

CSS Frameworks and Preprocessors

On a related note, we should use [schema.org](#) (?)

What is a CSS framework?

[Wikipedia](#) says "CSS frameworks are pre-prepared libraries that are meant to allow for easier, more standards-compliant styling of web pages using the Cascading Style Sheets language."

Currently we use *lite CSS "framework" but it does not provide grid layouts. It serves us well and is stable solid CSS base for many years in Tiki so far but it is just that - lightweight basic 3-column layout. So, do we need to extend it or pick some additional ?

CSS frameworks - the advantages (benefits)

- Currently in Tiki, on the up side:
 - *lite.css along with layout.css, etc has been providing a default layout.
 - There is a kind of badly organized framework with defaults provided for themes to inherit.
 - Theme stylesheets extend the default layout for customized sites.
 - Regarding grid systems, custom themes can introduce a grid layout by declaring div#fixedwidth the container, sizing side columns appropriately and providing div classes for wiki text content.
- On the other hand:
 - Tiki doesn't have any built-in grid awareness and it would be hard for a custom theme to cover all elements.
 - With no good organization or documentation of element design, there is a lot of redundancy and near-duplication in the CSS (200 colors in layout.css, design.css and the /css files, for example).
 - About out-of-the-box CSS frameworks, here's an argument against using them - <http://chrisepstein.github.com/blog/2009/02/15/a-sassy-response-to-css-framework-disadvantages/> - and using Sass CSS preprocessor instead.

Other points:

We need to think about [responsive web design](#), along with CSS frameworks and grid systems. An unresponsive grid isn't good enough these days. Do we build screen-size responsiveness into the CSS, or do we rely on jQuery to provide smaller views?

https://www.ohloh.net/tags/responsive_web_design

CSS preprocessors

Here is an introduction to [CSS pre-processors](#).

CSS preprocessors - the advantages (benefits)

Probably we should think about this on two levels:

Tiki core CSS (use by Tiki code contributors for basic layout, feature functionality, default design, etc.)

- Grid arrangements like Blueprint can be incorporated into some CSS preprocessors (Sass/Compass).
- Tiki CSS files need to be reorganized anyway, so conversion to Sass or Less would provide motivation and focus thinking.
- Hopefully, there would be fewer unique rules being added to the stylesheets if there is a logical and documented organization of Tiki CSS rules.

Tiki themes (best practices for additional/custom themes)

- Theme variants can be produced faster.
- Theme stylesheets would benefit if the default CSS becomes more modular.

Are there any problems or disadvantages using CSS preprocessors ?

- Specifically about Tiki, developers who contribute or edit CSS files would need to agree on which preprocessor to use, and get up to speed on using it.
- Presumably, in the revamped CSS directory, there would be a new directory containing the preprocessor files, which would be edited directly, and the compiled css files, which wouldn't be edited directly. This isn't a disadvantage, but just something to work out.

Just to name few from the article:

- Code duplication in compiled CSS
- Extend and mixins can be bad for maintenance
- Nested code can be harder to read

Please note that they are opinions of a single developer, who thinks that "A stylesheet full of mixins, if/else, loops, variables, functions, etc, will be as hard to maintain as a bloated hand-crafted stylesheet, if not harder".

Another opinion is here: <http://www.amberweinberg.com/januarys-12412-researching-less-sass/>

If yes, which CSS framework to pick ?

About picking a framework, here is a list:

https://www.ohloh.net/tags/css_framework

In order of number of active committers:

https://www.ohloh.net/p?query=css_framework&sort=active_committers

I can't tag this one as CSS framework but it would be on the list

above: <https://www.ohloh.net/p/blueprintcss>

If yes, which CSS preprocessor to pick?

Somebody said that currently Sass is to Less what jQuery was to Mootools a while back (that is, gaining the edge in mindshare). Here's one opinion: <http://css-tricks.com/sass-vs-less/> .

Tiki experiments

- [Using Less CSS with Tiki](#)

Suggestions

Gezza suggested [Twitter Bootstrap](#) which is [LESS](#) based and [Apache-Licensed](#) (so should be OK in [LGPL](#)?) Version 3 is licensed MIT so all is fine. ~~But it seems there could be problem because Twitter Bootstrap includes [Glyphicons](#) which are [Creative Commons Attribution 3.0 Unported \(CC BY 3.0\)](#) and require "you must always add a link to [glyphicons.com](#) in a prominent place (e.g. the footer of a website), include the CC-BY license and the reference to [glyphicons.com](#) on every page using [Glyphicons](#)."~~ [Font Awesome web-font icons](#) was added and also "[Glyphicons Halflings](#) are also a part of Bootstrap by Twitter, and are released under the same license as Bootstrap. While you are not required to include attribution on your Bootstrap-based projects, I'd certainly appreciate a visible link back to [glyphicons.com](#) in any place you find appropriate (footer, docs, etc)."

So what if we just use LESS to get the CSS pre-processing and [jQuery UI Bootstrap compatible "theme"](#) (because we use jQuery UI anyway already) to make it look nice and Twitter Bootstrap themes compatible ? Luci thinks it is not necessary to include all the Twitter Bootstrap stuff anyway. Gary agrees. Maybe Twitter Bootstrap could be done at the theme level.

[The big news among the many new features in this release is the mobile administration capabilities and the deep integration and support of Twitter's Bootstrap framework \(which enables much of the mobile admin\).](#)

jQuery UI Bootstrap author claims "With this theme, not only do you get the ability to use Bootstrap-themed widgets, but you can now also use (most) of Twitter Bootstrap side-by-wide with it without components breaking visually."

(There are also docs on how to use Sass/Compass with jQuery-UI, and there's a Sass Twitter Bootstrap implementation: <http://thesassway.com/projects/sass-twitter-bootstrap>.)

If we decide to use CSS preprocessor like LESS we will probably need to use an optional LESS compiler to turn LESS files into CSS and to avoid little delays less.js (which is client side JS compiler) can cause (load performance) - most probably [Assetic](#) or the PHP one: <http://leafo.net/lessphp> (About other compilers read the <http://fadeyev.net/2012/02/16/less-update/>) ?

(Note: CSS preprocessors are dev tools to create and manage CSS files. One workflow would be to work with the Sass or Less files in local dev machines, compile them locally as the CSS files in the Tiki package, and SVN commit both the Sass and compiled CSS files (after agreeing on the compile options for file format consistency regarding comments, compact or expanded, etc.). This way, there is no compiling done by the production server or in the browser.

Are you in favor of using CSS preprocessor ?

Accept	Undecided	Reject
1	4	0
<ul style="list-style-type: none">• jonnybradley	<ul style="list-style-type: none">• luci• gezza• chibaguy• eromneg	

Are you in favor of using even more comprehensive / advanced / complete / complex CSS framework ?

Accept	Undecided	Reject
2	1	1
<ul style="list-style-type: none">• jonnybradley• marclaporte	<ul style="list-style-type: none">• gezza	<ul style="list-style-type: none">• luci

Jonny's 2¢: *We already have a hugely complex CSS "framework", so actually why i am voting for this one really is to replace our chaotic and unique CSS system with a more standard and commonly used one.*

Are you in favor of simply focusing on the new native "CSS Variables" ?

- <http://www.sitepoint.com/native-css-variables/>
- http://developer.mozilla.org/en-US/docs/CSS/Using_CSS_variables
- <http://trac.webkit.org/changeset/120154>

Or even now in PHP (we could call our files styles/abc.css.php)

- <http://sperling.com/examples/pcss/>
- <http://davidwalsh.name/css-variables-php-dynamic>
- <http://icant.co.uk/articles/cssconstants/>

Accept	Undecided	Reject
0	2	0
	<ul style="list-style-type: none">• marclaporte• jonnybradley	

Comparisons

It's easy to compare them on Ohloh:

https://www.ohloh.net/p/compare?project_0=Blueprint&project_1=960+Grid+System&project_2=Compass+Stylsheet+Framework

This comparison is a little bit apples and oranges as Blueprint and 960gs are actually CSS class libraries and Compass is a production framework.

Comparison of Sass and less-css on Ohloh:

https://www.ohloh.net/p/compare?metric=Summary&project_0=Sass&project_1=lesscss&project_2=

Related links

- <https://github.com/kriswallsmith/assetic>
- <http://www.phpsass.com/>
- <http://code.google.com/p/phamlp/>
- <http://code.google.com/p/css-template-layout/>
- <http://ontwik.com/css/css-frameworks-make-the-right-choice/>
- <http://www.quora.com/What-is-the-best-open-source-CSS-framework>
- <http://webdesignerwall.com/trends/960-grid-system-is-getting-old>
- <http://lessframework.com/> CSS for Print
- <http://www.catswhocode.com/blog/8-css-preprocessors-to-speed-up-development-time>
- <http://net.tutsplus.com/tutorials/html-css-techniques/sass-vs-less-vs-stylus-a-preprocessor-shootout/>
 - <http://learnboost.github.com/stylus/>
- <http://the-echoplex.net/csscrush/>
- <http://blog.urbaninsight.com/2012/04/12/ten-reasons-you-should-be-using-css-preprocessor>
- <http://responsejs.com/>
- <http://code.google.com/p/jqcanvas/>
- <http://benbuckman.net/tech/11/11/migrating-static-960gs-grid-responsive-semantic-grid-lesscss>
- <http://fadeyev.net/2010/06/19/lessjs-will-obsolete-css/>
- <http://smacss.com/>
- <https://github.com/stubbornella/oocss>
- <http://news.ycombinator.com/item?id=4616528>
- <http://www.smashingapps.com/2012/12/26/16-useful-responsive-css-frameworks-and-boilerplates.html>
- <http://lifestyle.emmet.io/>

also related:

<http://modernizr.com/>

Front-end frameworks on Ohloh.net

Zurb Foundation

- <http://www.zurb.com/article/1152/coding-smarter-the-why-of-foundation-4>

Twitter Bootstrap

- <http://twitter.github.com/bootstrap/>
- <http://expo.getbootstrap.com/>
- <http://builtwithbootstrap.com/>
- <http://alistapart.com/article/building-twitter-bootstrap>
- <http://lightglitch.github.com/bootstrap-xtra/>
- <http://bootstraphero.com/the-big-badass-list-of-twitter-bootstrap-resources>
- <http://bootsnipp.com/>
- <http://addyosmani.github.com/jquery-ui-bootstrap/>
 - <https://github.com/lphuberdeau/jquery-ui-less>
 - <http://dev.twitter.com/blog/say-hello-to-bootstrap-2>
- <http://meta.ohloh.net/2012/05/ohloh-is-getting-bootstrapped/>
- <http://savvypanada.com/blog/what-is-twitter-bootstrap-and-how-will-it-change-joomla.html>
- <http://webdesign.tutsplus.com/tutorials/complete-websites/twitter-bootstrap-101-introduction/>
- <http://320press.com/wpbs/>
- <http://news.bootswatch.com/post/22193315172/bootswatch-api>
- <https://www.ohloh.net/p/yii-bootstrap>
- <http://bootsnipp.com/resources>
- <http://designshack.net/articles/css/20-awesome-resources-for-twitter-bootstrap-lovers/>
- <https://github.com/popular/starred>
- <http://lovebootstrap.com/>
- <http://coding.smashingmagazine.com/2013/03/12/customizing-bootstrap/>
- <https://github.com/thomas-mcdonald/bootstrap-sass>
- <http://plozi.github.com/jQDrawBootstrapGrid/>
- <http://wordsbyf.at/2012/03/08/why-less/>

Bootstrap themes repositories

- <https://wrapbootstrap.com/>
- <http://www.templatemonster.com/pack/bootstrap-website-templates/>
- <http://themeforest.net/search?utf8=%E2%9C%93&term=bootstrap>
- <http://bootswatch.com/>
- <http://bootstrappage.com/>

Bootstrap FOSS themes

- <http://usman.it/free-responsive-admin-template/>

Bootstrap generators theme and wireframe

- <http://pikock.github.com/bootstrap-magic/index.html>
- <http://paintstrap.com/>
- <http://stylebootstrap.info/>
- <http://www.boottheme.com/>

- <http://www.lavishbootstrap.com/>
- <http://bootswatchr.com/>
- <http://bootstrapdesigner.com/>
- <http://www.divshot.com/>
- <http://jetstrap.com/>
- <http://invokator.com/>

Videos about Bootstrap

Version 3

- <http://thenextweb.com/dd/2013/03/10/heres-an-early-look-at-bootstrap-3-rewritten-to-be-mobile-friendly-from-the-start/>
- <http://www.h-online.com/open/news/item/Mobile-first-Bootstrap-3-is-almost-ready-1866134.html>

LessPHP

<http://lesscss.org/>

LESS is included in Bootstrap

HTML Boilerplate

- <http://html5boilerplate.com/>
- <https://github.com/h5bp/mobile-boilerplate>

Gumby

<http://gumbyframework.com/>

Skeleton

- <http://www.getskeleton.com/>

960 Grid System

<http://960.gs/>

Unsemantic

<http://unsemantic.com/>

alias

-
- [Responsive web design](#)
 - [RWD](#)

- [CSS Framework](#)