

SIT Review Issues

Frederick Luehring

Indiana University

February 4, 2005

Presented to SIT Review Panel

Introduction

- This talk is not comprehensive!
 - It is an attempt to highlight a few hot topics
 - Each topic is given one slide
 - Each slide contains discussion points
 - Please consult the copious written material for full details
 - Intended to be short enough to allow plenty of time for discussion.
 - The topics are presented in alphabetical order and considered equally (except a first slide on the SIT mandate)

SIT Mandate

- The SIT is a technical organization
 - We enforce and occasionally suggest policies.
 - We do **not** make or approve policies.
 - Policies are made by Atlas management, SPMB, CMB, and EB.
- We suffer continuous pressure to expand into new areas without input of new resources.
 - The recent transfer of the Atlas mailing lists to Simba2 cost over week of people time to complete. It's not clear that mailing lists are in our mandate but who else would do it?
- Our activities are of three types:
 1. Keeping the existing system functioning.
 2. Improving the existing system.
 3. Introducing new capabilities.

Code Distribution

- The SIT is responsible for code distribution
 - The SIT also become responsible for validating the distribution kit.
 - The boundary between the SIT responsibilities and the Production Team responsibilities is a little vague.
- We have received many requests to provide a versioning system for the kits.
 - The kits get patched a number of times during the validation of the release.
 - People want to have a way to tell that they have the correct, most recent kit.
- Multiple platform support will be an ongoing issue.
 - Manpower will be needed to support multiple platforms.

Documentation

- We have been criticized for the state that the documentation is in. Some of this is deserved particularly about our own (SIT) documentation.
- The SIT does not provide the content of most documentation - the developers do (or should!).
- We are moving to solve the problems and have formed a Documentation Task Force which:
 - Is performing an inventory of the existing documentation.
 - Is introducing the framework for the “workbook” to provide help to users with all levels of experience but especially novices.
 - Is working to harmonize all of the documentation: Doxygen, web pages, Wikis, and html documents.

Help List and Bug Reporting

- We know we have a problem here:
 - The help list is almost not used and when it is, questions sometimes are unanswered.
 - Many users find the Savannah system as it is currently structured to be very intimidating and bugs are often not followed up on.
 - The developers list is used by many in place of the help list and Savannah.
- It seems clear that we need some sort of dedicated personnel to keep track of bug reports / help requests to ensure their timely resolution.
 - Currently with the manpower levels we have this is just not possible.

Manpower

- The SIT currently has 8.2 FTEs working on it.
 - The 8.2 FTEs are split between 21 people.
 - There is only one full time person at CERN (Emil).
 - Most of the SIT participants are volunteers.
 - Some holes: Training, Mac OS X, Help Mailing List
- The shortage of manpower severely restricts our ability to respond to requests from the community.
 - Our core tasks (release and distribution kit building, code management, testing) tie up most of our effort in keeping things running.
 - The shortage of manpower makes us operate in a reactive mode instead of being as proactive as we would like.
- An additional software librarian will probably be added soon (using CERN/Russian funding).

Project/Partial Builds

- Many people are enthusiastic about project builds.
 - The hope is to have a stable build environment for a group of related packages
 - It remains to be seen whether a project build is truly viable with the complex set of packages dependencies in the Atlas offline software
 - We worry that FTEs will be expended starting project build and the effort level won't be sustainable.
- Markus would like to explain further his ideas on using ordered building instead project builds.

Python / Scripting

- Python was the subject of much discussion.
 - Python is intended to be used to set job options and for analysis.
 - Python is not intended to replace C++ code in core software,
- The SIT has a limited role with respect to Python.
 - We provide the Python compiler.
 - We can help with technical problems with Python.
 - We don't set policy about Python.
 - We don't enforce coding Python coding rules.
- Python is primarily the responsibility of the core software team.

Release Building

- The SIT is pretty much at the limit of how many releases we can build with the current set of people (Emil, Zhongliang, David, Fred).
 - The SIT does not define the release schedule (except for limiting the rate of builds to what we can do with our people).
 - The release schedule is (mainly) defined by the SPMB.
- The SIT (Release Coordinator) does track the readiness to build a release and controls the process using the tag collector.
 - It has been extremely difficult to get accurate information about the readiness from the software developers and the production team for building/running a release.
 - This impairs our ability to plan and needs to be flagged as a problem at the highest level!
 - We need to find a replacement for Zhongliang!

QA/QC

- Need to decide when to enforce the automatic checks such as checkreq and RuleChecker.
 - The developer of checkreq has left HEP and we urgently need to find a new developer.
 - For RuleChecker need to define a core set of rules to be enforced.
 - Currently there are over 200 rules and many are just recommendations. We must avoid overwhelming developers with this information.
 - There are also lots of code metrics that we can make available.
- We are doing lots of reviews - we need to judge how effective they are.

Tag Collector

- The beta version of the new Tag Collector (Tag Collector 2 or TC2) is available for testing by all of the developers.
 - The TC2 team (Jerome & Fabian) have been working hard to respond to people's requests for changes (mostly changes to the interface).
 - We need to decide when to make the transition to TC2.
 - This is a decision for the SPMB.

Testing

- I was surprised at how little discussion there was about automated testing!
 - The need for testing is obvious since several recent productions have been compromised by problems that simple tests would have found.
- The testing group is working hard to provide the developers with a standard way to submit tests for their own packages.
- We want to do a thorough set of tests on the 10.0.x series before it is put into production.
- It is a policy set by the SPMB that will dictate what tests must be passed to go with a production release.
 - Using Kit Validation in the interim.