Debsources as a Platform
All your Debian source are belong to us!

Stefano Zacchirolı, Matthieu Caneill

August 18, 2015
DebConf15 (Heidelberg, Germany)
Acknowledgements

Code
Initially developed at IRILL, by Stefano Zacchirola and Matthieu Caneill. Many people have contributed code and ideas since then.

Infrastructure
Debsources’ servers are sponsored by IRILL.
Table of contents

1 Introduction

2 Features
   - Debsources’ features
   - What’s new?
   - Roadmap

3 Technologies

4 Research platform

5 Hacking
1 Introduction

2 Features
   - Debsources’ features
   - What’s new?
   - Roadmap

3 Technologies

4 Research platform

5 Hacking
What is Debsources?

- A web application to browse the source code of Debian packages
- The infrastructure behind: database, plugins, ...
What is Debsources?

- A web application to browse the source code of Debian packages
- The infrastructure behind: database, plugins, ...

Play with it!

Navigate to http://sources.debian.net
Debian Sources

All Debian source are belong to us  — Anonymous [1]

Browse through the source code of the Debian operating system. Read more...

**Browse** by prefix

0 1 2 3 4 5 6 7 8 9 a b c d e f g h i j k
lib, lib3 liba libb libc libd libe libf
libg libh libi libj libk libl libm libn libo
libp libq libr libs libt libu libv libw libx
liby libz m n o p q r s t u v w x y z

**Search**

by package name: [Search package]

the source code (via codesearch):
[Search code]
```perl
#!/usr/bin/perl

##
## Cowsay 3.03
##
## This file is part of cowsay. (c) 1999-2000 Tony Monroe.
##
use Text::Tabs qw(expand);
use Text::Wrap qw(wrap fill $columns);
use File::Basename;
use Getopt::Std;
use Cwd;

if (m{"^UTF8:locale"}) {
    binmode STDIN, ':utf8';
    binmode STDOUT, ':utf8';
    require Encode;
    eval { $_ = Encode::decode_utf8($_,1) } for @ARGV;
}

$version = "3.03";

Cowsay
See? Cowsay variables are declared here.

$profilename = basename($0);
$yes = 'on';
$tongue = 'i';
$cowpath = $ENV{'COWPATH'} || '/usr/share/cowsay/cows';
$message = ();
$thoughts = "";
```

---

**Source code display**

File: cowsay

```
#!/usr/bin/perl

##
## Cowsay 3.03
##
## This file is part of cowsay. (c) 1999-2000 Tony Monroe.
##
use Text::Tabs qw(expand);
use Text::Wrap qw(wrap fill $columns);
use File::Basename;
use Getopt::Std;
use Cwd;

if (m{"^UTF8:locale"}) {
    binmode STDIN, ':utf8';
    binmode STDOUT, ':utf8';
    require Encode;
    eval { $_ = Encode::decode_utf8($_,1) } for @ARGV;
}

$version = "3.03";

Cowsay
See? Cowsay variables are declared here.

$profilename = basename($0);
$yes = 'on';
$tongue = 'i';
$cowpath = $ENV{'COWPATH'} || '/usr/share/cowsay/cows';
$message = ();
$thoughts = "";
```
So what? Is this really useful?

“I want to check the source code of cowsay. What do?”

**The old way**

```bash
cd /tmp
apt-get source cowsay
cd cowsay-3.03+dfsg1
... 
cd ..
rm -r cowsay-3.03+dfsg1
```

Note that it only works on Debian(-based) systems.
So what?  
Is this really useful?

“I want to check the source code of cowsay. What do?”

**The old way**

```bash
cd /tmp
apt-get source cowsay
cd cowsay-3.03+dfsg1
...
cd ..
```

```
rm -r cowsay-3.03+dfsg1
```

Note that it only works on Debian(-based) systems.

**The new way**

```bash
lynx http://sources.debian.net/src/cowsay/
```

Almost runs on your typewriter.
Table of contents

1 Introduction

2 Features
   - Debsources’ features
   - What’s new?
   - Roadmap

3 Technologies

4 Research platform

5 Hacking
Source code browsing

Syntax highlighting

For all languages supported by highlight.js: C, C++, Java, Python, Ruby, Makefile… and 112 others.
Source code browsing

Syntax highlighting

For all languages supported by highlight.js: C, C++, Java, Python, Ruby, Makefile... and 112 others.

Included versions

Packages in

hamm, slink, potato, woody, sarge, etch, lenny, squeeze, oldoldstable, oldstable, stable, testing, unstable, experimental, oldoldstable-proposed-updates, oldstable-proposed-updates, proposed-updates, testing-proposed-updates, oldoldstable-updates, oldstable-updates, stable-updates, wheezy-backports, jessie-backports, squeeze-lts

are in Debsources.
Searching

You can search for:

- Packages
- Files
- File content
  - ctags
  - regular expressions through codesearch.debian.net

Content indexing

All search data is indexed in Postgres, resulting in pretty decent performances (in spite of sub-optimal disk I/O).
Advanced search

**Package search**

*search for a package and allow to browse through its source code*

package

filter by suite

*Search package*

**File search**

*search for a specific source code file and display it*

by SHA256 hash

sha256

within package (optional)

*Search files*

**Code search**

*search all available source code for occurrences of specific features (regex matches, identifiers, etc.)*

via [Debian code search](#)

regular expression

*Search regex*

by ctags

tag

within package (optional)

*Search tags*
Annotations and highlighting

Use cases

- I’m a developer: *I want to share a pointer to a precise location in the source code of package X.*
- I’m a user: *An errors point me to line 42 in file Y, and I look for support on IRC.*
- I’m a static source code checker: *I found an issue in file Z, line 51.*
Annotations and highlighting

http://sources.debian.net/src/
package/version/path/to/file.c?hl=a:b&msg=a:b:c#LXX

package: cowsay
version: 3.03-3
path: cowsay
highlight: 32:36
message: 30:Debian:rocks
Annotations and highlighting

Developer: I want to share a precise location in the source code of package X.

```php
15 $version = "3.03";
16 $progame = basename($0);
17 $eyes = "oo";

About the eyes...
Hey, should we change the default?

18 $tongue = " ";
19 $cowpath = $ENV{'COWPATH'} || '/usr/share/cowsay/cows';
20 @message = ();
21 $thoughts = "";
```
Annotations and highlighting

User: I can't compile software X, it fails at line 42 in the file Y.

```c
47  int main( int argc, char *argv[] );
48  void showDetails( MemoryStoragePtr );
49  void showHeader( CSAHeaderPtr )

Compilation error
My compiler fails here. What can I do?

50  void showGroups( CSAGrpHeaderPtr );
51  void showSemaphores( CSASemHeaderPtr );
```
Annotations and highlighting

Static analyzer: I found a semantic error in file Y.

```c
len *= 2;
}

if ((res == (size_t) -1) || (res == (size_t) - 1))
{
    /* The string cannot be converted. */
    if (use_malloc)
    {
        free (wmessage);
    }
```
Duplicated files

All the files are in the database, along with their checksum. The duplicates can be computed, for every file.

File: COPYING

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007
Duplicated files

Checksum: 8ceb4b9ee5adedde47b31e975c1d90c73ad27b6b165a1dc

4309 results:

- 3depict/0.0.10-1/COPYING
- 3depict/0.0.16-2.1/COPYING
- 3dldf/2.0.3+dfsg-2/COPYING
- 3dldf-doc/2.0.3+ndfsg-2/COPYING
- 4store/1.1.4-2/COPYING
- 4store/1.1.5-1/COPYING
- 4store/1.1.6-1/COPYING
- aac-tactics/0.2.pl2-7/COPYING
- aac-tactics/0.4-3/COPYING
- aaphoto/0.38-2/COPYING
- aaphoto/0.41-1.1/COPYING
- aaphoto/0.43.1-1/COPYING
- aaphoto/0.43.1-3/COPYING
- abby/0.4.7-1/COPYING
Integration in the Debian infrastructure

**Codesearch**

http://codesearch.debian.net is used for regular expression searches, and redirects back to Debsources to consult results.

Credits: Michael Stapelberg
Integration in the Debian infrastructure

**Codesearch**

http://codesearch.debian.net is used for regular expression searches, and redirects back to Debsources to consult results. 
Credits: Michael Stapelberg

**Package tracking systems**

The old PTS (http://packages.qa.debian.org) and the new tracker (http://tracker.debian.org) provide links to Debsources (“browse source code”).
Credits: Paul Wise, Raphael Hertzog
Integration in the Debian infrastructure

**Codesearch**

http://codesearch.debian.net is used for regular expression searches, and redirects back to Debsources to consult results.
Credits: Michael Stapelberg

**Package tracking systems**

The old PTS (http://packages.qa.debian.org) and the new tracker (http://tracker.debian.org) provide links to Debsources ("browse source code").
Credits: Paul Wise, Raphael Hertzog

**Need to embed code somewhere?**

<iframe>s embedding of files content is supported (see documentation).

(or talk to use for more proper integration)
Statistics

- Source code metrics for every package
- Plugins: disk size, ctags, sloccount
Statistics

- Source code metrics for every package
- Plugins: disk size, ctags, sloccount

```plaintext
package info [click to toggle]
chromium-browser 41.0.2272.118-1
- links: PTS, VCS
- area: main
- in suites: jessie, sid
- size: 2,241,900 kB
- ctags: 1,909,592
- SLOC: cpp: 9,691,826; anscic: 3,341,113; python: 712,689;
  asm: 518,779; xml: 208,926;
  java: 169,820; sh: 119,353;
  perl: 68,907; makefile: 28,311;
  yacc: 13,305; objc: 11,385; tcl:
  3,186; cs: 2,225; sql: 2,217;
  lex: 2,215; lisp: 1,349; pascal:
  1,256; awk: 407; ruby: 155;
  sed: 53; php: 14; exp: 11
```
Statistics

Aggregated statistics are available at http://sources.debian.net/stats/.

Metrics

- Disk usage
- SLOC (source lines of code)
- Number of source packages
- Number of files
- Number of ctags (symbols)
## Statistics

Currently in *sid*:

<table>
<thead>
<tr>
<th>Category</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source files</td>
<td>11,787,950</td>
</tr>
<tr>
<td>Source packages</td>
<td>23,846</td>
</tr>
<tr>
<td>Disk usage</td>
<td>228,599,736 kB</td>
</tr>
<tr>
<td>Ctags</td>
<td>127,884,129</td>
</tr>
<tr>
<td>Source lines of code</td>
<td>1,082,453,728</td>
</tr>
</tbody>
</table>


---

**Stefano Zacchioli, Matthieu Caneill**

Debsources as a Platform

**DebConf15, 18 August 2015**
## Statistics

Currently in *sid*:

<table>
<thead>
<tr>
<th>Category</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source files</td>
<td>11,787,950</td>
</tr>
<tr>
<td>Source packages</td>
<td>23,846</td>
</tr>
<tr>
<td>Disk usage</td>
<td>228,599,736 kB</td>
</tr>
<tr>
<td>Ctags</td>
<td>127,884,129</td>
</tr>
<tr>
<td>Source lines of code</td>
<td>1,082,453,728</td>
</tr>
</tbody>
</table>

1. C: 439,197,216
2. C++: 275,342,652
3. Python: 46,067,009

See [http://sources.debian.net/stats/sid/](http://sources.debian.net/stats/sid/) for more
Fancy graphs
All functionalities available via the Web UI, are also available via a JSON-based HTTP API:

**Examples**

```bash
curl http://sources.debian.net/api/ping/
{
  "status": "ok",
  "http_status_code": 200,
  "last_update": "Mon, 17 Aug 2015 10:43:27 -0000"
}

curl http://s.d.n/api/info/package/cowsay/3.03-3/
{
  "pkg_infos": {
    "suites": [
      "woody"
    ]
  }
```

Documentation at http://sources.debian.net/doc/api/.
What’s new?

Many new features since DebConf14

Outreachy student: Jingjie Jiang (sophiejjj)

GSoC students: Clément Schreiner (clemux), Orestis Ioannou (orestis)

Many new contributors:

▶ Stefano Zacchiroli: 721
▶ Matthieu Caneill: 483
▶ Orestis Ioannou: 118
▶ sophiejjj: 58
▶ Clément Schreiner: 19
▶ Jason Pleau: 12
▶ Akshita Jha: 7
▶ Jingjie Jiang: 5
▶ Paul Wise: 1
▶ tessy joseph: 1
▶ sodamatt: 1
▶ Luciano Bello: 1
▶ Christophe Siraut: 1
▶ James McCoy: 1
▶ Tapasweni Pathak: 1
What’s new?

- Many new features since DebConf14
What’s new?

- Many new features since DebConf14
- Outreachy student: Jingjie Jiang (sophiejjj)
What’s new?

- Many new features since DebConf14
- Outreachy student: Jingjie Jiang (sophiejjj)
- GSoC students: Clément Schreiner (clemux), Orestis Ioannou (orestis)
What’s new?

- Many new features since DebConf14
- Outreachy student: Jingjie Jiang (sophiejjj)
- GSoC students: Clément Schreiner (clemux), Orestis Ioannou (orestis)
- Many new contributors:
  - Stefano Zacchirolì: 721
  - Matthieu Caneill: 483
  - Orestis Ioannou: 118
  - sophiejjj: 58
  - Clément Schreiner: 19
  - Jason Pleau: 12
  - Akshita Jha: 7
  - Jingjie Jiang: 5
  - Paul Wise: 1
  - tessay joseph: 1
  - sodamatt: 1
  - Luciano Bello: 1
  - Christophe Siraut: 1
  - James McCoy: 1
  - Tapasweni Pathak: 1
What's new?

Multiple pop-up messages

Credits: Orestis Ioannou and Jason Pleau.
Application: generated code annotations.

```php
$version = "3.03";

// Cowsay
See? Cowsay variables are declared here.

$program = basename($0);
$eyes = "oo";

// Oh hello
I am just another dummy pop-up example

$tongue = "  ";
$cowpath = $ENV{`COWPATH`} || `/usr/share/cowsay/cows`;
$message = ()
$thoughts = "";
```
What’s new?
Blueprints support

Credits: Jingjie Jiang.
Application: new apps plugged.

Blueprints
- Flask apps embedded and plugged together (decentralization)
- Implied a big refactoring
- Enabled the development of new features (GSoC)
What’s new?

Detailed directory listing

Credits: Jingjie Jiang

Folder: 3.03+dfsg1-10

.. (parent)

drw-r-xr-x 4,096 cows

drw-r-xr-x 4,096 debian

-rw-r--r-- 931 ChangeLog

-rw-r--r-- 385 INSTALL

-rw-r--r-- 1,116 LICENSE

-rw-r--r-- 445 MANIFEST

-rw-r--r-- 1,610 README

-rw-r--r-- 879 Wrap.pm.diff

drw-r-xr-x 4,421 cowsay

-rw-r--r-- 4,693 cowsay.6

-rwxr-xr-x 2,275 install.sh

-rw-r--r-- 631 pgp_public_key.txt
What’s new?
File edition in-browser

Credits: Raphael Geissert

File edition
A plugin for Iceweasel and Chromium enables the edition of files in your browser.
A patch ready-to-be-sent™ is generated!
It makes the entire Debian archive editable from a browser.
What’s new?
Blueprint app: License information

Credits: Orestis Ioannou (GSoC student)

debian/copyright
Part of the archive uses machine-readable debian/copyright
What’s new?
Blueprint app: License information

Credits: Orestis Ioannou (GSoC student)

debian/copyright

Part of the archive uses **machine-readable debian/copyright**
- Parse and display this information in the web interface
What’s new?

Blueprint app: License information

Credits: Orestis Ioannou (GSoC student)

debian/copyright

Part of the archive uses machine-readable debian/copyright

- Parse and display this information in the web interface
- Compute statistics about license usage
What’s new?
Blueprint app: License information

Credits: Orestis Ioannou (GSoC student)

debian/copyright

Part of the archive uses **machine-readable debian/copyright**
- Parse and display this information in the web interface
- Compute statistics about license usage
- SPDX (generic license exchange format) export of the copyright file
What’s new?
Blueprint app: License information

Credits: Orestis Ioannou (GSoC student)

debian/copyright

Part of the archive uses machine-readable debian/copyright

- Parse and display this information in the web interface
- Compute statistics about license usage
- SPDX (generic license exchange format) export of the copyright file
- API
What's new?

Blueprint app: License information

Package: aircrack-ng / 1:1.2-0~beta3-4

Header

Format: http://www.debian.org/doc/packaging-manuals/copyright-format/1.0/
Upstream-Name: Aircrack-ng
Upstream-Contact: Thomas d'Otreppe <tdotreppe@aircrack-ng.org>
Source: http://download.aircrack-ng.org/aircrack-ng-1.2-beta3.tar.gz

Files

*  
Copyright: 2006-2014, Thomas d'Otreppe
License: GPL-2+

lib/csharp/MonoExample/NDesk-dbus/*
Copyright: 2006-2007, Alp Toker <alp@atoker.com>
License: MIT
What’s new?
Blueprint app: Patch tracker

Credits: Orestis Ioannou (GSoC student)

Tracking patches of a package
- Intended as a replacement of the old patch-tracker
What’s new?
Blueprint app: Patch tracker

Credits: Orestis Ioannou (GSoC student)

Tracking patches of a package
- Intended as a replacement of the old patch-tracker
- Currently supports 3.0 (quilt) format
What’s new?
Blueprint app: Patch tracker

Credits: Orestis Ioannou (GSoC student)

Tracking patches of a package
- Intended as a replacement of the old patch-tracker
- Currently supports 3.0 (quilt) format
- Syntax-highlighting
What’s new?
Blueprint app: Patch tracker

Credits: Orestis Ioannou (GSoC student)

Tracking patches of a package

- Intended as a replacement of the old patch-tracker
- Currently supports 3.0 (quilt) format
- Syntax-highlighting
- API
Package: beignet / 1.0.0-1

### Metadata

<table>
<thead>
<tr>
<th>Package</th>
<th>Version</th>
<th>Patches format</th>
</tr>
</thead>
<tbody>
<tr>
<td>beignet</td>
<td>1.0.0-1</td>
<td>3.0 (quilt)</td>
</tr>
</tbody>
</table>

### Patch series

<table>
<thead>
<tr>
<th>Patch</th>
<th>File delta</th>
<th>Description</th>
<th>View</th>
<th>Download</th>
</tr>
</thead>
<tbody>
<tr>
<td>Debian compliant compiler flags handling</td>
<td><code>CMakeLists.txt</code></td>
<td>7 +++--- respect cflags, etc and be verbose</td>
<td>View</td>
<td>download</td>
</tr>
<tr>
<td>Enhance debug output</td>
<td><code>src/cl_util.h</code></td>
<td>1 + 1 file changed, 1 insertion(+)</td>
<td>View</td>
<td>download</td>
</tr>
</tbody>
</table>

enhance debug output
What’s new?
Asynchronous updater

Credits: Clément Schreiner (GSoC student)

Asynchronous updater

- Rewrite of the stages of the updater (add-package, compute-stats, gc, etc)
- Using celery to spawn independent and asynchronous tasks.
For more information about these features

- Watch the video of the GSoC session (happened earlier this afternoon!)
- Features soon available at sources.debian.net (currently being merged and deployed).
What’s new?
And many many other features...

- Refactoring
What’s new?
And many many other features...

- Refactoring
  - Debsources as a top-level Python module
  - Configuration loader
  - Flake8 compliance (Zack, Jingjie Jiang, and others)
  - Test coverage (Jingjie Jiang, Clément Schreiner, and others)
  - 84% now!
  - Case-insensitive package name search (Akshita Jha)
  - Better statistics charts (Orestis Ioannou)
  - Python3 support (Zack)
What’s new?
And many many other features...

- Refactoring
  - Debsources as a top-level Python module
  - Configuration loader
What’s new?
And many many other features...

- Refactoring
  - Debsources as a top-level Python module
  - Configuration loader
  - Flake8 compliance (Zack, Jingjie Jiang, and others)
What’s new?
And many many other features...

- Refactoring
  - Debsources as a top-level Python module
  - Configuration loader
  - Flake8 compliance (Zack, Jingjie Jiang, and others)
- Test coverage (Jingjie Jiang, Clément Schreiner, and others)
What’s new?
And many many other features...

- **Refactoring**
  - Debsources as a top-level Python module
  - Configuration loader
  - Flake8 compliance (Zack, Jingjie Jiang, and others)

- **Test coverage** (Jingjie Jiang, Clément Schreiner, and others)
  - 84% now!
What’s new?
And many many other features...

- Refactoring
  - Debsources as a top-level Python module
  - Configuration loader
  - Flake8 compliance (Zack, Jingjie Jiang, and others)
- Test coverage (Jingjie Jiang, Clément Schreiner, and others)
  - 84% now!
- Case-insensitive package name search (Akshita Jha)
What’s new?
And many many other features...

- Refactoring
  - Debsources as a top-level Python module
  - Configuration loader
  - Flake8 compliance (Zack, Jingjie Jiang, and others)
- Test coverage (Jingjie Jiang, Clément Schreiner, and others)
  - 84% now!
- Case-insensitive package name search (Akshita Jha)
- Better statistics charts (Orestis Ioannou)
What’s new?
And many many other features...

- Refactoring
  - Debsources as a top-level Python module
  - Configuration loader
  - Flake8 compliance (Zack, Jingjie Jiang, and others)
- Test coverage (Jingjie Jiang, Clément Schreiner, and others)
  - 84% now!
- Case-insensitive package name search (Akshita Jha)
- Better statistics charts (Orestis Ioannou)
- Python3 support (Zack)
Roadmap

Static analysis

- Automatic runs of static analysis tools (e.g. clang, coccinelle) on all Debian packages
- Statistics gathering on bugs evolution
- → Debile, Firewoes
Roadmap
And many smaller items

- more live statistics (about licenses, patches, etc)
Roadmap
And many smaller items

- more live statistics (about licenses, patches, etc)
- file name search
Roadmap
And many smaller items

- more live statistics (about licenses, patches, etc)
- file name search
- binary package → source package redirection
Roadmap

And many smaller items

- more live statistics (about licenses, patches, etc)
- file name search
- binary package $\rightarrow$ source package redirection
- tarball-in-tarball support
Roadmap
And many smaller items

- more live statistics (about licenses, patches, etc)
- file name search
- binary package → source package redirection
- tarball-in-tarball support
- 100% test suite coverage (one day...)
Roadmap

And many smaller items

- more live statistics (about licenses, patches, etc)
- file name search
- binary package → source package redirection
- tarball-in-tarball support
- 100% test suite coverage (one day...)
- file-level deduplication
  - `select count(*) from checksums;` → 35'370'653
  - `select count(distinct sha256) from checksums;` → 15'822'745
  ⇒ deduplicated core: ≈ 45%
Table of contents

1 Introduction

2 Features
   • Debsources’ features
   • What’s new?
   • Roadmap

3 Technologies

4 Research platform

5 Hacking
Technologies
What languages and technologies do we use?

- **Code base**: (almost) entirely in Python
Technologies
What languages and technologies do we use?

- **Code base**: (almost) entirely in Python
- **Web application**: Flask framework, Jinja2 templates, HTML/CSS/Javascript
Technologies
What languages and technologies do we use?

- **Code base**: (almost) entirely in Python
- **Web application**: Flask framework, Jinja2 templates, HTML/CSS/Javascript
- **Database**: PostgreSQL
Technologies
What languages and technologies do we use?

- **Code base**: (almost) entirely in Python
- **Web application**: Flask framework, Jinja2 templates, HTML/CSS/Javascript
- **Database**: PostgreSQL
- Apache web server, SQLAlchemy, ...
### Overview

#### Data Model (Excerpt)

**Checksums**
- `id` (serial, PK)
- `package_id` (int, FK)
- `file_id` (int, FK)
- `sha256` (varchar)

**Files**
- `id` (serial, PK)
- `package_id` (int, FK)
- `path` (bytea)

**Packages**
- `id` (serial, PK)
- `version` (varchar)
- `name_id` (int, FK)
- `area` (varchar)
- `sticky` (boolean)

**CTags**
- `id` (serial, PK)
- `package_id` (int, FK)
- `tag` (varchar)
- `file_id` (int, FK)
- `line` (int)
- `kind` (varchar)
- `language` (lang_c_tags)

**Metrics**
- `id` (serial, PK)
- `package_id` (int, FK)
- `metric` (metric_type)
- `value_` (int)

**Sloccounts**
- `id` (serial, PK)
- `package_id` (int, FK)
- `language` (lang_sloc)
- `count` (int)

**Suites**
- `id` (serial, PK)
- `package_id` (int, FK)
- `suite` (varchar)

---

#### Table Rows

<table>
<thead>
<tr>
<th>Table</th>
<th>Rows</th>
</tr>
</thead>
<tbody>
<tr>
<td>suites_info</td>
<td>16</td>
</tr>
<tr>
<td>package_names</td>
<td>29,286</td>
</tr>
<tr>
<td>(source) packages</td>
<td>83,597</td>
</tr>
<tr>
<td>suites (mapping)</td>
<td>120,550</td>
</tr>
<tr>
<td>metrics (e.g., du)</td>
<td>83,597</td>
</tr>
<tr>
<td>sloccounts</td>
<td>298,360</td>
</tr>
<tr>
<td>checksums</td>
<td>35,370,653</td>
</tr>
<tr>
<td>ctags</td>
<td>358,773,259</td>
</tr>
</tbody>
</table>

---

1 snapshot, 31 July 2014
Disk usage

- unpacked sources: 805 GB
- PostgreSQL DB: 145 GB
- Source mirror: 135 GB

Hosting requirements: ≈ 1.1 TB

(17 August 2015)
Disk usage

- unpacked sources: 805 GB
- PostgreSQL DB: 145 GB
- Source mirror: 135 GB

Hosting requirements: \( \approx 1.1 \ \text{TB} \)

(17 August 2015)

Figure: unpacked sources trend (peek due to archive.d.o injection)
Table of contents

1 Introduction

2 Features
   - Debsources’ features
   - What’s new?
   - Roadmap

3 Technologies

4 Research platform

5 Hacking
Debsources is a huge software collection.

Homogeneous: all software follow Debian’s packaging format.

It is up-to-date.
## Facts
- Debsources is a **huge** software collection.
- Homogeneous: all software follow Debian’s packaging format.
- It is up-to-date.

## Software evolution
- 20 years of source code evolution.
- Plugins to compute stats.
Research platform

Facts

- Debsources is a huge software collection.
- Homogeneous: all software follow Debian’s packaging format.
- It is up-to-date.

Software evolution

- 20 years of source code evolution.
- Plugins to compute stats.

Nice charts can be computed with this!
Example: What are the trending programming languages?
Research platform
Software metrics evolution over Debian releases

[Graph showing the evolution of various software metrics over Debian releases from 1999 to 2015.]
Research platform
File size per language, evolution over Debian releases
Research platform

Absolute evolution of SLOC per language, over Debian releases

[Graph showing the absolute evolution of SLOC per language over Debian releases from 1999 to 2015. The graph uses a vertical bar chart with different colors and patterns to represent the SLOC in millions for various languages and releases.]
Research platform
Relative evolution of SLOC per language, over Debian releases


You can find the PDFs of the articles on http://sources.debian.net/doc/.
Table of contents

1 Introduction

2 Features
   • Debsources’ features
   • What’s new?
   • Roadmap

3 Technologies

4 Research platform

5 Hacking
Hacking
How can I contribute?

Step 1: clone Debsources git repository

```bash
git clone git://anonscm.debian.org/qa/debsources.git
```

Step 2: Set-up a development environment

Follow the instructions in the HACKING file, or use Docker!

```bash
bin/debsources-docker-build && bin/debsources-docker-run
```

This will setup a Docker container with all batteries included: dependencies, database, test data, configuration.

Step 3: open your editor and hack!

Bugs list:

- [https://bugs.debian.org/cgi-bin/pkgreport.cgi?pkg=qa.debian.org;tag=debsources](https://bugs.debian.org/cgi-bin/pkgreport.cgi?pkg=qa.debian.org;tag=debsources)
- or: implement your own plugin (see examples), add features, etc.
Step 1: clone Debsources git repository

```bash
git clone git://anonscm.debian.org/qa/debsources.git
```

Step 2: Set-up a development environment

- Follow the instructions in the HACKING file,
- Or use Docker! `bin/debsources-docker-build && bin/debsources-docker-run` will setup a Docker container with all batteries included: dependencies, database, test data, configuration.
Hacking

How can I contribute?

Step 1: clone Debsources git repository

git clone git://anonscm.debian.org/qa/debsources.git

Step 2: Set-up a development environment

- Follow the instructions in the HACKING file,
- Or use Docker! `bin/debsources-docker-build &&
  bin/debsources-docker-run` will setup a Docker container with all
  batteries included: dependencies, database, test data, configuration.

Step 3: open your editor and hack!

Bugs list: https://bugs.debian.org/cgi-bin/pkgreport.cgi?pkg=
qa.debian.org;tag=debsources
or: implement your own plugin (see examples), add features, etc.
Thanks! Questions?

Stefano Zacchirolì, Matthieu Caneill
info@sources.debian.net

Slides: Licensed CC-BY-SA.