

# 2007-11-19 OSP Conference Call

## Functional Team

At last week's Functional Team meeting a decision was made to require Usage Scenarios for all new features. This means that all work on new features should be done in a branch until the enhancement is approved by the community. What does this imply for developers? All work that hasn't had a usage scenario approved by the community should be developed in a branch. In the short term, anyone developing new features – develop in a branch, then submit usage scenario and give demo. We probably won't have time to review usage scenarios before mid-December.

Usage scenarios allow for transparency, whereas just developing means others don't know what's happening, what the implications might be.

In Indianapolis, everyone liked the work IU was doing on Goal Management, so they were informally okayed to develop in the trunk. It would be onerous to retroactively apply the scenario rule to them, so they should continue in the trunk. They expect to release at end of Dec., then QA. That means that we/they can use the January time frame to address community feedback and needs. In the meantime, John Gosney's 19-page specification of their work will serve as the Usage Scenario. John will upload a new version that reflects Sean Keesler's feedback and everyone should review it.

Noah is the "development liaison to the functional team".

It is urgent for the Functional Team to develop the complete process for enhancements. How to work as a team to have a common code base.

If stuff isn't happening in the trunk, no one can play around with the code easily. Is there a box someone can donate that would hold all the branches, so

Now that we are part of Sakai core, and since IU manages the trunk server, could they dedicate a server? Only works if there is only one branch. Maybe each institution should provide server for their branch

Status of new development by Institutions

Is rSmart developing new code? UM will have some development, but we aren't sure exactly what.

## Review Comments on rSmart Enhancements

Sean has put a lot of work into it.

1. Assignments imports enhancement (would need to vet with wider community)

Sean votes against, because it lets teachers edit post-hoc, which would pollute an assessment of what was NOT done.

2. Additional FCKEditors – what's new? Speed. 30 sec. to load 6 fckeditors. IE on PC faster? Redesign to make faster?

3. Add new assignment type. Sean: is it important to group like assignments in the list? Good question. Is there a real need for it?

Contact gradebook folks: ACTION ITEM -ME

4. Sean and Beth vote no. kludge. codes. Ros - option to turn on or off?

5. Lynn says they are already working on this for goal management. Wait and see what IU does.

6. Add view/hide content to rate all function of GMT. Just one more way to split the screen, not a way to show

We need to have a conversation about

a few fundamental things:

guided experience between steps

show content from different tools (e.g., side-by-side view of evaluation and content being evaluated)

Publish a patch for these? Is there time at rSmart to merge these features? Or should rSmart just publish patches and let the community take ownership as desired. rSmart will discuss. This was done against very old code, and not all features still needed. Make options you can switch on or off.

## Outstanding QA Issues

No info from Marist this week

Embedded Images in Resources

Since there are workarounds in place,

Out of the box current storage

John E will close

... from Sig Coordinator Login

Deleting a matrix throws a stack trace

There was an earlier problem that got fixed, but this problem only occurs on one server. Hard to fix if Beth can't reproduce

Resources seem to expire by themselves

Predictable reproducible model needed

Matrix crashes if it references a form that is hidden

Beth fixed it. Now if it's hidden, you don't see it. If it's unhidden, users see it again.