

Bug-Tracking and Quality Management tools and workflows

Henning Sprang

Silpion GmbH

04/16/2007 / Xen Summit 04/2007

Outline

- 1 Introduction
- 2 Status and target situation
- 3 What do we need (to do)?
- 4 Proposed Bug Handling Policy/Workflow
- 5 Open questions
- 6 Discussion

About the speaker

- Henning Sprang
- Working at Silpion IT solutions GmbH - <http://www.silpion.de>
- Development, sysadmin, QM and coordination work in many OS and commercial projects since 1997
- Co-developer of FAI - Fully Automatic Installation
- Involvement in the Xen community since about a year
- Started to help cleaning up bugzilla in March 2007
- Published a book about Xen in German(English version soon to come)

About Quality Management

- What is Quality?
 - The grade by which a set of inherent characteristics matches the requirements (ISO 9000:2000)
- What is the goal of Quality Management?
 - The goal is to continuously improve the quality of one's services and products, through improvement of the underlying processes.
- QM might be "boring" compared to hacking on hardware level and implementing great features
- But . . . essential!

Status and target situation

- Where are we?
- Where do we want to be?

State of the bug database

- (new) Bugs not always properly tracked
- A lot of bug reports not processed at all
- In numbers: NEW bugs (meaning their state never changed into a further processing state) as of 12th of April 2007: 177

Effects

- The bug tracking system is full of bug data that is not used, and not kept clean, up to date and useful for the future
- Some bugs more or less randomly reoccur (at least messages in the BTS say so)
- Closed bugs or problems that vanished in the course of development are not closed
- Important information and feedback from users and developers is unused

Desired situation

- Every occurring bug in Xen is being reported, fixed and never occurring again
- Very idealistic, though this would be really cool?!

What do we need?

- People
- A plan
- Tools and machines

People

- People who do the tidying work for handling incoming bugs (QA engineers)
- People who coordinate which developer should work on which bug and how (coordinators)
- People who actually fix bugs (developers)
- People making sure changes didn't introduce new problems (QA engineers)
- Last, but not least: people using Xen and reporting bugs (users, testers)

A plan

- A bug handling policy and workflow
- Defines what we should do with a bug, from occurring or being reported until being "closed"
- Should also define who is responsible for which piece of work
- See a more detailed proposal in the next section. . .

Tools and machines

- Bugzilla as bug tracking tool - does his job currently
- The people handling bugreports need access to a large variety of hardware and systems for being able to reproduce as many problems as possible
- An automated test system, in which a test is added when a bug is fixed, so it won't reoccur

Proposed Bug Handling Policy/Workflow I

- Whenever a bug occurs, users and testers should report it to the bug tracking system
- The bug should be checked by a QA engineer for:
 - An understandable description and enough useful information
 - Duplicates of this bug in the database
 - Reproducibility, and requirements for reproduction of the bug
- The bug should be assigned to a developer (if not closed because of the results of the checks made)

Proposed Bug Handling Policy/Workflow II

- A developer starts working on the bug:
 - Reproduce it
 - Find the program unit responsible for the bug
 - Develop a test case that fails as long as the bug occurs
 - Change the code until the test passes
 - Integrate the test in the xen test system, to make sure this bug doesn't reoccur
- Now, the solution to the bug is integrated in the code base, and the Xen automated test system.
- A last check is done by the QA team, to make sure the bug is really properly fixed.
- The bug is closed.

Additional notes to the workflow

- Take the bug lifecycle into account and set appropriate
 - We could start with the proposal from bugzilla - <http://www.bugzilla.org/docs/tip/html/lifecycle.html>
- To keep the policy useful and working, do regular checks for
 - Viability of the policy
 - Compliance to policy

Open and further questions I

- How can (better) bug handling be integrated in the current Xen QA workflow?
 - How does current QA work in Xen?
 - Are automated tests used efficiently, and can they easily be extended?
- How and by whom should the QA work be organized?
- How should bugs be assigned to developers after being checked by the QA engineers?
- About the people to do the work planned in the workflow
 - Are there enough available?
 - If not, write "job descriptions" or requirements to call for volunteers

Open and further questions II

- Can people outside of XenSource access the hardware testing infrastructure for their QA work?
- Some work has been started to close old bugs in the bug tracking system to get rid of old data - is this really a good idea?

Open and further questions III

- What exactly do we want and need to know about a bug? (e.g. for extensions of bugzilla)
 - Interesting thing the current and many systems don't track: How often is a bug occurring (apart from the usual: to whom, under which circumstances, while doing what)
 - What interesting things do we want to be reported from the available database - maybe regularly and automatically
 - Which values and data do we want and need to watch?

Discussion

- Any Questions?
- Volunteers for the (additional) QA work? - let's meet tomorrow!
- Proposals for doing things different/better than explained here?

Advertisement

- Silpion IT Solutions is sponsoring my work here, as well as the writing of the Xen Book
- It's a business of about 25 people working on systems and software development in many areas
- Linux server and embedded systems as well as Java development for servers and mobile devices are our key competences.
- see <http://www.silpion.de>