

OSCAR Package Compiler

May 15, 2007

INSTITUT NATIONAL
DE RECHERCHE
EN INFORMATIQUE
ET EN AUTOMATIQUE



Jean Parpaillon <jean.parpaillon@irisa.fr>
INRIA-IRISA

Geoffroy Vallée <valleegr@ornl.gov>
ORNL

Erich Focht <efocht@hpce.nec.com>
NEC HPC Europe

Agenda

- ✓ Introduction
- ✓ Original package management
- ✓ Limits
- ✓ opkgc
 - ✓ Concept
 - ✓ opkgc Specification
 - ✓ opkg Specification
 - ✓ Implementation
- ✓ Status
- ✓ Conclusion

Introduction

- ✓ OSCAR goals
 - ✓ Wizard based deployment of Linux clusters
 - ✓ Use distribution tools
 - ✓ Multi-distribution / multi-architecture
 - ✓ Cluster wide configuration

Need for a packaging API

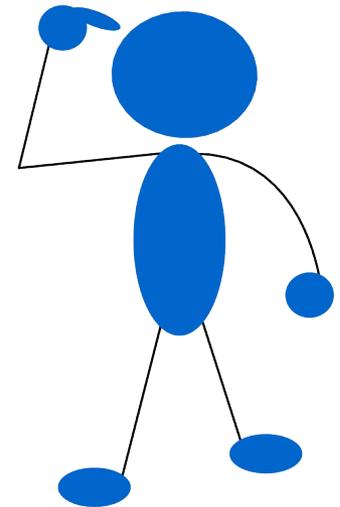
Original package management

```
config.xml
scripts/
    setup
    pre-configure
    post-configure
    post-clients
    post-install
    post-{server|client}-rpm-[un]install
testing/
doc/
SRPMS/
distro/
    <dist>-<arch>
    common-rpms
    common-debs
```



Limits

- ✓ An opkg (OSCAR Package) contains packages for all distributions: can be huge !
- ✓ As an original packaging system, we need to manage:
 - ✓ dependency relations
 - ✓ repositories
 - ✓ packages status (with database stuff and so on...)
- ✓ We reinvented the wheel !



opkgc – Concept

- ✓ Presented at OSCAR Annual Meeting '07 by Erich Focht
- ✓ Transform opkg description into native packages (deb or RPM)
- ✓ Use existing tools
 - ✓ package level: rpm, dpkg
 - ✓ repository level: yum, apt
- ✓ Compiler can check packages: files syntax, required files, etc.
- ✓ Ease new distribution support
 - ✓ most of work in opkgc

opkgc – Specification

- ✓ Produces 3 native packages:
 - ✓ `opkg-<name>`
 - ✓ Contains scripts for configuration and testing, documentation
 - ✓ Requires `oscar-base` package
 - ✓ `opkg-server-<name>`
 - ✓ Empty package
 - ✓ Requires `oscar-base-server` package
 - ✓ Other dependencies described in `config.xml`
 - ✓ `opkg-client-<name>`
 - ✓ Empty package
 - ✓ Requires `oscar-base-client` package
 - ✓ Other dependencies described in `config.xml`

opkg – Specifications

```
config.xml          # new XML schema
scripts/
    {api|server|client}-{pre|post}-[un]install
    # native packaging system scripts
    api-pre-configure
    api-post-configure
    api-post-image   # run on image, once created
    api-post-deploy  # run on nodes, once deployed
testing/
doc/
```

- ✓ **New** `config.xml` schema includes:
 - ✓ name, version, authors
 - ✓ changelog
 - ✓ dependencies: requires, provides, conflicts, suggests
 - ✓ version dependency informations (=, >, <, >=, <=)

opkgc – Implementation

- ✓ Python, object-oriented
- ✓ Use `lxml` + `Cheetah` templates system
 - ✓ Easy support for new distributions
- ✓ Compact (~1150 lines of code)
- ✓ Use config file (use standard python class `ConfigParser`)
- ✓ Validates input files (XML schema for `config.xml`)
- ✓ Produce packaging meta-files (`.spec`, Debian set of files)
- ✓ Build packages
- ✓ Usage: `cd <opkg dir>; opkgc --dist=<dist>`

Status

- ✓ Validated on Debian Sid and Fedora Core 5
- ✓ Released version 0.1
- ✓ Debian packages available
- ✓ Included `opkg-convert` migrates old packages
- ✓ Ongoing work:
 - ✓ Packages `oscar-base`, `oscar-base-{server|client}`
 - ✓ OSCAR package manager update
 - ✓ Repository management tools

Conclusion

- ✓ General goal of integrating OSCAR into distributions
 - ✓ Use (powerful) existing tools
 - ✓ Ease OSCAR development
- ✓ Use native packages keeping
 - ✓ cluster wide view
 - ✓ multi-distribution system
- ✓ Available in OSCAR 5.1
- ✓ Next step: provides OSCAR native packages then

```
apt-get install oscar | yum install oscar
```

- ✓ **Get opkgc:**
`http://oscar.gforge.inria.fr/downloads/opkgc-0.1.tar.gz`
- ✓ **Online wiki:**
`http://svn.oscar.openclustergroup.org/trac/oscar/wiki/DevelDocs`
- ✓ **Documentation included in package: docbook + samples**