Triaging Bugs in GNOME

Tobias Mueller

2009-12-05
FOSS.in 2009
1 Introduction
   • About us
   • Motivation

2 Bug Triaging
   • Description
   • GNOME Methodology
   • Beginners Guide
   • Best Practice Pitfalls

3 QnA
   • Status Fields
   • Triaging Diagram
   • Questions
**About us**

<table>
<thead>
<tr>
<th>Tobias</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Jabber</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td><strong>Mail</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>

рош Talk 30 min.

 setType Ask questions immediately

discussion at the end (and in between, if applicable)
Motivation

Why are we here today?

- Help People to start triaging
- Show best practices and common pitfalls
- Identify, discuss and solve current problems
- Get to know each other, aka map addresses to faces
Motivation
Why is GNOME here today?

- tidy up bug database
  - ~5 active triagers
  - ~1200 ~ 800 incoming bugs per week
  - ~800 closed bugs per week
  - ~24000 unanswered bugs
  - ~7000 bugs older than two years

- bringing triage community together
- teaching newcomers in triaging
Description

What is Bug Triaging anyway?

- Quality Assurance
- Manage Bugs
  - get enough information for devs to fix bugs
  - keep bugs in their actual state (NEEDINFO, OBSOLETE, Version, ...)
  - address severe bugs
- make users and devs happy : - )
GNOME Methodology
How does GNOME manage its bugs?

Bugzilla as Bug Database
- mature product
- awesome install base
- though old and kinda rusty

gnome-bugsquad@gnome.org as mailing-list
- Discuss Policies
- Organize Bugdays
- Answer Questions
- Subscribe http://mail.gnome.org/mailman/listinfo/gnome-bugsquad

#bugs on GIMPnet
- Easy Questions
- hang around
Beginners Guide

How do I triage bugs?
How do I triage bugs?

Create Bugzilla Account

To create a Bugzilla account, all you need to do is to enter a legitimate e-mail address. You will receive an email at this address to confirm the creation of your account. You will not be able to log in until you receive the email. If it doesn't arrive within a reasonable amount of time, you can contact the maintainer of this Bugzilla installation at bugmaster@gnome.org.

E-mail address: [ ]

Send
Beginners Guide (cont.)

How do I triage bugs?

Subscribe Mailinglist

Gnome-bugsquad Info Page

http://mail.gnome.org/mailman/listinfo/gnome-bugsquad

Gnome-bugsquad -- List to discuss bug maintenance in GNOME

About Gnome-bugsquad

This list is where we discuss how to manage bugs. Things like how to use bugzilla most effectively, how to respond to bugreports and so on.

To see the collection of prior postings to the list, visit the Gnome-bugsquad Archives.

Using Gnome-bugsquad

To post a message to all the list members, send email to gnome-bugsquad@gnome.org.

You can subscribe to the list, or change your existing subscription, in the sections below.

Subscribing to Gnome-bugsquad

Enter a Web address to open, or a phrase to search for
Beginners Guide (cont.)
How do I triage bugs?

Read Triage Guide

The GNOME Bugsquad

The Bugsquad is the Quality Assurance (QA) team for GNOME.
We keep track of current bugs in GNOME software and try to make sure that major bugs do not go unnoticed by developers.

You do not need any programming knowledge to join the Bugsquad; in fact it is a great way to return something to the GNOME community if you cannot program at all.

How bugs are tracked

We use the Bugzilla bug tracking system to keep track of bugs in GNOME software. Our Bugzilla database is hosted at [http://bugzilla.gnome.org/](http://bugzilla.gnome.org/).

How can I help?

The best way to help us triage bugs is to read the [Triage Guide](http://live.gnome.org/Bugsquad).
Beginners Guide (cont.)
How do I triage bugs?

Triage Bugs

Write Comments, Ask on IRC or Mailinglist
Beginners Guide (cont.)

How do I triage bugs?

Ask for permissions

On IRC or on Mailinglist
Beginners Guide (cont.)

How do I triage bugs?

Collect Points

Real Triagers are older in bugzilla points than in real life
Best Practice Pitfalls

How does Triaging look like?

Communication

- Be nice, be friendly, be happy
- Answer early, Answer often
- Be nice, be friendly, be happy
### Best Practice Pitfalls (cont.)

How does Triaging look like?

<table>
<thead>
<tr>
<th>Methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>🦅 Adjust Bugzilla Metadata (Version, Product, ...)</td>
</tr>
<tr>
<td>🦅 Ask for proper Stacktrace (C vs Mono vs Python vs ...)</td>
</tr>
<tr>
<td>🦅 Analyze Stacktrace (Ignore Glib/Gtk+ buzz, Find crashing function, ...)</td>
</tr>
<tr>
<td>🦅 look for Dups with <strong>Traceparser</strong></td>
</tr>
<tr>
<td>🦅 Rename Bug to include crashing function, filename, location</td>
</tr>
<tr>
<td>🦅 Watch out for modules that don’t like being triaged (Pango and vte)</td>
</tr>
</tbody>
</table>
Status Fields
What do all these fields mean, anyway?

**UNCONFIRMED**  This bug has recently been added to the database. Nobody has validated that this bug is true. Users who have obtained permissions from the bugsquad may confirm this bug, changing its state to → **NEW**. It may also be directly resolved and marked RESOLVED, or more information may be necessary, moving it to NEEDINFO.

**NEW**  This bug has recently been added to the assignee’s list of bugs and must be processed. Bugs in this state may be accepted and become → **ASSIGNED**, passed on to someone else and remain NEW, or resolved and marked RESOLVED.
ASSIGNED  This bug is not yet resolved, but is assigned to the proper person. From here bugs can be given to another person and become NEW, or resolved and become →RESOLVED.

NEEDINFO  More information from the reporter is needed to proceed further in fixing this bug. This should not be used when someone needs more information from a developer- a NEW or ASSIGNED bug implicitly needs more information from the developer.

RESOLVED  The bug has been resolved in some way. The resolution field should contain a secondary status describing the way in which it was resolved. See the Resolutions section for more details.
VERIFIED and

CLOSED GNOME does not substantially use VERIFIED or CLOSED. When used, they indicate that a third party has checked to see that a bug was properly resolved.
Triaging Diagram
Eh, what? How does it look like?

- NEW
- NEEDINFO
- UNCONFIRMED
- RESOLVED
- REOPENED
- Revert to previous status

- FIXED
- NOTABUG
- INCOMPLETE
- OBSOLETE
- NOTGNOME
- WONTFIX

- Gnome version older than 2 releases

[1] It's not always 6 weeks. Some maintainers explicitly asked to not close after 6 weeks

[2] Only for maintainers
Triaging Diagram

Normal Flow

- NEW
- NEEDINFO
- UNCONFIRMED
- RESOLVED
- REOPENED
- Revert to previous status

- NOTGNOME
- FIXED
- INCOMPLETE
- OBSOLETE
- NOTABUG

6 weeks without response [1]

Gnome version older than 2 releases

[1] It's not always 6 weeks. Some maintainers explicitly asked to not close after 6 weeks

ASSIGNED [2]

WONTFIX [2]

[2] Only for maintainers
Triaging Diagram

NEW

NEEDINFO

UNCONFIRMED

RESOLVED

REOPENED

Revert to previous status

NOTABUG

FIXED

INCOMPLETE

OBSOLETE

6 weeks without response [1]

Gnome version older than 2 releases

[1] It's not always 6 weeks. Some maintainers explicitly asked to not close after 6 weeks

ASSIGNED [2]

WONTFIX [2]

[2] Only for maintainers
Triaging Diagram

Resolutions

[1] It's not always 6 weeks. Some maintainers explicitly asked to not close after 6 weeks
[2] Only for maintainers
Triaging Diagram

Summary

- NEW
- NEEDINFO
- UNCONFIRMED
- RESOLVED
- REOPENED
- NOTABUG
- NOTGNOME
- FIXED
- OBSOLETE
- INCOMPLETE

Revert to previous status

6 weeks without response [1]

NOTGNOME

WONTFIX [2]

Gnome version older than 2 releases

ASSIGNED [2]

[2] Only for maintainers

[1] It's not always 6 weeks. Some maintainers explicitly asked to not close after 6 weeks
Conclusion
What we’ve just learned

- GNOME needs triagers
- It’s easy to start triaging
- It’s about people
Questions?