

Wikiprint Book

Title: Wiki プロセッサ

Subject: SilverFrost - WikiProcessors

Version: 3

Date: 12/20/25 13:32:39

## SilverFrost 目次

Wiki プロセッサ	3
プロセッサを使用する	3
これは raw HTML です	3
見出し	3
プロセッサ一覧	4
ソースコードの強調表示対応	4
上級トピック: プロセッサマクロを開発する	5

## Wiki プロセッサ

プロセッサは [Wiki マクロ](#) の一種です。[Trac Wiki エンジン](#) 以外のマークアップフォーマットを使用するために設計されています。プロセッサはユーザが編集したテキストを処理するマクロ機能 と考えることができます。

Trac 内の全ての Wiki テキストにおいて、Wiki エンジンにプロセッサを使わせることによって [Restructured Text](#), [raw HTML](#) および [textile](#) を使用することができます。

### プロセッサを使用する

テキストブロックでプロセッサを使うには、Wiki のコードブロックを使用し、プロセッサの名前を (Unix 系 OS のスクリプトのように) shebang 行 (#!) に続いて指定します。

例1 (Wiki テキストに raw HTML を挿入する):

```
{{{
#!html
<h1 style="color: orange">■■■■ raw HTML ■■■</h1>
}}}
```

結果、このように表示されます:

これは raw HTML です

Note that since 0.11, such blocks of HTML have to be self-contained, i.e. you can't start an HTML element in one block and close it later in a second block. Use div or span processors for achieving similar effect ([see WikiHtml](#)).

例2 (Wiki テキストに ReStructured テキストを挿入する):

```
{{{
#!rst
■■■■
-----
■■ [*]_ ■■■■ **text** ■■■■

.. [*] ■■■■■■■■
}}}
```

結果、このように表示されます:

### 見出し

脚注 [\[\\*\]](#) 付きの text です。

[\[\\*\]](#)

ここは脚注。

例3 (Wiki テキストに C 言語ソースコードブロックを挿入する):

```
{{{
#!c
int main(int argc, char *argv[])
{
    printf("Hello World\n");
    return 0;
}
}}}
```

結果、このように表示されます:

```
int main(int argc, char *argv[])
{
    printf("Hello World\n");
    return 0;
}
```

## プロセッサ一覧

以下のプロセッサが Trac の配布に含まれています:

- `html` -- Wiki ページにカスタム HTML を挿入する。[WikiHtml](#) を参照してください。
- `div` -- Wiki ページの内容を任意の `<div>` 要素でラップする (0.11 以降)。[WikiHtml](#) を参照してください。
- `span` -- Wiki ページの内容を任意の `<span>` 要素でラップする (0.11 以降)。[WikiHtml](#) を参照してください。
- `rst` -- Restructured テキスト を Trac で利用可能にする。[WikiRestructuredText](#) を参照してください。
- `textile` -- [Textile](#) をインストールすれば利用可能です。[Textile リファレンス](#) を参照してください。
- `comment` -- セクション内のテキストを処理しません。(セクションの内容はレンダリングされた Wiki ページには出力されず、プレーンテキストにのみ保持されます。)

## ソースコードの強調表示対応

Trac では、以下のプログラミング言語でのインライン [シンタックスハイライト](#) を行うプロセッサが組み込まれています。

- `c` -- C
- `cpp` -- C++
- `python` -- Python
- `perl` -- Perl
- `ruby` -- Ruby
- `php` -- PHP
- `asp` -- ASP
- `java` -- Java
- `js` -- Javascript
- `sql` -- SQL
- `xml` -- XML
- `sh` -- Bourne/Bash shell

Note: Trac はシンタックスの色付けを、外部のソフトウェアパッケージに依存しています。詳しくは [TracSyntaxColoring](#) をご覧下さい。

プロセッサとして `mime-type`

が指定されると、ソースコードをブラウズするときと同じく、対応されている言語であれば、シンタックスハイライトを行うことができます。例えば、以下のように

```
{{{
#!text/html
<h1>text</h1>
}}}
```

この結果はシンタックスハイライトされた HTML コードになります:

```
<h1>text</h1>
```

対応している全ての `mime-type` でも、これは同じく有効です。

ユーザによるプロセッサマクロの開発やコントリビュート:

- [ProcessorBazaar](#)
- [MacroBazaar](#)
- [Trac Hacks](#) コミュニティサイト

## 上級トピック：プロセッサマクロを開発する

プロセッサの開発は Wiki マクロと違いはありません。事実、これらは同じような機能を持ち、シンタックスだけが異なります。詳しくは [WikiMacros](#) を参照してください。

---

See also: [WikiMacros](#), [WikiHtml](#), [WikiRestructuredText](#), [TracSyntaxColoring](#), [WikiFormatting](#), [TracGuide](#)