

Automatic Installations and System updates with FAI

Overview, Functionality, Possibilities

Henning Sprang

Silpion IT Solutions GmbH/ LSP Hamburg

2007-07-11

Overview

- 1 Thoughts about systems installation
- 2 About FAI
- 3 Functions
- 4 Usage of FAI
- 5 Outlook and further applications/possibilities

Thoughts about system installation - starting position

- Manual installation has some some downsides:
 - Humans err when repeating the same task many times
 - Unefficient - install many systems takes a long time
- Solution: Automatisaton

disk images

- Create templates from “proper” installations
- Copy when needed and adjust them as needed
- Pros:
 - Low learning cost
 - Simple and fast implementation
- Cons:
 - Inflexibel - the smallest change requires rebuilding the image
 - Still manual work needed to get a installed system
 - Storage cost linear to number of different configurations

Installation with shell scripts

- Manual work replaced by shell scripts
- Pros:
 - Much lower storage cost than images
 - Higher flexibility
 - Tailor made
- Cons:
 - This is a real software development project (as you realize after some time when adding many small changes)
 - You solve every problem on your own, instead of reusing work of others

Usage of available auto-installers

There are ready solutions, that lighten the work to write all scripts on our own. . .

The most important projects and target distributions:

- Anaconda und Kickstart: Fedora-based
- Autoyast: SuSE-based
- Nlite/Unattend: Windows 2000, XP x86/x64 and 2003 x86/x64
- FAI: Debian-based ones, RHEL and other Fedora-based, SuSE, Mandriva, Windows, Solaris
- Others: Solaris Jumpstart, RedHat Cobbler/Koan

Why FAI?

- Flexible and easy to extend
- Simple, but powerful architecture – “everything is a shell script”
- Open development model
- Community support seasoned „Installers“
- Diverse client- and server- distributions
- Multiple installation types and system update
- Can be used for real hardware and virtualization systems

History

- Startet 1999 bei Thomas Lange at the University of cologne
- Base idea: structured and planned installation
- „Plan your installation, and FAI installs your plan“
- Part of the Debian Distribution
- Today about 10 active developers, small but nice community
- Since 2005 softupdate included – entwickelt an der FU Berlin

Who is using it for what?

- LiMux project in Munich installs and updates about 400, final 14000 Clients and Server
- Lycos-Europe installs 850 systems with FAI
- ComBots uses FAI for installing SLES9
 - 1200 systems
 - up to 300 at the same time
- Server for the OLPC Projekt updated with FAI
- Thomas Krenn(HW-vendor) installs customer systems with FAI
- Some top 500 High Performance Cluster
- Small home networks starting from 3 systems
- GRML is built with help of FAI

Overview

- FAI classes
 - In a class, system properties and actions that should be taken while installation, are defined.
 - A system can be assigned to multiple classes, and they can be combined arbitrary
- Server-distribution: Debian-based (Dependencies mainly NFS, TFTP, debootstrap)
- Target-distributions: Redhat, Debian, Ubuntu, SuSE, Mandriva, Fedora, Windows
- Different installation types
- Integrated versioning with subversion and CVS
- Softupdate for updates

Installation types

- Network installation with central install server
 - Client/Server architecture
- Directly calling dirinstall for chroots
- dirinstall with xen-tools for Xen domains
- fai-cd / fai-usb (for environments with no network connection to a mirror or an install server)

Process of the installation and update

- Preparations/manual actions:

- Boot via PXE/Bootfloppy/install-CD/USB Stick
- Mount the target directory and run *fai dirinstall*
- Update: run *fai softupdate*

here, the actual FAI tasks start(some internals neglected):

- *extrbase*: Unpack a minimal base image(cutom made or made by fai-setup)
- *defclass*: Class definition of the target system
- *partition*: Partitioning (not for softupdate)
- *debconf*: Debconf-preseedings
- *instsoft*: Software package installation
- *configure*: Run configuration scripts
- *savelog*: Push logfiles onto the install Server

Considerations – installation

- Decide the matching install type (net/cd/dirinstall)
- Plan your installation
 - Use cases
 - Network and environment
 - Software-packages
 - Configuration files and adjustments to the defaults
- Usage of local mirrors of software repositories

Considerations - Updates

- How and when should which patches be applied?
- Testing processes - where do I know from, which effect a patch/update has?
- Mirrors of security update repositories?
- Automatic(regular, timebased) or manual updates?

FAI setup and configuration

- Installation on Debian Etch:
 - *apt-get install fai-quickstart*
- Adjust install server setup in `/etc/fai`:
 - `fai.conf`: `LOGUSER=fai`, `LOGPROTO=ssh` for Logging via ssh
 - `apt/sources.list`: use local mirror if available
- FAI server is configured - create NFSroot:
 - *fai-setup*
- For PXE-Boot: *fai-chboot* to set boot-kernel and options
- Without PXE: *make-fai-bootfloppy*
- Installation from CD: *fai-cd*

Setup required infrastructure services

- For network install:
 - DNS entry for server and clients
 - DHCP config - Host/IP/MAC as usual
 - FAI-specific:

```
authoritative; # wegen IP_PNP_DHCP
option root-path "/usr/lib/fai/nfsroot ..."
server-name "faiserver"; # boot-server
next-server 172.20.2.64; # tftp server fuer kernel
filename "pxelinux.0";
```


Check and adjust FAI configspace

- The configurations of the install clienst are stored in the configspace, as text files
- The requirements from the installation plan are reflected here
- Structure of `/srv/fai/config`:

```
basefiles  
class  
debconf  
disk_config  
files  
hooks  
package_config  
scripts
```

Adjust configspace - class

- *class* contains class- and variable definitions.
- Simplest way: assign classes based on hostnames
- (Some)included sample classes: FAISERVER, GNOME, DEMO, XORG
- Every script that can echo a classname, can be used to assign classes
- That can be: check specific hardware, disk size, MAC or IP - everything

Adjust configspace – basefiles

- Contains minimal base images for special uses or non-Debian distributions
- At the start of the install, this directory is checked for images for defined classes
- You could also put an image here and skip the rest :)

Adjust configspace – disk_config

- Define haddisk configuration and mount points
- For paravirtualized Xen Domains often not necessary

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

```
disk_config disk1
primary / 150-300 rw,errors=remount-ro ; -c -j ext3
logical swap 40-500 rw
logical /var 90-1000 rw ; -m 5 -j ext3
logical /tmp 50-1000 rw ; -m 0 -j ext3
logical /usr 200-4000 rw ; -j ext3
logical /home 50- rw,nosuid ; -m 1 -j ext3
# logical /home preserve9 rw,nosuid ; -m 1 -j ext3
```

Adjust configspace - debconf

- Presets for package install scripts
- Only for dpkg-based distributions
- Works analog to Debian Installer

Adjust configspace - package_config

- Contents: files named by class names
- Purpose: Define packages to be installed
- Supports many installation methods:
 - install (apt-get)
 - aptitude
 - taskinst (Debian tasks=Package collections)
 - urpmi (mandriva)
 - yum (Fedora)
 - y2i (SuSE y2pms)
 - yast (SuSE yast -i)
 - Some more special ones...

Adjust configspace - package_config II

Example package_config/DEMO from simple examples:

```
PACKAGES aptitude  
fortune-mod fortunes  
rstat-client #rstatd  
rusers rusersd
```

```
# only when also class XORG is defined  
PACKAGES aptitude XORG  
bb frozen-bubble xpenguins
```

Adjust Configspace - scripts

- scripts contains scripts to be executed after package installation
- Usually shell-, Perl- and cfengine-scripts
- Need for others: just install Interpreter in the NFS-Root
- Naming scheme:
 <CLASSNAME>/<NUMBER>-<SCRIPTNAME>
- Number defines order of execution
- SCRIPTNAME arbitrary just for readability

scripts example:

```
.  
|-- AMD64  
|   |-- 99-discover-bug  
|-- DEMO  
|   |-- 10-misc  
|   |-- 30-demo  
|-- FAIBASE  
|   |-- 10-misc  
|   |-- 20-removable_media  
|   |-- 30-interface  
|   |-- 40-misc  
|-- FAISERVER  
|   |-- 10-conffiles  
|   |-- 20-copy-mirror  
|-- GRUB  
|   |-- 10-setup  
|-- LAST  
|-- 50-misc
```

Adjust configspace - files

- Structure of a filesystem, starting with /
- For usage with fcopy/ftar = classbases copy/unpack
- Copy single files explicitly, or recursive from / in a DEFAULT script

Adjust configspace - files II

- Target file=directory
- actually copied source file=CLASSNAME

```
.  
\-- etc  
|-- X11  
|   \-- xorg.conf  
|       |-- ATI  
|       \-- NVIDIA  
|-- apache2  
|   \-- conf.d  
|       \-- FAISERVER  
\-- fai  
\-- fai.conf  
\-- FAISERVER  
...
```

Adjust configspace - hooks

- Hook naming scheme: `.<CLASSNAME>[.source]`
- Execution before the according task
- Optional `skip_task`: The actual task will not be executed, for example for skipping partitioning of Xen Domains
- Example:

```
partition.XENU  
instsoft.FAIBASE  
savelog.LAST.source
```

Do the Installation

Depending on chosen install type:

- Start system via PXE
- Insert Bootfloppy/CD into the system and start
- Start Xen VM with `xm create vm-name.cfg install=1`
- `fai dirinstall <TARGETDIR>` into mounted blockdevice
- Call `dirinstall` in `xen-tools` from a `(xen-tools)` hook
- Start system with `fai-cd/fai-usb`

Inbetriebnahme

- Depending on number of packages system is installed in 3-30 minutes (Server/Desktop)
- Restart with production configuration (without `install=1` for Xen VM)
- Functionality tests (not scope of FAI)

Outlook and further applications/possibilities

- Helper scripts and configuration for other distributions as extra package *fai-multi-distribution*
- GOSA as LDAP and FAI GUI
- Automatic Tests of the installed systems:
 - hooks/scripts could check files and configurations
 - Crucible Test Framework
- Being worked on:
 - Build live-CD's with grml-live
 - Lightweight GUI without GOSA

Further informations

- WWW:
 - <http://www.informatik.uni-koeln.de/fai/>
 - <http://faiwiki.informatik.uni-koeln.de/>
 - <http://www.informatik.uni-koeln.de/fai/fai-guide.html>
 - <http://www.infrastructures.org/>
- Email und Chat:
 - IRC-Channel #fai im OFTC-Network
 - linux-fai-users und linux-fai-devel Mailinglisten

Fragen?

Questions?