I wrote Distromatch, shall we use it?

Feb 4, 2012

Enrico Zini (enrico@debian.org)

Enrico Zini enrico@debian.org

Enrico Zini enrico@debian.org

Introduction

Distromatch is
the right thing to do.

Enrico Zini enrico@debian.org

What it does

It converts package names from one distribution to any other.

Code: `git://gitorious.org/appstream/distromatch.git`
Data `tp://dde.debian.net/exports/distromatch-all.tar.gz`
Simple query: `http://dde.debian.net/distromatch-frontend.html`

Demo.

```
mkdir dist
cd dist
tar zxf distromatch-all.tar.gz
cd ..
./distromatch --datadir=dist --reindex --verbose
./distromatch --datadir=dist debian libreoffice-writer
```

Enrico Zini enrico@debian.org

History

In January 2011, in Nürnberg (Germany) there has been a cross-distribution meeting about application installers: `http://distributions.freedesktop.org/wiki/AppStream`

During the meeting I wrote a first prototype, with data from the Debian ecosystem.

Olivier Thauvin gave me access to Sophie's database (`http://sophie.zarb.org/`) with data from all the RPM ecosystem.

I continued to work on it for a month after the meeting, until the codebase has become stable.

I tried to get data exports from Gentoo but failed.

Enrico Zini enrico@debian.org

How it works: design goals

Some use cases worth supporting:

- Stealing screenshots, tags, ratings and reviews from each other.
- Supporting developers finding dependencies when repackaging for other distributions.
- Cross-linking bug tracking systems.

Enrico Zini enrico@debian.org

How it works: design goals

Considering the use cases, maintenance effort should be focused on popular packages, and user-oriented packages.

- Desktop applications should be matched *well*, even at the cost of some manual handling of corner cases.
- Useful packages should be matched as well as possible, at the cost of investing some time in fine tuning the matching algorithms.
- Shared libs do not deserve much investment of time.

Manual handling of corner cases can be performed simply by adding specially crafted .desktop files to the packages. TTBOMK, this has not been done yet.

Enrico Zini enrico@debian.org

How it works: data required

binsrc.gz, mapping source names to binary names:

```
afbackup afbackup
afbackup-client afbackup
afbackup-common afbackup
afbinit afbinit
affiche.app affiche
afflib-dbg afflib
afflib-tools afflib
```

For source-only distribution, this could just map source names to themselves, and still act as a content list for the distribution.

Enrico Zini enrico@debian.org

How it works: data required

interesting-files.gz, listing important files contained in each package:

```
afbackup man man8/__inc_link.8.gz
[...]
afbackup man man8/xserverstatus.8.gz
afbackup-client man man1/__descrpt.1.gz
[...]
afbackup-client man man8/xrestore.8.gz
afbinit man man8/afbinit.8.gz
affiche.app bin Affiche
affiche.app man man1/Affiche.1.gz
afflib-tools bin afcat
[...]
afflib-tools bin s3-afflib
afflib-tools man man1/afcat.1.gz
afflib-tools man man1/affcat.1.gz
```

Given a list of all files in each package, this can be generated using regexps found in distromatch itself.

Enrico Zini enrico@debian.org

How it works: data required

In a nutshell, for a distribution to be supported it requires:

- A source<->binary package name mapping
- A list of all files contained in each package

Nothing else is required, and this is enough to convert package names to/from **any** other distribution.

Enrico Zini enrico@debian.org

How it works: algorithms

Distromatch is really a framework for running several matching strategies. These are currently implemented:

- Package name is the same.
- 'stemmed' package name is the same.
- Packages contain the same 'interesting' files.

Stemming algorithms exist for perl modules, python modules, development libraries, shared libraries.

.desktop files and pkg-config .pc files are 'interesting' enough to provide exact mapping between packages.

Fuzzy matches by files in bin/ or lib/, manpages and .py modules.

Enrico Zini enrico@debian.org

How it works: making the query

Distribution data are acquired in Xapian full-text indices with a custom term layout.

The real name, stemmed name and names of interesting files are indexed as terms with well defined prefixes.

Matching a package works like this:

1. Query the package by real name in the source distribution.
2. Get its term list (stemmed name and interesting files).
3. Use them to query the indices of all the other distributions.

All these operations are *fast* when done using a full-text index, like Xapian.

Enrico Zini enrico@debian.org

Existing deployment

To allow anyone to test the code, the is a working and up to date index maintained at dde.debian.net.

It can be queried via a CORS-aware, REST-ish API, at
`http://dde.debian.net/dde/q/distromatch/match`
see `http://dde.debian.net/dde/` for details

Raw datafiles are published at
`http://dde.debian.net/exports/`

Simple demo: static html query page at
`http://dde.debian.net/distromatch-frontend.html`

Enrico Zini enrico@debian.org

Missing distributions

At least these distributions are missing because I have not managed (or tried) yet to find contacts who could set up a data feed for me:

- Gentoo
- Arch
- *BSD
- CPAN, PyPI, CRAN, PEAR: having a good mapping to/from module repositories would have very interesting applications

Note that distromatch does any-to-any matching, network scale economy applies: adding a single data source potentially doubles the value of the service.

Enrico Zini enrico@debian.org

Maintenance

I'm not good at marketing my work, but why this isn't actively maintained by a cross-distribution team is beyond me. [possible explanation: noone has heard about it yet.]

I want to see a loosely coupled group, with at least one person per distribution, with two goals:

1. keep distromatch (the Python command-line tool and the data exports) working and accurate;
2. use it to improve one's own distribution by "stealing" data from all the others.

Tasks are simple: maintain data exports, collect reports of false matches, tweak regexps, push cool new data to your community. Come find me later or tomorrow.

Enrico Zini enrico@debian.org

What this gives us

How to steal screenshots from each other:

- step 1: take screenshots form screenshots.debian.net, converting package names via distromatch;
- step 2: collect screenshots from your community, so they have proper branding;
- step 3: let the people of screenshots.debian.net know, so they can steal back from you.

Note that the code of screenshots.debian.net is free and available at `http://debshots.workaround.org/trac/`

An elegant solution would be to have a screenshots website that falls back on other distribution sites when an image is missing.

Enrico Zini enrico@debian.org

What this gives us

Do you need categories? Get a Debtags export for your distribution! I already have a script that generates it somewhere.

Does some distro have ratings I can steal for Debian?

Cross link bug trackers: “See [debian/ubuntu/fedora] bugs for this package”

Cross link webpages, to make it easier for maintainers in one distribution to find their counterparts in another.

Enrico Zini enrico@debian.org

Conclusion

It is a working prototype, development is done, only maintenance is needed.

It is good enough: the data is not perfect, but usable.

The quality of matches can easily be improved:

1. with more people bringing their know-how and tweaking the matching;
2. with feedback as distromatch actually starts to get used.

If you would like to take care of distromatch for your distribution of choice, come find me later, or tomorrow.

Enrico Zini enrico@debian.org