

1 Apache2::Directive - Perl API for manipulating the Apache configuration tree

1.1 Synopsis

```

use Apache2::Directive ();

my $tree = Apache2::Directive::conftree();

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

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

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

my $node = $tree;
while ($node) {
    print $node->as_string;

    #do something with $node

    my $directive = $node->directive;
    my $args = $node->args;
    my $filename = $node->filename;
    my $line_num = $node->line_num;

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

```

1.2 Description

`Apache2::Directive` provides the Perl API for manipulating the Apache configuration tree

1.3 API

`Apache2::Directive` provides the following functions and/or methods:

1.3.1 args

Get the arguments for the current directive:

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

- **obj:** `$node` (`Apache2::Directive` object)
- **ret:** `$args` (string)

Arguments are separated by a whitespace in the string.

- **since:** 2.0.00

For example, in *httpd.conf*:

```
PerlSwitches -M/opt/lib -M/usr/local/lib -wT
```

And later:

```
my $tree = Apache2::Directive::conftree();
my $node = $tree->lookup('PerlSwitches');
my $args = $node->args;
```

`$args` now contains the string "-M/opt/lib -M/usr/local/lib -wT"

1.3.2 as_hash

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

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

- **obj:** `$conftree` (`Apache2::Directive` object)

The config tree to stringify

- **ret:** `$config_hash` (HASH reference)
- **since:** 2.0.00

For example: in *httpd.conf*:

```
<Location /test>
  SetHandler perl-script
  PerlHandler Test::Module
</Location>
```

And later:

```
my $tree = Apache2::Directive::conftree();
my $node = $tree->lookup('Location', '/test/');
my $hash = $node->as_hash;
```

\$hash now is:

```
{
  'SetHandler' => 'perl-script',
  'PerlHandler' => 'Test::Module',
}
```

1.3.3 as_string

Get a string representation of the configuration node, in *httpd.conf* format.

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

- **obj:** `$node (Apache2::Directive object)`

The config tree to stringify

- **ret:** `$string (string)`
- **since:** 2.0.00

For example: in *httpd.conf*:

```
<Location /test>
  SetHandler perl-script
  PerlHandler Test::Module
</Location>
```

And later:

```
my $tree = Apache2::Directive::conftree();
my $node = $tree->lookup('Location', '/test/');
my $string = $node->as_string;
```

\$string is now:

```
SetHandler perl-script
PerlHandler Test::Module
```

1.3.4 conftree

Get the root of the configuration tree:

```
$conftree = Apache2::Directive::conftree();
```

- **obj:** `Apache2::Directive (class name)`
- **ret:** `$conftree (Apache2::Directive object)`
- **since:** 2.0.00

1.3.5 *directive*

Get the name of the directive in `$node`:

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

- **obj:** `$node` (**Apache2::Directive object**)
- **ret:** `$name` (**string**)
- **since:** **2.0.00**

1.3.6 *filename*

Get the *filename* the configuration node was created from:

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

- **obj:** `$node` (**Apache2::Directive object**)
- **ret:** `$filename` (**string**)
- **since:** **2.0.00**

For example:

```
my $tree = Apache2::Directive::conftree();
my $node = $tree->lookup('VirtualHost', 'example.com');
my $filename = $node->filename;
```

`$filename` is now the full path to the *httpd.conf* that `VirtualHost` was defined in.

If the directive was added with `add_config()`, the filename will be the path to the *httpd.conf* that triggered that Perl code.

1.3.7 *first_child*

Get the first child node of this directive:

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

- **obj:** `$node` (**Apache2::Directive object**)
- **ret:** `$child_node` (**Apache2::Directive object**)

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

- **since:** **2.0.00**

1.3.8 `line_num`

Get the line number in a *filename* this node was created at:

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

- **obj:** `$node (Apache2::Directive object)`
- **arg1:** `$lineno (integer)`
- **since:** 2.0.00

1.3.9 `lookup`

Get the node(s) matching a certain value.

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

- **obj:** `$conftree (Apache2::Directive object)`

The config tree to stringify

- **arg1:** `$directive (string)`

The name of the directive to search for

- **opt arg2:** `args (string)`

Optional args to the directive to filter for

- **ret:** `$string (string / ARRAY of HASH refs)`

In LIST context, it returns all matching nodes.

In SCALAR context, it returns only the first matching node.

If called with only `$directive` value, this method returns all nodes from that directive. For example:

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

returns all nodes for `Alias` directives.

If called with an extra `$args` argument, it returns only nodes where both the directive and the args matched. For example:

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

- **since:** 2.0.00

1.3.10 *next*

Get the next directive node in the tree:

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

- **obj:** `$node (Apache2::Directive object)`
- **ret:** `$next_node (Apache2::Directive object)`

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

- **since:** 2.0.00

1.3.11 *parent*

Get the parent node of this directive:

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

- **obj:** `$node (Apache2::Directive object)`
- **ret:** `parent_node (Apache2::Directive object)`

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

- **since:** 2.0.00

1.4 See Also

`mod_perl 2.0 documentation`.

1.5 Copyright

`mod_perl 2.0` and its core modules are copyrighted under The Apache Software License, Version 2.0.

1.6 Authors

The `mod_perl` development team and numerous contributors.

Table of Contents:

1	Apache2::Directive - Perl API for manipulating the Apache configuration tree	1
1.1	Synopsis	2
1.2	Description	2
1.3	API	2
1.3.1	args	3
1.3.2	as_hash	3
1.3.3	as_string	4
1.3.4	conftree	4
1.3.5	directive	5
1.3.6	filename	5
1.3.7	first_child	5
1.3.8	line_num	6
1.3.9	lookup	6
1.3.10	next	7
1.3.11	parent	7
1.4	See Also	7
1.5	Copyright	7
1.6	Authors	7