

## NAME

PSP::Config - the configuration object class for the Perl Script Pages (PSP) interface, engine and supporting modules.

## SYNOPSIS

```
use PSP::Config;
my $conf = new PSP::Config(
    file => $filename,
    hash => {
        key1 => $value1,
        key2 => $value2
    }
);

my $value = $conf->get($key);

my $value = $conf->{$key};

my $success = $conf->add($key,$value);

my $success = $conf->addHash(\%hash);

my $success = $conf->addHash(
    {
        key1      => 'value',
        key_array => ['value2', 'value3']
    }
);

my $success = $conf->addFile($filename);
```

## DESCRIPTION

This modules provides default configuration values, methods for setting and accessing configuration data and loading configuration from files. This module was created as a companion to the Perl Script Pages Engine. See *PSP::Engine* for more information.

## OVERVIEW

### Object Constructors

`new`

Creates a new PSP::Config object with default values. Optional a file to load configuration from or a configuration hash may be passed to the constructor. The new constructor is the only object constructor for this module. See *new object constructor* for details

### Object Methods

`get`

Receive a configuration value for a given key from the configuration data. See *get method* for details

`add`

Adds a single configuration key/value set to the configuration data. See *add method* for details

#### addHash

Adds a referenced hash or an anonymous hash to the configuration data. See *addHash method* for details

#### addFile

Loads data from a file and adds it to the object's configuration data. See *addFile method* for details

## SYNTAX

### new Object Constructor

Creates a new PSP::Config object with default values. Optional a file to load configuration from or a configuration hash may be passed to the constructor. For file format definitions see *addFile method*. For hash format definitions see *addHash method*.

```
use PSP::Config;
my $conf = new PSP::Config(
    file => $filename,
    hash => \%hash
);
```

```
use PSP::Config;
my $conf = new PSP::Config(
    hash => {
        key1 => $value1,
        key2 => $value2
    }
);
```

### Required Parameters

None.

### Optional Parameters

The following parameters are key/value hash parameters.

#### file

Filename to load configuration data from. For file format definitions see *addFile method*.

#### hash

A reference to a hash or an anonymous hash with configuration data. For hash format definitions see *addHash method*.

### get Method

Receive a configuration value for a given key from the configuration data.

```
my $value = $conf->get($key);
```

You may also access the configuration data directly via the object.

```
my $value = $conf->{$key};
```

### Required Parameters

#### \$key

Configuration key (directive) to receive configuration value for.

### Optional Parameters

None.

### Return Values

`$value`

Returns the value stored for the given configuration key or undef if not found.

### add Method

Adds a single configuration key/value set to the configuration data.

```
my $success = $conf->add($key,$value);
```

If `$key` describes a simple scalar configuration directive, `$value` replaces the default. If `$key` is an array configuration directive, `$value` is appended to the corresponding values array.

`$value` may contain special `@@string@@` macros that are replaced by environment variables, for example `@@DOCUMENT_ROOT@@` will be replaced by the content of the `$DOCUMENT_ROOT` environment variable, i.e. `"/var/www/html/"`.

### Required Parameters

`$key`

Configuration key used to store and identify the new value.

`$value`

Configuration data to store for `$key`.

### Optional Parameters

None.

### Return Values

`$success`

Returns 1 on success or undef on failure.

### addHash Method

Adds a referenced hash or an anonymous hash to the configuration data.

```
my $success = $conf->addHash($hashref);
```

```
my %hash = (  
    key1      => 'value',  
    key_array => ['value2', 'value3']  
);  
my $success = $conf->addHash(\%hash);
```

```
my $success = $conf->addHash(  
    {  
        key1      => 'value',  
        key_array => ['value2', 'value3']  
    }  
);
```

The hash elements can either be standard "key => \$value" parameters that call the *add method* once or "key => [\$value1, \$value2]" ordered array parameters that call the *add method* for every value in the array but always for the same key.

Values may contain environment variable macros. See *add method* for details.

### Required Parameters

\$hashref

A reference to a hash containing configuration data.

### Optional Parameters

None.

### Return Values

\$success

Returns 1 on success or undef on failure.

### addFile Method

Loads data from a file and adds it to the object's configuration data.

```
my $success = $conf->addFile($filename);
```

Configuration files are ordinary text files and may contain lines of the format `ThisIsTheKey=This is a value`, terminated by `\n` or `\r\n`. The system interpretes everything left of the first "=" character as the key, everything on the right as configuration value, so both key and value parts may contain i.e. spaces, value even may contain the "=" character without quoting or escaping. Empty lines and lines beginning with "#" (comments) are ignored.

Values may contain environment variable macros. See *add method* for details.

### Required Parameters

\$filename

Name of the file to load configuration data from.

### Optional Parameters

None.

### Return Values

\$success

Returns 1 on success or undef on failure.

### AUTHOR

Veit Wahlich

E-Mail: [cru \[at\] zodia \[dot\] de](mailto:cru[at]zodia[dot]de)

WWW: <http://home.ircnet.de/cru/>

### VERSION

v0.7 Wednesday, 18 January 2006

### COPYRIGHT/LICENSE

Copyright 2004-2006 Veit Wahlich

This software is distributed as free (libre) software under the terms of the GNU General Public License, version 2 <<http://www.gnu.org/copyleft/gpl.html>>. The author disclaims responsibility of any damage or harm caused directly or indirectly by usage of this software. Use only at your own risk.