



## ***Rocky Enterprise Linux 9.2 Manual Pages on command 'Font::TTF::XMLparse.3pm'***

**C:\>man Font::TTF::XMLparse.3pm**

Font::TTF::XMLparse(3pm) User Contributed Perl Documentation Font::TTF::XMLparse(3pm)

### NAME

Font::TTF::XMLparse - provides support for XML parsing. Requires Expat module

XML::Parser::Expat

### SYNOPSIS

```
use Font::TTF::Font;
```

```
use Font::TTF::XMLparse;
```

```
$f = Font::TTF::Font->new;
```

```
read_xml($f, $ARGV[0]);
```

```
$f->out($ARGV[1]);
```

### DESCRIPTION

This module contains the support routines for parsing XML and generating the TrueType font structures as a result. The module has been separated from the rest of the package in order to reduce the dependency that this would bring, of the whole package on XML::Parser. This way, people without the XML::Parser can still use the rest of the package.

The package interacts with another package through the use of a context containing

and element 'receiver' which is an object which can possibly receive one of the following messages:

#### XML\_start

This message is called when an open tag occurs. It is called with the context, tag name and the attributes. The return value has no meaning.

#### XML\_end

This messages is called when a close tag occurs. It is called with the context, tag name and attributes (held over from when the tag was opened). There are 3 possible return values from such a message:

undef This is the default return value indicating that default processing should occur in which either the current element on the tree, or the text of this element should be stored in the parent object.

#### \$context

This magic value marks that the element should be deleted from the parent. Nothing is stored in the parent. (This rather than " is used to allow 0 returns.)

#### anything

Anything else is taken as the element content to be stored in the parent.

In addition, the context hash passed to these messages contains the following keys:

xml This is the expat xml object. The context is also available as \$context->{'xml'}{' mycontext'}. But that is a long winded way of not saying much!

This is the base object that was passed in for XML parsing.

#### receiver

This holds the current receiver of parsing events. It may be set in associated application to adjust which objects should receive messages when. It is also stored in the parsing stack to ensure that where an object changes it during XML\_start, that that same object that received XML\_start will receive the corresponding XML\_end

#### stack

This is the parsing stack, used internally to hold the current receiver and attributes for each element open, as a complete hierarchy back to the root element.

#### tree

This element contains the storage tree corresponding to the parent of each element in the stack. The default action is to push undef onto this stack during XML\_start and then to resolve this, either in the associated application (by changing \$context->{'tree'}[-1]) or during XML\_end of a child element, by which time we know whether we are dealing with an array or a hash or what.

#### text

Character processing is to insert all the characters into the text element of the context for available use later.

## METHODS

## AUTHOR

Martin Hosken <<http://scripts.sil.org/FontUtils>>.

## LICENSING

Copyright (c) 1998-2016, SIL International (<http://www.sil.org>)

This module is released under the terms of the Artistic License 2.0. For details, see the full text of the license in the file LICENSE.

perl v5.24.1

2016-08-28

Font::TTF::XMLparse(3pm)