

Package search in modern package managers. NIH approach

Aleksey Cheusov
cheusov@NetBSD.org

Minsk LUG meeting, 24 Nov. 2012, Belarus

- **OBS**: more than 160.000 packages for 22 distributions
- **Debian/Linux**: about 29000 packages
- **FreeBSD**: more than 24000 ports
- **pkgsrc**: more than 12000 packages
- **AltLinux**: more than 12000 packages
- ...
- but package search facilities are still very rudimental

- **apt-cache(8)** — “It searches the package names and the descriptions for an occurrence of the regular expression...” ©. Why only regexp? Why only package name and description? How about file names? How about **apt-file(1)**?
- **yum(8)** — The search is limited to package names, summaries, descriptions, URL and regexp match.
- **pkgng(8)/pkg-search(8) (FreeBSD)** — Only a few search fields are available: one-line comment, description, origin, package name with/without version. The only available search strategy is regexp

- **pkgin(1) (NetBSD/pkgsrc)** — Package name and description again plus package category. Regexp search
- **pacman(8) (Arch Linux)** — It is also limited to package name and description.
- **pkgsrc pkg_info(8), dpkg(1), rpm(1), pacman(1)** and others are able to answer the following questions: “What installed package owns this file?”, “What installed package(s) depend on this one?”
- **aptitude(8)** — Much better than all of the above, but it is still limited

Four main questions about package search (I)

■ Where to search

- Binary package repository
- Installed packages
- Build specifications (rpm .spec, debian/, pkgsrc tree etc.)
- Online resources
- ...

■ What to match

- Package name
- One-line description of the package
- Full description of the package
- Package maintainer
- File lists
- Dependencies and dependants
- ...

Four main questions about package search (II)

- **How** to search
 - Exact match
 - Substring
 - Regexp search
 - Version matching (e.g. $\geq 5.14.0 < 5.16$)
 - More than one query in a single request (logical AND, OR and NOT, for example in CNF or DNF over simple queries)
 - ...
- **How much** information do we need
 - Package name or identifier
 - Short description of a package
 - Full information about a package
 - ...

Four main questions: conclusion

- A list of available search fields should be **variable** (package name, full description, maintainer etc.)
- A list of available search strategies should be **variable** (exact match, regular expression etc.)
- High level **AND**, **OR** and **NOT** constructions should be a part of **frontend** while actual search should be done in **backend**. Also, **backend** determines a list of supported search fields and search strategies

- Keeps a list of supported search fields and returns it on demand
- Keeps a list of supported search strategies and returns it on demand
- Given a search field, search strategies and a query, does a search and outputs identifiers of found packages
- Given a package identifier returns a description of the appropriate package

- Ask backend what search fields it support
- Ask backend what search strategies it support
- Call backend for actual search
- Provide support for AND, OR and NOT and operates on sets of package ids
- Provide short synonyms for search fields and strategies
- Do some postprocessing (formatting, coloring, GUI etc.)

The file `pkg_summary` contains information about each package in a binary package repository as a list of variable-value pairs. The variables describing different packages are separated by one empty line.

Source code

Makefile

```
PKGNAME=mrxvt-0.5.4nb6
DEPENDS=jpeg>=8nb1
DEPENDS=png>=1.5.0
COMMENT=Multi-tabbed terminal emulator with Xft support
PKGPATH=x11/mrxvt
HOMEPAGE=http://matern.sourceforge.net/
SIZE_PKG=420641
BUILD_DATE=2012-10-12 16:16:33 +0000
...
```

For the same purposes RFC822 is used in Debian and YAML in FreeBSD(`pkgng`)

- `pkg_digger_summary(1)` — search in `pkg_summary(5)` file
- `pkg_digger_installed(1)` — search in installed packages
- `pkg_online_client(1)` — search in `pkg_online` database provided by `dict://dict.mova.org:26280`

- **pkg_digger_summary -s**
output available search strategies
- **pkg_digger_summary -f**
output available search fields
- **pkg_digger_summary field:strat:query1 [f:s:q2...]**
search for packages and output packages ids
- **pkg_digger_summary -1|-3|-9|-i [-r] pkgid1 [pkgid2...]**
output description of the package
 - -1 output 1-line information about package
 - -3 output short information about package
 - -9|-i output full information about package
 - -r raw output in pkg_summary(5) format

Other backends have the same options

How it works

```
# pkg_digger_summary -s
exact    Match exactly
prefix  Match prefixes
suffix  Match suffixes
substring    Match substring
word     Match separate words
first    Match the first word
last     Match the last word
re       POSIX 1003.2 (modern) regular expressions
strfile  Match the words from file
strlist  Match the specified words
awk      Match using AWK expression
empty    Match an empty string
kw       "keyword" match
#
```

How it works

```
# export PKG_DIGGER_SUMMARY=/root/summary
# pkg_digger_summary -f
BUILD_DATE
CATEGORIES
COMMENT
CONFLICTS
DEPENDS
DESCRIPTION
...
PKGNAME
...
PKG_OPTIONS
PREV_PKGPATH
PROVIDES
REQUIRES
...
#
```

How it works

```
# pkg_digger_summary COMMENT:kw:'dns server'  
                COMMENT:kw:'dns proxy'  
net/maradns,maradns  
net/mydns-mysql,mydns-mysql  
net/mydns-pgsql,mydns-pgsql  
net/nsd,nsd  
net/rootprobe,rootprobe  
net/unbound,unbound  
net/totd,totd  
#
```

How it works

```
# pkg_digger_summary -1 net/unbound,unbound
net/unbound          - DNS resolver and recursive server
# pkg_digger_summary -3 net/unbound,unbound
-----
PKGNAME:             unbound-1.4.18
COMMENT:             DNS resolver and recursive server
CATEGORIES:         net
HOMEPAGE:           http://www.unbound.net/
PKGPATH:            net/unbound
DESCRIPTION:
    Unbound is an implementation of a DNS resolver.
    It provides a library...
    Homepage:
    http://www.unbound.net/

#
```


How it works

```
# pkg_digger_summary -9r net/unbound,unbound
PKGNAME=unbound-1.4.18
DEPENDS=ldns>=1.4
COMMENT=DNS resolver and recursive server
SIZE_PKG=6266495
BUILD_DATE=2012-11-09 19:13:30 +0000
CATEGORIES=net
HOMEPAGE=http://www.unbound.net/
LICENSE=modified-bsd

...
#
```

How it works

```
# pkg_online_client -s
    exact      Match headwords exactly
    prefix     Match prefixes
    nprefix    Match prefixes (skip, count)
    substring  Match substring occurring anywhere
              in a headword
    suffix     Match suffixes
    re         POSIX 1003.2 (modern) regular expressions
    regexp     Old (basic) regular expressions
    soundex    Match using SOUNDINDEX algorithm
    lev        Match headwords within
              Levenshtein distance one
    word       Match separate words within headwords
    first      Match the first word within headwords
    last       Match the last word within headwords
#
```

How it works

```
# pkg_online_client -f
PKGPATH
PKGNAME
PKGBASE
DEPENDS
BUILD_DEPENDS
CONFLICTS
HOMEPAGE
COMMENT
LICENSE
ONLYFOR
NOTFOR
MAINTAINER
CATEGORIES
PLIST
DESCRIPTION
#
```

How it works

```
# pkg_online_client PLIST:substring:bin/blockdiag
graphics/py-blockdiag,py26-blockdiag
graphics/py-blockdiag,py27-blockdiag
# pkg_online_client PKGBASE:prefix:dict
textproc/dict-client,dict-client
textproc/dict-dictionaries,dict-data
textproc/dict-mueller7,dict-mueller7
textproc/dict-server,dict-server
textproc/dictem,dictem
textproc/diction,diction
wip/dict-est-rus,dict-est-rus
wip/dict-freedict-eng-ara,dict-freedict-eng-ara
wip/dict-freedict-eng-fra,dict-freedict-eng-fra
#
```

Short synonyms for search fields

```
# export PKG_DIGGER_SUMMARY=/root/summary
# pkg_digger -f
synonym | full name
-----
C    CATEGORIES
c    COMMENT
d    DESCRIPTION
m    MAINTAINER
empty PKGBASE
n    PKGNAME
p    PKGPATH
f    PLIST
...
#
```

Short synonyms for search strategies

```
# pkg_digger -s
empty      exact      Match exactly
p          prefix    Match prefixes
u          suffix    Match suffixes
s          substring Match substring
w          word      Match separate words
f          first     Match the first word
l          last      Match the last word
r          re       POSIX 1003.2 (modern)
           regular expressions

#
```

Short synonyms for search fields/strategies

asrock# pkg_digger m:s:cheusov

```
archivers/heirloom-tar - Collection of standard Unix
audio/xmms-cdread      - XMMS input plugin that read...
audio/xmms-nas         - XMMS output plugin for Netw...
databases/libdbi       - Database Independent Abstra...
databases/libdbi-driver-mysql - MySQL driver for libdb...
databases/libdbi-driver-pgsql - PostgreSQL driver for ...
databases/libdbi-driver-sqlite - SQLite driver for lib...
databases/libdbi-driver-sqlite3 - SQLite3 driver for ...
devel/heirloom-getopt  - Collection of standard Unix...
devel/heirloom-libcommon - Collection of standard Unix...
devel/heirloom-what    - Collection of standard Unix...
devel/libjudy          - High-performance sparse dyn...
devel/libmaa           - General purpose data struct...
...
#
```

If neither field nor strategy was specified, fallback rules are used (by default PKGNAME:exact, PKGPATH:exact, PKGNAME:substring, COMMENT:word, COMMENT:substring). Frontend also implements operations on sets (Intersection and subtraction)

```
# pkg_digger dns server
net/dhisd - DynDNS server
net/fpdns - Fingerprinting DNS servers
net/maradns - Secure DNS server for *NIX...
net/mydns-mysql - MySQL-based DNS server
net/mydns-pgsql - PostgreSQL-based DNS serve...
net/p5-Net-DNSServer - Perl5 module to be used as...
net/powerdns - Modern, advanced and high ...
net/powerdns-recursor(pdns-recursor) - PowerDNS res...
net/rootprobe - Root DNS server performanc...
net/unbound - DNS resolver and recursive...
...
#
```


Finally! This approach was implemented in NIH

NIH is a package manager for pkgsrc (pkgtools/nih)

■ Where to search

- **nih search -b** — search in binary repository
- **nih search -l** — search in installed packages
- **nih search -o** — search in pkg_online database
- **nih search -p** — search in pkgsrc tree

■ What to match

- **nih search -f** — output supported search fields (-b|-l|-o|-p)

■ How to search

- **nih search -s** — output supported search strategies (-b|-l|-o|-p)
- if neither strategy nor field to search was specified we use fallback rules to make search easier and more efficient

■ How much information do we need

- **-1** — output 1-line information about found packages
- **-3** — output short information about found packages
- **-9|-i** — output full information about found packages

- Usage:

```
nih search -h
```

```
nih search [-b|-I|-o|-p] -s
```

```
nih search [-b|-I|-o|-p] -f
```

```
nih search [-b|-I|-o|-p] [-1|-3|-9|-i] query1 [query2..]
```

```
nih search [-b|-I|-o|-p] [-1|-3|-9|-i] query1 [query2..]
```

- Lines of code:

```
212 pkg_digger_summary
```

```
94 pkg_digger_installed
```

```
306 pkg_digger
```

```
330 pkg_online_client
```

```
250 pkg_grep_summary
```

```
254 ../share/runawk/pkg_grep_summary.awk
```

```
1446 total
```

- Reimplement **pkg_summary_installed** for better efficiency
- Partially reimplement **pkg_grep_summary** for better efficiency
- Currently **pkg_digger** is able to intersect and subtract sets of package ids. Support for union is needed.

- pkgtools/nih
(<https://github.com/cheusov/pkgnih>)
- pkgtools/pkg_summary-utils
(https://github.com/cheusov/pkg_summary-utils)
- pkgtools/pkg_online-client

The End