2010-06 Marc Laporte Interview Espirito Livre Magazine

This article has been published in Portuguese in http://espiritolivre.org

Table of contents

Introduce yourself to our readers. Where do you live and which area do you work these days?

My name is Marc Laporte and I live in Montréal, Canada. Since 2003, I have been full time on Tiki.

Introduce Tiki Wiki. What does "Tiki Wiki" mean?

Tiki can mean many things. Typically, it's referring to Tiki culture or the Kon-Tiki raft. The official name of the project is "Tiki Wiki CMS Groupware". The backronym is "Tightly Integrated Knowledge Infrastructure"

Tiki is a full-featured, web-based, multilingual (35+ languages), tightly integrated, all-in-one, open source Wiki+CMS+Groupware. Tiki can be used to create all kinds of Web applications, sites, portals, knowledge bases, intranets, and extranets. The underlying technology is PHP/MySQL/Zend Framework/Smarty/jQuery.

http://info.tiki.org/Fact+Sheet
http://doc.tiki.org/Features

How has the idea of creating a CMS born? Tiki Wiki is based on another CMS?

Tiki was started in 2002, not based on another CMS but making good use of available components, such as Smarty. The founders wanted a completely free CMS without license restrictions and a huge list of features, all in one package.

An early interview, when Tiki was awarded "Project of the Month" on SourceForge.net in July 2003, offers some interesting historical perspective:
http://sourceforge.net/potm/potm-2003-07.php

What are the differentials between Tiki Wiki and other CMS software? What public is Tiki Wiki indicated to?

Tiki is the Open Source Web Application with the most **built-in** features.

How? In a nutshell "Software made the Wiki Way". Wikipedia is using the wiki way to build a free body of knowledge. We are doing something similar but to build a web application.

The "Tiki model" consists of:

- Wiki community (do-ocracy)
- Wiki way participation to the code (400+ with full write access to core)
- Scheduled releases (2 major releases per year)
- All-in-one codebase (1 000 000 lines of code, with everything bundled. Each feature is optional)
  - Inherent synchronized releases (all features have to be ready at the same time)
- Lots of features, but no duplication (in a wiki, similar/related content is merged, so the same is applied to features)
- Dogfood (Tiki is a community recursively developing a community management system)
- Very open community. Everything is discussed. Even this interview was community-sourced

We feel that built-in features tend to be better than 3rd party extensions. And having everything in the core permits us to change things more rapidly between versions and offers tighter integration. It brings its own set of challenges (huge code base, learning curve, tons of options in the admin panels, huge effort at release time, etc.) but we prefer dealing with these than with dependency hell, duplicate 3rd party extensions, long release cycles to protect the extension ecosystem, etc.

For example, we

- added a search engine to find among the over 1000 (!) preferences & settings in the various admin panels.
- streamlined our release process.
- developed a profiles system, so admins can pick a pre-configured site, and not need to go through the admin panel. The many profiles are managed collaboratively in wiki pages and 4 profiles are "featured":
    1. Personal Blog and Profile: personal "about me" site
    2. Small Organization Web Presence: Publishing/Brochureware with User-generated content (UGC), but clear what is official vs user-contributed
    3. Collaborative Community: User profiles, collaboration, users register
    4. Company Intranet: Closed team collaboration, users do not register

In the core+extension model, the core stabilizes for a longer period so extensions can be updated. Tiki has a much faster release schedule (compared to similar applications), precisely because we don't have to worry about breaking third party extensions.

Most CMSs don't have a powerful wiki (As proof, they don't dogfood). And most Wikis are missing important features of a CMS (ex: no template engine). Tiki is one of the rare solutions which offers both, and it has since very early on. There is no set limitation on what Tiki should do. For example, Tiki5 saw the addition of a shopping cart, and there are discussions about accounting functionality. So this means just about anybody with a web project should take a look at Tiki.

http://tiki.org/Model
http://profiles.tiki.org/
http://tiki.org/Wiki+Landscape
http://tiki.org/CMS+Landscape

Tiki Wiki is licenced under what licence? Is there a corporate version? What are the prices? What are the differences between community and corporate versions?

Tiki is LGPL 2.1

There is no "corporate" version. While the license is permissive in terms of integrating with commercial applications, the culture of the community is to share all the available code in the core. If you want to invest funds in Tiki, the way to do it is to hire community members for support, training, etc. and to sponsor features (which become part of the core). You can also make a donation to the Tiki Software Community Association.

http://tiki.org/License

Is there a company that offer official support services to the product? What is it and why should users chose it?

We have a "community open source approach". Thus, we have a large number of consultants available for

hire. People will choose depending on the type of project, or various affinities (geography, language, etc.)

http://info.tiki.org/Consultants

How is your relationship with the open source community?

We are huge users of open source code. PHP, MySQL, Zend Framework, jQuery, Smarty, and many many others. We upstream our fixes when relevant.

See http://tiki.org/Partners

In the past, many CMS became known for their security troubles. How is the "security factor" treated in Tiki Wiki?

Security is a vast domain and risks depend on the use case. In an open community use case, it's important to protect users against each other (ex.: privilege escalation), however, in the brochureware use case, there are only a few trusted editors. By default, we ship with almost everything off and with the most secure defaults. Admins have a security admin panel to set the security features for their use case.

It's very important that site admins can upgrade quickly or turn off the feature which is vulnerable. The all-in-model makes upgrades easy. Site admins are warned of new versions in the admin panel (can be turned off). As part of our release procedures, we review that every executable file has a feature and/or permission check. Thus, the site admin can disable the vulnerable feature without having to upgrade or to take down the site.

By default, risky HTML &))JavaScript((is blocked which is more secure but not very user-friendly in the case of a blog, where only trusted users have logins. So we have different permissions so the site admin can decide who is allowed to use potentially risky features.

We have a security team, which responds to alerts and pro-actively improves various features according to our security dashboard.

http://security.tiki.org/

In your opinion, what is necessary to build a CMS with quality, elegant, safe and stable? Is it possible combine those four elements in just one solution?

With enough time and brainpower, it's possible to combine all four. Which is why a top priority is to get more people on the project.

Everyone can participate to any & all aspects of the code or the community:
http://tiki.org/SWOT

Who don't you encourage using Tiki Wiki?

Tiki has two sweet spots:

1. If you have a complex project and you want all parts to integrate well.
2. You have several diverse projects and you want to streamline on a common solution.

Tiki offers over 1000 settings & options in the control panel. It is fully themable with CSS and the Smarty Template engine. So with a custom theme and with configuration, you should be able to accomplish the vast majority of projects. The culture of the project is to address a specific challenge while making a generic feature, which is added to the main code base. The next stable version is never too far off and this

keeps upgrades easy.

So, don't use Tiki if you intend to modify it heavily without sharing back the code. There are 600-700 commits per month. Within one year, your Tiki instance will be out of sync with the rest of the project.

Try to explain or summarize Tiki Wiki in 3 words.

Wiki way webware

Do you think a CMS make web developers lazier and less concerned about programing itself?

The goal of a CMS is to reduce the technical barrier for content creators. With more available tools, the developer becomes more a solution finder than a problem solver.

It really depends on the use case.

What are the plans for the future? What comes next?

The continued growth of local Tiki ecosystems (user groups, consultants, events, translations, etc.) will be very interesting along with the new branding.

We will continue our fast-pace of releases. Tiki6 (October 2010) will become Long Term Support (LTS), and thus, we'll be in a position to do some re-architecture in Tiki7 (April 2011). For example, to upgrade to Smarty 3 and Zend Framework 2 (if it's ready by then)

Integrating with source control systems (SVN at first) will make Tiki a good all-round solution for software project management (we already have a ticket system, etc.)

Over the years, we have pretty much done most of the features possible with existing technology and we integrated with other open source projects which rely on Flash.))BigBlueButton((for real-time collaboration tool (Audio/Video/Screensharing/Chat) and Kaltura for web-based video editing. The next big thing will be to adopt HTML5 and do the missing bits like image manipulation. We tried with java applets but we never got something stable and user-friendly enough.

http://dev.tiki.org/Roadmap
http://branding.tiki.org/
http://dev.tiki.org/HTML5

For those who want to know more about Tiki Wiki, what should they do? Where should they search for? Where do they start?

You can download or use a demo from:
http://info.tiki.org/Get+Tiki

Leave some words to our readers at Espirito Livre Magazine.

Give someone a fish and you feed them for a day.
Teach someone to fish and you feed them for a lifetime.

Add that knowledge to a wiki
And countless people will be able to learn on their own

And share their knowledge
And many more will feed their families

And join http://br.tiki.org/ !