

make: Fatal error: Command failed for target `test_dynamic'

make: Fatal error: Command failed for target `test_dynamic'

Source: <http://coding.derkeiler.com/Archive/Perl/comp.lang.perl.modules/2005-06/msg00167.html>

- *From:* "Cathy Hui" <kaka.hui@xxxxxxxxxx>
 - *Date:* 30 Jun 2005 11:05:25 -0700
-

Hello,

I have trouble installing Template module using. Below is the installation logs. Could someone give me some clue what went wrong and how to fix it?

Some background info:

OS: Solaris 8
Perl: v5.8.3 (From SunFreeware)
DBI: 1.4.7 (from CPAN)
DBD: 2.1028 (from CPAN)
MySQL: 4.0.21 (From SunFreeware)
GCC: 3.4.2 (From SunFreeware)
Autoconf: 2.59 (From SunFreeware)
Automake: 1.9 (From SunFreeware)

Thanks in advance!

=====
perl -MCPAN -e 'install "Template"'

```
CPAN: Storable loaded ok
Going to read /.cpan/Metadata
Database was generated on Wed, 29 Jun 2005 09:58:01 GMT
Running install for module Template
Running make for A/AB/ABW/Template-Toolkit-2.14.tar.gz
CPAN: Digest::MD5 loaded ok
CPAN: Compress::Zlib loaded ok
Checksum for
/.cpan/sources/authors/id/A/AB/ABW/Template-Toolkit-2.14.tar.gz ok
Scanning cache /.cpan/build for sizes
x Template-Toolkit-2.14, 0 bytes, 0 tape blocks
x Template-Toolkit-2.14/lib, 0 bytes, 0 tape blocks
..
blahblahblah
..
```

make: Fatal error: Command failed for target `test_dynamic'

make: Fatal error: Command failed for target `test_dynamic'

x Template-Toolkit-2.14/TODO, 26388 bytes, 52 tape blocks
Removing previously used /.cpan/build/Template-Toolkit-2.14

CPAN.pm: Going to build A/AB/ABW/Template-Toolkit-2.14.tar.gz

Template Toolkit Version 2.14

=====

Using Unix defaults.

Run 'perl Makefile.PL TT_HELP' for a summary of options.

External Modules

The Template Toolkit includes a number of plugin modules, some of which interface to external Perl modules available from CPAN. All the plugins will be installed regardless so that they will automatically work as and when you install the relevant modules. The tests will be skipped for plugins that require external modules not currently available on your system.

Text::Autoformat [X] version 1.13 installed
GD [X] version 2.19 installed
GD::Text [X] version 0.86 installed
GD::Graph [X] version 1.43 installed
GD::Graph3d [X] version 0.63 installed
Image::Info [X] version 1.16 installed
Image::Size [X] version 2.992 installed
Date::Calc [X] version 5.4 installed
Pod::POM [X] version 0.15 installed
Tie::DBI [X] version 1.01 installed
XML::DOM [X] version 1.43 installed
XML::RSS [X] version 1.05 installed
XML::XPath [X] version 1.13 installed
DBI [X] version 1.47 installed, configuring tests

Do you want to run the DBI tests?
It requires access to an existing test database. [y] n

writing t/dbi_test.cfg

XS Stash: TT2 now twice as fast!

We have a new high speed version of the Template::Stash. It's a Perl XS module which can evaluate templates about twice as fast as the pure-Perl version. It is stable and reliable but be warned that it doesn't yet support access to tied hashes (e.g. Tie::DBI) so for

make: Fatal error: Command failed for target `test_dynamic'

make: Fatal error: Command failed for target `test_dynamic'

certain applications you may need to explicitly use the regular stash.

You can choose to build the XS stash module and enable it by default so that it is used automatically for all templates. If you build it but don't enable it by default then you can use it something like this:

```
use Template;  
use Template::Stash::XS;
```

```
my $tt = new Template ( { STASH => new Template::Stash::XS, ... } );
```

You can also enable the XS stash (or the regular stash if you opt to use the XS stash by default) by setting the \$STASH package variable in the Template/Config.pm module. See 'perldoc Template::Config' for further details.

Do you want to build the XS Stash module? [y]
Do you want to use the XS Stash for all Templates? [y]

LaTeX Support

TT2 supports PDF, DVI and PostScript output using the latex filter, implemented with the programs pdflatex, latex and dvips.

Because the latex filter runs latex and pdflatex, template authors could use this feature to include any arbitrary file in their latex input, or also open an arbitrary output file, independent of the ABSOLUTE or RELATIVE configuration settings. This might create a security concern at your site. If you don't trust your template authors then don't enable the latex filter.

I found the following locations for pdflatex, latex and dvips:

```
+ pdflatex =>  
+ latex =>  
+ dvips =>
```

Do you want to enable the latex filter? [n]

Optional Extras

In addition to the Perl modules and POD documentation installed in the usual way, the Template Toolkit distribution also contains a number of optional components:

make: Fatal error: Command failed for target `test_dynamic'

make: Fatal error: Command failed for target `test_dynamic'

- * Template libraries for basic HTML, Pod → HTML, and PostScript
- * Splash! – a stylish HTML user interface template library / widget set
- * HTML documentation – distributed in template form for customisation
- * Stylesheet templates to generate docs as vanilla HTML or using Splash!
- * Examples – numerous examples of using the template libraries

If you want to install these optional components then you'll need to specify a separate directory for them.

Do you want to install these components? [y]

You can choose any directory for the installation of the additional Template Toolkit components. The proposed default assumes a Unix flavour to your operating system (suggestions for suitable defaults for other platforms welcome).

Installation directory [/usr/local/tt2]

Splash!

The Splash! template library uses a number of (very) small images to build user interface components. These will be installed into the directory:

/usr/local/tt2/images

If you want to use the Splash! library then you'll need to copy these images, define an alias (e.g. in the httpd.conf) or create a symbolic link to them so that your web server can find them. Then you'll need to specify the resulting URL which can be used to retrieve them from the web server.

Typical values might be '/tt2/images', '/images/tt2',
'/~user/tt2/images'
or even something like '<http://www.yourhost.org/images/tt2>'.

(NOTE: If this is too much for you to think about right now, then accept the default below and read the Template::Library::Splash manpage at your leisure to find out more).

URL base for TT2 images? [/tt2/images]

make: Fatal error: Command failed for target `test_dynamic'

make: Fatal error: Command failed for target `test_dynamic'

HTML Documentation

The modules comprising the Template Toolkit contain comprehensive POD documentation which can be browsed using 'perldoc' or 'man' (if your system supports it). In addition, the distribution also includes a set of source templates and style elements for generating the same documentation in HTML format. These will be installed in the directory:

/usr/local/tt2/docs

The HTML documentation can be built for you at "make install" time in a plain and simple HTML format or using the Splash! library. You can see examples of these different styles and browse the documentation online at:

<http://www.template-toolkit.org/docs/>

Do you want to build the HTML documentation? [y]

If you want to build the HTML documentation using the Splash! library then you'll need to make sure you correctly defined the URL for the Splash! images above. Otherwise, answer 'n' to the next question to use plain HTML.

Do you want to use the Splash! library? [y]

Which Splash! colour scheme would you like to use to build the documentation? Acceptable values are:

Name Colours

default lilac/mauve
aqua aqua/marine
blue grey75/blue75
green grey75/green75
grey grey75/grey50
leon red75/orange/white/black
red grey75/red75

Enter name of colour scheme: [default]

HTML Examples

A number of examples showing use of the HTML, Splash! and PostScript libraries will be installed into:

/usr/local/tt2/examples

make: Fatal error: Command failed for target `test_dynamic'

make: Fatal error: Command failed for target `test_dynamic'

As with the documentation, the examples are provided in template form and can be automatically built into HTML pages during the "make install". These pages rely on the Splash! library and expect the images URL to be correctly defined for correct viewing.

Do you want to build the HTML example pages? [y]

Checking if your kit is complete...

Looks good

Checking if your kit is complete...

Looks good

Writing Makefile for Template::Stash::XS

Writing Makefile for Template

Configuration complete. You should now run 'make', 'make test' and then 'make install'. See the README file for further information.

Installation Notes

Please note that the installation of the optional components and building of the HTML documentation is performed at the "make install" stage as the effective user at that time. This implies that this user must have sufficient permission to install into the specified directory and that all created directories and files will be owned by them.

```
cp lib/Template/Plugin/HTML.pm blib/lib/Template/Plugin/HTML.pm
```

```
..  
blahblahblah
```

```
..  
cp lib/Template/Plugin/GD/Graph/lines.pm  
Manifying blib/man1/tpage.1
```

```
..  
blahblahblah
```

```
..  
Manifying blib/man3/Template::Plugin::GD::Graph::lines.3
```

```
/usr/ccs/bin/make -- OK
```

```
Running make test
```

```
PERL_DL_NONLAZY=1 /usr/local/bin/perl "-MExtUtils::Command::MM" "-e"  
"test_harness(0, 'blib/lib', 'blib/arch)' t/*.t
```

```
..  
blahblahblah bunch of test
```

```
..  
t/config.....ok
```

```
t/constants.....ok
```

```
t/iterator.....ok
```

```
t/latex2dvi.....skipped
```

```
all skipped: LaTeX not installed
```

make: Fatal error: Command failed for target `test_dynamic'

make: Fatal error: Command failed for target `test_dynamic'

t/latex2pdf.....skipped
all skipped: LaTeX not installed
t/latex2ps.....skipped
all skipped: LaTeX not installed
t/xpath.....ok 21/21
t/xpath.....ok
Failed Test Stat Wstat Total Fail Failed List of Failed

t/gd.t 11 1 9.09% 7
5 tests skipped.
Failed 1/90 test scripts, 98.89% okay. 1/2595 subtests failed, 99.96%
okay.
*** Error code 29
make: Fatal error: Command failed for target `test_dynamic'
/usr/ccs/bin/make test -- NOT OK
Running make install
make test had returned bad status, won't install without force

-
- Prev by Date: [**Re: Google API using SOAP Lite doesn't like '&' in query**](#)
 - Next by Date: [**Failing to install IPTables::IPv4 perl module**](#)
 - Previous by thread: [**Problem with Net::SSH Perl**](#)
 - Next by thread: [**Failing to install IPTables::IPv4 perl module**](#)
 - Index(es):
 - ◆ [**Date**](#)
 - ◆ [**Thread**](#)

make: Fatal error: Command failed for target `test_dynamic'