

Packaging GNOME in Debian

A desktop that just works

Josselin Mouette

Debian GNOME team

2–3 April 2011

GNOME.Asia Summit 2011, Bangalore



Outline

1 The Debian GNOME team

2 The Debian distribution workflow

3 How to help

A long time ago, in a very different situation...

As the GNOME desktop version 2 lacks large parts of GNOME 1.4's functionality, and suffers from incompatibilities, i'm willing to make it possible to install Gnome 1 on a Debian system, without conflicting with Gnome 2.

01 November 2002

→ Does it remind of anything?

- At that time, **cluttered** packaging
- **Low collaboration** between involved people
- Personality issues
- Bad **communication** with the GNOME project

A disastrous result for only ~20 packages

A long time ago, in a very different situation. . .

As the GNOME desktop version 2 lacks large parts of GNOME 1.4's functionality, and suffers from incompatibilities, i'm willing to make it possible to install Gnome 1 on a Debian system, without conflicting with Gnome 2.

01 November 2002

→ Does it remind of anything?

- At that time, **cluttered** packaging
- **Low collaboration** between involved people
- Personality issues
- Bad **communication** with the GNOME project

A disastrous result for only ~20 packages

A long time ago, in a very different situation...

As the GNOME desktop version 2 lacks large parts of GNOME 1.4's functionality, and suffers from incompatibilities, i'm willing to make it possible to install Gnome 1 on a Debian system, without conflicting with Gnome 2.

01 November 2002

→ Does it remind of anything?

- At that time, **cluttered** packaging
- **Low collaboration** between involved people
- Personality issues
- Bad **communication** with the GNOME project

A disastrous result for only ~20 packages

A long time ago, in a very different situation...

As the GNOME desktop version 2 lacks large parts of GNOME 1.4's functionality, and suffers from incompatibilities, i'm willing to make it possible to install Gnome 1 on a Debian system, without conflicting with Gnome 2.

01 November 2002

→ Does it remind of anything?

- At that time, **cluttered** packaging
- **Low collaboration** between involved people
- Personality issues
- Bad **communication** with the GNOME project

A disastrous result for only ~20 packages

The Debian GNOME team today

- Collective responsibility of all packages for a number of skilled maintainers
- Synchronization between all desktop modules
- Close to the release team

Now the norm for any large subsystem in Debian: KDE, X, kernel, installer, Perl, Python...

<http://wiki.debian.org/Teams>

Some figures:

- 218 source packages, amounting for 663 binary packages
- + Specific teams for Evolution, GStreamer and Freedesktop
- 28 registered contributors, ~5 major contributors
- ~1 month to package a full GNOME version

The Debian GNOME team today

- **Collective responsibility** of all packages for a number of **skilled** maintainers
- Synchronization between all desktop modules
- Close to the **release team**

Now the norm for any large subsystem in Debian: KDE, X, kernel, installer, Perl, Python...

<http://wiki.debian.org/Teams>

Some figures:

- 218 source packages, amounting for 663 binary packages
- + Specific teams for Evolution, GStreamer and Freedesktop
- 28 registered contributors, ~5 major contributors
- ~1 month to package a full GNOME version

The Debian GNOME team today

- **Collective responsibility** of all packages for a number of **skilled** maintainers
- Synchronization between all desktop modules
- Close to the **release team**

Now the norm for any large subsystem in Debian: KDE, X, kernel, installer, Perl, Python...

<http://wiki.debian.org/Teams>

Some figures:

- 218 source packages, amounting for 663 binary packages
- + Specific teams for Evolution, GStreamer and Freedesktop
- 28 registered contributors, ~5 major contributors
- ~1 month to package a full GNOME version

The Debian GNOME team today

- **Collective responsibility** of all packages for a number of **skilled** maintainers
- Synchronization between all desktop modules
- Close to the **release team**

Now the norm for any large subsystem in Debian: KDE, X, kernel, installer, Perl, Python...

<http://wiki.debian.org/Teams>

Some figures:

- 218 source packages, amounting for 663 binary packages
- + Specific teams for Evolution, GStreamer and Freedesktop
- 28 registered contributors, ~5 major contributors
- ~1 month to package a full GNOME version

Our goals

1 A desktop for everyone

- ▶ Stable releases that can last **3 years** for enterprise users
- ▶ A rolling release, **CUT** (*Constantly Usable Testing*)
- ▶ The latest stable platform for developers in *unstable*
- ▶ Technology previews in *experimental*

2 No really, we mean everyone

- ▶ Support for all **14 Debian architectures**
- ▶ Includes support for *kfreebsd* and (sometimes) for *Hurd*

3 Close to upstream

- ▶ **100%** of blessed modules are packaged
- ▶ No large-scale changes
- ▶ All patches are (*should*) be forwarded upstream

4 Completeness

- ▶ Installation tasks & **metapackages**
- ▶ 4 layers: session, core (1 CD), desktop, full
- ▶ Consistency: limited choices

Our goals

1 A desktop for everyone

- ▶ Stable releases that can last **3 years** for enterprise users
- ▶ A rolling release, **CUT** (*Constantly Usable Testing*)
- ▶ The latest stable platform for developers in *unstable*
- ▶ Technology previews in *experimental*

2 No really, we mean everyone

- ▶ Support for all **14 Debian architectures**
- ▶ Includes support for *kfreebsd* and (sometimes) for *Hurd*

3 Close to upstream

- ▶ **100%** of blessed modules are packaged
- ▶ No large-scale changes
- ▶ All patches are (*should*) be forwarded upstream

4 Completeness

- ▶ Installation tasks & **metapackages**
- ▶ 4 layers: session, core (1 CD), desktop, full
- ▶ Consistency: limited choices

Our goals

1 A desktop for everyone

- ▶ Stable releases that can last **3 years** for enterprise users
- ▶ A rolling release, **CUT** (*Constantly Usable Testing*)
- ▶ The latest stable platform for developers in *unstable*
- ▶ Technology previews in *experimental*

2 No really, we mean everyone

- ▶ Support for all **14 Debian architectures**
- ▶ Includes support for kfreebsd and (sometimes) for Hurd

3 Close to upstream

- ▶ **100%** of blessed modules are packaged
- ▶ No large-scale changes
- ▶ All patches are (*should*) be forwarded upstream

4 Completeness

- ▶ Installation tasks & **metapackages**
- ▶ 4 layers: session, core (1 CD), desktop, full
- ▶ Consistency: limited choices

Our goals

1 A desktop for everyone

- ▶ Stable releases that can last **3 years** for enterprise users
- ▶ A rolling release, **CUT** (*Constantly Usable Testing*)
- ▶ The latest stable platform for developers in *unstable*
- ▶ Technology previews in *experimental*

2 No really, we mean everyone

- ▶ Support for all **14 Debian architectures**
- ▶ Includes support for kfreebsd and (sometimes) for Hurd

3 Close to upstream

- ▶ **100%** of blessed modules are packaged
- ▶ No large-scale changes
- ▶ All patches are (*should*) be forwarded upstream

4 Completeness

- ▶ Installation tasks & **metapackages**
- ▶ 4 layers: session, core (1 CD), desktop, full
- ▶ Consistency: limited choices

Our goals

1 A desktop for everyone

- ▶ Stable releases that can last **3 years** for enterprise users
- ▶ A rolling release, **CUT** (*Constantly Usable Testing*)
- ▶ The latest stable platform for developers in *unstable*
- ▶ Technology previews in *experimental*

2 No really, we mean everyone

- ▶ Support for all **14 Debian architectures**
- ▶ Includes support for kfreebsd and (sometimes) for Hurd

3 Close to upstream

- ▶ **100%** of blessed modules are packaged
- ▶ No large-scale changes
- ▶ All patches are (*should*) be forwarded upstream

4 Completeness

- ▶ Installation tasks & **metapackages**
- ▶ 4 layers: session, core (1 CD), desktop, full
- ▶ Consistency: limited choices

Outline

1 The Debian GNOME team

2 The Debian distribution workflow

3 How to help

The Debian suites

- 1 oldstable
 - ▶ Previous stable version
 - ▶ Only security support, rare bug fixes
- 2 stable
 - ▶ One every ~ 2 years, lasts ~ 3 years
 - ▶ Only **important** bug fixes
- 3 testing
 - ▶ The next stable release, constantly releasable
 - ▶ New goal: **constantly usable**
 - ▶ Packages flow in from unstable
 - ★ after 10 days (simple changes)
 - ★ batches of self-contained **transitions** (e.g. one library SONAME change)
- 4 unstable
 - ▶ Developers usually upload here
 - ▶ Packages must be **release quality**
- 5 experimental
 - ▶ Testbed where you can break anything
 - ▶ Not a suite per se, but an overlay on unstable
 - ▶ You don't need all of experimental

The Debian suites

1 oldstable

- ▶ Previous stable version
- ▶ Only security support, rare bug fixes

2 stable

- ▶ One every ~2 years, lasts ~3 years
- ▶ Only **important** bug fixes

3 testing

- ▶ The next stable release, constantly releasable
- ▶ New goal: **constantly usable**
- ▶ Packages flow in from unstable
 - ★ after 10 days (simple changes)
 - ★ batches of self-contained **transitions** (e.g. one library SONAME change)

4 unstable

- ▶ Developers usually upload here
- ▶ Packages must be **release quality**

5 experimental

- ▶ Testbed where you can break anything
- ▶ Not a suite per se, but an overlay on unstable
- ▶ You don't need all of experimental

The Debian suites

- 1 oldstable
 - ▶ Previous stable version
 - ▶ Only security support, rare bug fixes
- 2 stable
 - ▶ One every ~ 2 years, lasts ~ 3 years
 - ▶ Only **important** bug fixes
- 3 testing
 - ▶ The next stable release, constantly releasable
 - ▶ New goal: **constantly usable**
 - ▶ Packages flow in from unstable
 - ★ after 10 days (simple changes)
 - ★ batches of self-contained **transitions** (e.g. one library SONAME change)
- 4 unstable
 - ▶ Developers usually upload here
 - ▶ Packages must be **release quality**
- 5 experimental
 - ▶ Testbed where you can break anything
 - ▶ Not a suite per se, but an overlay on unstable
 - ▶ You don't need all of experimental

The Debian suites

- 1 oldstable
 - ▶ Previous stable version
 - ▶ Only security support, rare bug fixes
- 2 stable
 - ▶ One every ~ 2 years, lasts ~ 3 years
 - ▶ Only **important** bug fixes
- 3 testing
 - ▶ The next stable release, constantly releasable
 - ▶ New goal: **constantly usable**
 - ▶ Packages flow in from unstable
 - ★ after 10 days (simple changes)
 - ★ batches of self-contained **transitions** (e.g. one library SONAME change)
- 4 unstable
 - ▶ Developers usually upload here
 - ▶ Packages must be **release quality**
- 5 experimental
 - ▶ Testbed where you can break anything
 - ▶ Not a suite per se, but an overlay on unstable
 - ▶ You don't need all of experimental

The Debian suites

- 1 oldstable
 - ▶ Previous stable version
 - ▶ Only security support, rare bug fixes
- 2 stable
 - ▶ One every ~ 2 years, lasts ~ 3 years
 - ▶ Only **important** bug fixes
- 3 testing
 - ▶ The next stable release, constantly releasable
 - ▶ New goal: **constantly usable**
 - ▶ Packages flow in from unstable
 - ★ after 10 days (simple changes)
 - ★ batches of self-contained **transitions** (e.g. one library SONAME change)
- 4 unstable
 - ▶ Developers usually upload here
 - ▶ Packages must be **release quality**
- 5 experimental
 - ▶ Testbed where you can break anything
 - ▶ Not a suite per se, but an overlay on unstable
 - ▶ You don't need all of experimental

The Debian suites

- 1 oldstable
 - ▶ Previous stable version
 - ▶ Only security support, rare bug fixes
- 2 stable
 - ▶ One every ~ 2 years, lasts ~ 3 years
 - ▶ Only **important** bug fixes
- 3 testing
 - ▶ The next stable release, constantly releasable
 - ▶ New goal: **constantly usable**
 - ▶ Packages flow in from unstable
 - ★ after 10 days (simple changes)
 - ★ batches of self-contained **transitions** (e.g. one library SONAME change)
- 4 unstable
 - ▶ Developers usually upload here
 - ▶ Packages must be **release quality**
- 5 experimental
 - ▶ Testbed where you can break anything
 - ▶ Not a suite per se, but an overlay on unstable
 - ▶ You don't need all of experimental

Debian GNOME packages workflow

In GNOME development cycles

- Packaging of core libraries in experimental
 - ▶ Mostly glib, pango, gtk, GStreamer
 - ▶ Work shared with Ubuntu
- Later, core desktop in experimental

Released GNOME

- Packages go in unstable in order
- **Scheduling** of transitions with the Debian release team
 - Tricks to avoid **entangled transitions**
- All stable GNOME updates get packaged

Debian GNOME packages workflow

In GNOME development cycles

- Packaging of core libraries in experimental
 - ▶ Mostly glib, pango, gtk, GStreamer
 - ▶ Work shared with Ubuntu
- Later, core desktop in experimental

Released GNOME

- Packages go in unstable in order
- **Scheduling** of transitions with the Debian release team
 - Tricks to avoid **entangled transitions**
- All stable GNOME updates get packaged

Debian GNOME packages workflow

In GNOME development cycles

- Packaging of core libraries in experimental
 - ▶ Mostly glib, pango, gtk, GStreamer
 - ▶ Work shared with Ubuntu
- Later, core desktop in experimental

Released GNOME

- Packages go in unstable in order
- **Scheduling** of transitions with the Debian release team
 - Tricks to avoid **entangled transitions**
- All stable GNOME updates get packaged

Preparing a Debian release

Choice of a GNOME version with the release team

- *squeeze*: GNOME 2.30 with bits of 2.32

Conservative technology choices

- No PulseAudio by default
- No GSettings
- In lenny (GNOME 2.20/2.22): no gvfs

During the *freeze*, long *polishing* phase

- Lots of bug fixes stolen upstream
- Patches for targeted UI issues
- No large-scale changes (most of the time)

After the release

- A few more targeted fixes, mostly crashers
- Security support starts

Preparing a Debian release

Choice of a GNOME version **with the release team**

- *squeeze*: GNOME 2.30 with bits of 2.32

Conservative technology choices

- No PulseAudio by default
- No GSettings
- In lenny (GNOME 2.20/2.22): no gvfs

During the *freeze*, long **polishing** phase

- Lots of bug fixes stolen upstream
- Patches for targeted UI issues
- No large-scale changes (most of the time)

After the release

- A few more targeted fixes, mostly crashers
- Security support starts

Preparing a Debian release

Choice of a GNOME version **with the release team**

- *squeeze*: GNOME 2.30 with bits of 2.32

Conservative technology choices

- No PulseAudio by default
- No GSettings
- In lenny (GNOME 2.20/2.22): no gvfs

During the *freeze*, long **polishing** phase

- Lots of bug fixes stolen upstream
- Patches for targeted UI issues
- No large-scale changes (most of the time)

After the release

- A few more targeted fixes, mostly crashers
- Security support starts

Preparing a Debian release

Choice of a GNOME version **with the release team**

- *squeeze*: GNOME 2.30 with bits of 2.32

Conservative technology choices

- No PulseAudio by default
- No GSettings
- In lenny (GNOME 2.20/2.22): no gvfs

During the *freeze*, long **polishing** phase

- Lots of bug fixes stolen upstream
- Patches for targeted UI issues
- No large-scale changes (most of the time)

After the release

- A few more targeted fixes, mostly crashers
- Security support starts

Preparing a Debian release

Choice of a GNOME version **with the release team**

- *squeeze*: GNOME 2.30 with bits of 2.32

Conservative technology choices

- No PulseAudio by default
- No GSettings
- In lenny (GNOME 2.20/2.22): no gvfs

During the *freeze*, long **polishing** phase

- Lots of bug fixes stolen upstream
- Patches for targeted UI issues
- No large-scale changes (most of the time)

After the release

- A few more targeted fixes, mostly crashers
- Security support starts

Our tools

Integrating with Debian **packages**

- `dh_gconf` for registering schemas and defaults
- `dh_girepository` + GI policy & naming scheme
- **CDBS** classes to ease packaging
 - Small packages are trivial
- Triggers for registering desktop files and icons

Keeping close to **upstream**

- Automated download scripts
- Information flowing from Bugzilla: BTS link
- New upstream releases: DEHS
- The GNOME status **summary**

Our tools

Integrating with Debian **packages**

- `dh_gconf` for registering schemas and defaults
- `dh_girepository` + GI policy & naming scheme
- **CDBS** classes to ease packaging
 - Small packages are trivial
- Triggers for registering desktop files and icons

Keeping close to **upstream**

- Automated download scripts
- Information flowing from Bugzilla: BTS link
- New upstream releases: DEHS
- The GNOME status **summary**

Our tools

Integrating with Debian **packages**

- `dh_gconf` for registering schemas and defaults
- `dh_girepository` + GI policy & naming scheme
- **CDBS** classes to ease packaging
 - Small packages are trivial
- Triggers for registering desktop files and icons

Keeping close to **upstream**

- Automated download scripts
- Information flowing from Bugzilla: BTS link
- New upstream releases: DEHS
- The GNOME status **summary**



Status of GNOME 3.0 in Debian

17 out of 116 packages are up to date

GNOME

	Upstream	Debian	Team	Archs
core				
atk	PTS 1.91.92	1.32.0-2	-	all
atkmm	PTS 2.22.3	2.22.2-1	-	all
brasero	PTS 2.91.91	2.30.3-2	-	13/0
cantarell-fonts	0.0.5			
clutter	PTS 1.6.10	1.6.10-3	-	6/13
clutter-gtk	PTS 0.91.8	0.91.8-2	-	7/9
dconf	PTS 0.7.3	0.7.3-1	-	2/13
empathy	PTS 2.91.92	2.91.92-1	-	3/13
eog	PTS 2.91.92	2.91.91-4	-	all
epiphany	PTS 2.91.92	2.91.91-2	-	all
evince	PTS 2.91.92	2.30.3-3	-	14/0
evolution-data-server	PTS 2.91.92	2.91.91-1	-	all
gcalctool	PTS 5.91.92	5.91.91-1	-	11/13
GConf	PTS 2.32.1	2.32.1-2	-	all
gdk-pixbuf	PTS 2.23.1	2.23.0-2	-	all
gjs	PTS 0.7.13	0.7.13-2	-	7/12
glib	PTS 2.28.4	2.28.2-1	-	11/0
glib-networking	PTS 2.28.4	2.28.4-1	-	9/0
glibmm	PTS 2.27.98	2.24.2-1	-	1/0
gnome-backgrounds	PTS 2.91.92	2.32.0-1	-	all
gnome-bluetooth	PTS 2.91.92	2.91.5-1	-	all
gnome-control-center	PTS 2.91.92	1:2.91.92-1	-	2/12
gnome-desktop	PTS 2.91.92	2.91.93-1	-	9/13
gnome-disk-utility	PTS 2.91.7	2.30.1-2	-	11/0
gnome-doc-utils	PTS 0.20.5	0.20.4-1	-	all

Up-to-date in testing

Up-to-date in unstable

Up-to-date in

experimental

Not up-to-date, lagging
by a minor versionNot up-to-date, lagging
by a major version

Not in Debian

 Display arch details

Outline

- 1 The Debian GNOME team
- 2 The Debian distribution workflow
- 3 How to help**

Why we need help

Simple... for everything we're **bad** at.

- Packaging when GNOME is in RC or released is **late**
 - ▶ We discover **important issues** too late
 - ▶ **Patches** don't make it before the next GNOME release
 - ▶ Packaging development releases is **rough**
 - Targeted to specific packages
- Polishing of specific issues
 - ▶ We have to **arbitrate** between all problems
 - ▶ Example in *squeeze*: package management **inconsistencies**
- **Long-term maintenance** of given GNOME releases
 - ▶ Decisions **constrained** by release plannings
 - ▶ **Sharing the load** with other distributions
 - ▶ Currently, mostly BZ as an exchange media for patches

Why we need help

Simple... for everything we're **bad** at.

- Packaging when GNOME is in RC or released is **late**
 - ▶ We discover **important issues** too late
 - ▶ **Patches** don't make it before the next GNOME release
 - ▶ Packaging development releases is **rough**
 - Targeted to specific packages
- Polishing of specific issues
 - ▶ We have to **arbitrate** between all problems
 - ▶ Example in *squeeze*: package management **inconsistencies**
- **Long-term maintenance** of given GNOME releases
 - ▶ Decisions **constrained** by release plannings
 - ▶ **Sharing the load** with other distributions
 - ▶ Currently, mostly BZ as an exchange media for patches

Why we need help

Simple... for everything we're **bad** at.

- Packaging when GNOME is in RC or released is **late**
 - ▶ We discover **important issues** too late
 - ▶ **Patches** don't make it before the next GNOME release
 - ▶ Packaging development releases is **rough**
 - Targeted to specific packages
- Polishing of specific issues
 - ▶ We have to **arbitrate** between all problems
 - ▶ Example in *squeeze*: package management **inconsistencies**
- **Long-term maintenance** of given GNOME releases
 - ▶ Decisions **constrained** by release plannings
 - ▶ **Sharing the load** with other distributions
 - ▶ Currently, mostly BZ as an exchange media for patches

Why we need help

Simple... for everything we're **bad** at.

- Packaging when GNOME is in RC or released is **late**
 - ▶ We discover **important issues** too late
 - ▶ **Patches** don't make it before the next GNOME release
 - ▶ Packaging development releases is **rough**
 - Targeted to specific packages
- Polishing of specific issues
 - ▶ We have to **arbitrate** between all problems
 - ▶ Example in *squeeze*: package management **inconsistencies**
- **Long-term maintenance** of given GNOME releases
 - ▶ Decisions **constrained** by release plannings
 - ▶ **Sharing the load** with other distributions
 - ▶ Currently, mostly BZ as an exchange media for patches

Why we need help

Simple... for everything we're **bad** at.

- Packaging when GNOME is in RC or released is **late**
 - ▶ We discover **important issues** too late
 - ▶ **Patches** don't make it before the next GNOME release
 - ▶ Packaging development releases is **rough**
 - Targeted to specific packages
- Polishing of specific issues
 - ▶ We have to **arbitrate** between all problems
 - ▶ Example in *squeeze*: package management **inconsistencies**
- **Long-term maintenance** of given GNOME releases
 - ▶ Decisions **constrained** by release plannings
 - ▶ **Sharing the load** with other distributions
 - ▶ Currently, mostly BZ as an exchange media for patches

Bug triage

Still the shortest path to **being helpful**

- **Anyone** can help, you only need a machine running *unstable*.
- Getting **useful** reports
 - ▶ Is the description **accurate**?
 - ▶ Can the bug be **reproduced**?
 - ▶ For crashers, does it have a usable **backtrace**?
Unfortunately, still no **debugging packages**
- **Forwarding** bugs upstream
 - ▶ Does it already have a fix in git?
 - ▶ Does it have a duplicate in Bugzilla?
Plea: we want the duplicate finder back
- Getting it **closed**
 - ▶ Unreproducible/old reports
 - ▶ Propose packaging changes
 - ▶ Pointers to upstream patches

→ Get your Subversion access in **no time!**

Bug triage

Still the shortest path to **being helpful**

- **Anyone** can help, you only need a machine running *unstable*.
- Getting **useful** reports
 - ▶ Is the description **accurate**?
 - ▶ Can the bug be **reproduced**?
 - ▶ For crashers, does it have a usable **backtrace**?
Unfortunately, still no **debugging packages**
- **Forwarding** bugs upstream
 - ▶ Does it already have a fix in git?
 - ▶ Does it have a duplicate in Bugzilla?
Plea: we want the duplicate finder back
- Getting it **closed**
 - ▶ Unreproducible/old reports
 - ▶ Propose packaging changes
 - ▶ Pointers to upstream patches

→ Get your Subversion access in **no time!**

Bug triage

Still the shortest path to **being helpful**

- **Anyone** can help, you only need a machine running *unstable*.
- Getting **useful** reports
 - ▶ Is the description **accurate**?
 - ▶ Can the bug be **reproduced**?
 - ▶ For crashers, does it have a usable **backtrace**?
Unfortunately, still no **debugging packages**
- **Forwarding** bugs upstream
 - ▶ Does it already have a fix in git?
 - ▶ Does it have a duplicate in Bugzilla?
Plea: we want the duplicate finder back
- Getting it **closed**
 - ▶ Unreproducible/old reports
 - ▶ Propose packaging changes
 - ▶ Pointers to upstream patches

→ Get your Subversion access in **no time!**

Bug triage

Still the shortest path to **being helpful**

- **Anyone** can help, you only need a machine running *unstable*.
- Getting **useful** reports
 - ▶ Is the description **accurate**?
 - ▶ Can the bug be **reproduced**?
 - ▶ For crashers, does it have a usable **backtrace**?
Unfortunately, still no **debugging packages**
- **Forwarding** bugs upstream
 - ▶ Does it already have a fix in git?
 - ▶ Does it have a duplicate in Bugzilla?
Plea: we want the duplicate finder back
- Getting it **closed**
 - ▶ Unreproducible/old reports
 - ▶ Propose packaging changes
 - ▶ Pointers to upstream patches

→ Get your Subversion access in **no time!**

Bug triage

Still the shortest path to **being helpful**

- **Anyone** can help, you only need a machine running *unstable*.
- Getting **useful** reports
 - ▶ Is the description **accurate**?
 - ▶ Can the bug be **reproduced**?
 - ▶ For crashers, does it have a usable **backtrace**?
Unfortunately, still no **debugging packages**
- **Forwarding** bugs upstream
 - ▶ Does it already have a fix in git?
 - ▶ Does it have a duplicate in Bugzilla?
Plea: we want the duplicate finder back
- Getting it **closed**
 - ▶ Unreproducible/old reports
 - ▶ Propose packaging changes
 - ▶ Pointers to upstream patches

→ Get your Subversion access in **no time!**

Bug triage

Still the shortest path to **being helpful**

- **Anyone** can help, you only need a machine running *unstable*.
- Getting **useful** reports
 - ▶ Is the description **accurate**?
 - ▶ Can the bug be **reproduced**?
 - ▶ For crashers, does it have a usable **backtrace**?
Unfortunately, still no **debugging packages**
- **Forwarding** bugs upstream
 - ▶ Does it already have a fix in git?
 - ▶ Does it have a duplicate in Bugzilla?
Plea: we want the duplicate finder back
- Getting it **closed**
 - ▶ Unreproducible/old reports
 - ▶ Propose packaging changes
 - ▶ Pointers to upstream patches

→ Get your Subversion access in **no time!**

Bug triage

Still the shortest path to **being helpful**

- **Anyone** can help, you only need a machine running *unstable*.
- Getting **useful** reports
 - ▶ Is the description **accurate**?
 - ▶ Can the bug be **reproduced**?
 - ▶ For crashers, does it have a usable **backtrace**?
Unfortunately, still no **debugging packages**
- **Forwarding** bugs upstream
 - ▶ Does it already have a fix in git?
 - ▶ Does it have a duplicate in Bugzilla?
Plea: we want the duplicate finder back
- Getting it **closed**
 - ▶ Unreproducible/old reports
 - ▶ Propose packaging changes
 - ▶ Pointers to upstream patches

→ Get your Subversion access in **no time!**

Communication with upstream

Interested by the state of a module?

- Subscribe to the development **mailing list**
- Subscribe to the **PTS** for your module:

<http://packages.qa.debian.org/>

→ The example of totem

Checking that patches are forwarded

- Debian → Upstream
 - ▶ <http://patch-tracker.debian.org/>
- GNOME → downstream
 - ▶ Tell distributions@gnome.org when a change should be **backported**
 - ▶ Tell us when a package generates a lot of **duplicate** crashers

Communication with upstream

Interested by the state of a module?

- Subscribe to the development **mailing list**
- Subscribe to the **PTS** for your module:

<http://packages.qa.debian.org/>

→ The example of totem

Checking that patches are forwarded

- Debian → Upstream
 - ▶ <http://patch-tracker.debian.org/>
- GNOME → downstream
 - ▶ Tell distributions@gnome.org when a change should be **backported**
 - ▶ Tell us when a package generates a lot of **duplicate** crashers

Communication with upstream

Interested by the state of a module?

- Subscribe to the development **mailing list**
- Subscribe to the **PTS** for your module:

<http://packages.qa.debian.org/>

→ The example of totem

Checking that patches are forwarded

- Debian → Upstream
 - ▶ <http://patch-tracker.debian.org/>
- GNOME → downstream
 - ▶ Tell distributions@g.o when a change should be **backported**
 - ▶ Tell us when a package generates a lot of **duplicate** crashers

Getting in touch

Primary contact: IRC

→ #debian-gnome @OFTC

Mailing lists:

- Discussions at debian-gtk-gnome@lists.debian.org (low traffic)
- Bugs/uploads:
pkg-gnome-maintainers@lists.alioth.debian.org (high traffic)

And don't forget: we do not bite. Most of the time.

Thank you for your attention.
Any questions?

Getting in touch

Primary contact: IRC

→ #debian-gnome @OFTC

Mailing lists:

- Discussions at `debian-gtk-gnome@lists.debian.org` (low traffic)
- Bugs/uploads:
`pkg-gnome-maintainers@lists.alioth.debian.org` (high traffic)

And don't forget: we do not bite. Most of the time.

Thank you for your attention.
Any questions?

Getting in touch

Primary contact: IRC

→ #debian-gnome @OFTC

Mailing lists:

- Discussions at `debian-gtk-gnome@lists.debian.org` (low traffic)
- Bugs/uploads:
`pkg-gnome-maintainers@lists.alioth.debian.org` (high traffic)

And don't forget: we do not bite. Most of the time.

Thank you for your attention.
Any questions?

Getting in touch

Primary contact: **IRC**

→ #debian-gnome @OFTC

Mailing lists:

- Discussions at `debian-gtk-gnome@lists.debian.org` (low traffic)
- Bugs/uploads:
`pkg-gnome-maintainers@lists.alioth.debian.org` (high traffic)

And don't forget: we do not bite. Most of the time.

Thank you for your attention.
Any questions?