Testing the CDT

Testing Status - JUnits

- 3882 JUnits run as part of the nightly builds
  - 3237 core suite
  - 13 managed builder ui suite
  - 632 ui suite
- Others fail
  - Managed builder core suite crashes on build machine
  - Debug tests had failures so simply removed them
- We haven't measured coverage of this suite
  - I'd guess around 20%

Testing Status - Ad hoc testing

- A good number of the community take milestone builds and do test with them
  - Received a good number of bug reports from non-committers
- Once GA is announced then we get a lot of testing
- Vendors adopting the CDT test to put it in their products
  - Usually happens late in the game (post GA?)

Problems

- We have no real idea of the test coverage we get at GA
- We have no co-ordination of the test effort to ensure coverage
  - And to avoid duplication of coverage
- Very few products are released on the .0 version of a release
  - Everyone waits for the maintenance release (or two)

Problems

- Not all platforms tested equally
  - Who's testing on Solaris, Mac, ia64?
- We don't really specify what tool chain line-ups we support
  - Gcc, gdb, but what versions, what derivatives
  - What are people using when they test?
- Fixed deadlines, over committed development
  - Not sure all development plans include adequate time for testing
  - Fixed deadline, fixed resources, quality has to adjust

Possible solutions

- Grow our automated test base
  - And measure the coverage
  - Add in GUI tests to this mix
- Use an on-line test management tool to co-ordinate
  - Something like a Testzilla
  - Encourage community to contribute tests and results
- Slow down development
  - Make sure we allow adequate time for testing
  - Allocate time to write automated tests

Possible Solutions

- Better project scheduling
  - Especially milestones