

When are developers allowed to do their work directly on the trunk?

The LSST project expects that all significant development take place on ticket branches; these tickets (and their emergence on the trunk) are under the control of the LSST workflow.

There are, however, three circumstances in which a change may be made on the trunk:

1. The initial implementation of some piece of functionality
2. A "trivial change", which is defined to mean one that it:
 - ◆ does not change the API;
 - ◆ does not change the documented behavior of the module
 - ◆ takes less than 30 lines of code alteration (n.b. a context diff would show a few more lines)
 - ◆ has a (new) unit test that failed before the fix and passes now (this isn't counted in the 30 lines)
3. With the permission of the Package Guru. In this case:
 - ◆ the API may change
 - ◆ the documented behaviour may change
 - ◆ fewer than 50 lines of code alteration
 - ◆ a unit test is still required

In all three cases, a ticket must still be opened.

KT comments:

1. The initial implementation of some piece of functionality

I used to believe in this, but I am now starting to think that

even initial implementations of entire packages should have (ticket) branches. This provides ease-of-use when strict checkin/ticket control goes into effect, and it could conceivably allow automated trunk/HEAD builds of all packages, whether released or not. For pieces of functionality smaller than a package, I think separate branches should be mandatory, or else the trunk/HEAD builds of released packages will be broken more than necessary.

As I see it:

tags: checkins (actually copies) must follow version compatibility
trunk: checkins must build but may not be compatible with other packages
branch: checkins need not even build

Initial implementations typically spend a lot of time in this last stage.

Comment by rhl on Tue 05 May 2009 08:20:22 AM CDT

By "The initial implementation of some piece of functionality" I meant a new package (e.g. ip/diffim). As such, no-one will know about it and I don't see a gain from a ticket branch. On the other hand, the only downside of a ticket is that it's less visible to busybodies like me; but this is a downside --- e.g. it's convenient to think, "Ah, RHL's working on creating meas/algorithms, I wonder where he's got to" and do an `svn co`

`$SVNROOT/meas/algorithms`

Add comment

Your email or username: