GalaxiaWorkflowDev

The information below may or many to be outdated. All the relevant info needs to be copy/pasted to the new site. Please see http://workflow.tiki.org

---

Status/RoadMap

We have a good activity-based Workflow engine. We need to build a team to test/debug/improve Galaxia. We need to gather a collection of workflow apps which can easily be downloaded and installed on Tiki sites.

TikiTeam

Dayton Clark UserPagedaytonclark
please add your name here!

Trackers

RFEs

- 1.7 Galaxia: Dynamic Mapping of Users/Groups to Role
- galaxia: output/input from tiki

Competition and standards

Galaxia is the only (AFAIK) PHP based open source activity WorkFlow.

CVS Doc section

Important work was done since 1.7.4...

Discussion/participation

Re-usability of the Galaxia Workflow Engine

Since Galaxia looks like a good piece of work still waiting to really take off, I tried using it for a workflow module in Xaraya. Here's a little summary of that experience, in case anyone else is thinking of re-using Galaxia outside of TikiWiki. 2003-09-29 updated with the latest CVS changes

1. overall, you can simply copy the lib/Galaxia directory and get started (use the CVS version for this, and not 1.7.1.1)
2. the config.php file specifies the configuration of the Galaxia Workflow Engine, and you may need to customize that for your environment
3. table names all start with GALAXIA_TABLE_PREFIX, so if you want to keep some common table naming convention or have several instances working on the same database, you can change that now
4. all hard-coded lib/Galaxia paths have been replaced with the GALAXIA_LIBRARY constant, so you can place the Galaxia lib anywhere you want, basically
5. processes are stored in GALAXIA_PROCESSES/*, and activity templates are also copied to GALAXIA_TEMPLATES/* if that constant isn't empty, so you can move the directories that require write access outside of your code base now
6. the Galaxia lib is pretty much CMS-agnostic overall as long as you work with ADODB, but you'll want to specify how activity errors should be shown with the galaxia_show_error() function. Some examples are given in the config.*.php files.
7. regarding ADODB, you may need to define('DB_FETCHMODE_ASSOC', 2) and do a $db->SetFetchMode(DB_FETCHMODE_ASSOC) before you pass the database handler to Galaxia
8. you can specify a template header to be added at the top of new templates via the GALAXIA_TEMPLATE_HEADER constant. For Smarty templates, you could use {*Smarty template*} for instance
9. file read operations typically relied on a single fread(...,filesize(...)) call, which often fails to retrieve the complete file on some platforms (e.g. Windows), so that code/template updating was unreliable. Those have been replaced in several places, but some might have been missed. And if you do your own PHP error handling, you'll probably want to verify all @ operations and add some file or directory checking if necessary.
10. in src/API/Instance.php, non-interactive activities are executed by making an call to the function galaxia_execute_activity(). In Tikiwiki, this is done via an external web call to tiki-g-run_activity.php, but you'll want to replace that with some API function call, or whatever is most appropriate for your environment.

In short, pretty good re-usability overall - you should count about half a day to handle any portability issues for the Galaxia library itself.

If you want to re-use the tiki-g-* scripts and templates from Tikiwiki as well, to get you started on the GUI side of things, you'll need to :

1. specify the current user in $user
2. create some fake Userslib that can handle get_users(), get_groups() and get_group_users() calls, and provide a global $userlib variable
3. provide global $tikilib and $dbTiki variables pointing to your db connection
4. set $maxRecords and some $feature_* and $tiki_p_* variables as well
5. insert stripslashes() in a few places in case magic quotes are on, and htmlspecialchars() when templates are being edited

Integrating the scripts, converting the templates and getting things to work overall may take another day or so depending on your environment.

I hope that my modest contribution can help a bit in spreading the use of Galaxia to other PHP environments. It's a great piece of work

Mapping of CMS user groups to Galaxia process roles

One of the issues with the current Galaxia library is that you can only map individual users from your CMS (TikiWiki, Xaraya, ...) to Galaxia process roles, not user groups. True, it's possible to get all the current users from a particular group mapped to some process role via the admin interface, but this is a one-shot operation, and any changes you make to that user group afterwards will not be reflected in the user-role mapping of Galaxia.

So new users will not be able to use any workflow processes unless you map them to a process role, or if a user is assigned to a new group, he won't be mapped to a corresponding Galaxia role either.

Ideally, it should therefore be possible to permanently map user groups from your CMS to Galaxia process roles in the future. Some design considerations are discussed below...

Assumptions about the user groups in your CMS

- users can belong to more than one group
- it is possible to retrieve the list of groups the current user belongs to via some simple API (to be included in config.php)
- group identifiers may overlap with user identifiers

Assumptions about how Galaxia should deal with groups

- we can re-use the same tables that currently exist, i.e. no group_roles table is required
- group mappings can be distinguished by Galaxia, e.g. because group identifiers stored in the user_roles table start with group_
- it is not necessary that all users belonging to a group must be known to Galaxia
- if the current user belongs to any group that has been mapped to a role, he will be granted access
- instances/workitems can be assigned to users that are not explicitly known to Galaxia (is this possible ?)

The discussion is now open