECRET_KEY`
- Region und Credential wählen, dann Hybridfox-Tab neu öffnen



## Auslieferung des VM-Abbilds

- Erstellen des VM-Pakets („Bundle“)

```
$ cd /proj/i4mw/<benutzer>
$ euca-bundle-image -i image.raw -d . --arch i386
```
- VM-Paket an Eucalyptus senden

```
$ euca-upload-bundle -b <bucket_name> -m image.raw.manifest.xml
```
- Registrierung des VM-Pakets

```
$ euca-register <bucket_name>/image.raw.manifest.xml
```

→ ID des VM-Abbilds (`vm_abbild_id`)
- Eigene, registrierte VM-Pakete auflisten

```
$ euca-describe-images -o <benutzer>
```
- Attribute eines VM-Pakets bearbeiten

```
$ euca-modify-image-attribute -l -r all <vm_abbild_id>
```
- Löschen eines VM-Pakets

```
$ euca-deregister <vm_abbild_id>
$ euca-delete-bundle -b <bucket_name>
```



## Betrieb der virtuellen Maschine

### ■ Instanziierung einer virtuellen Maschine

```
$ euca-run-instances [-t <typ>] [-n <anzahl>] <vm_abbild_id>
```

→ Ausgabe: ID der Instanz(en) (*instanz\_id*)

### ■ Statusabfrage (Zustand und IP-Adresse)

```
$ euca-describe-instances
```

### ■ Ausgabe der Instanz-Konsole

```
$ tail -f /proj/i4mw/pub/console-log/guest-<instanz_id>.log
```

### ■ Passwortloser Zugriff mit SSH

```
$ ssh-keygen -R <instanz_ip> # Alten Host-Key entfernen
$ ssh -i <gruppen_name> cloud@<instanz_ip>
```

→ Schlüssel aus letzter Tafelübung, Instanz-IP von `euca-describe-instances`

→ Anderes VM-Image unter selber IP erfordert evtl. Zurücksetzen von Host-Key

### ■ Instanzen beenden

```
$ euca-terminate-instances <instanz_id>
```



↔ **Alternative:** Zugriff über das Firefox-Plugin

## Private-Cloud-Umgebung des Lehrstuhls

### ■ Persistente Modifikationen des VM-Abbilds mit `qemu`

- Installation weiterer Software-Pakete
- Anpassung der Start-Skripte
- System-Konfiguration

### ■ Limitationen der Cloud-Umgebung des Lehrstuhls

- Ressourcen der vier Node-Controller sind **beschränkt**
- Beenden von nicht (mehr) benötigten Instanzen
- Jederzeit auf faire Verwendung achten

### ■ Infrastruktur

- Bitte sendet bei Problemen oder Ungereimtheiten schnellstmöglichst eine E-Mail an [mw@i4.informatik.uni-erlangen.de](mailto:mw@i4.informatik.uni-erlangen.de)

