

A discussion about ...

our integrated gui

Goals

- Quantify the costs and benefits of keeping the GUI tightly integrated to the project
- Quantify technical requirements why the gui needs/should be tightly integrated
- Look for rough consensus on the medium-term plan
- Try and agree concrete next steps

Considerations

- Do the benefits outweigh the costs?
- Are we delivering the best experience to our users?
- Are we on the right path?

Benefits of status quo

- Exists today
- Works well
- Good feature set
- Purposefully-designed for minimising errors in use case
- Multiple translated languages
- More responsive vs non-integrated? (but some global locking/perf issues exist)
- It's an “official” part of the Bitcoin Core project

Costs

- Low and unmonitored

Generated by: LCOV version 1.16

Fuzzing

- no fuzz tests


LCOV - code coverage report

Current view: top level

Test: fuzz_coverage.info

Date: 2024-04-04 03:49:55

Lines:	51015	78801	Coverage	64.7 %
Functions:	11949	16923		70.6 %
Branches:	53411	155614		34.3 %

Directory	Line Coverage ↕	Functions ↕	Branches ↕
src	 68.9 % 17149 / 24890	 69.0 % 5934 / 8606	 35.2 % 15409 / 43739
src/common	 61.8 % 645 / 1043	 65.5 % 116 / 177	 36.1 % 708 / 1960
src/compat	 61.1 % 33 / 54	 76.2 % 16 / 21	 0.0 % 0 / 4
src/consensus	 99.2 % 243 / 245	 98.5 % 65 / 66	 72.4 % 223 / 308
src/crypto	 80.0 % 2469 / 3085	 74.3 % 182 / 245	 64.0 % 393 / 614
src/index	 1.3 % 12 / 906	 5.5 % 7 / 127	 0.2 % 2 / 1298
src/init	 65.9 % 58 / 88	 87.5 % 7 / 8	 31.5 % 117 / 371
src/interfaces	 35.9 % 23 / 64	 13.3 % 8 / 60	 0.0 % 0 / 2
src/kernel	 82.3 % 673 / 818	 76.2 % 109 / 143	 42.4 % 405 / 955
src/logging	 96.7 % 29 / 30	 50.0 % 4 / 8	 61.8 % 21 / 34
src/node	 54.1 % 1473 / 2723	 55.3 % 257 / 465	 28.2 % 1163 / 4131
src/policy	 91.3 % 1020 / 1117	 91.6 % 109 / 119	 60.0 % 876 / 1460
src/primitives	 97.5 % 275 / 282	 92.4 % 304 / 329	 56.8 % 172 / 303
src/rpc	 70.5 % 5968 / 8460	 80.7 % 481 / 596	 37.7 % 10570 / 28005
src/script	 94.0 % 5542 / 5896	 86.9 % 1012 / 1164	 63.9 % 7993 / 12513
src/support	 86.0 % 166 / 193	 76.5 % 26 / 34	 49.1 % 52 / 106
src/support/allocators	 98.0 % 96 / 98	 98.1 % 102 / 104	 71.4 % 140 / 196
src/test/fuzz	 96.9 % 8852 / 9132	 98.1 % 1703 / 1736	 57.6 % 9582 / 16633
src/test/fuzz/util	 70.1 % 216 / 308	 69.8 % 30 / 43	 56.9 % 182 / 320
src/test/util	 58.0 % 505 / 871	 68.7 % 79 / 115	 35.1 % 591 / 1686
src/univalue/include	 95.7 % 112 / 117	 64.9 % 61 / 94	 59.5 % 88 / 148
src/univalue/lib	 92.8 % 506 / 545	 90.7 % 39 / 43	 71.7 % 405 / 565
src/util	 74.8 % 1550 / 2071	 75.2 % 664 / 883	 55.8 % 1316 / 2359
src/wallet	 25.8 % 2527 / 9779	 34.4 % 445 / 1292	 12.7 % 2072 / 16361
src/wallet/rpc	 3.0 % 154 / 5130	 23.4 % 67 / 286	 1.0 % 197 / 20104
src/wallet/test	 22.4 % 37 / 165	 35.8 % 19 / 53	 10.3 % 20 / 195
src/wallet/test/fuzz	 98.7 % 682 / 691	 97.2 % 103 / 106	 57.4 % 714 / 1244

Costs

- Adds code complexity in unrelated components in our codebase
- Adds complexity in guix build system
- Dependencies (see xz attack)
- Prevents us using a fully static release flow, without having two release processes

Issues

- We ship with known bugs
 - If we don't fix these, does it damage the reputation of the project?
- Same regressions multiple times due to lack of testing
- Reduces importance for bitcoind to improve supporting other GUIs
- Is lower levels of activity result of a barrier to entry for new gui devs

Discussion points

- Can we clarify/solidify what we are doing with the gui as a project?
- Is fixing up an existing gui easier than making a new one?
 - Other GUI examples exist, Bitcoin Design Community, Sparrow, Liana etc.
- Is C++/QML the best tool for the job?
 - Is it limiting new contributor pool
 - Will being seperated and modernised attract more new developers?
- Can we Make RPC/Cap'n'Proto Great (enough?)
 - Reliability / Timeouts
 - Long-running RPCs / progress monitoring
 - Polling
 - Daemon error propagation
 - Versioning
- Is status quo hurting the project reputation
- What am I missing?

Stats

- 371671 LoC (~30k cpp/h directly)

```
will@ubuntu in ~/src/bitcoin-feature on ȳ new-feature [$?] : 3.9.18  
$ git diff HEAD..master --shortstat  
655 files changed, 371671 insertions(+), 930 deletions(-)
```

- 460 files in repo

```
will@ubuntu in ~/src/bitcoin-feature on ȳ new-feature [$?] : 3.9.18 took 2s  
$ git diff master..HEAD --diff-filter D --pretty="format:" --name-only | wc -l  
460
```