

patConfiguration



Contents

Package patExampleGen Procedural Elements	2
patExampleGen.php	2
Package patExampleGen Classes	3
Class patExampleGen	3
Method displayError	3
Method displayExample	4
Method displayFooter	4
Method displayFrameset	4
Method displayGuide	4
Method displayHead	5
Method displayNavigation	5
Method displayOutput	5
Method displayOverview	6
Method displayPHPSource	6
Method displayText	6
Method displayXMLSource	6
Method displayXMLString	7
Method file_get_contents	7
Method getDefaultTab	8
Method getExample	8
Method getExampleSection	8
Method getRequestVars	9
Method getTabs	9
Method highlightXML	9
Method process	9
Method setAppDescription	10
Method setAppForumId	10
Method setAppName	10
Method setSections	11
Method setTabs	11
Method url_get_contents	12
fileUsage.php	13
_examples_config.php	14
_examples_customFunctions.php	15
_examples_prepend.php	16
Package patConfiguration Procedural Elements	18
patConfiguration.php	18
Define PATCONFIGURATION_ERROR_CACHEDIR_NOT_WRITEABLE	18
Define PATCONFIGURATION_ERROR_CONFIGDIR_NOT_FOUND	18
Define PATCONFIGURATION_ERROR_CONFIG_INVALID	18
Define PATCONFIGURATION_ERROR_CORRUPT_DRIVER	18
Define PATCONFIGURATION_ERROR_DRIVERDIR_NOT_FOUND	19
Define PATCONFIGURATION_ERROR_DRIVER_NOT_WORKING	19

Define PATCONFIGURATION_ERROR_FILE_NOT_FOUND	19
Define PATCONFIGURATION_ERROR_FILE_NOT_READABLE	19
Define PATCONFIGURATION_ERROR_FILE_NOT_WRITABLE	19
Define PATCONFIGURATION_ERROR_UNKNOWN_DRIVER	19
Define PATCONFIGURATION_WARNING_CONFIGVALUE_WITHOUT_NAME	19
Package patConfiguration Classes	20
Class patConfiguration	20
Var \$cacheDir	20
Var \$conf	20
Var \$configDir	20
Var \$driverDir	21
Var \$drivers	21
Var \$encoding	21
Var \$systemVars	21
Constructor patConfiguration	21
Constructor __construct	22
Method clearConfigValue	22
Method getConfigDir	22
Method getConfigValue	23
Method getDriverDir	23
Method getFullPath	23
Method getOption	24
Method loadCachedConfig	24
Method loadConfig	24
Method parseConfigFile	25
Method setCacheDir	25
Method setConfigDir	26
Method setConfigValue	26
Method setConfigValues	26
Method setDriverDir	27
Method setEncoding	27
Method setOption	27
Method writeConfigFile	28
example api array.php	29
example api basic.php	30
example api cache.php	31
example api cache option.php	32
example api create.php	33
example api errors.php	34
example api objects.php	35
example api save.php	36
example define auto.php	37
example define basic.php	38
example define children.php	39
example define constants.php	40
Define FOO	40
example define content.php	41
example define external.php	42
example define object.php	43

example define reserved.php	44
example formats ini.php	45
example formats php.php	46
example formats wddx.php	47
example formats xml.php	48
example misc whitespace.php	49
example misc xinc.php	50
index.php	51
patConfigurationSource.php	52
examples sections.php	53
examples sections api.php	54
examples sections define.php	55
examples sections formats.php	56
examples sections misc.php	57
Class Foo	57
Constructor Foo	57
Class page	57
Constructor page	58
Reader.php	59
INI.php	60
WDDX.php	61
XML.php	62
Define PATCONFIGURATION READER XML ERROR RESERVED TAG	62
Class patConfiguration Reader	62
Var \$configObj	62
Method loadConfigFile	62
Method setConfigReference	63
Class patConfiguration Reader INI	63
Method loadConfigFile	64
Class patConfiguration Reader WDDX	64
Method loadConfigFile	65
Class patConfiguration Reader XML	65
Var \$conf	65
Var \$currentNamespace	65
Var \$currentTag	66
Var \$data	66
Var \$defineStack	66
Var \$externalFiles	66
Var \$includedFiles	66
Var \$nsStack	66
Var \$parsers	66
Var \$path	67
Var \$reservedTags	67
Var \$tagStack	67
Var \$valDepth	67
Var \$valStack	67
Var \$whitespace	67
Var \$xmlFiles	67
Method appendData	67

Method characterData	68
Method createParser	68
Method endElement	68
Method externalEntity	68
Method getConfigValue	69
Method loadConfigFile	69
Method parseXMLFile	69
Method setDefinedTags	69
Method startElement	70
autopackage.php	71
package.php	72
Writer.php	73
INI.php	74
PHP.php	75
WDDX.php	76
XML.php	77
Class patConfiguration_Writer	77
Var \$configOb	77
Method serializeConfig	77
Method setConfigReference	78
Class patConfiguration_Writer_INI	78
Class patConfiguration_Writer_PHP	79
Method serializeConfig	79
Class patConfiguration_Writer_WDDX	80
Var \$xmlSpecialchars	80
Method replaceXMLSpecialchars	80
Class patConfiguration_Writer_XML	81
Var \$openTags	81
Var \$xmlSpecialchars	81
Method replaceXMLSpecialchars	81
Appendices	82
Appendix A - Class Trees	83
patConfiguration	83
patExampleGen	84

Package patExampleGen Procedural Elements

patExampleGen.php

The **pat examples framework class** - manages the navigation and functionality of the **examples collection of our tools**.

\$Id: patExampleGen.php 29 2005-03-04 21:25:29Z schst \$

- **Package** patExampleGen
- **Author** Sebastian Mordziol < argh@php-tools.net >

Package patExampleGen Classes

Class patExampleGen

[line 27]

The pat examples framework class - manages the navigation and functionality of the examples collection of our tools.

\$Id: patExampleGen.php 29 2005-03-04 21:25:29Z schst \$

- **Package** patExampleGen
- **Link** <http://www.php-tools.de>
- **License** LGPL
- **Author** Sebastian Mordziol < argh@php-tools.net>
- **Version** 0.9

object error function patExampleGen::displayError(&\$error) *[line 229]*

Function Parameters:

- *object error* **&\$error** object

Custom error handler that is set automatically to display any patError error objects within the examples collection.

Displays:

- Error level
- Error Message
- Error info
- Error file
- Error line
- plus the call stack that lead to the error

- **Static**
- **Access** public
- **Author** Sebastian Mordziol < argh@php-tools.net>

function patExampleGen::displayExample() [*line 338*]

Displays an example page along with all defined tabs

- **Access** public

function patExampleGen::displayFooter() [*line 804*]

Displays the footer for any example page

- **Access** public

function patExampleGen::displayFrameset() [*line 609*]

Displays the examples main frameset

- **Access** public

function patExampleGen::displayGuide(\$tabId, \$file, \$guide) [*line 461*]

Function Parameters:

- **\$tabId**
- **\$file**

- **\$guide**

function patExampleGen::displayHead(\$area, [\$onload = null]) [*line 781*]

Function Parameters:

- *string* **\$area** The area we are in - is added to the body style
- **\$onload**

Displays the header for any example page

- **Access public**

function patExampleGen::displayNavigation() [*line 638*]

Displays the main examples navigation

- **Access public**

function patExampleGen::displayOutput(\$tabId, \$file) [*line 456*]

Function Parameters:

- *string* **\$tabId** The ID of the tab
- *string* **\$file** The file to show

Display an example's output in an example tab via an iframe

- **Access public**

function patExampleGen::displayOverview() [*line 712*]

Displays the examples overview page

- **Access** public

function patExampleGen::displayPHPSource(\$tabId, \$file) [*line 520*]

Function Parameters:

- *string* **\$tabId** The ID of the tab
- *string* **\$file** The file to highlight

Display PHP source code in an example tab

- **Access** public

function patExampleGen::displayText(\$tabId, \$text) [*line 534*]

Function Parameters:

- *string* **\$tabId** The ID of the tab
- *string* **\$text** The text to display

Display a free text in a tab

- **Access** public

function patExampleGen::displayXMLSource(\$tabId, \$file) [*line 548*]

Function Parameters:

- *string* **\$tabId** The ID of the tab
- *string* **\$file** The file to highlight

Display XML source code in an example tab

- **Access** public

function patExampleGen::displayXMLString(\$xml) [*line 568*]

Function Parameters:

- *string* **\$xml** The XML string

Hilights an xml string and displays it

- See [patExampleGen::highlightXML\(\)](#)
- **Access** public

mixed function patExampleGen::file_get_contents(\$filename) [*line 907*]

Function Parameters:

- *string* **\$filename** filename

Replacement for file_get_contents

- **Static**
- **Access** public

string function patExampleGen::getDefaultTab(\$tabs) [*line 196*]

Function Parameters:

- **\$tabs**

Retrieves the default example view tab, the one that is active when the page is loaded.

- **Access public**

string function patExampleGen::getExample(\$id) [*line 835*]

Function Parameters:

- *string* **\$id** example id

Get an example's details

- **Access public**

string function patExampleGen::getExampleSection(\$id) [*line 817*]

Function Parameters:

- *string* **\$id** example id

get the example section

- **Access public**

array function patExampleGen::getRequestVars() [*line 133*]

Retrieves the request variables from POST and GET

- **Access** public

array function patExampleGen::getTabs(\$id) [*line 866*]

Function Parameters:

- *string* **\$id** The example id

Get the tabs for an example (per example tabs can be modified/added).

- **Access** public

string function patExampleGen::highlightXML(\$xml) [*line 580*]

Function Parameters:

- *string* **\$xml** The XML string

Highlights an xml string and returns the highlighted source

- **Access** public

function patExampleGen::process() [*line 99*]

Main process method that dispatches needed tasks according to the given action.

- **Access** public

function patExampleGen::setAppDescription(\$desc) [*line 171*]

Function Parameters:

- *string* **\$desc** The description text

Sets the description of the application

- **See** \$appDescription
- **Access** public

function patExampleGen::setAppForumId(\$id) [*line 184*]

Function Parameters:

- *int* **\$id** The forum ID

Sets the ID of the application's forum. Needed to display the correct link to the forum.

- **See** \$appForumId
- **Access** public

function patExampleGen::setAppName(\$appName) [*line 159*]

Function Parameters:

- *string* **\$appName** The name

Sets the name of the application

- **See** \$appName
- **Access** public

```
function patExampleGen::setSections($sections, &$sections) [line 628]
```

Function Parameters:

- *array* **&\$sections** The sections definition
- **\$sections**

Sets the sections from which to generate the examples framework

- **Access** public

```
function patExampleGen::setTabs($tabs) [line 147]
```

Function Parameters:

- *array* **\$tabs** The tab configuration

Sets the default tab configuration that will be used for each example page. Additional tabs can be defined for each example within the section definition.

- **See** \$tabs
- **Access** public

mixed function patExampleGen::url_get_contents(\$url) [*line 938*]

Function Parameters:

- *string* **\$url** The Url to get the content from

Simple url content loader

- **Static**
- **Access** public

fileUsage.php

Helper file for the examples collection that displays a list of example files that may be missing or are not referenced to in the examples section definitions.

\$Id: fileUsage.php 29 2005-03-04 21:25:29Z schst \$

- **Package** patExampleGen
- **Sub-Package** Examples
- **Link** <http://www.php-tools.net>
- **Author** Sebastian Mordziol < s.mordziol@metrix.de>
- **Version** 0.1
- **Access** public

require_once '**prepend.php**' [*line 20*]

Main prepend file

`__examples_config.php`

Configuration for the examples - edit this file to set the paths to the needed classes as they should be on your server.

\$Id: __examples_config.php 34 2005-03-05 16:55:02Z schst \$

- **Package** patExampleGen
- **Sub-Package** Examples
- **Author** Stephan Schmidt < schst@php-tools.net>

`_examples_customFunctions.php`

Custom functions for the current examples collection - the functions defined here are available for all examples.

\$Id: `_examples_customFunctions.php` 34 2005-03-05 16:55:02Z schst \$

- **Package** patExampleGen
- **Sub-Package** Examples
- **Link** <http://www.php-tools.net>
- **Author** Stephan Schmidt < schst@php-tools.net>
- **Version** 0.1
- **Access** public

`_examples_prepend.php`

Main prepend file for the examples framework, sets error reporting and initializes the `patExampleGen` class that manages the whole framework.

\$Id: _examples_prepend.php 29 2005-03-04 21:25:29Z schst \$

- **Package** patExampleGen
- **Sub-Package** Examples
- **Author** Sebastian Mordziol < argh@php-tools.net>

`include_once 'examples_customFunctions.php'` [line 34]

Custom functions for the current examples collection

`include_once 'examples_config.php'` [line 19]

The examples needed files list

`include_once 'patExampleGen/patExampleGen.php'` [line 29]

The examples manager class patExampleGen

`include_once 'examples_sections.php'` [line 24]

The examples definitions

`require_once $neededFiles['patErrorManager']` [line 50]

patErrorManager class

Package patConfiguration Procedural Elements

patConfiguration.php

patConfiguration

\$Id: patConfiguration.php 54 2006-02-03 08:01:11Z argh \$

Abstracts access to configuration files. Currently XML, WDDX, INI and PHP files are supported.

WARNING: patConfiguration 2.0.0 is not backwards compatible to older versions. It supports patError and is now driverbased. Furthermore, the extensions have been dropped. They could possibly be integrated in the final release.

- **Package** patConfiguration
- **Link** <http://www.php-tools.net>
- **Author** Stephan Schmidt < schst@php-tools.net >
- **Version** 2.0.0

PATCONFIGURATION_ERROR_CACHEDIR_NOT_WRITEABLE = 20006 *[line 51]*
cache file could not be written

PATCONFIGURATION_ERROR_CONFIGDIR_NOT_FOUND = 20008 *[line 61]*
config directory does not exist

PATCONFIGURATION_ERROR_CONFIG_INVALID = 20010 *[line 66]*
config file has invalid syntax

PATCONFIGURATION_ERROR_CORRUPT_DRIVER = 20004 *[line 41]*
Driver file does not contain driver class

PATCONFIGURATION_ERROR_DRIVERDIR_NOT_FOUND = 20007 *[line 56]*

config directory does not exist

PATCONFIGURATION_ERROR_DRIVER_NOT_WORKING = 20005 *[line 46]*

driver is not working

PATCONFIGURATION_ERROR_FILE_NOT_FOUND = 20001 *[line 26]*

File not found

PATCONFIGURATION_ERROR_FILE_NOT_READABLE = 20002 *[line 31]*

File not readable

PATCONFIGURATION_ERROR_FILE_NOT_WRITABLE = 20030 *[line 76]*

configuration file could not be written

PATCONFIGURATION_ERROR_UNKNOWN_DRIVER = 20003 *[line 36]*

Driver file not found

PATCONFIGURATION_WARNING_CONFIGVALUE_WITHOUT_NAME = 20020 *[line 71]*

config value without a name

Package patConfiguration Classes

Class patConfiguration *[line 89]*

patConfiguration

Main class, aggregates readers and writers to access config files.

- **Package** patConfiguration
- **Author** Stephan Schmidt < schst@php-tools.net>
- **Version** 2.0
- **Access** public

patConfiguration::\$cacheDir

string = 'cache' [line 118]

directory where cache files are located

patConfiguration::\$conf

array = array() [line 106]

array that stores configuration

patConfiguration::\$configDir

string = [line 112]

directory, where the configuration files are stored

patConfiguration::\$driverDir

string = NULL [*line 124*]

directory where drivers are stored

patConfiguration::\$drivers

array = array() [*line 130*]

drivers that have already been loaded

patConfiguration::\$encoding

string = 'ISO-8859-1' [*line 136*]

default encoding

- **Var** encoding

patConfiguration::\$systemVars

```
array = array(  
    'appName' => 'patConfiguration',  
    'appVersion' => '2.0.0',  
    'author' => array(  
        'Stephan Schmidt <schst@php-tools.net>'  
    )) [line 95]
```

information about the project

Constructor function patConfiguration::patConfiguration([\$options = array()]) [*line 154*]

Function Parameters:

- *array* **\$options**

constructor

- **Access** public

Constructor function patConfiguration::__construct([\$options = array()]) [*line 165*]

Function Parameters:

- *array* **\$options**

constructor for PHP 5

- **Access** public

function patConfiguration::clearConfigValue([\$path = ""]) [*line 913*]

Function Parameters:

- *string* **\$path** path, where the value is stored

clears a config value

if no path is given, the complete config will be cleared

- **Access** public

string function patConfiguration::getConfigDir() [*line 210*]

get the directory, where all config files are stored

- **Access** public

mixed function patConfiguration::getConfigValue([\$path = "]) [*line 839*]

Function Parameters:

- *string* **\$path** path, where the value is stored

returns a configuration value

if no path is given, all config values will be returned in an array

- **Access public**

string function patConfiguration::getDriverDir() [*line 948*]

get driver directory

- **Access public**

function patConfiguration::getFullPath(\$file, [\$relative = NULL], [\$check = true]) [*line 453*]

Function Parameters:

- *string* **\$file** filename
- *string* **\$relative** get path relative to this file
- **\$check**

get the full path of a file

- **Access public**

mixed function patConfiguration::getOption(\$name) [*line 282*]

Function Parameters:

- *string* **\$name** name of the option

get an option

This is only used by the drivers.

- **Access** public

mixed function patConfiguration::loadCachedConfig(\$file, [\$options = array('mode' => 'w')], \$mode) [*line 369*]

Function Parameters:

- *string* **\$file** name of config file
- *string* **\$mode** options for loading the config
- **\$options**

load a configuration from a cache

if cache is not valid, it will be updated automatically

- **See** [patConfiguration::loadConfig\(\)](#)
- **Access** public

function patConfiguration::loadConfig(\$file, [\$options = array('mode' => 'w')], \$mode) [*line 304*]

Function Parameters:

- *string* **\$file** name of config file
- *string* **\$mode** options for loading the config
- **\$options**

load a configuration

this loads any config file, alias for parseConfigFile

Possible options are:

- mode (a|w)
- filetype

- See [patConfiguration::loadCachedConfig\(\)](#), [patConfiguration::parseConfigFile\(\)](#)
- Access public

function patConfiguration::parseConfigFile(\$file, [\$options = array('mode' => 'w')], \$mode) [*line 441*]

Function Parameters:

- *string* **\$file** name of the configuration file
- *string* **\$mode** options for loading the config
- **\$options**

parse a configuration file

please use loadConfig instead

- **Deprecated** since v2.0
- See [patConfiguration::loadConfig\(\)](#), [patConfiguration::loadCachedConfig\(\)](#)
- Access public

boolean function patConfiguration::setCacheDir(\$cacheDir) [*line 240*]

Function Parameters:

- *string* **\$cacheDir** name of the directory

set the directory, where all cache files are stored

- **Throws** patError
- **Access** public

function patConfiguration::setConfigDir(\$configDir) [*line 195*]

Function Parameters:

- *string* **\$configDir** name of the directory

set the directory, where all config files are stored

- **Access** public

function patConfiguration::setConfigValue(\$path, \$value, [\$type = 'leave']) [*line 883*]

Function Parameters:

- *string* **\$path** path, where the value will be stored
- *mixed* **\$value** value to store
- **\$type**

set a config value

- **Access** public

function patConfiguration::setConfigValues(\$values) [*line 895*]

Function Parameters:

- *array* **\$values** assoc array containing paths and values

sets several config values

- **Access** public

boolean function patConfiguration::setDriverDir(\$driverDir) [*line 223*]

Function Parameters:

- *string* **\$driverDir** name of the directory

set the directory, where all drivers are stored

- **Throws** patError
- **Access** public

function patConfiguration::setEncoding(\$encoding) [*line 183*]

Function Parameters:

- *string* **\$encoding**

set the XML encoding

- **Access** public

mixed function patConfiguration::setOption(\$option, \$value) [*line 263*]

Function Parameters:

- *string* **\$option** name of the option
- *mixed* **\$value** value of the option

set any option

This method checks, whether there is a method that should be called for the specific option

- **Access** public

function patConfiguration::writeConfigFile(\$filename, [\$options = NULL], [\$oldOptions = NULL], \$format) [*line 676*]

Function Parameters:

- *string* **\$filename** name of the configfile
- *string* **\$format** format of the config file
- *array* **\$options** options, see the writer driver for details
- **\$oldOptions**

write a configfile

- **Access** public

example_api_array.php

patConfiguration example

\$Id: example_api_array.php 29 2005-03-04 21:25:29Z schst \$

- **Package** patConfiguration
- **Sub-Package** Examples
- **Author** Stephan Schmidt < schst@php-tools.net>

require_once ['./patConfiguration.php'](#) [*line 25*]

require main class

require_once **'pat/patErrorManager.php'** [*line 18*]

requires patErrorManager

make sure that it is in your include path

example_api_basic.php

patConfiguration example

\$Id: example_api_basic.php 29 2005-03-04 21:25:29Z schst \$

- **Package** patConfiguration
- **Sub-Package** Examples
- **Author** Stephan Schmidt < schst@php-tools.net>

require_once ['./patConfiguration.php'](#) *[line 25]*

require main class

require_once **'pat/patErrorManager.php'** *[line 18]*

requires patErrorManager

make sure that it is in your include path

example_api_cache.php

patConfiguration example

\$Id: example_api_cache.php 29 2005-03-04 21:25:29Z schst \$

- **Package** patConfiguration
- **Sub-Package** Examples
- **Author** Stephan Schmidt < schst@php-tools.net>

require_once ['./patConfiguration.php'](#) *[line 25]*

require main class

require_once **'pat/patErrorManager.php'** *[line 18]*

requires patErrorManager

make sure that it is in your include path

example_api_cache_option.php

patConfiguration example

\$Id: example_api_cache_option.php 29 2005-03-04 21:25:29Z schst \$

- **Package** patConfiguration
- **Sub-Package** Examples
- **Author** Stephan Schmidt < schst@php-tools.net>

require_once ['./patConfiguration.php'](#) *[line 25]*

require main class

require_once **'pat/patErrorManager.php'** *[line 18]*

requires patErrorManager

make sure that it is in your include path

example_api_create.php

patConfiguration example

\$Id: example_api_create.php 30 2005-03-04 21:35:42Z schst \$

- **Package** patConfiguration
- **Sub-Package** Examples
- **Author** Stephan Schmidt < schst@php-tools.net>

require_once ['./patConfiguration.php'](#) *[line 25]*

require main class

require_once **'pat/patErrorManager.php'** *[line 18]*

requires patErrorManager

make sure that it is in your include path

example_api_errors.php

patConfiguration example

\$Id: example_api_errors.php 29 2005-03-04 21:25:29Z schst \$

- **Package** patConfiguration
- **Sub-Package** Examples
- **Author** Stephan Schmidt < schst@php-tools.net>

require_once ['./patConfiguration.php'](#) *[line 23]*

require main class

require_once **'pat/patErrorManager.php'** *[line 18]*

requires patErrorManager

make sure that it is in your include path

example_api_objects.php

patConfiguration example

\$Id: example_api_objects.php 33 2005-03-05 15:28:08Z schst \$

- **Package** patConfiguration
- **Sub-Package** Examples
- **Author** Stephan Schmidt < schst@php-tools.net>

require_once ['./patConfiguration.php'](#) *[line 36]*

require main class

require_once **'pat/patErrorManager.php'** *[line 18]*

requires patErrorManager

make sure that it is in your include path

example_api_save.php

patConfiguration example

\$Id: example_api_save.php 29 2005-03-04 21:25:29Z schst \$

- **Package** patConfiguration
- **Sub-Package** Examples
- **Author** Stephan Schmidt < schst@php-tools.net>

require_once ['./patConfiguration.php'](#) *[line 25]*

require main class

require_once **'pat/patErrorManager.php'** *[line 18]*

requires patErrorManager

make sure that it is in your include path

example_define_auto.php

patConfiguration example

\$Id: example_define_auto.php 29 2005-03-04 21:25:29Z schst \$

- **Package** patConfiguration
- **Sub-Package** Examples
- **Author** Stephan Schmidt < schst@php-tools.net>

require_once ['./patConfiguration.php'](#) *[line 25]*

require main class

require_once **'pat/patErrorManager.php'** *[line 18]*

requires patErrorManager

make sure that it is in your include path

example_define_basic.php

patConfiguration example

\$Id: example_define_basic.php 29 2005-03-04 21:25:29Z schst \$

- **Package** patConfiguration
- **Sub-Package** Examples
- **Author** Stephan Schmidt < schst@php-tools.net>

require_once ['./patConfiguration.php'](#) *[line 25]*

require main class

require_once **'pat/patErrorManager.php'** *[line 18]*

requires patErrorManager

make sure that it is in your include path

example_define_children.php

patConfiguration example

\$Id: example_define_children.php 43 2005-04-05 12:08:26Z schst \$

- **Package** patConfiguration
- **Sub-Package** Examples
- **Author** Stephan Schmidt < schst@php-tools.net>

require_once ['./patConfiguration.php'](#) *[line 25]*

require main class

require_once **'pat/patErrorManager.php'** *[line 18]*

requires patErrorManager

make sure that it is in your include path

example_define_constants.php

patConfiguration example

\$Id: example_define_constants.php 29 2005-03-04 21:25:29Z schst \$

- **Package** patConfiguration
- **Sub-Package** Examples
- **Author** Stephan Schmidt < schst@php-tools.net>

FOO = 54633 *[line 35]*

require_once ['./patConfiguration.php'](#) *[line 25]*

require main class

require_once **'pat/patErrorManager.php'** *[line 18]*

requires patErrorManager

make sure that it is in your include path

example_define_content.php

patConfiguration example

\$Id: example_define_content.php 29 2005-03-04 21:25:29Z schst \$

- **Package** patConfiguration
- **Sub-Package** Examples
- **Author** Stephan Schmidt < schst@php-tools.net>

require_once ['./patConfiguration.php'](#) *[line 25]*

require main class

require_once **'pat/patErrorManager.php'** *[line 18]*

requires patErrorManager

make sure that it is in your include path

example_define_external.php

patConfiguration example

\$Id: example_define_external.php 29 2005-03-04 21:25:29Z schst \$

- **Package** patConfiguration
- **Sub-Package** Examples
- **Author** Stephan Schmidt < schst@php-tools.net>

require_once ['./patConfiguration.php'](#) *[line 25]*

require main class

require_once **'pat/patErrorManager.php'** *[line 18]*

requires patErrorManager

make sure that it is in your include path

example_define_object.php

patConfiguration example

\$Id: example_define_object.php 29 2005-03-04 21:25:29Z schst \$

- **Package** patConfiguration
- **Sub-Package** Examples
- **Author** Stephan Schmidt < schst@php-tools.net>

require_once ['./patConfiguration.php'](#) *[line 43]*

require main class

require_once **'pat/patErrorManager.php'** *[line 36]*

requires patErrorManager

make sure that it is in your include path

example_define_reserved.php

patConfiguration example

\$Id: example_define_reserved.php 29 2005-03-04 21:25:29Z schst \$

- **Package** patConfiguration
- **Sub-Package** Examples
- **Author** Stephan Schmidt < schst@php-tools.net>

require_once ['./patConfiguration.php'](#) *[line 25]*

require main class

require_once **'pat/patErrorManager.php'** *[line 18]*

requires patErrorManager

make sure that it is in your include path

example_formats_ini.php

patConfiguration example

\$Id: example_formats_ini.php 48 2005-04-05 21:27:48Z schst \$

- **Package** patConfiguration
- **Sub-Package** Examples
- **Author** Stephan Schmidt < schst@php-tools.net>

require_once ['./patConfiguration.php'](#) *[line 25]*

require main class

require_once **'pat/patErrorManager.php'** *[line 18]*

requires patErrorManager

make sure that it is in your include path

example_formats_php.php

patConfiguration example

\$Id: example_formats_php.php 29 2005-03-04 21:25:29Z schst \$

- **Package** patConfiguration
- **Sub-Package** Examples
- **Author** Stephan Schmidt < schst@php-tools.net>

require_once ['./patConfiguration.php'](#) *[line 25]*

require main class

require_once **'pat/patErrorManager.php'** *[line 18]*

requires patErrorManager

make sure that it is in your include path

example_formats_wddx.php

patConfiguration example

\$Id: example_formats_wddx.php 48 2005-04-05 21:27:48Z schst \$

- **Package** patConfiguration
- **Sub-Package** Examples
- **Author** Stephan Schmidt < schst@php-tools.net>

require_once ['./patConfiguration.php'](#) *[line 25]*

require main class

require_once **'pat/patErrorManager.php'** *[line 18]*

requires patErrorManager

make sure that it is in your include path

example_formats_xml.php

patConfiguration example

\$Id: example_formats_xml.php 48 2005-04-05 21:27:48Z schst \$

- **Package** patConfiguration
- **Sub-Package** Examples
- **Author** Stephan Schmidt < schst@php-tools.net>

require_once ['./patConfiguration.php'](#) *[line 25]*

require main class

require_once **'pat/patErrorManager.php'** *[line 18]*

requires patErrorManager

make sure that it is in your include path

example_misc_whitespace.php

patConfiguration example

\$Id: example_misc_whitespace.php 29 2005-03-04 21:25:29Z schst \$

- **Package** patConfiguration
- **Sub-Package** Examples
- **Author** Stephan Schmidt < schst@php-tools.net>

require_once ['./patConfiguration.php'](#) *[line 25]*

require main class

require_once **'pat/patErrorManager.php'** *[line 18]*

requires patErrorManager

make sure that it is in your include path

example_misc_xinc.php

patConfiguration example

\$Id: example_misc_xinc.php 29 2005-03-04 21:25:29Z schst \$

- **Package** patConfiguration
- **Sub-Package** Examples
- **Author** Stephan Schmidt < schst@php-tools.net>

require_once ['./patConfiguration.php'](#) *[line 25]*

require main class

require_once **'pat/patErrorManager.php'** *[line 18]*

requires patErrorManager

make sure that it is in your include path

index.php

Main examples dispatcher script that manages the whole examples navigation and display framework via the [patExampleGen](#) class.

\$Id: index.php 29 2005-03-04 21:25:29Z schst \$

- **Package** patConfiguration
- **Sub-Package** Examples
- **Author** Stephan Schmidt < schst@php-tools.net>

include_once '[examples_prepend.php](#)*[line 17]*

The examples prepend file

patConfigurationSource.php

Helper script that displays the source string or file

\$Id: patConfigurationSource.php 31 2005-03-04 21:40:04Z schst \$

You may pass the following GET parameters to the script:

- example => id of the example

- **Package** patConfiguration
- **Sub-Package** Examples
- **Author** Stephan Schmidt

_examples_sections.php

Definitions of all included examples

\$Id: _examples_sections.php 34 2005-03-05 16:55:02Z schst \$

- **Package** patConfiguration
- **Sub-Package** Examples
- **Author** Stephan Schmidt < schst@php-tools.net>

include '[_examples_sections_formats.php](#)'*[line 48]*

include '[_examples_sections_misc.php](#)'*[line 47]*

include '[_examples_sections_define.php](#)'*[line 46]*

include '[_examples_sections_api.php](#)'*[line 45]*

`_examples_sections_api.php`

Definitions for the examples on the patConfiguration API

\$Id: `_examples_sections_api.php` 33 2005-03-05 15:28:08Z schst \$

- **Package** patConfiguration
- **Sub-Package** Examples
- **Author** Stephan Schmidt < schst@php-tools.net>

`_examples_sections_define.php`

Definitions for the examples of the `<define/>` tag

\$Id: `_examples_sections_define.php` 43 2005-04-05 12:08:26Z schst \$

- **Package** patConfiguration
- **Sub-Package** Examples
- **Author** Stephan Schmidt < schst@php-tools.net>

`_examples_sections_formats.php`

Definitions for the examples of the different formats

\$Id: `_examples_sections_formats.php` 34 2005-03-05 16:55:02Z schst \$

- **Package** patConfiguration
- **Sub-Package** Examples
- **Author** Stephan Schmidt < schst@php-tools.net>

`_examples_sections_misc.php`

Definitions for the examples on misc features of patConfiguration

\$Id: _examples_sections_misc.php 34 2005-03-05 16:55:02Z schst \$

- **Package** patConfiguration
- **Sub-Package** Examples
- **Author** Stephan Schmidt < schst@php-tools.net>

Class Foo

[line 22]

- **Package** patConfiguration
- **Sub-Package** Examples

Constructor function Foo::Foo(\$values) *[line 24]*

Function Parameters:

- **\$values**

Class page

[line 19]

dummy class

patConfiguration will instantiate this for you

- **Package** patConfiguration
- **Sub-Package** Examples

Constructor function page::page(\$props) [*line 21*]

Function Parameters:

- **\$props**

Reader.php

patConfiguration reader base class

\$Id: Reader.php 45 2005-04-05 19:48:28Z schst \$

- **Package** patConfiguration
- **Sub-Package** Reader
- **Copyright** PHP Application Tools
- **Link** <http://www.php-tools.net>
- **Author** Stephan Schmidt < schst@php-tools.net>

INI.php

patConfiguration reader for INI files

used by the patConfiguration object to read INI config files

\$Id: INI.php 22 2005-02-05 15:50:56Z schst \$

- **Package** patConfiguration
- **Sub-Package** Reader
- **Author** Stephan Schmidt < schst@php-tools.net>

WDDX.php

patConfiguration WDDX Reader

used by the patConfiguration object to read WDDX config files

\$Id: WDDX.php 22 2005-02-05 15:50:56Z schst \$

- **Package** patConfiguration
- **Sub-Package** Reader
- **Author** Stephan Schmidt < schst@php-tools.net>

XML.php

patConfiguration XML Reader

used by the patConfiguration object to read XML config files

\$Id: XML.php 43 2005-04-05 12:08:26Z schst \$

- **Package** patConfiguration
- **Sub-Package** Reader
- **Author** Stephan Schmidt < schst@php-tools.net>

PATCONFIGURATION_READER_XML_ERROR_RESERVED_TAG = 'PCF::R::XML::1' [*line 17*]

Trying to define reserved tag

Class patConfiguration_Reader

[*line 23*]

patConfiguration reader base class

\$Id: Reader.php 45 2005-04-05 19:48:28Z schst \$

- **Package** patConfiguration
- **Sub-Package** Reader
- **Author** Stephan Schmidt < schst@php-tools.net>

patConfiguration_Reader::\$configObj

object patConfiguration = NULL [*line 29*]

reference to patConfiguration object

array function patConfiguration_Reader::loadConfigFile(\$configFile, [\$options = array()]) [*line 51*]

Function Parameters:

- *string* **\$configFile** full path of the config file
- *array* **\$options** various options, depending on the reader

load configuration from a file

- **Abstract Element**
- **Access** public

function patConfiguration_Reader::setConfigReference(&\$config) [*line 37*]

Function Parameters:

- *object* **&\$config**

set reference to the patConfiguration object

- **Access** public

Class patConfiguration_Reader_INI

[*line 25*]

patConfiguration reader for INI files

used by the patConfiguration object to read INI config files

\$Id: INI.php 22 2005-02-05 15:50:56Z schst \$

- **Package** patConfiguration
- **Sub-Package** Reader
- **Author** Stephan Schmidt < schst@php-tools.net>

array function patConfiguration_Reader_INI::loadConfigFile(\$configFile, [\$options = array()]) [*line 35*]

Function Parameters:

- *string* **\$configFile** full path of the config file
- *array* **\$options** various options, depending on the reader

load configuration from a file

- **Access** public

Class patConfiguration_Reader_WDDX

[*line 25*]

patConfiguration WDDX Reader

used by the patConfiguration object to read WDDX config files

\$Id: WDDX.php 22 2005-02-05 15:50:56Z schst \$

- **Package** patConfiguration
- **Sub-Package** Reader
- **Author** Stephan Schmidt < schst@php-tools.net>

array function patConfiguration_Reader_WDDX::loadConfigFile(\$configFile, [\$options = array()]) *[line 36]*

Function Parameters:

- *string* **\$configFile** full path of the config file
- *array* **\$options** various options, depending on the reader

load configuration from a file

- **Access** public

Class patConfiguration_Reader_XML

[line 30]

patConfiguration XML Reader

used by the patConfiguration object to read XML config files

\$Id: XML.php 43 2005-04-05 12:08:26Z schst \$

- **Package** patConfiguration
- **Sub-Package** Reader
- **Author** Stephan Schmidt < schst@php-tools.net >

patConfiguration_Reader_XML::\$conf

array = array() [line 42]

array that stores configuration from the current file

patConfiguration_Reader_XML::\$currentNamespace

string = '_none' [line 96]

current namespace for define

patConfiguration_Reader_XML::\$currentTag

string = false [line 102]

current tag for define

patConfiguration_Reader_XML::\$data

string = array() [line 78]

current CDATA found

patConfiguration_Reader_XML::\$defineStack

array = array() [line 108]

stack for define tags

patConfiguration_Reader_XML::\$externalFiles

array = array() [line 120]

list of all files that were needed

patConfiguration_Reader_XML::\$includedFiles

array = array() [line 114]

files that have been included

patConfiguration_Reader_XML::\$nsStack

array = array() [line 54]

stack of the namespaces

patConfiguration_Reader_XML::\$parsers

array = array() [line 48]

array that stores all xml parsers

patConfiguration_Reader_XML::\$path

```
array = array() [line 36]
```

store path information

patConfiguration_Reader_XML::\$reservedTags

```
array = array('configuration', 'configValue', 'define', 'getConfigValue', 'path') [line 126]
```

reserved tags

patConfiguration_Reader_XML::\$tagStack

```
array = array() [line 60]
```

stack for tags that have been found

patConfiguration_Reader_XML::\$valDepth

```
int = 1 [line 72]
```

current depth of the stored values, i.e. array depth

patConfiguration_Reader_XML::\$valStack

```
array = array() [line 66]
```

stack of values

patConfiguration_Reader_XML::\$whitespace

```
string = array('default') [line 90]
```

treatment of whitespace

patConfiguration_Reader_XML::\$xmlFiles

```
array = array() [line 84]
```

all open files

```
function patConfiguration_Reader_XML::appendData($data) [line 608]
```

Function Parameters:

- *mixed* **\$data** data to be appended

append Data to the current data

function patConfiguration_Reader_XML::characterData(\$parser, \$data) [line 545]

Function Parameters:

- *int* **\$parser** resource id of the current parser
- *string* **\$data** character data, that was found

handle character data if the character data was found between tags using namespaces, the appropriate namespace handler will be called

object function patConfiguration_Reader_XML::createParser() [line 1044]

create a parser

function patConfiguration_Reader_XML::endElement(\$parser, \$name) [line 388]

Function Parameters:

- *resource* **\$parser** resource id of the current parser
- *string* **\$name** name of the element

handle end element

if the end element contains a namespace calls the appropriate handler

function patConfiguration_Reader_XML::externalEntity(\$parser, \$openEntityNames, \$base, \$systemId, \$publicId) [line 860]

Function Parameters:

- *int* **\$parser** resource id of the current parser
- *string* **\$openEntityNames** space-separated list of the names of the entities that are open for the parse of this entity (including the name of the referenced entity)
- *string* **\$base** currently empty string
- *string* **\$systemId** system identifier as specified in the entity declaration
- *string* **\$publicId** publicId, is the public identifier as specified in the entity declaration, or an empty string if none was specified; the whitespace in the public identifier will have been normalized as required by the XML spec

parse an external entity

mixed function patConfiguration_Reader_XML::getConfigValue([\$path = "]) [*line 913*]

Function Parameters:

- *string* **\$path** path, where the value is stored

returns a configuration value

if no path is given, all config values will be returned in an array

- **Access** public

array function patConfiguration_Reader_XML::loadConfigFile(\$configFile, [\$options = array()]) [*line 136*]

Function Parameters:

- *string* **\$configFile** full path of the config file
- *array* **\$options** various options, depending on the reader

load configuration from a file

- **Access** public

boolean function patConfiguration_Reader_XML::parseXMLFile(\$file) [*line 953*]

Function Parameters:

- *string* **\$file** filename, without dirname

parse an external xml file

function patConfiguration_Reader_XML::setDefinedTags(\$tags, [\$ns = null]) [*line 160*]

Function Parameters:

- *array* **\$tags**
- **\$ns**

set defined tags

- **Access public**

function patConfiguration_Reader_XML::startElement(\$parser, \$name, \$attributes) [*line 178*]

Function Parameters:

- *resource* **\$parser** resource id of the current parser
- *string* **\$name** name of the element
- *array* **\$attributes** array containing all attributes of the element

handle start element if the start element contains a namespace calls the appropriate handler

autopackage.php

package.xml generation file for patConfiguration

This file is executed by createSnaps.php to automatically create a package that can be installed via the PEAR installer.

\$Id: autopackage.php 54 2006-02-03 08:01:11Z argh \$

- **Package** patConfiguration
- **Sub-Package** Tools
- **Author** Stephan Schmidt < schst@php-tools.net>

require_once 'PEAR/PackageFileManager.php' [*line 21*]

uses PackageFileManager

package.php

package.xml generation file for patConfiguration

This file is executed by createSnaps.php to automatically create a package that can be installed via the PEAR installer.

\$Id: package.php 54 2006-02-03 08:01:11Z argh \$

- **Package** patConfiguration
- **Sub-Package** Tools
- **Author** Stephan Schmidt < schst@php-tools.net>

require_once 'PEAR/PackageFileManager.php' [*line 21*]

uses PackageFileManager

Writer.php

patConfiguration Writer

abstract base class for config writers

\$Id: Writer.php 35 2005-03-06 10:01:02Z schst \$

- **Package** patConfiguration
- **Sub-Package** Writer
- **Author** Stephan Schmidt < schst@php-tools.net >
- **Abstract Element**

INI.php

patConfiguration reader for INI files

used by the patConfiguration object to write INI config files

\$Id: INI.php 22 2005-02-05 15:50:56Z schst \$

- **Package** patConfiguration
- **Sub-Package** Writer
- **Author** Stephan Schmidt < schst@php-tools.net>

PHP.php

patConfiguration writer for PHP files

used by the patConfiguration object to write PHP config files

\$Id: PHP.php 23 2005-02-05 16:06:09Z schst \$

- **Package** patConfiguration
- **Sub-Package** Writer
- **Author** Stephan Schmidt < schst@php-tools.net>

WDDX.php

patConfiguration writer for WDDX files

used by the patConfiguration object to write INI config files

\$Id: WDDX.php 23 2005-02-05 16:06:09Z schst \$

- **Package** patConfiguration
- **Sub-Package** Writer
- **Author** Stephan Schmidt < schst@php-tools.net>

XML.php

patConfiguration writer for XML files

used by the patConfiguration object to write XML config files

\$Id: XML.php 23 2005-02-05 16:06:09Z schst \$

- **Package** patConfiguration
- **Sub-Package** Writer
- **Author** Stephan Schmidt < schst@php-tools.net>

Class patConfiguration_Writer

[line 26]

patConfiguration Writer

abstract base class for config writers

\$Id: Writer.php 35 2005-03-06 10:01:02Z schst \$

- **Package** patConfiguration
- **Sub-Package** Writer
- **Author** Stephan Schmidt < schst@php-tools.net>

patConfiguration_Writer::\$configOb

object patConfiguration = NULL [line 32]

reference to patConfiguration object

string function patConfiguration_Writer::serializeConfig(\$config, \$options) [line 54]

Function Parameters:

- *array* **\$config** config to serialize
- *array* **\$options** options for the serialization

serialize the config

- **Access** public
- **Abstract Element**

function patConfiguration_Writer::setConfigReference(&\$config) [*line 40*]

Function Parameters:

- [object patConfiguration](#) **&\$config**

set reference to the patConfiguration object

- **Access** public

Class patConfiguration_Writer_INI

[*line 25*]

patConfiguration writer for INI files

used by the patConfiguration object to write INI config files

\$Id: INI.php 22 2005-02-05 15:50:56Z schst \$

- **Package** patConfiguration
- **Sub-Package** Writer
- **Author** Stephan Schmidt < schst@php-tools.net>

Class patConfiguration_Writer_PHP

[line 25]

patConfiguration writer for PHP files

used by the patConfiguration object to write PHP config files

\$Id: PHP.php 23 2005-02-05 16:06:09Z schst \$

- **Package** patConfiguration
- **Sub-Package** Writer
- **Author** Stephan Schmidt < schst@php-tools.net>

string function patConfiguration_Writer_PHP::serializeConfig(\$config, \$options) [line 36]

Function Parameters:

- *array* **\$config** config to serialize
- *array* **\$options** options for the serialization

create a php representation of the current config

- **Access** public

Class patConfiguration_Writer_WDDX

[line 25]

patConfiguration writer for WDDX files

used by the patConfiguration object to write INI config files

\$Id: WDDX.php 23 2005-02-05 16:06:09Z schst \$

- **Package** patConfiguration
- **Sub-Package** Writer
- **Author** Stephan Schmidt < schst@php-tools.net>

patConfiguration_Writer_WDDX::\$xmlSpecialchars

```
array = array(  
    "&" => "&",  
    "'" => "&apos;",  
    "\" => "\"",  
    "<" => "<",  
    ">" => ">"  
)  
[line 31]
```

table used for translation of xml special chars

string function patConfiguration_Writer_WDDX::replaceXMLSpecialchars(\$string, [\$table = array()]) [line 82]

Function Parameters:

- *string* **\$string** string, where special chars should be replaced
- *array* **\$table** table used for replacing

replace XML special chars

Class patConfiguration_Writer_XML

[line 25]

patConfiguration writer for XML files

used by the patConfiguration object to write XML config files

\$Id: XML.php 23 2005-02-05 16:06:09Z schst \$

- **Package** patConfiguration
- **Sub-Package** Writer
- **Author** Stephan Schmidt < schst@php-tools.net>

patConfiguration_Writer_XML::\$openTags

```
array = array() [line 43]
```

open tags

patConfiguration_Writer_XML::\$xmlSpecialchars

```
array = array(  
    "&" => "&",  
    "'" => "&apos;",  
    "\"" => "\"",  
    "<" => "<",  
    ">" => ">"  
)  
[line 31]
```

table used for translation of xml special chars

string function patConfiguration_Writer_XML::replaceXMLSpecialchars(\$string, [\$table = array()]) [line 214]

Function Parameters:

- *string* **\$string** string, where special chars should be replaced
- *array* **\$table** table used for replacing

replace XML special chars

Appendices

Appendix A - Class Trees

Package patConfiguration

Foo

- [Foo](#)

page

- [page](#)

patConfiguration

- [patConfiguration](#)

patConfiguration_Reader

- [patConfiguration Reader](#)
 - [patConfiguration Reader INI](#)
 - [patConfiguration Reader WDDX](#)
 - [patConfiguration Reader XML](#)

patConfiguration_Writer

- [patConfiguration Writer](#)
 - [patConfiguration Writer INI](#)
 - [patConfiguration Writer PHP](#)
 - [patConfiguration Writer WDDX](#)
 - [patConfiguration Writer XML](#)

Package patExampleGen

patExampleGen

- [patExampleGen](#)

Index

A

autopackage.php	71
<i>package.xml generation file for patConfiguration</i>	

C

constructor page::page()	58
constructor Foo::Foo()	57
constructor patConfiguration::__construct()	22
<i>constructor for PHP 5</i>	
constructor patConfiguration::patConfiguration()	21
<i>constructor</i>	

E

example define_reserved.php	44
<i>patConfiguration example</i>	
example define_object.php	43
<i>patConfiguration example</i>	
example define_external.php	42
<i>patConfiguration example</i>	
example define_content.php	41
<i>patConfiguration example</i>	
example formats_ini.php	45
<i>patConfiguration example</i>	
example formats_php.php	46
<i>patConfiguration example</i>	
example misc_xinc.php	50
<i>patConfiguration example</i>	
example misc_whitespace.php	49
<i>patConfiguration example</i>	
example formats_xml.php	48
<i>patConfiguration example</i>	
example formats_wddx.php	47
<i>patConfiguration example</i>	
example define_constants.php	40
<i>patConfiguration example</i>	
example define_children.php	39
<i>patConfiguration example</i>	
example api_create.php	33
<i>patConfiguration example</i>	
example api_cache_option.php	32
<i>patConfiguration example</i>	

example_api_cache.php	31
<i>patConfiguration example</i>	
example_api_basic.php	30
<i>patConfiguration example</i>	
example_api_errors.php	34
<i>patConfiguration example</i>	
example_api_objects.php	35
<i>patConfiguration example</i>	
example_define_basic.php	38
<i>patConfiguration example</i>	
example_define_auto.php	37
<i>patConfiguration example</i>	
example_api_save.php	36
<i>patConfiguration example</i>	
example_api_array.php	29
<i>patConfiguration example</i>	

F

Foo	57
FOO	40
fileUsage.php	13
<i>Helper file for the examples collection that displays a list of example files that may be missing or are not referenced to in the examples section definitions.</i>	

I

INI.php	74
<i>patConfiguration reader for INI files</i>	
INI.php	60
<i>patConfiguration reader for INI files</i>	
index.php	51
<i>Main examples dispatcher script that manages the whole examples navigation and display framework via the patExampleGen class.</i>	

P

patConfiguration_Reader_XML::\$currentTag	66
<i>current tag for define</i>	
patConfiguration_Reader_XML::\$data	66
<i>current CDATA found</i>	
patConfiguration_Reader_XML::\$currentNamespace	65
<i>current namespace for define</i>	
patConfiguration_Reader_XML::\$conf	65
<i>array that stores configuration from the current file</i>	
patConfiguration_Reader_WDDX::loadConfigFile()	65
<i>load configuration from a file</i>	
patConfiguration_Reader_XML	65

patConfiguration XML Reader	66
patConfiguration Reader XML::\$defineStack	66
<i>stack for define tags</i>	
patConfiguration Reader XML::\$externalFiles	66
<i>list of all files that were needed</i>	
patConfiguration Reader XML::\$reservedTags	67
<i>reserved tags</i>	
patConfiguration Reader XML::\$tagStack	67
<i>stack for tags that have been found</i>	
patConfiguration Reader XML::\$path	67
<i>store path information</i>	
patConfiguration Reader XML::\$parsers	66
<i>array that stores all xml parsers</i>	
patConfiguration Reader XML::\$includedFiles	66
<i>files that have been included</i>	
patConfiguration Reader XML::\$nsStack	66
<i>stack of the namespaces</i>	
patConfiguration Reader WDDX	64
<i>patConfiguration WDDX Reader</i>	
patConfiguration Reader INI::loadConfigFile()	64
<i>load configuration from a file</i>	
patConfiguration::writeConfigFile()	28
<i>write a configfile</i>	
patConfigurationSource.php	52
<i>Helper script that displays the source string or file</i>	
patConfiguration::setOption()	27
<i>set any option</i>	
patConfiguration::setEncoding()	27
<i>set the XML encoding</i>	
patConfiguration::setConfigValues()	26
<i>sets several config values</i>	
patConfiguration::setDriverDir()	27
<i>set the directory, where all drivers are stored</i>	
page	57
<i>dummy class</i>	
PATCONFIGURATION READER XML ERROR RESERVED TAG	62
<i>Trying to define reserved tag</i>	
patConfiguration Reader::setConfigReference()	63
<i>set reference to the patConfiguration object</i>	
patConfiguration Reader INI	63
<i>patConfiguration reader for INI files</i>	
patConfiguration Reader::loadConfigFile()	62
<i>load configuration from a file</i>	
patConfiguration Reader::\$configObj	62
<i>reference to patConfiguration object</i>	
patConfiguration Reader	62
<i>patConfiguration reader base class</i>	
patConfiguration Reader XML::\$valDepth	67
<i>current depth of the stored values, i.e. array depth</i>	
patConfiguration Reader XML::\$valStack	67
<i>stack of values</i>	
patConfiguration Writer PHP	79
<i>patConfiguration writer for PHP files</i>	

patConfiguration Writer PHP::serializeConfig()	79
<i>create a php representation of the current config</i>	
patConfiguration Writer INI	78
<i>patConfiguration writer for INI files</i>	
patConfiguration Writer::setConfigReference()	78
<i>set reference to the patConfiguration object</i>	
patConfiguration Writer::\$configOb	77
<i>reference to patConfiguration object</i>	
patConfiguration Writer::serializeConfig()	77
<i>serialize the config</i>	
patConfiguration Writer WDDX	80
<i>patConfiguration writer for WDDX files</i>	
patConfiguration Writer WDDX::\$xmlSpecialchars	80
<i>table used for translation of xml special chars</i>	
patConfiguration Writer XML::\$xmlSpecialchars	81
<i>table used for translation of xml special chars</i>	
patConfiguration Writer XML::replaceXMLSpecialchars()	81
<i>replace XML special chars</i>	
patConfiguration Writer XML::\$openTags	81
<i>open tags</i>	
patConfiguration Writer XML	81
<i>patConfiguration writer for XML files</i>	
patConfiguration Writer WDDX::replaceXMLSpecialchars()	80
<i>replace XML special chars</i>	
patConfiguration Writer	77
<i>patConfiguration Writer</i>	
PHP.php	75
<i>patConfiguration writer for PHP files</i>	
patConfiguration Reader XML::createParser()	68
<i>create a parser</i>	
patConfiguration Reader XML::endElement()	68
<i>handle end element</i>	
patConfiguration Reader XML::characterData()	68
<i>handle character data</i>	
<i>if the character data was found between tags using namespaces, the appropriate namespace handler will be called</i>	
patConfiguration Reader XML::appendData()	67
<i>append Data to the current data</i>	
patConfiguration Reader XML::\$whitespace	67
<i>treatment of whitespace</i>	
patConfiguration Reader XML::\$xmlFiles	67
<i>all open files</i>	
patConfiguration Reader XML::externalEntity()	68
<i>parse an external entity</i>	
patConfiguration Reader XML::getConfigValue()	69
<i>returns a configuration value</i>	
patConfiguration Reader XML::startElement()	70
<i>handle start element</i>	
<i>if the start element contains a namespace calls the appropriate handler</i>	
package.php	72
<i>package.xml generation file for patConfiguration</i>	
patConfiguration Reader XML::setDefinedTags()	69
<i>set defined tags</i>	

patConfiguration_Reader_XML::parseXMLFile()	69
<i>parse an external xml file</i>	
patConfiguration_Reader_XML::loadConfigFile()	69
<i>load configuration from a file</i>	
patConfiguration::setConfigValue()	26
<i>set a config value</i>	
patConfiguration::setConfigDir()	26
<i>set the directory, where all config files are stored</i>	
patExampleGen::getTabs()	9
<i>Get the tabs for an example (per example tabs can be modified/added).</i>	
patExampleGen::highlightXML()	9
<i>Highlights an xml string and returns the highlighted source</i>	
patExampleGen::getRequestVars()	9
<i>Retrieves the request variables from POST and GET</i>	
patExampleGen::getExampleSection()	8
<i>get the example section</i>	
patExampleGen::getDefaultTab()	8
<i>Retrieves the default example view tab, the one that is active when the page is loaded.</i>	
patExampleGen::getExample()	8
<i>Get an example's details</i>	
patExampleGen::process()	9
<i>Main process method that dispatches needed tasks according to the given action.</i>	
patExampleGen::setAppDescription()	10
<i>Sets the description of the application</i>	
patExampleGen::setTabs()	11
<i>Sets the default tab configuration that will be used for each example page. Additional tabs can be defined for each example within the section definition.</i>	
patExampleGen::url_get_contents()	12
<i>Simple url content loader</i>	
patExampleGen::setSections()	11
<i>Sets the sections from which to generate the examples framework</i>	
patExampleGen::setAppName()	10
<i>Sets the name of the application</i>	
patExampleGen::setAppForumId()	10
<i>Sets the ID of the application's forum. Needed to display the correct link to the forum.</i>	
patExampleGen::file_get_contents()	7
<i>Replacement for file_get_contents</i>	
patExampleGen::displayXMLString()	7
<i>Hilights an xml string and displays it</i>	
patExampleGen::displayFrameset()	4
<i>Displays the examples main frameset</i>	
patExampleGen::displayGuide()	4
patExampleGen::displayFooter()	4
<i>Displays the footer for any example page</i>	
patExampleGen::displayExample()	4
<i>Displays an example page along with all defined tabs</i>	
patExampleGen	3
<i>The pat examples framework class - manages the navigation and functionality of the examples collection of our tools.</i>	

patExampleGen::displayError()	3
<i>Custom error handler that is set automatically to display any patError error objects within the examples collection.</i>	
patExampleGen::displayHead()	5
<i>Displays the header for any example page</i>	
patExampleGen::displayNavigation()	5
<i>Displays the main examples navigation</i>	
patExampleGen::displayText()	6
<i>Display a free text in a tab</i>	
patExampleGen::displayXMLSource()	6
<i>Display XML source code in an example tab</i>	
patExampleGen::displayPHPSource()	6
<i>Display PHP source code in an example tab</i>	
patExampleGen::displayOverview()	6
<i>Displays the examples overview page</i>	
patExampleGen::displayOutput()	5
<i>Display an example's output in an example tab via an iframe</i>	
patConfiguration.php	18
<i>patConfiguration</i>	
PATCONFIGURATION_ERROR_CACHEDIR_NOT_WRITEABLE	18
<i>cache file could not be written</i>	
patConfiguration::getConfigDir()	22
<i>get the directory, where all config files are stored</i>	
patConfiguration::getConfigValue()	23
<i>returns a configuration value</i>	
patConfiguration::clearConfigValue()	22
<i>clears a config value</i>	
patConfiguration::\$systemVars	21
<i>information about the project</i>	
patConfiguration::\$drivers	21
<i>drivers that have already been loaded</i>	
patConfiguration::\$encoding	21
<i>default encoding</i>	
patConfiguration::getDriverDir()	23
<i>get driver directory</i>	
patConfiguration::getFullPath()	23
<i>get the full path of a file</i>	
patConfiguration::parseConfigFile()	25
<i>parse a configuration file</i>	
patConfiguration::setCacheDir()	25
<i>set the directory, where all cache files are stored</i>	
patConfiguration::loadConfig()	24
<i>load a configuration</i>	
patConfiguration::loadCachedConfig()	24
<i>load a configuration from a cache</i>	
patConfiguration::getOption()	24
<i>get an option</i>	
patConfiguration::\$driverDir	21
<i>directory where drivers are stored</i>	
patConfiguration::\$configDir	20
<i>directory, where the configuration files are stored</i>	
PATCONFIGURATION_ERROR_DRIVER_NOT_WORKING	19
<i>driver is not working</i>	

PATCONFIGURATION ERROR FILE NOT FOUND	19
<i>File not found</i>	
PATCONFIGURATION ERROR DRIVERDIR NOT FOUND	19
<i>config directory does not exist</i>	
PATCONFIGURATION ERROR CORRUPT DRIVER	18
<i>Driver file does not contain driver class</i>	
PATCONFIGURATION ERROR CONFIGDIR NOT FOUND	18
<i>config directory does not exist</i>	
PATCONFIGURATION ERROR CONFIG INVALID	18
<i>config file has invalid syntax</i>	
PATCONFIGURATION ERROR FILE NOT READABLE	19
<i>File not readable</i>	
PATCONFIGURATION ERROR FILE NOT WRITABLE	19
<i>configuration file could not be written</i>	
patConfiguration::\$cacheDir	20
<i>directory where cache files are located</i>	
patConfiguration::\$conf	20
<i>array that stores configuration</i>	
patConfiguration	20
<i>patConfiguration</i>	
PATCONFIGURATION WARNING CONFIGVALUE WITHOUT NAME	19
<i>config value without a name</i>	
PATCONFIGURATION ERROR UNKNOWN DRIVER	19
<i>Driver file not found</i>	
patExampleGen.php	2
<i>The pat examples framework class - manages the navigation and functionality of the examples collection of our tools.</i>	

R

Reader.php	59
<i>patConfiguration reader base class</i>	

W

WDDX.php	76
<i>patConfiguration writer for WDDX files</i>	
Writer.php	73
<i>patConfiguration Writer</i>	
WDDX.php	61
<i>patConfiguration WDDX Reader</i>	

X

XML.php	77
<i>patConfiguration writer for XML files</i>	
XML.php	62
<i>patConfiguration XML Reader</i>	

_examples_sections_define.php	55
<i>Definitions for the examples of the <define/> tag</i>	
_examples_sections_formats.php	56
<i>Definitions for the examples of the different formats</i>	
_examples_sections_misc.php	57
<i>Definitions for the examples on misc features of patConfiguration</i>	
_examples_sections_api.php	54
<i>Definitions for the examples on the patConfiguration API</i>	
_examples_sections.php	53
<i>Definitions of all included examples</i>	
_examples_customFunctions.php	15
<i>Custom functions for the current examples collection - the functions defined here are available for all examples.</i>	
_examples_prepend.php	16
<i>Main prepend file for the examples framework, sets error reporting and initializes the patExampleGen class that manages the whole framework.</i>	
_examples_config.php	14
<i>Configuration for the examples - edit this file to set the paths to the needed classes as they should be on your server.</i>	