

Voll automatische Linux Installationen

Thomas Lange

Institut für Informatik

Universität zu Köln

lange@informatik.uni-koeln.de

Linuxtag Chemnitz 2004

p.1/22

Administration - bei Ihnen auch so?

- ▶ Hauptproblem der Sysadmins: **Keine Zeit!**
- ▶ Gewachsene Systeme, sehr unterschiedliche Hard- und Software
- ▶ Man kauft Hardware ohne vorher zu planen
- ▶ Kaufe ein Rechner, installiere das System, lass es ewig laufen
- ▶ Zwischendurch viele kleine Änderungen an jedem einzelnen System
- ▶ Nur Notfalladministration, da keine Zeit
- ▶ Keine Zeit für Fortbildung und Infrastrukturplan
- ▶ Trotz schnellster Computer, wird zu viel per Hand gemacht
- ▶ Bändigen Sie auch ihre vielen Pinguine so?

p.2/22

- ▷ Eine gute Computerinfrastruktur ist so wichtig wie ...?
- ▷ Gute laufende Rechner sind ihr Kapital
- ▷ Datensicherung (Backup) ist nur ein Teil
- ▷ Schlechte Installation -> unbrauchbare Rechner
- ▷ Schlechte Installation -> manuelle Nachbesserung -> keine Zeit
- ▷ Schlechte Installation -> unproduktive Benutzer
- ▷ Keine Updates, keine Patches, keine Sicherheit!!!
- ▷ Betrachte nicht einzelne Rechner sondern die ganze Infrastruktur
- ▷ Arbeiten in der Systemadministration müssen skalieren !
- ▷ Zeitaufwand für 5, 50 oder 500 Rechner sollte nicht linear steigen
- ▷ Haben Sie einen Plan für ihre Computer Infrastruktur?

p.3/22

Manuelle Installation

- ▷ Manuelle Installationen und Konfiguration dauern viele Stunden
- ▷ Viele Fragen sind zwischendurch zu beantworten
- ▷ Gleiche Daten müssen bei jedem Rechner erneut eingegeben werden
- ▷ Manuelle Installationen können nicht parallel durchgeführt werden
- ▷ "No simple sysadmin task is fun more than twice"
- ▷ Wiederholende Arbeit ist stupide und führt zu Fehlern
- ▷ **Ein Installation per Hand skaliert nicht !**
- ▷ Ziel: Keine Zeit mit stupiden Arbeiten vergeuden

p.4/22

Wer möchte diese Rechner per Hand installieren?



p.5/22

Warum nicht voll automatisch?

- ▶ Automatische Installationen dauern nur wenige Minuten
- ▶ Identische Installationen garantiert (auch nach Monaten)
- ▶ Nach Hardwaredefekt ist der Rechner in kürzester Zeit wieder mit identischer Konfiguration einsatzbereit (Desaster recovery)
- ▶ Heterogene Hardware und unterschiedliche Konfigurationen einfach
- ▶ Cluster, Serverfarmen und Pools ideal
- ▶ Ein Befehl installiert hunderte Rechner gleichzeitig
- ▶ Junior Admins können FAI anwenden
- ▶ Es kann sehr viel Zeit gespart werden !

p.6/22

- ▶ FAI macht alles, was ihr Systemadministrator zu tun hat, bevor der Benutzer das erste Mal auf einem neuen Rechner arbeiten kann
- ▶ Serverbasiertes Tool zur skriptgesteuerten vollautomatischen Installation von Debian GNU/Linux
- ▶ Kein Master Image notwendig
- ▶ Es kann das ganze Betriebssystem und die Anwendungsprogramme installieren und konfigurieren
- ▶ Sehr flexibel und einfach erweiterbar durch eigene Skripte (hooks)
- ▶ Es ist weder ein Cluster Management Tool noch ein Batch System
- ▶ Es kann die Installation nicht planen :-(), aber
- ▶ **Plane die Installation und FAI installiert deinen Plan! :-)**

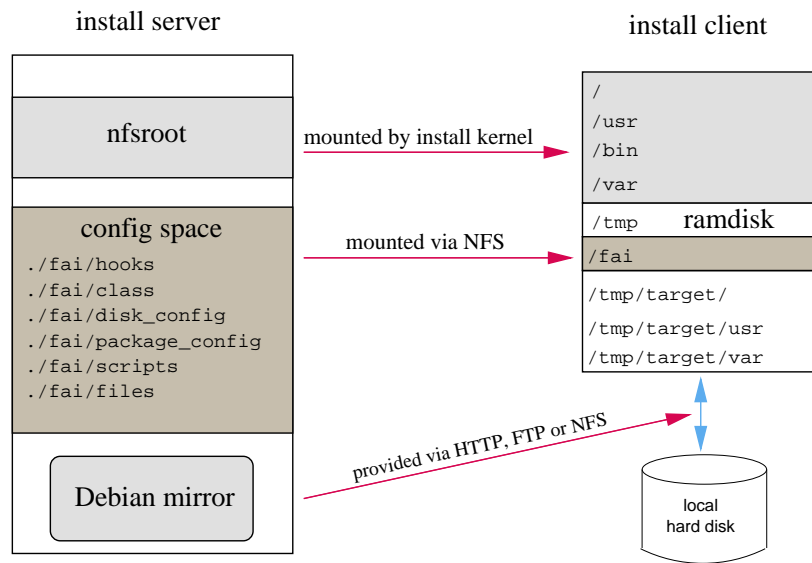
p.7/22

Wie funktioniert FAI ?



Ein Systemadministrator während der automatischen Installation

p.8/22



- ▶ Die Konfiguration liegt auf dem Install server
- ▶ Die Installation läuft auf dem Klienten
- ▶ Remotekontrolle während der Installation via ssh
- ▶ Nutzt Standard Tools zur Installation

p.9/22

Was braucht FAI?

- ▶ Server mit DHCP oder BOOTP, NFS und TFTP (install server)
- ▶ Rechner mit Netzwerkkarte (install client)
- ▶ Booten geht aber auch von Floppy oder CD-ROM
- ▶ Nicht nötig: Floppy, CD-ROM, Tastatur, Grafikkarte
- ▶ Lokaler Spiegel von Debian (NFS, FTP oder HTTP)
- ▶ Plattenplatz auf dem Server:

| | | |
|----------------|--------|---|
| FAI Paket | 13 MB | Kernel, Skripte, Konfigurationsdateien |
| nfsroot | 160 MB | erzeugt mit <code>make-fai-nfsroot</code> |
| Debian Spiegel | 5.6 GB | Debian 3.0 (nur i386) |
- ▶ Alle Install Clients nutzen die gleichen Verzeichnisse
- ▶ **Konstanter Plattenplatz**

p.10/22

- ▷ Plane deine Installation!
- ▷ Booten via PXE, dann Kernel via TFTP holen
- ▷ Rechner startet vollständiges Linux, ohne lokale Platte zu benutzen
- ▷ Hardwareerkennung und Kernel Module laden
- ▷ Klassen und Variablen definieren
- ▷ Festplatten partitionieren
- ▷ Dateisysteme erzeugen und mounten
- ▷ Software Pakete installieren
- ▷ Betriebssystem und Anwendungen konfigurieren
- ▷ Protokolldateien lokal und auf Install Server speichern
- ▷ Neu installiertes System booten

p.11/22

Das Klassenkonzept

- ▷ Ein Rechner gehört zu mehreren Klassen
- ▷ Priorität von niedrig nach hoch
- ▷ Beispiel: `DEFAULT SMALL_IDE GRUB GNOME demohost LAST`
- ▷ Klassen werden über Skripte in `/fai/class` definiert
- ▷ Alle Teile der Installation nutzen das Klassenkonzept
- ▷ Konfigurationsdateien werden anhand der Klassennamen ausgewählt
- ▷ Mit `fcopy` wird klassenbasiert ein Template kopiert
- ▷ Erfahrener Admin kreiert die Klassen
- ▷ Junior Admin ordnet die Klassen den Rechnern zu
- ▷ Junior Admin installiert die Rechner
- ▷ Junior Admin ~~installiert die Rechner~~ lässt installieren ;-)

p.12/22

Verzeichnisse im Config Space

```
|-- class
|   |-- 01alias
|   |-- 06hwdetect.source
|   |-- 24nis
|   |-- CS_KOELN.var
|   |-- DEFAULT.var
|   `-- wwwkiosk
|-- disk_config
|   |-- ATOMCLIENT
|   |-- SMALL_IDE
|   `-- vitamalz
|-- package_config
|   |-- BOWULF
|   |-- DEBIAN_DEVEL
|   |-- DEMO
|   |-- GERMAN
|   |-- GNOME
|   |-- WWWKIOSK
|   `-- nucleus
```

p.13/22

Verzeichnisse im Config Space

```
|-- files
|   |-- etc
|       |-- X11
|       |   `-- XF86Config-4
|       |       |-- ATI_ACER
|       |       |-- MATROX
|       |       `-- nucleus
|       `-- nsswitch.conf
|           |-- NIS
|           `-- NONIS
`-- scripts
    |-- BOOT
    |-- DEFAULT
    |   |-- S01
    |   |-- S20
    |   |-- S21
    |   `-- kueppers
    |-- NETWORK
    |   |-- S10
    |   `-- S40
    `-- kueppers
```

p.14/22

DEFAULT.var:

```
FAI_KEYMAP=de-latin1-nodeadkeys
```

```
UTC=yes
```

```
time_zone=Europe/Berlin
```

```
addpackages="kernel-image-2.4.24-atoms"
```

```
# root password for the new installed linux system;
```

```
rootpw="1234NVqX514t0f"
```

```
# Beowulf: define NFS server for /home and /usr/local
```

```
hserver=atom00
```

```
bserver=atom00
```

► Die Konfigurationsskripte in `/fai/scripts/*` nutzen diese

p.15/22

Plattenpartitionierung

```
# <type> <mountpoint> <size in mb> [mount options] [;extra options]
```

```
disk_config hda
```

```
primary /          30-100    rw,errors=remount-ro ;-c -j ext3
```

```
primary /fai-boot  7         rw        ; -j ext3
```

```
logical swap      50-500    rw
```

```
logical /var      50-1000   rw        ; -m 5  -j ext3
```

```
logical /tmp      50-1000   rw        ; -m 0  -j ext3
```

```
logical /usr      300-4000  rw        ; -j ext3
```

```
logical /home     50-4000   rw,nosuid ; -m 1  -j ext3
```

```
logical /scratch  0-        rw,nosuid ; -m 0  -i 50000 -j ext3
```

```
#logical /scratch preserve10 rw,nosuid ; -m 0  -i 50000 -j ext3
```

p.16/22

Beispiel: /fai/package_config/BEOWULF:

```
# packages for Beowulf clients

PACKAGES install
fping jmon autofs
rsh-client rsh-server rstat-client rstatd rusers rusersd
lam-runtime lam3 lam3-dev libpvm3 pvm-dev mpich scalapack-mpich-dev
dsh update-cluster-hosts update-cluster etherwake

PACKAGES taskinst
c-dev
```

- ▶ Aktionen wie bei `apt-get`: `install`, `remove` and `taskinst`
- ▶ Abhängigkeiten innerhalb der Pakete werden aufgelöst
- ▶ `dpkg -get-selections` auch möglich

p.17/22

Konfigurationsskripte

```
# create NIS/NONIS config
fcopy -M /etc/nsswitch.conf /etc/host.conf
fcopy -i /etc/ypserv.securenets # only for yp server
ifclass NONIS && rm -f $target/etc/defaultdomain
if ifclass NIS; then
    echo $YPPDOMAIN > $target/etc/defaultdomain
    rm -f $target/etc/yp.conf
    for s in $YPSRVR; do
        echo "ypserver $s" >> $target/etc/yp.conf
    done
fi

ifclass USR_LOCAL_COPY && {
    mount -o ro $bserver:/usr/local /usr/local
    cp -a /usr/local $target/usr
}
fcopy -M /etc/X11/XF86Config-4 && rm -f $target/etc/X11/XF86Config
```

p.18/22

```
files:
  any::
    ${target}/dev include=fd* mode=666  action=fixall r=1

editfiles:
  any::
    { ${target}/etc/fstab
      AppendIfNoSuchLine "none /proc/bus/usb usbdevfs defaults"
      AppendIfNoSuchLine "/dev/fd0 /floppy auto users,noauto 0 0"
    }
    { ${target}/etc/inittab
      ReplaceAll "/sbin/getty" With "/sbin/getty -f /etc/issue.linuxlogo"
    }
HOME_CLIENT::
  { ${target}/etc/fstab
    HashCommentLinesContaining "/home "
    AppendIfNoSuchLine "${hserver}:/home /home nfs rw,nosuid 0 0"
  }
```

p.19/22

Installationszeiten

| Rechner | RAM | Platte | Software | Zeit |
|--------------------|--------|--------|----------|--------|
| Pentium 4 2.80GHz | 1024MB | IDE | 948 MB | 5 min |
| Athlon XP1600+ | 896MB | SCSI | 1 GB | 6 min |
| AMD-K7, 500MHz | 320MB | IDE | 780 MB | 12 min |
| PentiumPro 200MHz | 128MB | IDE | 800 MB | 28 min |
| Pentium III 850MHz | 256MB | IDE | 820 MB | 10 min |
| Pentium III 850MHz | 256MB | IDE | 180 MB | 3 min |

Installation mehrerer Knoten in einem Beowulf Cluster:

| Knoten | Sekunden |
|--------|----------|
| 1 | 337 |
| 5 | 340 |
| 10 | 345 |
| 20 | 379 |

12% mehr Zeit bei 20 Rechnern.

p.20/22

- ▷ Deutsches Elektronen-Synchrotron, DESY, 60+
- ▷ McConnell Brain Imaging Centre, Montreal, 50+
- ▷ Danmarks Meteorologiske Institut, 85+ hosts
- ▷ IFW-Dresden, Germany, 100+ hosts, Cluster
- ▷ Physics department (FU Berlin), 139+ hosts
- ▷ University of New Orleans, 72 node Beowulf cluster
- ▷ Brown University, Dep. of Computer Science, 300+ hosts
- ▷ University of West Bohemia, Czech Republic, 180+
- ▷ Anonyme Firma, Electronic Payment Management
- ▷ Anonyme Firma, IDS boxen
- ▷ Anonyme Firma, Suchmaschine, 200+
- ▷ Institut für Physik, Universität Augsburg, 80+
- ▷ Institut für Theoretische Physik, Saarbrücken, 2 Cluster mit je 20 Knoten
- ▷ University Rekenentrum Rijksuniversiteit Groningen, Workstations und Server
- ▷ weitere Anwender mit heterogene Umgebungen, Desktops und Servern
- ▷ detaillierte Anwendungsberichte auf der FAI Webseite

p.21/22

Zusammenfassung

- ▷ Homepage: www.informatik.uni-koeln.de/fai
- ▷ aktive Maillingliste: linux-fai@uni-koeln.de
- ▷ CVS ro Zugriff, Beispiele der Log Dateien
- ▷ FAI läuft auf i386, IA64, SPARC, PowerPC
- ▷ Installiert auch Solaris SUN Sparc
- ▷ Centibots Projekt installierte 100 Roboter mit FAI

▷ FAI Demo in der Halle

p.22/22