TikiPlan19

Holmes Chapel, Wed Oct 20

TikifestUK has been going now for some days, mainly featuring Damian and Mose, with the support of Olisb, Berry, Taryn and Isabel. More friends will join in the next few days, like Gary Alexander, Michael Linton, Tom Salfield, and more. (see http://uk.tiki.org/gallery10)

Our main objective is to push the 1.9 tikiwiki version to a state that can be released, as well as setting up the context for a nice future evolution of the community collaborative development,

The 1.9 release cycle has begun long time ago (june 2004) first rc was shipped in , and the freezing of features is not easy to sustain. 1.10 HEAD branch is not stable enough mainly because not used in production environment. The developers population is huge (263 people) and the environment, that is so important in our development rules, is more and more difficult to embrace.

We can notice some problems in the actual development scenario.

First we have 4 branches, many cvs modules, which make contribution not always easy for the newcomer. That issue should be sorted out by the use of the mods installer/deinstaller, providing an unique entry point for unqualified or not known contributions. The merges have been nice for a time, but now with the double merge more and more things become hazardeous, and it also breaks on some changes that should be necessary to do, like moving code around, fixing the indentation, and such things.

Second we have more and more specific features that are not obviously of a general interest, some requiring extra applications, not available on shared hostings, like tikimaps, tikisheet, homework, jukebox. We value those contribution as really kewl ones and that's also what makes tiki such an interesting framework, but taking them in account in the general-purpose development is now heavy and threatens with staticity, hindering further evolution or improvement of the general code base. By using mods projects, then spliting the development in subgroups, it will probably bring more fluidity in the evolution of the main trunk as well as on the various developped features.

So, we are proposing to change radically our development strategy, by shifting some subprojects to mods.tiki.org and by that way also improving it drastically for being an improved replacement of the sourceforge platform, at least for the mods for now, maybe later for the whole tikiwiki development.

That change can't be too much brutal and will be progressive, by moving things to mods progressively. There is a set of easily moveable features that we can already manage in mods, as listed below. So 1.9 will be halfway to the goal we talk about, and 1.10 will probably have more things moved to mods if it appears that this idea is valid under the pressure of the ongoing work.

1.9 preparation process

As lot of time passed, we are going to backport from 1.10 the tested improvements, in more than moving mods to mods.

1.9 timeline

Sunday 25th october is the planned deadline for adding stuff into BRANCH-1-9, then we can quickly debug on new features added and improvements.

After that date, nothing, and really nothing should be added and the only activity expected on 1.9 should be about debugging existing stuff, as well as translating. Then we expect to be able to release the 1.9dr4 on november 1st. That release is aimed at developers and translators only and should not be used in production environment.

2 weeks will be necessary for fixing the last translations and work on documentation as far as possible.

Then we can have a final 1.9 Release, with a good stability level, around sunday 14th in november,

Mods project management and distribution

The mods feature make possible to split the updating of tikiwiki in parts, by using http://mods.tiki.org as a reference provider of new features for testing or using in production. The mods installer is almost done and is the priority for the 1.9 release, as well as the tikiproject feature that will manage the project aspect and the publication side of each subproject development.
That mods feature is expected to work in a way similar to portage or apt : a remote server is used as a reference for getting index of new available features, plugins, themes, whatever. A set of metafile will handle the operations to be done at install and at removal, in SQL statements, scripts and configuration options.
The mainstream server would be mods.tw.o for spreading the tikiwiki community development, but arbitrary servers can also be used to manage self-publication of new features, or even internal releasing of home-made features in a network, intranet or VPN.
The mods is not following the branches schemes, as many features are not depending on the version used. An extensive documentation on the Mods system will provide proper information about how it's working.

moving to mods

The tiki-project feature, developed by damian, will be ready before the freeze (this sunday) and will be able to handle the sub-group development on specific features or addons in tikiwiki. The main goal is to reduce the tikiwiki mainstream tarball to a lighter profile, and puts all the optional and one-time used features in mods. Of course we need to discuss about what will go in mods and what will remain in tikiwiki main trunk.
Here is a list of features that we think appropriate to move to mods before 1.9 final release.

- **TikiSheet**, that also contains licence doubty libs, and is still in development
- **Wikigraph** plugin and graphviz lib (also subject to licence diff)
- **Galaxia** is a big chunk and is quite independant, it should go to mods
- **phpCAS** feature is also independant, and very specific in its use as you need a special server environment
- **maps**_ feature requires as well some special server setup and is not useable by everyone, that should go to mods
- the **homework** feature is maybe appropriate to move to mods, but that's not decided yet
- the **themes** are numerous and should be limitied to a small number of theme into the core release. The mods is very adapted to distribute more and more themes.
- Most **languages** supported should move to mods as well, depending their level of completion. Actually the languages should deserve a dedicated module in the cvs, Ultimately all languages should go to mods.
- **fonts** for pdflib are heavy and not necessarily used, and we can have them in mods too, except one that would be the default one (probably helvetica).
- tiki jukebox needs to be finished and will become a mod.
- *... if you think about some more, please suggest them below...*

Some "one time" features don't need to be in the trunk, and could be installed when needed, and removed when not anymore.

- **Import** phpwiki and other import filters are used only once usually and could be installed, used, and then uninstalled.
- the **SQL upgrade** scripts are one-time used, they should go to mods, as the mods feature can execute sql queries as well as phpfiles. the main advantage is that it enables the possibility to downgrade tiki (which is not very easy right now)
- the converted **database files** can also be in mods, as most of time they are not used (in regard of the proportion of mysql users compared to others in tikiwiki population)

- *... if you think about some more occational features, please add them below...*

Main trunk fixes and changes

What remains in tiki main trunk and requires an intensive attention in next days, in more than the declared bugs on ReleaseProcess19 as well as in sf bugtracker.

- rewriting of **htmlMimeMail** lib (or getting agreement from author for use in an LGPL release). If rewriting is done, mose will handle it.
  - Marc contacted the author that said it was a BSD licenced lib, so we have no trouble and can keep it !
- replacement of **Date** and **Netsocket** libs (a rumour says redflo could handle that)
- replacement of **POP3** GPL library
- we need to be sure that all **help-links** point to a valid page, by at least changing them to go to valid pages on tikiwiki.org (everyone is invited to help on this, that's not high level technical work, just a matter of changing some urls in templates).
- the **search** feature had some recent improvement (about permissions checking) and it needs to be tested extensively (batawata did work on it, we need more eyes to test and validate it).
- we need to have **http metadata** more fitted to the actual content of each page (eg. page description for wiki pages, heading of articles can be in meta description) damian plans to work on that.
- we need to know what have been done in **quizzes** features and check if it's useable. ggeller worked on it, his input is primary on there, but he worked with dgd that now uses moodle, we think that the motivator is not so intense anymore)
- **FAQ** needs a per-item permission handling, which is easy to add, and very useful. either mose or damian will manage that trivial issue.
- we need to check up the permissions handling into **multiprint** and pdf generation features.
- *... if you know more issues, please list them below ...*

What is new and will come in cvs in next days

- **polls** feature has got some enhancements by mose
- gary (gg) have hacked the **calendar** for adding the possibility of events management, where people can attend, and get notices by mail. Quite nice work, and at last the calendar will evolve with some new blood.
  - Gary committed his work, please test it and feedback !
- *... if you think about some more, please suggest them below...*

Other points of interest

The third party libs need to be upgraded to recently released version

- smarty
- adodb
- phplayers
- jscalendar
- ...

SourceForge patches and other sources

- we need a review of what have been uploaded the recentlly, to see in the week what can jump into 1.9 or in mods

And we also have some less immediate plans, that will be filled if times permit it

- we absolutely need a rationalization of the **comments** include, which is used by forums as well as for comments. It needs to be split, and comments, as the categories, deserve a specific code not linked to forums.
- the **tikiMarket** project is a special workshop, that could enable the community matching between wishes and offers, possibly but not necessarily involving hard cash transfers and third party referee. It includes the necessity for a voting system for taking in account the biggest part possible from the community.
- The **wikifarm** is an old idea but still very much wanted.

The list above is absolutely not exhaustive at all, it's mainly the result of a brainstorming between damian and mose. Please add more input if you have.

-- mose and damian with the help of Berryweb