

Hacking Pacman

The Arch Linux package manager in its gory details, how it gets the job done, and the supporting ecosystem behind our distro's most popular tool.

Dan McGee, dan@archlinux.org



About Me

- Arch Developer since early 2007
- Arch user since mid 2006
- Started by contributing to pacman
- Freelance software developer based in Chicago, IL



Basis for this presentation

The critical piece of any distro is the packages provided and how those packages are managed. While sometimes working with a package manager can be an exercise in frustration, many Arch users find pacman is straightforward and it does exactly what they want. This talk will look briefly at pacman at a high level, then dive down into the guts to see what it does and how it works. We may also touch on makepkg and other scripts. Most importantly, I will show you how the development process works if you are interested in getting involved or have a particular feature you want to see in pacman that doesn't exist yet.



Facts and Stats

```
dmcgee@galway ~/projects/pacman (master)
$ sloccount lib/ src/ scripts/ contrib/ test/
```

```
SLOC Directory SLOC-by-Language (Sorted)
8207    lib          ansic=8207
4350    src          ansic=4350
3761    test         python=3693,sh=68
2328    scripts       sh=2328
751     contrib      sh=630,perl=121
```

Totals grouped by language (dominant language first):

```
ansic:      12557 (64.74%)
python:     3693 (19.04%)
sh:         3026 (15.60%)
perl:       121 (0.62%)
```

```
Total Physical Source Lines of Code (SLOC)                = 19,397
Development Effort Estimate, Person-Years (Person-Months) = 4.50 (53.99)
  (Basic COCOMO model, Person-Months = 2.4 * (KSLOC**1.05))
Schedule Estimate, Years (Months)                          = 0.95 (11.38)
  (Basic COCOMO model, Months = 2.5 * (person-months**0.38))
Estimated Average Number of Developers (Effort/Schedule)  = 4.74
Total Estimated Cost to Develop                             = $ 607,803
  (average salary = $56,286/year, overhead = 2.40).
SLOCCount, Copyright (C) 2001-2004 David A. Wheeler
```



archcon2010
A simple, lightweight linux conference

Facts and Stats

- As of mid-July 2010
- 3022 revisions in GIT
- 99 authors total, 20 authors with >10 commits, and 6 authors with >100 commits
- 197 automated pactests, 58 vercmp tests
- 121 open bug or feature requests
- Pacman developers love candy



Big Current and Future Projects

- FS#8586, Tar database backend
 - For sync databases, read directly from db.tar.gz
 - Local DB won't change...for now
- FS#5331, Signed packages
 - Everyone has wanted them, but few have contributed good and useful work
- FS#16872, VCS packaging handling
 - We didn't do this perfectly the first time around, so it could use a bunch of improvement from what we've learned

Overall Architecture

- Split in two major components
 - Libalpm – backend, intended to be useful for more than just pacman purposes
 - Pacman – frontend, a wrapper around the API
- The backend API is...rough

Let's go to the code

- The all-important main() method
- Important backend structs and types
- The all-important sync.c
- DB reading and writing
- Dependency and delta resolving using graphs

“Normal” bug coding procedure

- If applicable, make a (failing) pactest
 - `make check`, your test will fail
- Fix the code
- Run the tests again, all should now pass
- Submit a patch including the test

“Normal” feature procedure

- Read the existing Flyspray issue report
- Work on the feature, pose questions to the ML as necessary
- Submit to the ML assuming you will have to make changes to get it accepted
- Refine the feature and patch until most people like it

Not a great C coder?

- Translations
 - Pacman/libalpm is translated to 19 languages
- Scripts
 - Outside of pacman, much is written in bash/python
- Documentation
 - Manpages all written using AsciiDoc so are very easy to work with
 - A volunteer to get API documentation working (through Doxygen?) would be appreciated

Contact Info

- Dan McGee
- Website: <http://www.toofishes.net/>
- Email: dan@archlinux.org
- Jabber: [toofishes@jabber.org](jabber:toofishes@jabber.org)
- IRC: toofishes (please don't PM me!)

