

# The GDA software project

Richard Woolliscroft  
NoBUGS October 2010



- Background
- Past Team and Tools
- Current Team and Tools
- Communication

# Background...

- Originally developed for two new stations in 2001 at SRS, Daresbury, UK.
  - a PX station (14.2)
  - a Materials (SAXS and powder diffraction) station (MPW6.2)
- Became a joint project between SRS and Diamond in 2003.
- Diamond took 'ownership' in 2007 after the user programme started and SRS stations began to shutdown.

# 2003-2007

- Joint development between SRS (Daresbury) and Diamond (RAL)
- Regular workshops
  - Twice a year
  - To outline new developments to all developers
  - Plan new major developments
- Communication methods:
  - CVS
  - Bugzilla
  - Pair Programming for refactoring
  - Mailing list

Successful: ~ third of SRS beamlines switched to GDA and at Diamond the GDA was ready before the hardware 😊

But...

- Fast paced development, so the workshops were not frequent enough
- Poor culture of writing documentation and unit tests
- Did not have enough VCs
- Informal communications would be missed by the other site



# Since 2007

- SRS stations were being shutdown, less developers and less development from that site
- GDA code repository and issue tracking systems moved to Diamond
  - SVN code repository
  - Jira (originally Trac) issue tracking
  - Trac wiki

# Oct 2010 Diamond Team

Bill Pulford – Group Leader

Paul Gibbons (Team Manager - GDA)	Alun Ashton (Team Manager - SciSoft)
Jun Aishima	Peter Chang
Mark Basham	Joachim Diepstraten
Mark Booth	Ghita Mostefaoui
Chris Coles	Irakli Sikharulidze
Richard Fearn	Duncan Sneddon
Paul Hathaway	Graeme Winter
Graham Lee	
Karl Levik	
Vasanthi Nagalingam	
Eric Ren	
Tobias Richter	
Rob Walton	
Matthew Webber	
Richard Woolliscroft	
Fajin Yuan	



# Current collaborators

- Collaborators with commit rights to our svn:
  - Geoff Mant (STFC, Daresbury)
  - John Hammond (Argonne)
  - Kenneth Evans (APS)
  - Matt Gerring (ESRF)
- Open Source since Oct 2008
- Need standards in the tools and methods used
- Need best possible communication with collaborators

# Working as a team...

- Same IDE
  - A standard Eclipse installation which we build against
  - Useful as we don't all develop on the same platform (so use same Eclipse version and set of plugins)
- Coding standards
  - Encourages everyone to contribute to, and fix bugs in, all parts of the code
  - Reduces sense of code 'ownership'
  - New packages/classes 'look' like others, increasing readability
- Daily 'stand-up' meetings
  - Get immediate advice/feedback from whole group
  - Advertise your work to everyone
- Issue tracking system
  - New bugs assigned to a mailing list so all developers can see
  - Commit messages also sent to another (now external) mailing list

# Testing, testing, testing...

- Test manager since Jan 09
  - Why didn't we do this before!?
  - One resource dedicated to testing and deployment
- All new code should be unit tested
  - New devices should have a simulation version
  - Old code gradually being unit tested
- A 'base' configuration to test against, and to use in demonstrations / courses
  - All beamlines have their simulation version in the code repository
- Gives confidence when refactoring sub-systems
- Required a change in the team culture to achieve this

# Continuous Build server

- Hudson
- Repeatedly builds the source, run units tests and tests if beamline simulations start
  - Any problems are emailed within a few minutes to committers

The screenshot shows the Hudson web interface. On the left is a sidebar with navigation links like 'People', 'Build History', and 'Project Relationship'. The main content area is titled 'GDA automated build and test for trunk'. Below the title is a table of build jobs. The table has columns for 'S' (status), 'W' (weather icon), 'Job', 'Last Stable', 'Last Success', 'Last Failure', and 'Last Duration'. The jobs listed include 'GDA\_trunk.deploy\_test', 'GDA\_trunk.diffcalc', 'GDA\_trunk.documentation.javadoc', 'GDA\_trunk.documentation.javadoc.Win', 'GDA\_trunk.documentation.manuals', 'GDA\_trunk.environment', 'GDA\_trunk.simulation.b16', 'GDA\_trunk.simulation.i20', 'GDA\_trunk.tests.RH5\_32', 'GDA\_trunk.tests.RH5\_64', 'GDA\_trunk.tests.Ubuntu\_32', 'GDA\_trunk.tests.WinXP\_32', 'Scisoft\_build', and 'Subversion.post-commit'.

S	W	Job ↓	Last Stable	Last Success	Last Failure	Last Duration
●	☀	<a href="#">GDA_trunk.deploy_test</a>	42 min (#662)	42 min (#662)	2 days 16 hr (#657)	24 min
●	☀	<a href="#">GDA_trunk.diffcalc</a>	12 days (#47)	12 days (#47)	N/A	5.7 sec
●	☀	<a href="#">GDA_trunk.documentation.javadoc</a>	18 hr (#1763)	18 hr (#1763)	N/A	13 min
●	☀	<a href="#">GDA_trunk.documentation.javadoc.Win</a>	6 hr 55 min (#299)	6 hr 55 min (#299)	N/A	17 min
●	☀	<a href="#">GDA_trunk.documentation.manuals</a>	1 hr 57 min (#112)	1 hr 57 min (#112)	N/A	1 min 52 sec
●	☀	<a href="#">GDA_trunk.environment</a>	32 min (#799)	32 min (#799)	2 mo 1 day (#679)	9 min 8 sec
●	☁	<a href="#">GDA_trunk.simulation.b16</a>	2 mo 24 days (#120)	14 hr (#194)	20 hr (#193)	21 min
●	☀	<a href="#">GDA_trunk.simulation.i20</a>	14 hr (#487)	14 hr (#487)	22 hr (#486)	11 min
●	☀	<a href="#">GDA_trunk.tests.RH5_32</a>	5 days 22 hr (#4214)	1 hr 52 min (#4269)	21 hr (#4263)	31 min
●	☀	<a href="#">GDA_trunk.tests.RH5_64</a>	5 days 22 hr (#3054)	17 hr (#3083)	22 hr (#3081)	34 min
●	☀	<a href="#">GDA_trunk.tests.Ubuntu_32</a>	5 days 22 hr (#1792)	1 hr 17 min (#1830)	22 hr (#1826)	30 min
●	☀	<a href="#">GDA_trunk.tests.WinXP_32</a>	5 days 22 hr (#2033)	18 hr (#2063)	22 hr (#2061)	37 min
●	☀	<a href="#">Scisoft_build</a>	2 hr 17 min (#55)	2 hr 17 min (#55)	N/A	5 min 22 sec
●	☀	<a href="#">Subversion.post-commit</a>	8.1 sec (#10856)	8.1 sec (#10856)	N/A	3.1 sec

# Cross-site collaboration

- Code base has been split into many plugins using the Eclipse framework
  - Separate the 'core' code from Diamond-specific code
  - Separate the data analysis and visualisation components from data acquisition
- Each release has a 'base' version of GDA shared on the website
  - core plugins with a simple configuration
- Developer documentation and detailed release notes are important to keep everyone updated
  - Better than a wiki!
- Forum <http://opengda.org>
- External code repository for the 'core' (git - coming soon)
  - Internal git repository for Diamond-specific plugins and configurations
- Mailing lists <https://lists.opengda.org/>
  - Announce
  - Developers
  - Commits
- Developer workshops – one at BNL later this week
  - GDA introduction
  - 'build a beamline' workshop

# Take 'The Joel Test'

- <http://www.joelonsoftware.com/articles/fog0000000043.html>
- (we score 9.5/12, I think)

## **The Joel Test**

1. Do you use source control?
2. Can you make a build in one step?
3. Do you make daily builds?
4. Do you have a bug database?
5. Do you fix bugs before writing new code?
6. Do you have an up-to-date schedule?
7. Do you have a spec?
8. Do programmers have quiet working conditions?
9. Do you use the best tools money can buy?
10. Do you have testers?
11. Do new candidates write code during their interview?
12. Do you do hallway usability testing?

# Conclusion

- We are aiming for a culture of:
  - Writing tests
  - Writing documentation
  - High levels of communication
- This takes effort, but its worth it!