System Imaging with KIWI

Jan-Christoph Bornschlegel <jcborn@suse.de>

SUSE Linux Products GmbH - Build Service Team

12th February 2008

System Imaging with `kiwi` – Overview

where are we

What kiwi is and what it's not

KIWI is:

- A command line based toolkit
- Usable as part of a process chain
- Usable as base tool for a high level application

KIWI is not:

- A product

kiwi history

- originated by **Marcus Schäfer**
- original purpose was creating “system on a stick”
- **James Willcox** (snorp) joins active development for Thin Client (SLETC)
- **Jigish Gohil** (CyberOrg) joins active development: LTSP project
- I join active development for Autobuild extension

I’ll get back to the last point later

Current project status

Used for the following products:

- SLEPOS – SuSE Linux Point of Sale
- SLETC – SuSE Linux Thin Client
- Hardware vendors for preload images

Community projects:

- Developers who deliver Live DVDs (KDE, openSUSE, . . .)
- users who want to make images containing their application
- . . . ? (You tell me)

where are we

Help Wanted!

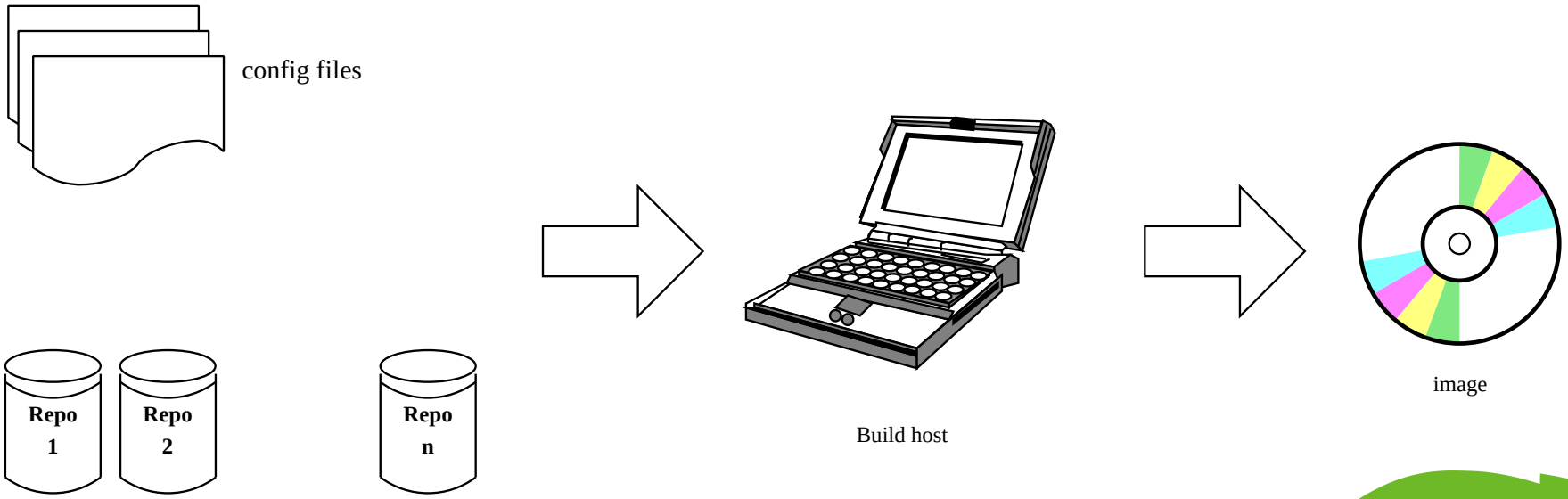
Documentation is available thoughout the web in various places

- `http://www.suse.de/~jcborn/kiwi-links.html`
- official documentation delivered with kiwi package

Setting up the buildhost

- Install **kiwi**, **kiwi-tools** and **kiwi-desc-*** packages
- create an image description file *config.xml*, or
- get and modify an existing image description

Buildsystem overview



Buildsystem overview II

- Package repositories (local, network)
- Configuration file(s)
- Buildhost with sufficient resources (esp. hdd)

where are we

Contents of the Configuration Directory

config.xml	contains every necessary image information(packages, repositories, settings, . . .
config.sh	customise the image after the packages are installed
image.sh	ditto
root/	contains overlay files which are included in the image or needed in scripts
other	special YaST files and others

The main thing to do is tweaking the *config.xml*.

where are we

Requirements and Regressions

- You developed a distributed application as Diploma Work (for example)
- You have to give a “public” demonstration
- You have virtually no control over the hardware for the demo

where are we

One Possible Solution

Your program is available as RPM package:

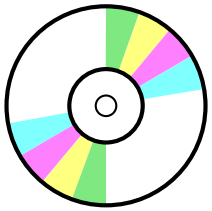
- add a plain RPM folder to your repositories
- add the package name to the <packages> section
- Build your image

One Possible Solution II

Your program is *not* available as RPM package:

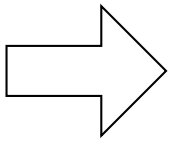
- Get a BuildService account and package it ;)
- put all needed files into your image manually
- Build your image

Repository Configuration Example

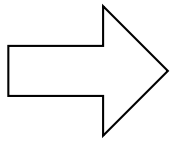


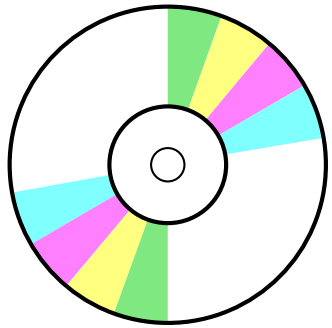
mounted DVD (base repo)

your home project



Build host





your own image

where are we

Purpose

- Autobuild is current internal package and media factory
- openSUSE BuildService will be the next generation package factory
- `kiwi` can not yet create installation sources
- not *yet*...

How it works

Autobuild is a distributed system

- Build clients build single RPMs based on a central scheduler and source base
- every employee’s machine can (and should) be a build host
- scheduler collects built rpm files to internal “full trees” for each codebase and architecture
- metadata is created

where are we

Current Product Creation

- full trees for target architectures are sync'd to dedicated machines
- rpm files are selected and collected to one repository
- metadata for this particular repository is created
- finally all sorts of media are made:
  - ftp repositories
  - CD, DVD, torrent, . . .

Product Creation with `kiwi`

Collecting the target repository must be integrated into `kiwi`.

- Expansion of the config.xml syntax
- add module for repository creation
- allow priority value for repositories
- allow exceptions
- implement script hooks

Autobuild knowledge is necessary to create package lists and scripts

Questions?

If your time (or hunger) allows. . .

Another talk is over

Thank you for your attention!
See you on

- `irc.freenode.net` #opensuse-kiwi
- for kiwi issues: `<kiwi-users@lists.berlios.de>`
- for packaging issues: `<opensuse-packaging@opensuse.org>`
- Bugzilla for kiwi: product “opensuse.org”, component “system imaging”