

## **Euro Tcl 2006**

### **nwikit - an extended wiki implementation for Tcl**

*Dr. Detlef Groth, MPIMG Berlin, Germany, dgroth(at)molgen.mpg.de*

Abstract: The current Tcl-Wiki implementation has some serious drawbacks thereof limiting its usage for documentation and building of webapplications. Some of the features which are currently missing are: no support for delivery of static content by the inbuild webserver, no support for tables, nested lists and structured markup like headers. Here we describe a small, extended Wikit implementation which overcomes those limitations via extending the inbuild webserver and utilizing a rewritten scanner for the Wiki markup.

Wiki implementations are a hot topic in the internet. Wikis can be used as a content management system for websites, as blogging facilities and as discussion platforms. The fast and simple add/edit/fix/update cycle is responsible for the success of Wiki-pages like wikipedia.org and others. The mostly simple markup language allows also non experienced users to enter content. Errors can be easily fixed and the Wiki is updated immediately. In the programming community there are existing a huge range of Wiki implementations. As a google search for and Wiki suggests, most Wiki implementations are done in php. Also wikipedia is implemented in php via the MediaWiki. Tcl currently has to my knowledge only one Wiki implementation which is used by more people than the author itself, the Wikit on. Of course the famous Tcl'ers Wiki is implemented in Wikit.

Currently however Wikit is missing some important features which could improve its usability and its usage by the community.

- no static content delivery from the inbuild webserver
- no support for tables and tabular data
- no support for structured markup like h1,h2 table of contents
- limited navigation facilities
- no default stylesheet for improved layout
- default design does not encourage usage

The current problems with those missing Wikit features have been handled in two ways: At first I directly changed the Wikit parser. The old underlying parser of the Wikit package is not easily to modify without breaking other parts of the application. I wrote a new scanner/parser via utilising the ifickle scanner generator. Of course there should be always a compromise between simplicity/usability vs. features. Furthermore the code should be easier to maintain and to extend in the future. The new scanner can also be used as a commandline application to convert wiki pages into html pages. Second, some none essential parts have been implemented in javascript thereof giving the user more choice to allow or disable certain features by just switching between various, stylesheets and javascript files.

Discussion: We were targeting some of the old Wikit limitations with an extended webserver implementation, able to serve as well static content, a new scanner implementation using the ifickle scanner generator which is easier to maintain and to extend. The current scanner supports in addition to the markup supported by wikit: tables, nested lists, named links, more structured markup like headers, tabboxes or a dynamic table of contents. Some features like tabboxes and the table of contents are only available with modern javascript enabled browsers. Our extension of the markup language is mostly compatible with the Tk-based Wikit-Editor, thereof the pages can be rendered as well in the Tk-Help System. It might be possible to change the Tk-scanner as well to an ifickle based scanner. Furthermore a print mode was added to properly print the content of the wiki page.

## References

- MediaWiki <http://www.mediawiki.org/wiki/MediaWiki>
- Wikit <http://www.equi4.com/wikit.html>
- nwikit <http://goblet.molgen.mpg.de/nwikit>
- fickle <http://tcl.jtang.org/fickle/>
- ifickle <http://goblet.molgen.mpg.de/ifickle>

## Markup Rules

### References

- links to the same wiki, but another page with square brackets like this [PAGE]
- URLs are automatically recognized and underlined like here <http://mini.net/tcl>
- URLs in square brackets will be shown as tiny reference as here: [ <http://mini.net/tcl>] [1]
- **new:** URLs which should become a named linked has to be written as [URL|name] like [ <http://mini.net/tcl>|Tclers Wiki] becomes Tclers Wiki
- URLs from images are automatically recognized, if put into square brackets they are linked, if put into square brackets they are inlined

### Headers

- **new:** text surrounded by triples of single quotes makes it a header type h2 if it is the only text one the line

### Highlights

- text surrounded by triples of single quotes becomes **bold** text
- text surrounded by doubles of single quotes becomes *italic* text

### Structure

- lines of text are joined, empty lines divide paragraphs
- lines beginning with three spaces, a "\*\*\*", and another space are shown as bulleted items. The entire item must be on one line.
- **new:** nested lists can be done by using three spaces and two "\*\*\*" like

here:

```
* list
** sublist item
** sublist item
```

- Lines starting with three spaces, a "1.", and another space are shown as a numbered list.
- Lines starting with three spaces, item tag name, ":", three spaces, and then the item tag body (entered as 1 logical line) are shown as tagged lists.
- add horizontal separator is done for four dashes on one line like ----
- any other lines starting with a white space are shown as they are, now wrapping, no references, no formatting

## Tables

**New:** Tables start with one space and pipes as delimiters for each column:

```
| col1 | col2 | col3 |
| val1 | val2 | val3 |
```

This code will will be rendered as:

<b>col1</b>	<b>col2</b>	<b>col3</b>
val1	val2	val3

## Tabboxes

**New:** Tabs can be generated as well via :

```
--- Sectionname ---
```