URL Rewriting Revamp

2013-01-28 A PHP route.php file now replaces the rewrite rules from _htaccess
http://sourceforge.net/p/tikiwiki/code/44661/ for Tiki11

Very important: adding to tiki log or error log so we catch them all through usage

At the TikiFest, we decided to Update all links to have friendly URLs

Lets attempt to solve URL rewriting problem in Tiki once and for all. Here are goals:

1. Server-agnostic enabling Tiki users to deploy any web server.
    - Apache, IIS, Nginx, Lighttpd, Hiawatha, Cherokee, LiteSpeed, etc.
        - IIS
        - Hiawatha
        - Nginx
        - Lighttpd
2. Reduce maintenance server specific config files .htaccess vs web.config, etc
3. Have great SEO
4. Nice to read for humans
5. To work with Canonical URLs
    - For existing ones (wiki, blog, articles) and future ones (forums, trackers, etc.)
6. New feature: if something changes to a wiki page, article, blog post, etc. and this breaks the URL, make it an option (default on) to have a redirect 301
    - Articles & blogs: nothing needs to be done because it's well handled by canonicals
    - Wiki pages: could offer to append a page alias or something else?
7. Be able to deal with a URL being chopped off accidentally (in an email message line breakage) or intentionally to shorten URL
    - Current SEFURL for articles and blogs make this possible
8. Seemless upgrade path
    - All old URLs should have 301 redirects to the new ones
    - All currently accepted characters should continue to be. Ex.: a-z . _
9. When a URL is broken (a site moves from another tech to Tiki), all info should be thrown to the search engine to try to find new URL for the old page
    - So this would bypass Apache error pages?
10. Work for non-Latin character set
    - Test with Korean pages on doc.tiki.org
    - Tiki blog post with title "ä, ë, ï, ö, ü, ÿ Ä, Ë, Ï, Ö, Ü, Ÿ å, Å æ, Æ œ, Œ ç, Ç ð, Ð ø, Ø ¿ ¡ ß" becomes blogpost3-a-e-i-o-u-A-E-I-O-U-a-A-ae-AE-c-C-D-o-O-s
    - luci reported that in Czech language, "Most" and "mošt" are two very different things (a bridge and a cider). In the blog post, this is not a problem because there is a blogpost ID. But if you were making a glossary in Czech language, there would be a collision. This is an edge case. Perhaps Tiki could detect this and offer a disambiguation page?
        - I just checked in trunk on demo and there is no way to create "Most" and "mošt" as different

pages. So a manual disambiguation page is already necessary.

11. Pollution of URLs by sending relative links to inexisting subdirectories
12. Be consistent thoughout Tiki features
    - For example, ¥ · £ · € · ¢ · ₡ · ₲ · ₣ · £ · ₥ · ₦ · ₧ · ₨ · ₩ · ₪ · ₫ · ₭ · ₮ · ₯ · ₹ is OK but the same characters are ignored for article and blog post SEFURL
13. non-ambiguous correspondance back and forth Jyhem: can you explain?
14. all the rewriting could be done in index.php and use the database, so everyone could customise their sefurls to suit the site
    - Perhaps this could address http://tiki.org/Infrastructure+Team#Rewrite_Rule_collisions
15. Performance
    - Be good for Reverse proxy
      - May need unique URLs per language
16. Make it possible to have a new optional feature where URL is different than wiki page name, blog post title, etc. Ex.: blog title: "Best Practices for Optimizing URL Structure" : url: best-practice-url-structure
    - And this is used for the canonical
    - Someone could use this for the old URL before a migration to Tiki
17. Should there be any link with Sitemap?
18. Should be able to handle when
    - Tiki is not in the root directory
    - Other apps than Tiki are installed in the same root as Tiki or in a subdirectory
19. Ideally, we'd like to make it possible to put everything that needs writable access in one directory (temp, templates_c, etc)
20. Ideally, we'd like that site customizations could be all done in one directory so you can easily see what is modified, without SVN. All .php/.tpl/.css/.png files there should "magically" overrride Tiki defaults. Historically, for a Tiki theme, you need to put some files in templates/styles/abc/* and styles/abc/* and styles/abc.css
    - custom/tiki-calendar.php
    - custom/templates/tiki-calendar.tpl
    - custom/templates/styles/abc/tiki-calendar.tpl
    - etc.
21. Future-proof
    - What about mobile?

Issues with status quo

- While researching about the possibility to use other than Apached webservers for deploying Tiki, we found the following post post explaining that:

  > “...according to RFC 3986 using ‘+’ outside of the scheme or query string in a URI as the Tiki shortlinks do is invalid...and Tiki is clearly at fault for generating non-compliant URIs...”

Steve Streeting wrote:

> *My involvement was just trying to get it working as best I could with the minimum of changes, and my hack was pretty nasty:*

+++ tiki-index.php 2012-05-20 12:06:13.000000000 +0100
@@ -98,6 +98,11 @@

$info = null;

+// SJS Hack for Nginx compatibility +// Nginx encodes '+' characters when going through rewrite, so replace them +$_REQUEST%22page%22 = strtr($_REQUEST%22page%22, "+", " "); +// End SJS Hack + $structs_with_perm = array();
if($prefs'feature_wiki_structure' == 'y') {
$structure = 'n';

So basically, this undoes Nginx's (strictly correct) encoding of the '+' characters and puts them back as spaces, which then works when Tiki looks up the page. However, if any page name actually includes a REAL '+' character (say if a page was called 'C++'), then those pages would fail to be resolved, because we'd turn them into spaces - but we can't tell the difference by the time we receive them. There aren't that many pages with actual '+' characters in the name though, so this hack fixed more things than it broke. Good enough for government work ;)

A permanent fix which is compliant with the stricter rules Nginx is adhering to would be awesome though.

Cheers
Steve

Who

- Gour
- Marc Laporte
- You?

What

The present Tiki's problem with URL rewriting for non-Apache server is nicely explained by Jean-Marc-Libs in this post:

> “More precisely, no RFC ever made + a *replacement* for space. + is actually an alternate, visually friendly *url-encoding* for space.
> The classic, visually unfriendly one being %20.
>
> So, if the Tiki page is called "Tiki page", the URL should be sent as tiki-index.php?page=Tiki+page or tiki-index.php?page=Tiki%20page (equivalent ref RFC). And that is what the web server should get, post rewriting rules.
> If the page is called "Tiki+page", then the RFC says it should be tiki-index.php?page=Tiki%2Bpage

> Since this is a very common misconception among developpers, apache does allow for replacing space with +.
>
> For similar reasons, firefox will detect %20 or + in the url and display space instead. You can still see it's really %20 if you cut-paste the urls field of the browser to an editor.
>
> Since the Tiki confuses both spaces and +, Tiki does not allow + in page names. That's not a very good way of dealing with the issue.
>
> Hope this makes things clearer."

The possible solution which would be server-agnostic and therefore much easier to maintain was suggested by the developer of Hiawatha server who wrote (see this thread):

> "But, I seriously think that there is a better way to solve your issue. Create a new PHP file, say route.php and rewrite every request for a non-existing file to that route.php. In route.php, you determine what other PHP file should have been called. At the end of route.php, simply include that PHP file. In other words, transfer the routing business logic to where it should be: inside your application and away from the webserver's configuration."

Some participants in the tiki-devel thread expressed concern about backward-compatibility for the present TIki sites and Hugo Leisink - main developer of Hiawatha webserver addressed it with:

> "That should also not be a concern, because I'm also very sure that it's perfectly possible to create a PHP script that has the same functionality as the URL rewrite rules.
>
> The only thing current tiki-users have to do is to place the new route.php on their server and change the current URL rewriting rules with the rewrite-requests-for-non-existing-files-to-route.php one."

Related things to think about

- Canonical links
- Character substitutions -> There are some very important questions on this page
- Bad characters
- htaccess

When

- route.php done for Tiki11

Nice to have

- route.php could log errors to Tiki logs

Handling 404

Now that we have route.php, we could easily support

- Link Checker
- http://notfound.org/ (more languages and countries would be better)
- https://isc.sans.edu/404project/
- URL Rewriting Revamp: adding to Tiki log or error log so we catch them all through usage
- Tighter integration with Apache's > .htaccess (error messages, etc.)

Ideal rewrite rules

https://www.google.com/search?q=best+practices+for+SEO+in+URLs

- So hypen is reportedly a better separator (Matt Cutts on hyphens).
- Should we keep case or put to all lowercase?

Examples

Wiki

In trunk, we have a feature to prevent certain characters in page names. If you try, you get:

```
The page name specified contains unallowed characters. It will not be possible to save the page until those are removed: /?#[]@$&+;=<>
```

We would want all variants

- space:http://dev.tiki.org/tiki-index.php?page=Search engine optimization -> http://dev.tiki.org/Search-engine-optimization
- hyphen: http://dev.tiki.org/tiki-index.php?page=Search-engine-optimization -> http://dev.tiki.org/Search-engine-optimization
- plus: http://dev.tiki.org/tiki-index.php?page=Search+engine+optimization -> http://dev.tiki.org/Search-engine-optimization
- underscore: http://dev.tiki.org/tiki-index.php?page=Search_engine_optimization -> http://dev.tiki.org/Search_engine_optimization

Blog

- Blog title: http://dev.tiki.org/tiki-view_blog_post.php?postId=22 with a title of "The module-ificaton of Tiki themes & easy module export via profiles in Tiki 9.1" ->

http://dev.tiki.org/blogpost22-The-module-ificaton-of-Tiki-themes-easy-module-export-via-profiles-in-Tiki-9-1

- Notice how period (.) and ampersand (&) become hyphens

Perhaps example.com/test+page could become a search for those two terms and lead to example.com/test-page or search results?

Related links

- https://github.com/auraphp/Aura.Router#readme
- https://github.com/auraphp/Aura.Uri#readme
- Infrastructure Revamp
- https://wiki.mozilla.org/Support/Tiki/UrlHandling
- https://github.com/chriso/klein.php#readme

alias

- URL handling