WikiFormatting

Error: Macro TracGuideToc(None) failed

```
'NoneType' object has no attribute 'find'
```

Wiki markup is a core feature in Trac, tightly integrating all the other parts of Trac into a flexible and powerful whole.

Trac has a built in small and powerful wiki rendering engine. This wiki engine implements an ever growing subset of the commands from other popular Wikis, especially ?MoinMoin.

This page demonstrates the formatting syntax available anywhere WikiFormatting is allowed.

Font Styles

The Trac wiki supports the following font styles:

```
 * '''bold''', '''!''' can be bold too''', and '''! '''
 * ''italic''
 * '''''bold italic'''''
 * __underline__
 * {{{monospace}}} or `monospace`
 * ~~strike-through~~
 * ^superscript^
 * ,,subscript,,
```

Display:

- **bold**, **''' can be bold too**, and **!**
- *italic*
- ***bold italic***
- underline
- `monospace` or `monospace`
- ~~strike-through~~
- superscript
- subscript

Notes:

- `{{{...}}}` and `` `...` `` commands not only select a monospace font, but also treat their content as verbatim text, meaning that no further wiki processing is done on this text.
- `!` tells wiki parser to not take the following characters as wiki format, so pay attention to put a space after !, e.g. when ending bold.

Headings

You can create heading by starting a line with one up to five *equal* characters ("=") followed by a single space and the headline text. The line should end with a space followed by the same number of = characters. The heading might optionally be followed by an explicit id. If not, an implicit but nevertheless readable id will be generated.

Example:

```
= Heading =
```

```
== Subheading ==
=== About ''this'' ===
=== Explicit id === #using-explicit-id-in-heading
```

Display:

Heading

Subheading

About *this*

Explicit id

Paragraphs

A new text paragraph is created whenever two blocks of text are separated by one or more empty lines.

A forced line break can also be inserted, using:

```
Line 1[[BR]]Line 2
```

Display:

Line 1
Line 2

Lists

The wiki supports both ordered/numbered and unordered lists.

Example:

```
 * Item 1
   * Item 1.1
      * Item 1.1.1
      * Item 1.1.2
      * Item 1.1.3
   * Item 1.2
 * Item 2

 1. Item 1
   a. Item 1.a
   a. Item 1.b
      i. Item 1.b.i
      i. Item 1.b.ii
 1. Item 2
And numbered lists can also be given an explicit number:
 3. Item 3
```

Display:

- Item 1
  - Item 1.1

            - Item 1.1.1
            - Item 1.1.2
            - Item 1.1.3
        - Item 1.2
- Item 2

1. Item 1
    1. Item 1.a
    2. Item 1.b
        1. Item 1.b.i
        2. Item 1.b.ii
2. Item 2

And numbered lists can also be given an explicit number:

3. Item 3

Note that there must be one or more spaces preceding the list item markers, otherwise the list will be treated as a normal paragraph.

Definition Lists

The wiki also supports definition lists.

Example:

```
llama::
  some kind of mammal, with hair
ppython::
  some kind of reptile, without hair
  (can you spot the typo?)
```

Display:

llama
: some kind of mammal, with hair

ppython
: some kind of reptile, without hair (can you spot the typo?)

Note that you need a space in front of the defined term.

Preformatted Text

Block containing preformatted text are suitable for source code snippets, notes and examples. Use three *curly braces* wrapped around the text to define a block quote. The curly braces need to be on a separate line. Example:

```
{{{
 def HelloWorld():
     print "Hello World"
}}}
```

Display:

```
def HelloWorld():
    print "Hello World"
```

Blockquotes

In order to mark a paragraph as blockquote, indent that paragraph with two spaces.

Example:

```
  This text is a quote from someone else.
```

Display:

> This text is a quote from someone else.

Discussion Citations

To delineate a citation in an ongoing discussion thread, such as the ticket comment area, e-mail-like citation marks (">", ">>", etc.) may be used.

Example:

```
>> Someone's original text
> Someone else's reply text
My reply text
```

Display:

> > > Someone's original text
>
> > Someone else's reply text
>
> My reply text

Note: Some WikiFormatting elements, such as lists and preformatted text, are lost in the citation area. Some reformatting may be necessary to create a clear citation.

Tables

Simple tables can be created like this:

```
||Cell 1||Cell 2||Cell 3||
||Cell 4||Cell 5||Cell 6||
```

Display:

Cell 1	Cell 2	Cell 3
Cell 4	Cell 5	Cell 6

Note that more complex tables can be created using reStructuredText.

Links

Hyperlinks are automatically created for WikiPageNames and URLs. WikiPageLinks can be disabled by prepending an exclamation mark "!" character, such as `!WikiPageLink`.

Example:

```
TitleIndex, http://www.edgewall.com/, !NotAlink
```

Display:

> TitleIndex, ?http://www.edgewall.com/, NotAlink

Links can be given a more descriptive title by writing the link followed by a space and a title and all this inside square brackets. If the descriptive title is omitted, then the explicit prefix is discarded, unless the link is an external link. This can be useful for wiki pages not adhering to the WikiPageNames convention.

Example:

```
* [http://www.edgewall.com/ Edgewall Software]
* [wiki:TitleIndex Title Index]
* [wiki:ISO9000]
```

Display:

- ?Edgewall Software
- Title Index
- ISO9000?

Trac Links

Wiki pages can link directly to other parts of the Trac system. Pages can refer to tickets, reports, changesets, milestones, source files and other Wiki pages using the following notations:

```
* Tickets: #1 or ticket:1
* Reports: {1} or report:1
* Changesets: r1, [1] or changeset:1
* ...
```

Display:

- Tickets: #1 or ticket:1
- Reports: {1} or report:1
- Changesets: r1, [1] or changeset:1
- ...

There are many more flavors of Trac links, see TracLinks for more in-depth information.

Escaping Links and WikiPageNames

You may avoid making hyperlinks out of TracLinks by preceding an expression with a single "!" (exclamation mark).

Example:

```
!NoHyperLink
!#42 is not a link
```

Display:

NoHyperLink #42 is not a link

Images

Urls ending with `.png`, `.gif` or `.jpg` are no longer automatically interpreted as image links, and converted to `<img>` tags.

You now have to use the [[Image]] macro. The simplest way to include an image is to upload it as attachment to the current page, and put the filename in a macro call like `[[Image(picture.gif)]]`.

In addition to the current page, it is possible to refer to other resources:

- `[[Image(wiki:WikiFormatting:picture.gif)]]` (referring to attachment on another page)
- `[[Image(ticket:1:picture.gif)]]` (file attached to a ticket)
- `[[Image(htdocs:picture.gif)]]` (referring to a file inside project htdocs)
- `[[Image(source:/trunk/trac/htdocs/trac_logo_mini.png)]]` (a file in repository)

Example display:

See WikiMacros for further documentation on the `[[Image()]]` macro.

Macros

Macros are *custom functions* to insert dynamic content in a page.

Example:

```
[[RecentChanges(Trac,3)]]
```

Display:

Feb 23, 2013

- TracRepositoryAdmin (diff)
- TracWorkflow (diff)
- TracInterfaceCustomization (diff)

See WikiMacros for more information, and a list of installed macros.

Processors

Trac supports alternative markup formats using WikiProcessors. For example, processors are used to write pages in reStructuredText or HTML.

Example 1:

```
{{{
#!html
<h1 style="text-align: right; color: blue">HTML Test</h1>
}}}
```

Display:

HTML Test

Example:

```
{{{
#!python
class Test:

    def __init__(self):
        print "Hello World"
if __name__ == '__main__':
   Test()
}}}
```

Display:

```
class Test:
    def __init__(self):
        print "Hello World"
if __name__ == '__main__':
   Test()
```

Perl:

```
my ($test) = 0;
if ($test > 0) {
    print "hello";
}
```

See WikiProcessors for more information.

Comments

Comments can be added to the plain text. These will not be rendered and will not display in any other format than plain text.

```
{{{
#!comment
Your comment here
}}}
```

Miscellaneous

Four or more dashes will be replaced by a horizontal line (<HR>)

Example:

```
 ----
```

Display:

See also: TracLinks, TracGuide, WikiHtml, WikiMacros, WikiProcessors, TracSyntaxColoring.