EVALUATION OF ISSUE TRACKING AND PROJECT MANAGEMENT TOOLS FOR USE ACROSS ALL CSIRO RADIO TELESCOPES FACILITIES

Juan Carlos Guzman | ATNF Software and Computing Group Leader

ICALEPCS 2013, 8th October 2013, San Francisco, USA

Motivation 1

Software Team Restructure

2012 and before: Two relatively independent software teams

ATNF Scientific Computing and Archives

ASKAP Computing

Motivation 1

Software Team Restructure

2012 and before: Two relatively independent software teams

ATNF Scientific Computing and Archives

ATNF IT

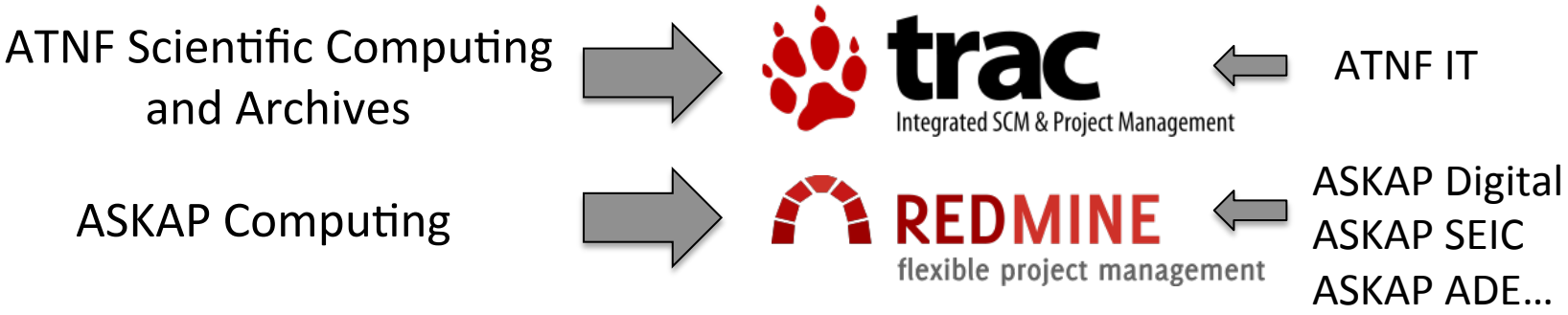
ASKAP Computing

ASKAP Digital
ASKAP SEIC
ASKAP ADE…

Motivation 1

Software Team Restructure

2012 and before: Two relatively independent software teams



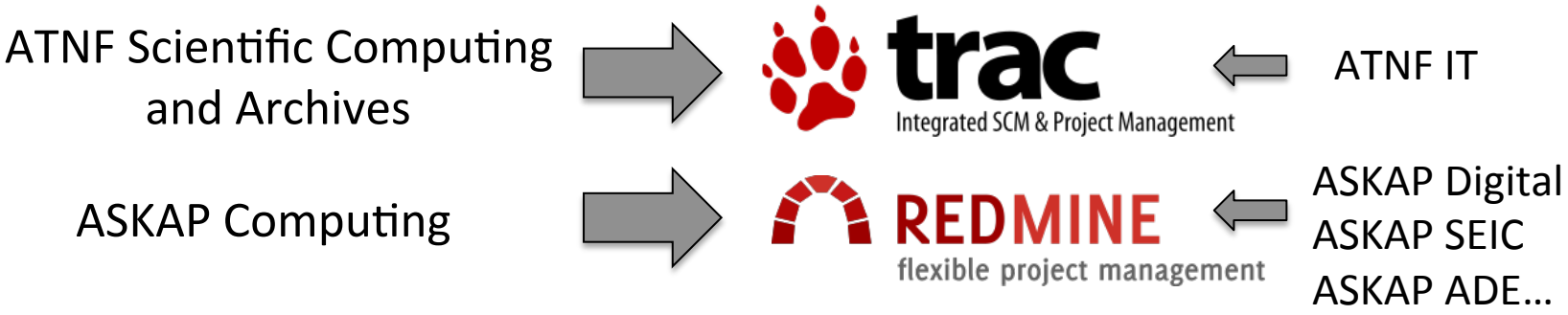
In early 2013, two software teams merged into single ATNF Software and Computing group, part of Operations

- Heterogeneous software maintenance and development processes and tools
- Some old software packages are “track by hand”

Motivation 1

Software Team Restructure

2012 and before: Two relatively independent software teams



In early 2013, two software teams merged into single ATNF Software and Computing group, part of Operations

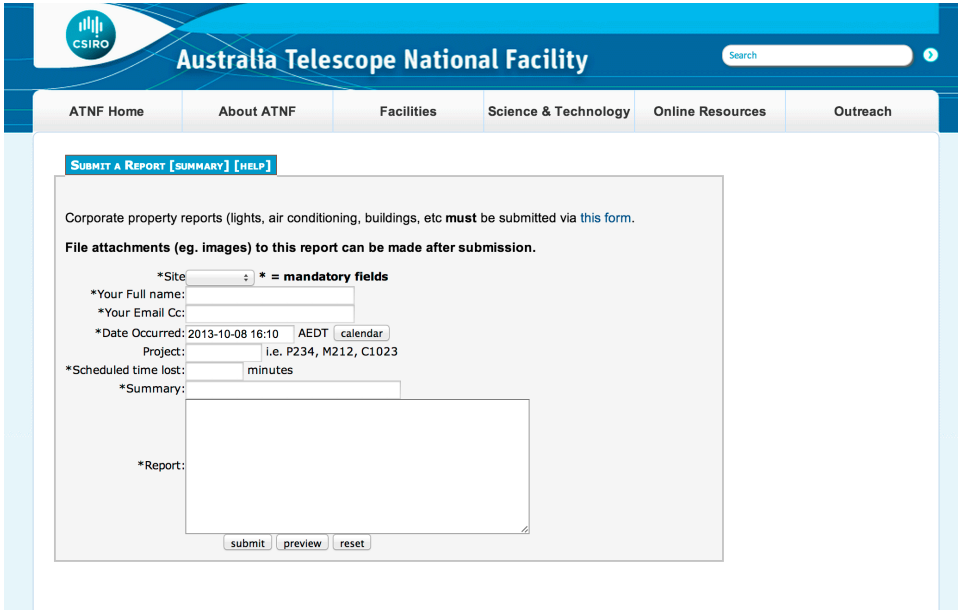
- Heterogeneous software maintenance and development processes and tools
- Some old software packages are “track by hand”

Goal: Homogenise processes and tools

Motivation 2

Upgrade of ATNF Fault Report System?

- Head of Engineering Operations asked for help to review the Fault Report System and evaluate alternatives.. Why?
- The ATNF Fault Report System
  - In-house development, running for 15 years (PHP/ MySQL)
  - Supports Parkes, ATCA and Mopra
  - Need to adapt to ASKAP and its operation model
  - Issues with categories, limited reporting and flexibility
  - Does not link with other issue-tracking systems



Requirements Analysis

Helpdesk/Support Tracking System

- Track Issues
- Email notification
- Support for watchers
- Multiple telescopes (or projects)
- Custom categories and sub-categories
- Custom fields
- Custom workflows
- Administration of users/ groups
- Access Security
- API
- Import from other issue tracking (CSV)
- Searching issues (common queries)
- Custom queries
- Basic reporting
- Custom reports
- Web-based UI
- Easy to use
- Easy to install/administer

Requirements Analysis

Software Issues Tracking System

- Track Issues
- Email notification
- Support for watchers
- Multiple telescopes (or projects)
- Custom categories and sub-categories
- Custom fields
- Custom workflows
- Administration of users/ groups
- Access Security
- API
- Import from other issue tracking (CSV)
- Searching issues (common queries)
- Custom queries
- Basic reporting
- Custom reports
- Web-based UI
- Easy to use
- Easy to install/administer
- Integration with source VCS
- Code statistics/visualisation

Implementation Options

Many, many off-the-shelf Alternatives

FogBugz

JIRA

kayako

MANTIS

trac
Integrated SCM & Project Management

REDMINE
flexible project management

Implementation Options

Many, many off-the-shelf Alternatives

kayako

trac

Integrated SCM & Project Management

REDMINE

flexible project management

…and many more in wikipedia

http://en.wikipedia.org/wiki/Comparison_of_issue_tracking_systems

Implementation Options

The shortlist

In use in other facilities and other CSIRO divisions

Already in use in ASKAP

Comparison Tests

- Analysed user documentation and books
- Play with JIRA 5.2 (trial version) and Redmine 2.3.2
  - Ask a couple of more developers to play with them
- Usability
  - Survey to Operations staff (in the context of Fault Report System), including playing with the tool without spending too much time
- Score on each category from (no weight):
  - 1 = not supported or don't like it
  - …
  - 5 = fully supported or feature is awesome

Comparison Results

- Result: JIRA 118, Redmine 109
- Differences and findings
  - JIRA provides support for sub-categories (up to two level). Redmine provides sub-categories via plugin (untested)
  - Redmine provides hierarchical project structure. JIRA only has two levels.
  - JIRA provides graphical workflow editor
  - JIRA has many built-in reports: user workload, version workload, time tracking, created vs resolved, etc. Redmine only provides time spent report
  - JIRA provides better documentation on how to create custom reports
  - Slight preference in terms of UI of JIRA compared with Redmine
  - Add-on JIRA FishEye provides more code statistics and nicer visualisation. There might be Redmine plugins that add on similar functionality (unverified)

Recommendation

- Replace Fault Report System with off-the-shelf issue tracking
  - JIRA is (slightly) more user friendly than Redmine
  - JIRA has a larger use base, including several large physics facilities
  - JIRA has better record as a Helpdesk/Support system compared to Redmine
  - Using JIRA is less riskier than using Redmine
- Evaluation report (internal) in the context of the Fault Report System only, released to the Operations staff for consultation in Sep 2013
- Are you using JIRA now? What happened to Redmine?

Recommendation

- Replace Fault Report System with off-the-shelf issue tracking
  - JIRA is (slightly) more user friendly than Redmine
  - JIRA has a larger use base, including several large physics facilities
  - JIRA has better record as a Helpdesk/Support system compared to Redmine
  - Using JIRA is less riskier than using Redmine
- Evaluation report (internal) in the context of the Fault Report System only, released to the Operations staff for consultation in Sep 2013
- Are you using JIRA now? What happened to Redmine?
  - Not implementing it yet
  - Updating the report with comments I received + add analysis of impact on other projects
  - Decision by end of October 2013
  - Driving change is difficult and challenging… get staff involved

Conclusions

- Use an issue tracking tool, it will make your life easier both as a developer and manager, especially in medium to large teams
- Single project, small team, open source “fundamentalist” and/or very little budget
  - Consider Trac (or others)
- Multiple project, small team (< 5), open source “fundamentalist” or very little budget
  - Consider Redmine
- Multiple project, medium to large team (distributed geographically), don’t mind spending a modest license fee
  - Consider JIRA, Confluence for wiki, FishEye for code visualisation

Conclusions

- Use an issue tracking tool, it will make your life easier both as a developer and manager, especially in medium to large teams
- Single project, small team, open source “fundamentalist” and/or very little budget
  - Consider Trac (or others)
- Multiple project, small team (< 5), open source “fundamentalist” or very little budget
  - Consider Redmine
- Multiple project, medium to large team (distributed geographically), don’t mind spending a modest license fee
  - Consider JIRA, Confluence for wiki, FishEye for code visualisation

Remember: These are just suggestions/views from the author
Use it at your own risk

Thank you
and see you in Melbourne 2015!

CSIRO Astronomy and Space Science
Juan Carlos Guzman

ATNF Software and Computing Group Leader

t +61 2 9372 4457
E juan.guzman@csiro.au
w www.csiro.au