

# **1 Apache::Directive - Perl API for manipulating Apache configuration tree**

## 1.1 Synopsis

```
use Apache::Directive ();

my $tree = Apache::Directive->conftree;

my $documentroot = $tree->lookup('DocumentRoot');

my $vhost = $tree->lookup('VirtualHost', 'localhost:8000');
my $servername = $vhost->{'ServerName'};

print $tree->as_string;

use Data::Dumper;
print Dumper($tree->as_hash);

my $node = $tree;
while ($node) {

    #do something with $node

    if (my $kid = $node->first_child) {
        $node = $kid;
    }
    elsif (my $next = $node->next) {
        $node = $next;
    }
    else {
        if (my $parent = $node->parent) {
            $node = $parent->next;
        }
        else {
            $node = undef;
        }
    }
}
}
```

## 1.2 Description

`Apache::Directive` allows its users to search and navigate the internal Apache configuration.

Internally, this information is stored in a tree structure. Each node in the tree has a reference to its parent (if it's not the root), its first child (if any), and to its next sibling.

## 1.3 Class Methods

### 1.3.1 *conftree()*

Returns the root of the configuration tree.

```
$tree = Apache::Directive->conftree();
```

- **arg1: Apache::Directive (class name)**
- **ret: \$tree (Apache::Directive)**

## 1.4 Object Methods

### 1.4.1 *as\_hash()*

Returns a hash representation of the configuration tree, in a format suitable for inclusion in the <Perl> sections.

```
$config_hash = $conftree->as_hash();
```

- **arg1: \$conftree (Apache::Directive)**

The config tree to stringify

- **ret: \$config\_hash (HASH)**

### 1.4.2 *as\_string()*

Returns a string representation of the configuration tree, in httpd.conf format.

```
$string = $conftree->as_string();
```

- **arg1: \$conftree (Apache::Directive)**

The config tree to stringify

- **ret: \$string (string)**

### 1.4.3 *lookup()*

Returns node(s) matching a certain value.

```
$node = $conftree->lookup($directive, $args);
@nodes = $conftree->lookup($directive, $args);
```

- **arg1: \$conftree (Apache::Directive)**

The config tree to stringify

#### 1.4.4 walk\_config

- **arg2: \$directive (string)**
- **opt arg3: args (string)**
- **ret: \$string (string)**

In list context, it will return all matching nodes.

In scalar context, it will return only the first matching node.

If called with only \$directive value, this will return all nodes from that directive. For example:

```
@Alias = $conftree->lookup('Alias');
```

would return all nodes for Alias directives.

If called with an extra \$args argument, this will return only nodes where both the directive and the args matched. For example:

```
$VHost = $tree->lookup('VirtualHost', '_default_:8000');
```

If called with only one \$directive value, this will return all nodes from that directive:

```
@Alias = $tree->lookup('Alias');
```

Would return all nodes for Alias directives.

If called with an extra \$args argument, this will return only nodes where both the directive and the args matched:

```
$VHost = $tree->lookup('VirtualHosts', '_default_:8000');
```

### ***1.4.4 walk\_config***

META: Autogenerated - needs to be reviewed/completed

Walk a config tree and setup the server's internal structures

```
$ret = $conftree->walk_config($cmdparms, $section_vector);
```

- **arg1: \$conftree (Apache::Directive)**  
The config tree to walk
- **arg2: \$cmdparms (Apache::CmdParms)**  
The cmd\_parms to pass to all functions
- **arg3: \$section\_vector (Apache::ConfVector)**  
The per-section config vector.

- **ret: \$ret (string)**

Error string on error, undef otherwise

### ***1.4.5 directive***

Returns the name of the directive in \$node.

```
$name = $node->directive();
```

- **arg1: \$node (Apache::Directive)**
- **ret: \$directive (string)**

### ***1.4.6 args***

The arguments for the current directive, stored as a space separated list

```
$args = $node->args();
```

- **arg1: \$node (Apache::Directive)**
- **ret: \$args (string)**

### ***1.4.7 next***

The next directive node in the tree

```
$next_node = $node->next();
```

- **arg1: \$node (Apache::Directive)**
- **ret: \$next\_node (Apache::Directive)**

Returns the next sibling of \$node, undef if there is none

### ***1.4.8 first\_child***

The first child node of this directive

```
$subtree = $node->first_child;
```

- **arg1: \$node (Apache::Directive)**
- **ret: \$child\_node (Apache::Directive)**

Returns the first child node of \$node, undef if there is none

## ***1.4.9 parent***

META: Autogenerated - needs to be reviewed/completed

The parent node of this directive

```
$parent_node = $node->parent();
```

- **arg1: \$node (Apache::Directive)**
- **ret: parent\_node (Apache::Directive)**

Returns the parent of \$node, undef if this node is the root node

## ***1.4.10 data***

META: Autogenerated - needs to be reviewed/completed

directive's module can store add'l data here

```
$ret = $conftree->data($newval);
```

- **arg1: \$conftree (Apache::Directive)**
- **arg2: \$newval XXX**
- **ret: XXX**

## ***1.4.11 filename***

Returns the filename the configuration node was created from

```
$filename = $node->filename();
```

- **arg1: \$node (Apache::Directive)**
- **ret: \$filename (string)**

## ***1.4.12 line\_num***

Returns the line number in filename this node was created from

```
$lineno = $node->line_num();
```

- **arg1: \$node (Apache::Directive)**
- **arg2: \$lineno (integer)**

## **1.5 See Also**

mod\_perl 2.0 documentation.

## **1.6 Copyright**

mod\_perl 2.0 and its core modules are copyrighted under The Apache Software License, Version 1.1.

## **1.7 Authors**

The mod\_perl development team and numerous contributors.



## Table of Contents:

1	Apache::Directive - Perl API for manipulating Apache configuration tree . . . . .	1
1.1	Synopsis . . . . .	2
1.2	Description . . . . .	2
1.3	Class Methods . . . . .	2
1.3.1	conftree() . . . . .	3
1.4	Object Methods . . . . .	3
1.4.1	as_hash() . . . . .	3
1.4.2	as_string() . . . . .	3
1.4.3	lookup() . . . . .	3
1.4.4	walk_config . . . . .	4
1.4.5	directive . . . . .	5
1.4.6	args . . . . .	5
1.4.7	next . . . . .	5
1.4.8	first_child . . . . .	5
1.4.9	parent . . . . .	6
1.4.10	data . . . . .	6
1.4.11	filename . . . . .	6
1.4.12	line_num . . . . .	6
1.5	See Also . . . . .	7
1.6	Copyright . . . . .	7
1.7	Authors . . . . .	7