Why Wiki Syntax Is Important

Please also see the dedicated micro-site on this topic: http://wysiwygproblems.com/

This page discusses the importance of Wiki markup (or syntax) for Tiki.
Wiki markup has actually two sides :

1. It can be used as a user data entry method for formatting the page (as opposition to a WYSIWYG editor where you click on buttons)
2. It is also often used as the storage format of the page when the HTML page is not cached in the database.

Related: WYSIWYG vs Wiki

Caution : This section previously contained essentially the opinion of one specific user, while the subject has spurred many debates already. In addition, it also mixed different levels of discussion, sometimes opposing Wikimarkup with WYSIWYG editors, sometimes with HTML, sometimes with Flash. my edits aim at providing a more balanced content. Feel free to edit further, in the wiki way ...

Advantages of Wiki Markup

- **Rock-solid** and **cross-browser** - unlike WYSIWYG editors, it doesn't need any additional technology, like JavaScript or Flash and it works equally well on all kinds of web browsers, making use of any additional features they might offer (like checking spelling or using an external editor, for example).
  - This assumes the parser (which interprets wiki syntax) doesn't change, which is something that can happen between major versions of Tiki.
- **Readable** - with wiki markup, the text is still readable by less technical users (more than HTML), you can just ignore all the funny characters and read the text alone. On the other hand, HTML does not render newlines, while some Wikis do. This impairs the readability.
- **More Accessible than Flash** - for blind people, for software, for searching and indexing.
- **Secure** - because it's just text, you can't really put any malicious code in there, like you can in Flash and/or JavaScript.
- **Better for styling** - because there is no style information in the text itself, you can easily associate a style via CSS.
- **Extensible** - Syntax can be extended to do more things (access external services, etc).
- It can even be used like a **mini-programming language**
  - You can have variables such as TWikiVariables and basic logic (if/then/else). With some wiki engines, such as TWiki and XWiki_old, you can build quite complex applications from Wiki Syntax. For Tiki, here is an example of basic if/then/else:

Example of PluginGroup

```
{GROUP(groups=>Registered)}
some text
{ELSE}
other text
{GROUP}
```

    - Wiki syntax can generate different things per context (ex.: web vs mobile)
    - Have syntax like Print / no print
- **Future proof** - you can determine a wiki syntax and change the HTML output centrally. For example, if you have a syntax for icons or smilies. Can produce evolving HTML over time. As standards evolve, wiki parser can evolve from <b> to style='strong' -> https://dev.tiki.org/item5712
- **Easy/Difficult to learn** - The Wiki syntax is, in theory, easier to learn than HTML: for example, a star as the first character of the line means a bullet. However, the multiplicity of wiki markup dialects and its density makes it unattractive for many users. In general, it is also very poorly documented, and people tend to learn it from examples/others (how they used the syntax).
- **Fast typing** - (after initial learning period). Due to its density, wiki markup is much faster to type than alternatives requiring the selection of text and the use the mouse for formatting. Key shortcuts in WYSIWYG editors are also hard to learn and do not provide good feedback.
- **Fast page load** - Wiki markup is just a text box, thus page load is much faster than JavaScript WYSIWYG editors.
- **Fault tolerant** - Most parsers tolerate small faults in input markup. Rendering won't be perfect then, but also not fail completely usually. If everything else fails, you can just look at the raw source and see what the author meant.
- **Easy to split** - good for diff display (try that with HTML)
- **Easy linking**
- **Mobile** - it suits mobile phone keypads really well
- **User-controlled** *the user is in full control of what gets saved in the wiki* in the edit mode nothing is hidden, nothing is automatically added or removed. There is no converting or other magic working behind the user's back. There might be exceptions like automatic dates or signatures, but these are rare.

Disadvantages of Wiki markup

- One big disadvantage of Wiki markup, on the other hand, is that each wiki has a different, often incompatible markup. This means that content is not portable. While there have been efforts to standardize wiki markup, WikiCreole is not adopted widely. Worse, WikiCreole is not seen as important by many Wiki designers, and migration to WikiCreole would break most of the pages on other wikis because of the conflicts.
- The **Density** of Wiki markup doesn't make it look appealing to most people. This is the flip side of the "readability" advantage above. it is far from comfortable.
- Many Wikis get poor acceptance by users because of a lack of the presence of a WYSIWYG editor. However, this is worth a separate debate.
- Wiki markup is not adapted for page design or manipulating large data sets (like really wide tables)
- The interpretation at rendering time may differ from the intended result, e.g. with unwanted links. Hence editing can be a trial and error process. In Tiki, Live Preview was added to offset this issue.

```
Ex.: In Tiki [This] will create an unwanted hyperlink to the page "This" if it exists
```

References

- http://davidgerard.co.uk/notes/2011/01/04/what-you-see-is-for-the-win/

- http://www.pmwiki.org/wiki/PmWiki/WYSIWYG

- Wikibook DTD

- "Wiki markup has no future" is an article from Column Two that advocates against Wiki markup. Unfortunately, due to some internal "remodeling", the comments are not available anymore.

- MediaWiki has an interesting page on limitations of WYSIWYG editors for wikis, stating :

MediaWiki site wrote:

> *In 2009, there is no available 'ready-to-go' package for incorporating full WYSIWYG into the MediaWiki software.*
> *The problem is that any WYSIWYG editor would have to know wikitext grammar, and no full grammar for wikitext exists - the "parser" doesn't parse, it's a twisty series of regular expressions. So present WYSIWYG editors either have to (a) reverse-engineer as much of a grammar as they can, or (b) forget wikitext and just write HTML.*

"Seeing as I've spent the last few months writing a product that uses Wiki markup as its basis, I thought I might come to the markup's defense."

- http://fishbowl.pastiche.org/2004/03/19/in_defense_of_wiki_markup

"The author is distracted from the proper business of composing text, in favor of making typographical choices in relation to which she may have no expertise (``fiddling with fonts and margins'' when she should be concentrating on content)."
http://ricardo.ecn.wfu.edu/~cottrell/wp.html

"The simplification process of text markup has therefore reached a point where you can't make it much simpler; and as it turns out, it seems to be finally simple enough for end users. A Canadian study among 4th-grade students (age 8 to 9) showed that indeed wiki markup can even be used by children after a 15-minute introduction [Désilets 06]. The study shows that the wiki markup itself is by far not the biggest usability issue with wikis. The real issues that were revealed by this study are the understanding of the hypertext nature of the Internet and related to that, link creation and management. The success of wikis and Wikipedia in particular therefore indicates that we seem to have the missing simple markup finally to get the masses involved."
http://www.i3g.hs-heilbronn.de/attach/Ver%C3%B6ffentlichungen/What+you+see+is+Wiki.pdf

RFCWiki
http://www.wikicreole.org/wiki/TikiWikiCMSGroupware

For all people that say WYSIWYG is important, I have a challenge for you. I am asking you to put your time where your mouth is :-)

http://www.codinghorror.com/blog/2006/05/invisible-formatting-tags-are-evil.html
http://www.codinghorror.com/blog/2012/03/what-you-cant-see-you-cant-get.html
http://girtby.net/archives/2005/06/13/this-is-what-you-see-this-is-what-you-get/

Other

Why version history is crucial

- People learn from others and can experiment.
- If something goes wrong, it's easy to roll back to the previous page version.

When/why a wiki should be the central part of your information system:

- It's a superior way of handling certain types of information.
- It's a more flexible way to deal with "content chaos".
- See the success of Wikipedia as an example.
- It's the best method in many Use Cases.

Page aliases

- Why Wiki Syntax?
- Wiki Syntax Advantages
- Wiki Syntax Benefits
- Wiki Markup is important