Quality Team

As per Where, this page should be moved to Quality Team

This pages needs a major update in light of: Quality Team - How to reduce the workload ^

Intro

This team exists to monitor backport proposals from trunk to the proposals branches, vote on them and finally commit them to the corresponding branch to be released in the final Tiki (LTS version).

The current members of the team are:

- Nyloth
- SylvieG
- JonnyB
- Sept_7
- Pkdille
- luci

Reasons for existence

The Quality Team was first conceived during the TikiFestMontreal2009 so developers can keep adding new things to trunk, and the best of them can be merged into forthcoming minor releases and the release branch doesn't become stagnant (as happened with 2.x)

Only the Quality Team and the Security Team are permitted to commit directly to the stable release branch, except for translations (where anyone can).

How it works

Devs should work on their fixes and enhancements in trunk (or an experimental branch *first* if it's really "out there"), and then commit that to the **branches/proposals/4.x** branch when tested thoroughly. If you also want to get your change into 3.x ("LTS" version) you should then commit the work (with suitable modifications where necessary) to **branches/proposals/3.x**

These "proposed" fixes should be done as a single commit (even if the original work in trunk took several goes to get right) so the whole change can be reviewed in one go, and if needs be, rolled back cleanly.

The commit message on branches/proposed should contain the revision number(s) and commit message(s) from trunk where this was first committed. If possible, wishlist id's when the commit fixes, or addresses any previously logged bugs. This is important because the Quality Team has the responsibility of checking that all commits to proposals has indeed been done in trunk first, and appropriately tested. It is also useful for all developers to have a clear log of who did what where, and why. A good explanation of the improvement will also help obtain a favorable decision for your work.

The team will discuss and vote on the proposed commits on the SVN Mailing List (which you should subscribe to!) to make the process as transparent, efficient and simple as possible.

The Quality Team will be alerted of new patches through the CVS/SVN mailing list after each commit. They will discuss the change together, evaluate the change by majority decision, and then approve and merge, or reject and rollback.

What will be considered

In general,

- Backports should be to solve a specific issue, not to address a global concern.
- Backports should be a quite limited number of files/line changes, to reduce risk of regression and ease review. A typical review should be 10-15 minutes.
- All feature enhancements must be optional and be "disabled" by default so users should not notice any change in operation without activating them.
- Major features will not be considered for back-porting - obviously the definition of *major* is subjective and debatable. Risk of introducing unexpected problems is a big decision factor. Developer time should mostly be focused on trunk. The Quality Team will judge and as a community, we can document a common understanding of what this is.
- Alterations to the database should be avoided: thus only critically needed and proven fixes should be proposed.
- Bug fixes should preserve existing behaviour as much as possible - The Three Rules still apply!
- All files needed for the enhancement or fix must be committed in a single revision. If you miss a file, or find that a subsequent fix is needed, than rollback the initial commit, and re-commit the whole lot in one go. This will make the members of the Quality Team smile on you!

Release process

The Quality Team will manage minor releases and hopes to release often (for example every month or three depending on activity). These releases will be "dot releases" e.g. 3.1, 3.2 etc.

The Security Team may need to release security updates between these releases and will operate in parallel with the Quality Team. These will be "dot dot releases" - e.g. 3.1.2

Marc Laporte wrote:

> *For simplicity, I think a security release should just be a dot release (which will presumably have other fixes). Basically, a security issue prompts a release.*

Soft mode and strict mode

Typically between x.0 and x.1, the Quality Team is in soft mode, and thus, you can commit on that branch (also see: Semi-automatic merging period), and the Quality Team can provide feedback and even rollback your commit. Once x.1 is released, the Quality Team goes in strict mode (which is in general what this page describes)

Conduct and recruitment

The membership of the team will be reviewed at every new release (every "six" months) and should be agreed by consensus through the tikiwiki-devel list, TikiFests and The Wiki-Way.

Members of the team cannot vote on their own back-ports, these must be approved by a majority of the other members of the team.

Questions

- Is there a timeframe to deal with proposed commits? (Is there a "maximum" or "expected" time delay?)
- If a commit is rejected in the stable branch, what about trunk? (Someone should check if they should also be rejected in trunk)

See also

- http://en.wikipedia.org/wiki/Change_control_board
- Quality Team on tiki.org
- Quality Team - How to reduce the workload
- http://code.tiki.org

Alias

- Quality Group
- Quality
- QT