

Dokumentationsportal generieren



Historie

Datum	Autor	Änderung
17.09.2008	Simon Wilper	Ersterstellung

Inhaltsverzeichnis

1 -<u>Voraussetzungen</u>	3
1.1 Installation von Perl und Template::Toolkit.....	3
1.1.1 Windows.....	3
1.1.2 Linux.....	5
2 -<u>doc-generate</u>	6
2.1 Format für die Eingabedateien.....	6
2.2 Template-Prozessor starten.....	7
2.3 Aufbau der Vorlagen.....	8
2.3.1 index.tt.....	8

1 - Voraussetzungen

Um das Dokumentationsportal mit Hilfe des Tools "doc-generate.pl" zu generieren, sind folgende Programme erforderlich:

- Perl 5.10.0 mit:
 - Template::Toolkit

1.1 Installation von Perl und Template::Toolkit

1.1.1 Windows

Laden Sie das ActivePerl-MSI-Archiv von [activestate.com](http://www.activestate.com) herunter: <http://www.activestate.com/Products/activeperl/index.mhtml>

Nach der Installation starten Sie unter Programme – Active Perl den "Perl Package Manager":

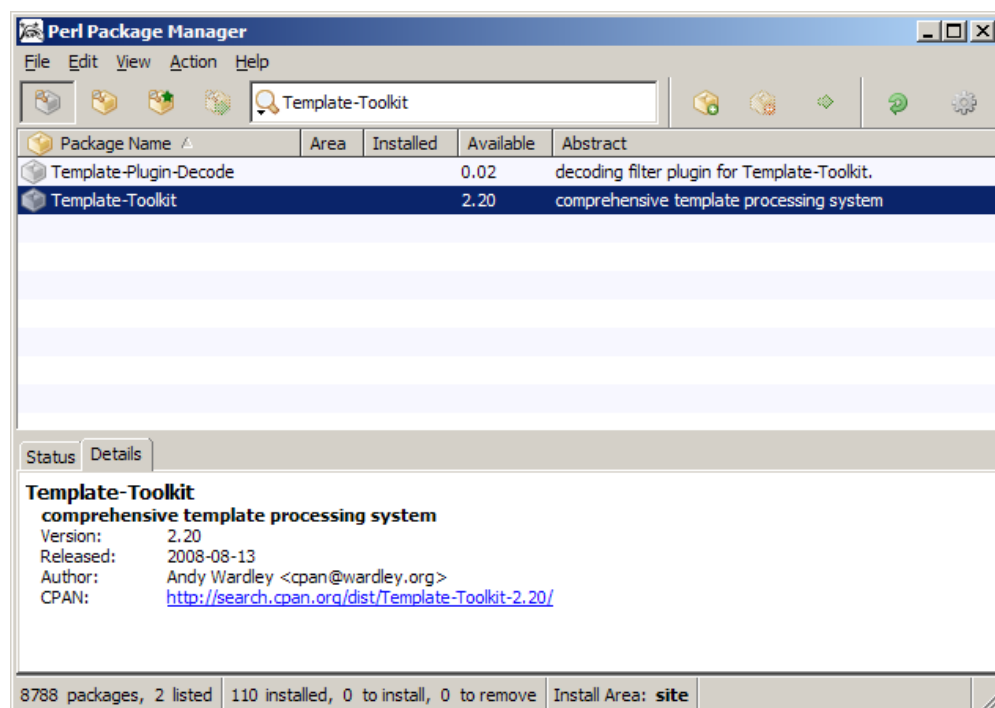


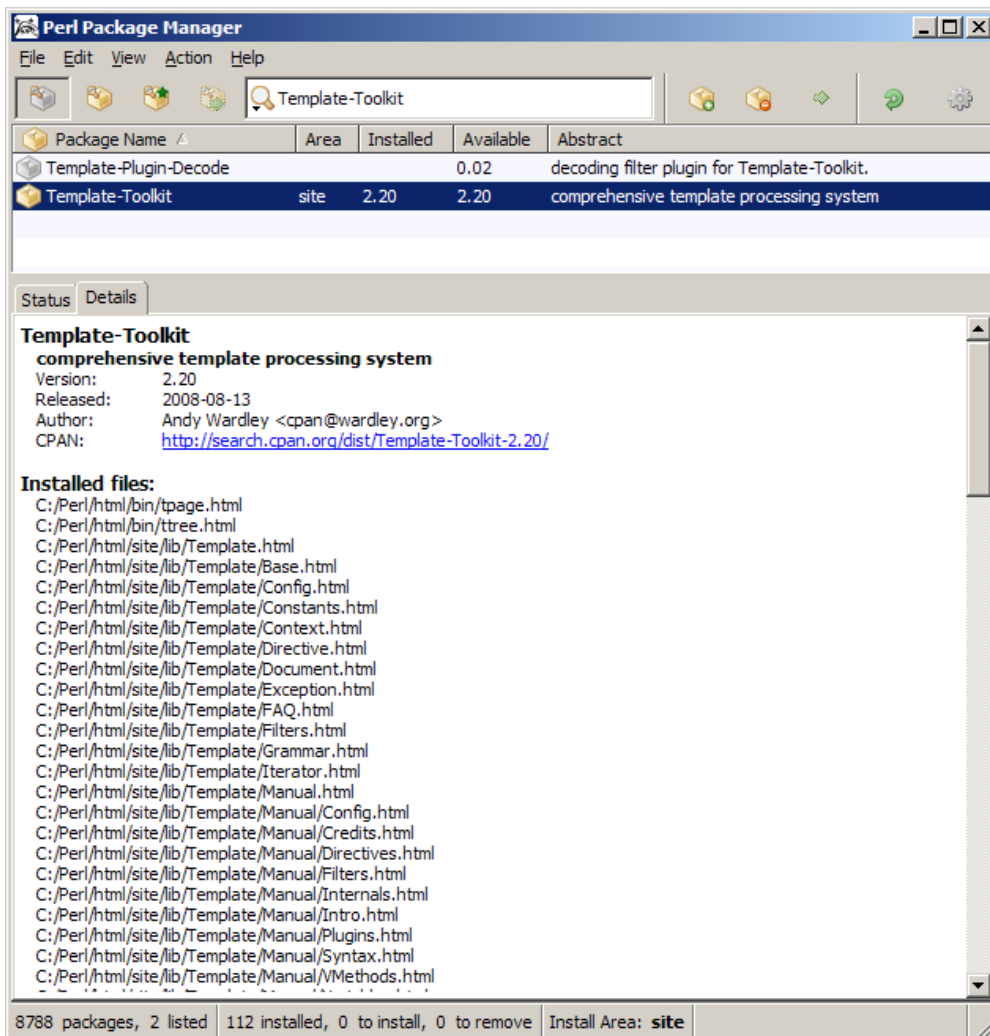
Abbildung 1: Perl Package Manager

Geben Sie in das Suchfeld "Template-Toolkit" ein und wählen Sie in der Toolbar den linken Knopf "View all Packages".

Das Packet "Template-Toolkit" sollte auftauchen.

Durch Klick auf "Mark for Install" und gefolgt von "Run marked actions"

wird das Paket heruntergeladen und installiert:



4

Abbildung 2: Package Manager - Template-Toolkit wurde installiert

Öffnen Sie nun eine Kommandozeile "cmd" und geben Sie ein:

```
1 C:\>perl -MTemplate::Toolkit -e1  
2 C:\>_
```

Sollte keine Ausgabe erscheinen und wieder der Kommando-Prompt, ist das Modul vorhanden und alles in Ordnung.

Bei einer Meldung der Art

```
1 | Can't locate Template.pm in @INC (@INC contains:  
2 | C:/Perl/site/lib C:/Perl/lib ).  
3 |  
4 | BEGIN failed--compilation aborted
```

konnte das Modul nicht gefunden werden. Bitte überprüfen Sie in dem Fall Ihre Perl-Installation

1.1.2 Linux

Perl sollte unter Linux bereits installiert sein.

Je nach Distribution existiert ein Paket, z.B. "perl-template-toolkit", das einfach über den Paketmanager installiert werden kann.

Sollte dies nicht möglich sein, gibt es die Alternative, das Modul über das CPAN zu installieren:

```
1 | # cpan  
2 | CPAN: File::HomeDir loaded ok (v0.69)  
3 |  
4 | cpan shell -- CPAN exploration and modules installation  
5 | (v1.9205)  
6 | ReadLine support enabled  
7 |  
8 |  
9 | cpan[1]> install Template  
10 | ...
```

2 - doc-generate

Der Inhalt des doc-generate-Tools ist wie folgt:

```
1 | doc-generate
2 | |-- doc-generate.pl
3 | |-- in
4 |     |-- all.in
5 |     |-- base.in
6 |     |-- calc.in
7 |     |-- content-developer.in
8 |     |-- developers.in
9 |     |-- draw.in
10 |    |-- impress.in
11 |    |-- math.in
12 |    |-- writer.in
13 | |-- templates
14 |     |-- grid-contents.tt
15 |     |-- index.tt
```

- **doc-generate.pl**

Das Perl-Programm, das ausgeführt wird. Das generierte HTML-Markup wird auf die Standardausgabe geschrieben.

- **in**

Verzeichnis mit allen Eingabedateien. Für jedes OpenOffice.org-Modul existiert eine Eingabedatei (z.B. writer.in)

- **templates**

Verzeichnis mit allen Vorlagen. Momentan sind zwei Vorlagen in Verwendung:

- index.tt:
Vorlage, die das ganze Portal repräsentiert
- grid-contents.tt:
Vorlage für ein Grid (Tabelle pro OOo-Modul)

2.1 Format für die Eingabedateien

Die Eingabedateien unterhalb des Verzeichnisses "in" weisen folgendes Format auf:

```

1 | key: value
2 | key: value
3 | key: value
4 |
5 | key: value
6 | key: value
7 | key: value

```

Jede Dokumentation wird durch einen Block repräsentiert, der durch eine Leerzeile von den anderen getrennt ist.

Jede Zeile besteht aus einem Schlüssel und einem Wert, die jeweils durch einen Doppelpunkt gefolgt von einem Leerzeichen voneinander getrennt sind.

Folgende Schlüssel-Wert-Paare sind definiert:

title	Dokument-Titel
author	Autor Wenn eine Text in eckigen Klammern angegeben wird, etwa: <div style="text-align: center; margin: 5px 0;"> Simon Wilper[simonaw@openoffice.org] </div> wird (in diesem Fall die E-Mail-Adresse) dem Template-Prozessor als zugewiesener Wert zum Schlüssel "authorlink" übergeben.
date	Datum
version	Dokumentation relevant für OOo-Version
pdf	PDF-Link
html	HTML-Link
odt	ODT-Link
indent	Einrückung (für die Handbücher z.B.)
type	Momentan nur "group" als Wert möglich. Wird dieser Schlüssel benutzt, wird nur "title" ausgewertet und fügt eine größere Überschrift ein.

2.2 Template-Prozessor starten

doc-generate.pl muss im Verzeichnis ausgeführt werden, in dem sich das Programm befindet, da relativ dazu auf die "in"- und "templates"-Verzeichnisse zugegriffen wird.

```
1 | $ ./doc-generate.pl
```

Das HTML -Markup sollte nun auf der Standardausgabe erscheinen.

Um die Ausgabe in eine Datei (z.B. index.html) zu schreiben, muss sie umgeleitet werden:

```
1 | $ ./doc-generate.pl > /home/work/de/www/doc/index.html
```



Durch Anwendung des einfachen Umleitungsoperators '>' wird die Datei, die angegeben wird ohne Vorwarnung überschrieben.

Seien Sie also vorsichtig!

2.3 Aufbau der Vorlagen

Die Syntax der Template-Sprache ist im Detail beschrieben auf <http://template-toolkit.org>.

8

2.3.1 index.tt

Damit die Sektionen auf- und zugeklappt werden können, übernimmt ein kleines JavaScript die Änderung der CSS-Eigenschaften zur Laufzeit:


```

1  var GridStates = new Array();
2
3  GridStates['grid-all']      = 0;
4
5  GridStates['grid-writer']   = 0;
6  GridStates['grid-calc']     = 0;
7  GridStates['grid-draw']     = 0;
8  GridStates['grid-base']    = 0;
9  GridStates['grid-impress']  = 0;
10 GridStates['grid-math']     = 0;
11
12 GridStates['grid-dev']      = 0;
13 GridStates['grid-author']   = 0;
14
15 function toggleGridVisible( gridname ) {
16     var obj = document.getElementById( gridname );
17     if ( GridStates[gridname] == 0 ) {
18         obj.style.display = 'block';
19         GridStates[gridname] = 1;
20     } else {
21         obj.style.display = 'none';
22         GridStates[gridname] = 0;
23     }
24 }

```

Die Funktion `toggleGridVisible` nimmt einen Parameter (`gridname`), schaut im globalen Array "GridStates" nach, ob die Sektion ausgeklappt ist (1) oder nicht (0).

Der Status wird entsprechend geändert und die Sektion mittels

```
obj.style.display = 'block';
```

bzw.

```
obj.style.display = 'none';
```

angezeigt bzw. ausgeblendet.

Im weiteren Markup werden in den a-hrefs Aufrufe zu `toggleGridVisible` getätigt:

```
<a href="javascript:toggleGridVisible(
'grid-all' )" >OpenOffice.org Allgemein</a>
```

In der `index.tt` existiert ein Template-Aufruf zu `'grid-contents.tt'` für jedes Modul:

```
[% INCLUDE 'grid-contents.tt' elements=all %]
```

Hier wird jedes Mal, wenn grid-contents.tt inkludiert wird eine Referenz mit den Werten übergeben, die für die jeweilige Sektion eingefügt werden.