

Warum die Wiki Syntax wichtig ist

Bitte beachten Sie auch die spezielle Microsite zu diesem Thema: <http://wysiwygproblems.com/>

Auf dieser Seite wird die Bedeutung des Wiki-Markups (oder der Syntax) für Tiki diskutiert. Wiki-Markup hat eigentlich zwei Seiten:

1. Es kann als Eingabemethode für Benutzerdaten zur Formatierung der Seite verwendet werden (im Gegensatz zu einem WYSIWYG-Editor, bei dem Sie auf Schaltflächen klicken).
2. Es wird auch oft als Speicherformat der Seite verwendet, wenn die HTML-Seite nicht in der Datenbank zwischengespeichert wird.

Verwandt: [WYSIWYG vs Wiki](#)

Achtung: Dieser Abschnitt enthielt bisher im Wesentlichen die Meinung eines bestimmten Benutzers, während das Thema bereits viele Debatten ausgelöst hat. Darüber hinaus vermischte es auch verschiedene Ebenen der Diskussion, manchmal gegen Wikimarkup mit WYSIWYG-Editoren, manchmal mit HTML, manchmal mit Flash. Meine Bearbeitungen zielen darauf ab, einen ausgewogeneren Inhalt zu liefern. Du kannst sie gerne weiter bearbeiten, auf dem Wiki-Weg 😊 ...

Vorteile des Wiki Markups

- **Solide wie Felsen** und „**Cross-Browser**“ - im Gegensatz zu WYSIWYG-Editoren benötigt es keine zusätzliche Technologie, wie JavaScript oder Flash, und es funktioniert genauso gut in allen Arten von Webbrowsern, indem es alle zusätzlichen Funktionen nutzt, die sie anbieten (wie z.B. Rechtschreibprüfung oder Verwendung eines externen Editors).
 - Dies setzt voraus, dass sich der Parser (der die Wiki-Syntax interpretiert) nicht ändert, was zwischen den wichtigsten Versionen von Tiki passieren kann.
 - **Lesbar** - mit Wiki-Markup ist der Text für weniger technische Benutzer lesbar (mehr als HTML), Sie können einfach alle lustigen Zeichen ignorieren und den Text allein lesen. Andererseits rendert HTML keine Zeilenumbrüche, während einige Wikis dies tun. Dies beeinträchtigt die Lesbarkeit.
 - **Besser zugänglich als Flash** - für Blinde, für Software, für die Suche und Indizierung.
 - **Sicher** - da es sich nur um Text handelt, können Sie keinen böartigen Code einfügen, wie in Flash und/oder JavaScript.
 - **Besser für Styling** - da es keine Style-Informationen im Text selbst gibt, können Sie einen Style einfach per CSS zuordnen.
 - **Erweiterbar** - Die Syntax kann erweitert werden, um mehr Dinge zu tun (Zugriff auf externe Dienste, etc.).
 - Es kann sogar wie eine **mini-programmierende Sprache** verwendet werden.
 - Sie können Variablen wie [TWikiVariables](#) und grundlegende Logik (if/then/else) haben. Mit einigen Wiki-Engines, wie Tiki-vs-TWiki und XWiki_ oldkönnen Sie recht komplexe Anwendungen aus der Wiki-Syntax erstellen. Für Tiki folgt ein Beispiel für ein exemplarisches if/then/else:

♣ **Beispiel einer PluginGroup**
- ```
{GROUP(groups=>Registered)} etwas Text {ELSE} anderer Text {GROUP}
```
- Die Wiki-Syntax kann verschiedene Dinge pro Kontext generieren (z.B.: Web vs. Mobile).
  - Es bietet Syntax wie Drucken / kein Drucken
- **Zukunftssucher** - Sie können eine Wiki-Syntax festlegen und die HTML-Ausgabe zentral ändern. Zum Beispiel, wenn Sie eine Syntax für Symbole oder Smilies haben. Kann im Laufe der Zeit sich entwickelndes HTML erzeugen. Wenn sich Standards entwickeln, kann sich der Wiki-Parser von `<b>` zu `style='strong'` -> <https://dev.tiki.org/item5712> entwickeln.
- **Einfach/schwierig zu lernen** - Die Wiki-Syntax ist theoretisch einfacher zu erlernen als HTML: z.B.

ein Stern als erstes Zeichen der Zeile bedeutet einen Punkt. Die Vielzahl der Wiki-Markup-Dialekte und ihre Dichte machen sie jedoch für viele Benutzer unattraktiv. Im Allgemeinen ist es auch sehr schlecht dokumentiert, und die Leute neigen dazu, es von Beispielen/anderen zu lernen (wie sie die Syntax verwendet haben).

- **Schnelles Tippen** - (nach der ersten Einarbeitungszeit). Aufgrund seiner Dichte ist das Wiki-Markup viel schneller zu tippen als Alternativen, die die Auswahl von Text und die Verwendung der Maus zur Formatierung erfordern. Tastenkombinationen in WYSIWYG-Editoren sind ebenfalls schwer zu erlernen und bieten kein gutes Feedback.
- **Schneller Seitenaufbau** - Wiki-Markup ist nur ein Textfeld, daher ist das Laden der Seite viel schneller als bei JavaScript WYSIWYG-Editoren.
- **Fehler tolerant** - Die meisten Parser tolerieren kleine Fehler im Input Markup. Das Rendern wird dann nicht perfekt sein, aber auch nicht ganz scheitern. Wenn alles andere fehlschlägt, können Sie sich einfach die Rohdatenquelle ansehen und sehen, was der Autor gemeint hat.
- **Leicht zu splitten** - gut für die Diff-Anzeige (versuchen Sie das mal mit HTML)
- Einfache Verknüpfung\_\_
- **Mobile** - es passt sehr gut zu Handy-Tastaturen.
- **Benutzer kontrolliert** - Der Benutzer hat die volle Kontrolle darüber, was im Wiki gespeichert wird. Im Bearbeitungsmodus wird nichts versteckt, nichts wird automatisch hinzugefügt oder entfernt. Es gibt keine Konvertierung oder andere Magie, die hinter dem Rücken des Benutzers funktioniert. Es kann Ausnahmen wie automatische Daten oder Unterschriften geben, aber diese sind selten.

#### Nachteile des Wiki-Markups

- Ein großer Nachteil des Wiki-Markups ist dagegen, dass jedes Wiki ein anderes, oft inkompatibles Markup hat. Das bedeutet, dass Inhalte nicht portabel sind. Obwohl es Bemühungen zur Standardisierung des Wiki-Markups gegeben hat, wird [WikiCreole](#) nicht weit verbreitet. Schlimmer noch, WikiCreole wird von vielen Wiki-Designern nicht als wichtig angesehen, und die Migration zu WikiCreole würde die meisten Seiten anderer Wikis wegen der Konflikte abbrechen.
- Die **Dichte** des Wiki-Markups lässt es nicht für die meisten Leute ansprechend aussehen. Dies ist die Kehrseite des obigen Vorteils der "Lesbarkeit". Es ist alles andere als bequem.
- Viele Wikis erhalten eine schlechte Akzeptanz bei den Nutzern, da es keinen WYSIWYG-Editor gibt. Dies ist jedoch eine separate Debatte wert.
- Wiki-Markup ist nicht geeignet für das Seitendesign oder die Manipulation großer Datensätze (wie z.B. wirklich große Tabellen).
- Die Interpretation zum Zeitpunkt des Renderings kann vom beabsichtigten Ergebnis abweichen, z.B. bei unerwünschten Links. Daher kann die Bearbeitung ein Trial-and-Error-Prozess sein. In Tiki wurde [LivePreview](#) hinzugefügt, um dieses Problem zu beheben.

- Beispiel: In Tiki [This] wird ein unerwünschter Hyperlink zur Seite „This“ erstellt, wenn er existiert

---

#### Referenzen

- <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 hat keine Zukunft](#)" ist ein Artikel aus [Column Two](#), der sich gegen Wiki Markup ausspricht. Leider sind die Kommentare aufgrund einiger interner "Umbauten" nicht mehr verfügbar.
- MediaWiki hat eine interessante Seite über die Einschränkungen von [WYSIWYG-Editoren für Wikis](#) mit folgenden Angaben:

MediaWiki site wrote:

*Im Jahr 2009 gibt es kein fertiges Paket zur Integration des vollständigen WYSIWYG in die MediaWiki-Software.*

*Das Problem ist, dass jeder WYSIWYG-Editor die Wikitext-Grammatik kennen müsste und keine vollständige Grammatik für Wikitext existiert - der "Parser" analysiert nicht, es ist eine kurvenreiche Reihe von regulären Ausdrücken. So müssen WYSIWYG-Editoren entweder (a) so viel Grammatik wie möglich zurückentwickeln, oder (b) Wikitext vergessen und einfach HTML schreiben.*

"Da ich die letzten Monate damit verbracht habe, ein Produkt zu schreiben, das Wiki-Markup als Grundlage verwendet, dachte ich, ich könnte zur Verteidigung des Markups kommen."

- [http://fishbowl.pastiche.org/2004/03/19/in\\_defense\\_of\\_wiki\\_markup](http://fishbowl.pastiche.org/2004/03/19/in_defense_of_wiki_markup)

"Die Autorin wird von der eigentlichen Aufgabe der Texterstellung abgelenkt, um typografische Entscheidungen zu treffen, für die sie vielleicht keine Expertise hat ( `Fummeln an Schriften und Rändern", wenn sie sich auf den Inhalt konzentrieren sollte)."

<http://ricardo.ecn.wfu.edu/~cottrell/wp.html>

"Der Vereinfachungsprozess der Textmarkierung hat daher einen Punkt erreicht, an dem man es nicht viel einfacher machen kann; und wie sich herausstellt, scheint es endlich einfach genug für den Endbenutzer zu sein. Eine kanadische Studie unter Schülern der 4. Klasse (Altersgruppe 8 bis 9 Jahre) zeigte, dass Wiki-Markup tatsächlich auch von Kindern nach einer 15-minütigen Einführung verwendet werden kann [Désilets 06]. Die Studie zeigt, dass das Wiki-Markup selbst bei weitem nicht das größte Usability-Problem bei Wikis ist. Die wirklichen Probleme, die durch diese Studie aufgedeckt wurden, sind das Verständnis der Hypertext-Natur des Internets und die damit verbundene Linkerstellung und -verwaltung. Der Erfolg von Wikis und insbesondere von Wikipedia deutet daher darauf hin, dass wir das fehlende einfache Markup haben, um endlich die Massen einzubeziehen."

<http://www.i3g.hs-heilbronn.de/attach/Ver%C3%B6ffentlichungen/What+you+see+is+Wiki.pdf>

RFCWiki

<http://www.wikicreole.org/wiki/TikiWikiCMSGroupware>

Für alle, die sagen, dass WYSIWYG wichtig ist, habe ich eine Herausforderung für dich. Ich bitte dich, deine Zeit dort zu verbringen, wo dein Mund ist:-)

<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/>

Sonstiges

Warum die Versionshistorie entscheidend ist

- Menschen lernen von anderen und können experimentieren.
- Wenn etwas schief geht, ist es einfach, zur vorherigen Seitenversion zurückzukehren.

Wann/Warum sollte ein Wiki der zentrale Bestandteil Ihres Informationssystems sein?

- Es ist eine überlegene Art, mit bestimmten Arten von Informationen umzugehen.

- Es ist eine flexiblere Art, mit dem "Inhaltschaos" umzugehen.
- Sehen Sie sich den Erfolg von Wikipedia als Beispiel an.
- Es ist die beste Methode von vielen [Anwendungsfällen](#).

#### Seiten-Aliase

- [Wiki Syntax Vorteile](#)
- [Wiki Syntaxvorteile](#)
- [Wiki Markup ist wichtig](#)

(Übersetzt mit dem neuronalen Netzwerk von DeepL: <https://www.deepl.com>)