

patSession



Contents

Package patSessoin Procedural Elements	2
package.php	2
Package patSession Procedural Elements	4
startsession.php	4
example secure fork.php	5
patSession.php	6
Package patSession Classes	7
Class patSession	7
Class Constant ERROR_DRIVER_NOT_FOUND	7
Class Constant ERROR_NOT_ACTIVE	7
Class Constant ERROR_NOT_DESTROYED	7
Method create	8
Method setDriverDir	8
Method singleton	8
example basic count.php	10
example basic customdriver.php	11
example basic destroy.php	12
example basic factory.php	13
example basic first.php	14
example basic namespace.php	15
example basic options.php	16
example basic parameter.php	17
example basic restart.php	18
example basic singleton.php	19
example secure expire.php	20
example secure fix.php	21
example secure fixbrowser.php	22
example secure fixip.php	23
example secure fixreferer.php	24
example secure useToken.php	25
example spl array.php	26
example spl iterator.php	27
example spl property.php	28
example test cli.php	29
example test driver.php	30
example test null.php	31
example test toString.php	32
index sections.php	33
_error.php	34
_viewExample.php	35
Function buildNavigation	35
Function getExample	35
Function getExampleSection	36

Joker.php	37
Storage.php	38
Cli.php	39
Native.php	40
Null.php	41
Class patSession Storage	41
Var \$ allowedReferer	42
Var \$ cookieParams	42
Var \$ emptyReferer	42
Var \$ expire	43
Var \$ id	43
Var \$ idPrefix	43
Var \$ security	43
Var \$ sess	44
Var \$ sessAttributes	44
Var \$ state	44
Var \$ tokenMode	45
Var \$ tokenName	45
Var \$ tokenValue	45
Constructor construct	45
Method clear	46
Method current	46
Method destroy	46
Method fork	47
Method get	47
Method getContainerId	47
Method getCounter	47
Method getId	48
Method getName	48
Method getQueryString	48
Method getSID	48
Method getState	49
Method getTimer	49
Method has	49
Method isNew	49
Method key	50
Method next	50
Method offsetExists	50
Method offsetGet	51
Method offsetSet	51
Method offsetUnset	51
Method restart	52
Method rewind	52
Method set	52
Method valid	53
Method checkExpire	53
Method checkFixBrowser	53
Method checkFixIp	53
Method checkReferer	54

Method checkSecurity	54
Method checkToken	54
Method createId	55
Method initCounter	55
Method setOptions	55
Method start	56
Method suspend	56
Method destruct	56
Method get	56
Method toString	57
Class patSession Storage Cli	57
Method setOptions	58
Method start	58
Method suspend	58
Class patSession Storage Joker	58
Class Constant STORAGE DIR	59
Var \$ sessFile	59
Var \$ sessId	59
Var \$ sessName	59
Method handleFork	60
Method start	60
Method suspend	60
Class patSession Storage Native	61
Var \$ sessId	61
Var \$ sessName	61
Method handleFork	61
Method setOptions	62
Method start	62
Method suspend	62
Class patSession Storage Null	63
Method setOptions	63
Method start	63
Method suspend	64
Package patTemplate Procedural Elements	66
index.php	66
index_main.php	67
index_nav.php	68
autopackage2.php	69
Appendices	70
Appendix A - Class Trees	71
patSession	71
patTemplate	71
patSessoin	71
Appendix D - Todo List	72

Package patSessoin Procedural Elements

package.php

package.xml generation file for patSession

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

\$Id\$

- **Package** patSessoin
- **Sub-Package** Tools
- **Author** Stephan Schmidt < schst@php-tools.net>
- **Author** gERD Schaufelberger < gerd@php-tools.net>

include **__FILE__** *[line 23]*

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

require_once **'PEAR/PackageFileManager2.php'** *[line 20]*

uses PackageFileManager Version 2

Package patSession Procedural Elements

startsession.php

- **Package** patSession

include_once **\$incPath.'/patSession.php'** [*line 19*]

include_once **\$incPath.'../patError/patErrorManager.php'** [*line 15*]

Test the CLI-Storage driver

- **Copyright** 2003
- **Version** \$Id: startsession.php 26 2004-11-03 07:12:01Z gerd \$

example_secure_fork.php

- **Package** patSession
- **License** LGPL
- **Copyright** 2004 <http://www.php-tools.net>
- **Author** gERD Schaufelberger < gerd@php-tools.net>
- **Version** \$Id: example_secure_fork.php 46 2006-03-15 22:15:57Z gerd \$

include_once ['./patSession.php'](#)*[line 14]*

include_once ['./ error.php'](#)*[line 11]*

patSession.php

patSession

\$Id: patSession.php 53 2007-03-23 18:10:56Z gerd \$

- **Package** patSession
- **License** LGPL
- **Copyright** 2004-2006 <http://www.php-tools.net>
- **Author** gERD Schaufelberger < gerd@php-tools.net>
- **Version** 2.0.2

Package patSession Classes

Class patSession

[line 20]

patSession allows easy access to session values

- usable in static or object mode
- support for singleton

- **Package** patSession

patSession::ERROR_DRIVER_NOT_FOUND

= 2 *[line 30]*

error definition: could not load driver

patSession::ERROR_NOT_ACTIVE

= 1 *[line 25]*

error definition: session not active

patSession::ERROR_NOT_DESTROYED

= 3 *[line 35]*

error definition: session not destroyed

object function patSession::create([\$sid = 'session'], [\$storage = 'Native'], [\$options = array()]) *[line 118]*

Function Parameters:

- *string* **\$sid** name-prefix used for internal storage of session-data
- *string* **\$storage** named storage driver
- *array* **\$options** optional parameter passed to the storage driver

factory Allows to create multiple instances of patSession Each instance must differ in name of prefix, otherwise this function return a previously created object.

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

int function patSession::setDriverDir(\$dir, [\$add = true]) *[line 66]*

Function Parameters:

- *string* **\$dir** of custom drivers
- *bool* **\$add**

Custom driver dir to include your own classes

Set or add a folder as custom driver directory

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

object function patSession::singleton([\$sid = 'session'], [\$storage = 'Native'], [\$options = array()]) *[line 87]*

Function Parameters:

- *string* **\$sid** name-prefix used for internal storage of session-data
- *string* **\$storage** named storage driver
- *array* **\$options** optional parameter passed to the storage driver

Singleton create a single instance of patSession driver

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

example_basic_count.php

example_basic_counter.php

- **Package** patSession
- **Sub-Package** Examples
- **License** LGPL
- **Copyright** 2004 <http://www.php-tools.net>
- **Author** gERD Schaufelberger < gerd@php-tools.net>
- **Version** \$Id: example_basic_count.php 49 2006-09-05 21:00:01Z gerd \$

include_once ['./patSession.php'](#)*[line 18]*

include_once ['./ error.php'](#)*[line 15]*

example_basic_customdriver.php

example_basic

- **Package** patSession
- **Sub-Package** Examples
- **License** LGPL
- **Copyright** 2004 <http://www.php-tools.net>
- **Author** gERD Schaufelberger < gerd@php-tools.net>
- **Version** \$Id\$

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

include_once ['./ error.php'](#)*[line 15]*

example_basic_destroy.php

example_basic_destroy.php

- **Package** patSession
- **Sub-Package** Examples
- **License** LGPL
- **Copyright** 2004 <http://www.php-tools.net>
- **Author** gERD Schaufelberger < gerd@php-tools.net>
- **Version** \$Id: example_basic_destroy.php 46 2006-03-15 22:15:57Z gerd \$

include_once ['./patSession.php'](#)*[line 18]*

include_once ['./ error.php'](#)*[line 15]*

example_basic_factory.php

example_basic_factory.php

- **Package** patSession
- **Sub-Package** Examples
- **License** LGPL
- **Copyright** 2004 <http://www.php-tools.net>
- **Author** gERD Schaufelberger < gerd@php-tools.net>
- **Version** \$Id: example_basic_factory.php 46 2006-03-15 22:15:57Z gerd \$

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

include_once ['./ error.php'](#)*[line 15]*

example_basic_first.php

example_basic_first.php

- **Package** patSession
- **Sub-Package** Examples
- **License** LGPL
- **Copyright** 2004 <http://www.php-tools.net>
- **Author** gERD Schaufelberger < gerd@php-tools.net>
- **Version** \$Id: example_basic_first.php 46 2006-03-15 22:15:57Z gerd \$

include_once ['./patSession.php'](#)*[line 18]*

include_once ['./ error.php'](#)*[line 15]*

example_basic_namespace.php

example_basic_options.php

- **Package** patSession
- **Sub-Package** Examples
- **License** LGPL
- **Copyright** 2004 <http://www.php-tools.net>
- **Author** gERD Schaufelberger < gerd@php-tools.net>
- **Version** \$Id: example_basic_namespace.php 46 2006-03-15 22:15:57Z gerd \$

include_once ['./patSession.php'](#)*[line 18]*

include_once ['./ error.php'](#)*[line 15]*

example_basic_options.php

example_basic_options.php

- **Package** patSession
- **Sub-Package** Examples
- **License** LGPL
- **Copyright** 2004 <http://www.php-tools.net>
- **Author** gERD Schaufelberger < gerd@php-tools.net>
- **Version** \$Id: example_basic_options.php 46 2006-03-15 22:15:57Z gerd \$

include_once ['./patSession.php'](#)*[line 18]*

include_once ['./ error.php'](#)*[line 15]*

example_basic_parameter.php

example_basic_parameter.php

- **Package** patSession
- **Sub-Package** Examples
- **License** LGPL
- **Copyright** 2004 <http://www.php-tools.net>
- **Author** gERD Schaufelberger < gerd@php-tools.net>
- **Version** \$Id: example_basic_parameter.php 46 2006-03-15 22:15:57Z gerd \$

include_once ['./patSession.php'](#)[line 17]

include_once ['./ error.php'](#)[line 15]

example_basic_restart.php

example_basic_counter.php

- **Package** patSession
- **Sub-Package** Examples
- **License** LGPL
- **Copyright** 2004 <http://www.php-tools.net>
- **Author** gERD Schaufelberger < gerd@php-tools.net>
- **Version** \$Id: example_basic_restart.php 46 2006-03-15 22:15:57Z gerd \$

include_once ['./patSession.php'](#)*[line 18]*

include_once ['./ error.php'](#)*[line 15]*

example_basic_singleton.php

example_basic_singleton.php

- **Package** patSession
- **Sub-Package** Examples
- **License** LGPL
- **Copyright** 2004 <http://www.php-tools.net>
- **Author** gERD Schaufelberger < gerd@php-tools.net>
- **Version** \$Id: example_basic_singleton.php 46 2006-03-15 22:15:57Z gerd \$

include_once ['./patSession.php'](#)*[line 18]*

include_once ['./ error.php'](#)*[line 15]*

example_secure_expire.php

example_secure_expire.php

- **Package** patSession
- **Sub-Package** Examples
- **License** LGPL
- **Copyright** 2004 <http://www.php-tools.net>
- **Author** gERD Schaufelberger < gerd@php-tools.net>
- **Version** \$Id: example_secure_expire.php 46 2006-03-15 22:15:57Z gerd \$

include_once ['./patSession.php'](#)*[line 18]*

include_once ['./ error.php'](#)*[line 15]*

example_secure_fix.php

example_secure_fix.php

- **Package** patSession
- **Sub-Package** Examples
- **License** LGPL
- **Copyright** 2004 <http://www.php-tools.net>
- **Author** gERD Schaufelberger < gerd@php-tools.net>
- **Version** \$Id: example_secure_fix.php 46 2006-03-15 22:15:57Z gerd \$

include_once ['./patSession.php'](#)*[line 18]*

include_once ['./ error.php'](#)*[line 15]*

example_secure_fixbrowser.php

example_secure_fix.php

- **Package** patSession
- **Sub-Package** Examples
- **License** LGPL
- **Copyright** 2004 <http://www.php-tools.net>
- **Author** gERD Schaufelberger < gerd@php-tools.net>
- **Version** \$Id: example_secure_fixbrowser.php 46 2006-03-15 22:15:57Z gerd \$

include_once ['./patSession.php'](#)*[line 18]*

include_once ['./ error.php'](#)*[line 15]*

example_secure_fixip.php

example_secure_fix.php

- **Package** patSession
- **Sub-Package** Examples
- **License** LGPL
- **Copyright** 2004 <http://www.php-tools.net>
- **Author** gERD Schaufelberger < gerd@php-tools.net>
- **Version** \$Id: example_secure_fixip.php 46 2006-03-15 22:15:57Z gerd \$

include_once ['./patSession.php'](#)*[line 18]*

include_once ['./ error.php'](#)*[line 15]*

example_secure_fixreferer.php

example_secure_fix.php

- **Package** patSession
- **Sub-Package** Examples
- **License** LGPL
- **Copyright** 2004 <http://www.php-tools.net>
- **Author** gERD Schaufelberger < gerd@php-tools.net>
- **Version** \$Id: example_secure_fixreferer.php 46 2006-03-15 22:15:57Z gerd \$

include_once ['./patSession.php'](#) [line 21]

include_once ['../patError/patErrorManager.php'](#) [line 15]

example_secure_useToken.php

example_secure_fix.php

- **Package** patSession
- **Sub-Package** Examples
- **License** LGPL
- **Copyright** 2004 <http://www.php-tools.net>
- **Author** gERD Schaufelberger < gerd@php-tools.net>
- **Version** \$Id: example_secure_useToken.php 46 2006-03-15 22:15:57Z gerd \$

include_once ['./patSession.php'](#)*[line 18]*

include_once ['./ error.php'](#)*[line 15]*

example_apl_array.php

example_spl_array.php

- **Package** patSession
- **Sub-Package** Examples
- **License** LGPL
- **Copyright** 2004-2006 <http://www.php-tools.net>
- **Author** gERD Schaufelberger < gerd@php-tools.net>
- **Version** \$Id\$

include_once ['./patSession.php'](#)*[line 18]*

include_once ['./ error.php'](#)*[line 15]*

example_spl_iterator.php

example_apl_array.php

- **Package** patSession
- **Sub-Package** Examples
- **License** LGPL
- **Copyright** 2004 <http://www.php-tools.net>
- **Author** gERD Schaufelberger < gerd@php-tools.net>
- **Version** \$Id\$

include_once ['./patSession.php'](#)*[line 18]*

include_once ['./ error.php'](#)*[line 15]*

example_spl_property.php

example_apl_array.php

- **Package** patSession
- **Sub-Package** Examples
- **License** LGPL
- **Copyright** 2004 <http://www.php-tools.net>
- **Author** gERD Schaufelberger < gerd@php-tools.net>
- **Version** \$Id\$

include_once ['./patSession.php'](#)*[line 18]*

include_once ['./ error.php'](#)*[line 15]*

example_test_cli.php

example_basic_first.php

- **Package** patSession
- **Sub-Package** Examples
- **License** LGPL
- **Copyright** 2004 <http://www.php-tools.net>
- **Author** gERD Schaufelberger < gerd@php-tools.net>
- **Version** \$Id: example_test_cli.php 46 2006-03-15 22:15:57Z gerd \$

include_once ['./patSession.php'](#)*[line 18]*

include_once ['./ error.php'](#)*[line 15]*

example_test_driver.php

example_basic_first.php

- **Package** patSession
- **Sub-Package** Examples
- **License** LGPL
- **Copyright** 2004 <http://www.php-tools.net>
- **Author** gERD Schaufelberger < gerd@php-tools.net>
- **Version** \$Id: example_test_driver.php 49 2006-09-05 21:00:01Z gerd \$

include_once ['./patSession.php'](#) [line 19]

include_once ['../patError/patErrorManager.php'](#) [line 15]

example_test_null.php

example_basic_first.php

- **Package** patSession
- **Sub-Package** Examples
- **License** LGPL
- **Copyright** 2004 <http://www.php-tools.net>
- **Author** gERD Schaufelberger < gerd@php-tools.net>
- **Version** \$Id: example_test_null.php 46 2006-03-15 22:15:57Z gerd \$

include_once ['./patSession.php'](#)*[line 18]*

include_once ['./ error.php'](#)*[line 15]*

example_test_tostring.php

example_apl_array.php

- **Package** patSession
- **Sub-Package** Examples
- **License** LGPL
- **Copyright** 2004 <http://www.php-tools.net>
- **Author** gERD Schaufelberger < gerd@php-tools.net>
- **Version** \$Id\$

include_once ['./patSession.php'](#)*[line 18]*

include_once ['./ error.php'](#)*[line 15]*

index_sections.php

Definitions of all included examples

\$Id: index_sections.php 49 2006-09-05 21:00:01Z gerd \$

- **Package** patSession
- **Sub-Package** Examples
- **Author** Sebastian Mordziol < argh@php-tools.net>
- **Author** gERD Schaufelberger < gerd@php-tools.net>

`_error.php`
helper script that loads and initializes error handler
\$Id\$

- **Package** patSession
- **Sub-Package** Examples
- **Author** gERD Schaufelberger

`include_once '../patError/patErrorManager.php' [line 15]`

error handling

`_viewExample.php`

helper script that displays the examples including the template and PHP source.

\$Id: _viewExample.php 46 2006-03-15 22:15:57Z gerd \$

You may pass the following GET parameters to the script:

- `example => id` if the example

- **Package** patSession
- **Sub-Package** Examples
- **Author** Stephan Schmidt
- **Author** gERD Schaufelberger

function buildNavigation() [*line 115*]

builds the tab navigation and displays it

- **Access** public

string function getExample(\$id) [*line 159*]

Function Parameters:

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

get the example section

- **Access** public

string function `getExampleSection($id)` [*line 142*]

Function Parameters:

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

get the example section

- **Access** public

`require_once` ['index_sections.php'](#) [*line 19*]

Joker.php

patSession_Storage_Joker

Implements the Joker-Driver for patSession. Simple test custom drivers

\$Id\$

- **Package** patSession
- **Sub-Package** Storage
- **License** LGPL
- **Copyright** 2004 <http://www.php-tools.net>
- **Author** gERD Schaufelberger < gerd@php-tools.net>
- **Version** 1.0.0

Storage.php

patSession Storage driver

\$Id: Storage.php 57 2008-10-21 00:24:17Z gerd \$

- **Package** patSession
- **Sub-Package** Storage
- **License** LGPL
- **Copyright** 2004-2006 <http://www.php-tools.net>
- **Author** gERD Schaufelberger < gerd@php-tools.net>
- **Version** 2.3.1

Cli.php

patSession_Storage_Cli

Implements the Cli-Driver for patSession.

\$Id: Cli.php 46 2006-03-15 22:15:57Z gerd \$

- **Package** patSession
- **Sub-Package** Storage
- **License** LGPL
- **Copyright** 2004 <http://www.php-tools.net>
- **Author** gERD Schaufelberger < gerd@php-tools.net>
- **Version** 2.0.0

Native.php

patSession_Storage_Native

Implements the Native-Driver for patSession. Native uses PHP-Session usually stored in `$_SESSION`

\$Id: Native.php 56 2008-02-06 11:12:46Z gerd \$

- **Package** patSession
- **Sub-Package** Storage
- **License** LGPL
- **Copyright** 2004 <http://www.php-tools.net>
- **Author** gERD Schaufelberger < gerd@php-tools.net >
- **Version** 2.0.1

Null.php

patSession_Storage_Null

Implements the Null-Driver for patSession. Null does not store any value at all, so this is a quite useless driver

\$Id: Null.php 49 2006-09-05 21:00:01Z gerd \$

- **Package** patSession
- **Sub-Package** Storage
- **License** LGPL
- **Copyright** 2004 <http://www.php-tools.net>
- **Author** gERD Schaufelberger < gerd@php-tools.net>
- **Version** 2.0.0

Class patSession_Storage

[line 48]

patSession Storage driver

Base class for all storage drivers. The base class implements some basic features used by any storage driver. Furthermore the security checks are implemented here.

The derived drivers must implement the methods marked as abstract. The constructor acts as template function to start the session. Therefore the driver must implement the following functions:

`_start()` start or restart session. Make storage container available. Set session attributes.

`_suspend()` stop session, make storage container persistend.

Usually the following function are required for functional drivers, too:

`_handleDestroy()` do all necessary steps to required to destroy current session.

`_handleFork()`

`_setOptions()`

- **Package** patSession
- **Sub-Package** Storage
- **Abstract Element**

patSession_Storage::\$_allowedReferer

array = array() [line 113]

list of allowed referers

- **Access** protected

patSession_Storage::\$_cookieParams

array = array() [line 137]

current cookie parameter

- **Access** protected

patSession_Storage::\$_emptyReferer

string = null [line 119]

allow empty referer

- **Var** 'deny' is default
- **Access** protected

patSession_Storage::\$_expire

string = null [line 88]

maximum age of unused session

- **Var** minutes
- **Access** protected

patSession_Storage::\$_id

mixed = null [line 67]

id string

- **Access** protected
- **See** [patSession_Storage::getContainerId\(\)](#)

patSession_Storage::\$_idPrefix

mixed = null [line 73]

string added to session id

- **Access** protected

patSession_Storage::\$_security

array = array('fix_referer', 'fix_ip', 'fix_browser') [line 107]

security policy

Default values:

- fix_ip
- fix_browser

- `fix_referer`
- `use_token`

- **Var** list of checks that will be done.
- **Access** protected

patSession_Storage::\$_sess

mixed = null *[line 54]*

session variables

- **Var** temporary storage container
- **Access** protected

patSession_Storage::\$_sessAttributes

array = array() *[line 60]*

session attributes

- **Access** protected

patSession_Storage::\$_state

string = 'new' *[line 81]*

internal state

- **Var** one of 'new'|'active'|'destroyed'
- **Access** protected
- **See** [patSession_Storage::getState\(\)](#)

patSession_Storage::\$_tokenMode

string = 'cookie' [*line 95*]

accept tokens send by

- **Access** protected

patSession_Storage::\$_tokenName

string = 'patSessionToken' [*line 125*]

patSession Security Token

- **Access** protected

patSession_Storage::\$_tokenValue

string = null [*line 131*]

disposable controll value for each request

- **Access** protected

Constructor function patSession_Storage::__construct([\$id = 'session'], [\$options = array()]) [*line 164*]

Function Parameters:

- *string* **\$id** name-prefix used for internal storage of session-data

- *array* **\$options** additional session parameter and configuration values

Start the session

- pass the configuration options
- start session
- perform security checks
- increase session counter

- **Access** public

mixed function `patSession_Storage::clear($name)` [line 394]

Function Parameters:

- *string* **\$name** name of variable

unset data from session

- **Access** public

boolean function `patSession_Storage::current()` [line 996]

implement Iterator interface

- **See** `Iterator::current()`

boolean function `patSession_Storage::destroy()` [line 418]

unset session variables and destroy session

- **Access public**

boolean function patSession_Storage::fork() [*line 480*]

create a new session and copy variables from the old one

- **Access public**

mixed function patSession_Storage::get(\$name) [*line 352*]

Function Parameters:

- *string* **\$name** name of variable

get data from session

- **Access public**

string function patSession_Storage::getContainerId() [*line 292*]

recieve name of session container

- **Access public**

int function patSession_Storage::getCounter() [*line 505*]

get session usage counter

- **Access** public

string function `patSession_Storage::getId()` [*line 312*]

recieve id of this session

- **Access** public

string function `patSession_Storage::getName()` [*line 302*]

recieve name of this session

- **Access** public

string function `patSession_Storage::getQueryString()` [*line 282*]

recieve url-encoded string The query string contains the session name and id and can be used the same way as the constant SID

- **Access** public
- **See** [patSession_Storage::getSID\(\)](#)
- **Deprecated** in favor of `getSID()`

string function `patSession_Storage::getSID()` [*line 263*]

recieve url-encoded string The query string contains the session name and id and can be used the same way as the constant SID

- **Access** public

string function patSession_Storage::getState() [*line 251*]

get current state of session

- **Access** public

array function patSession_Storage::getTimer() [*line 552*]

get session timers

returns a list of available timers in seconds since 1. January 1970

- **Access** public

boolean function patSession_Storage::has(\$name) [*line 375*]

Function Parameters:

- *string* **\$name** name of variable

check whether a session value exists

- **Access** public

boolean function patSession_Storage::isNew() [*line 519*]

check whether this session is currently created

- **Access** public

boolean function patSession_Storage::key() [*line 985*]

implement Iterator interface

- **Access** public
- **See** Iterator::valid()

boolean function patSession_Storage::next() [*line 1007*]

implement Iterator interface

- **Access** public
- **See** Iterator::next()

boolean function patSession_Storage::offsetExists(\$offset) [*line 1021*]

Function Parameters:

- *mixed* **\$offset**

offsetExists

implement ArrayAccess interface

- **Access** public
- **See** ArrayAccess::offsetExists()

boolean function patSession_Storage::offsetGet(\$offset) [line 1035]

Function Parameters:

- *mixed* **\$offset**

offsetGet

implement ArrayAccess interface

- **Access** public
- **See** ArrayAccess::offsetGet()

boolean function patSession_Storage::offsetSet(\$offset, \$value) [line 1049]

Function Parameters:

- *mixed* **\$offset**
- **\$value**

offsetSet

implement ArrayAccess interface

- **Access** public
- **See** ArrayAccess::offsetSet()

boolean function patSession_Storage::offsetUnset(\$offset) [line 1063]

Function Parameters:

- *mixed* **\$offset**

offsetUnset

implement ArrayAccess interface

- **Access** public
- **See** `ArrayAccess::offsetUnset()`

boolean function `patSession_Storage::restart()` [*line 449*]

restart a destroyed or locked session

- **Access** public
- **See** [patSession_Storage::destroy\(\)](#)

boolean function `patSession_Storage::rewind()` [*line 949*]

implement Iterator interface

- **Access** public
- **See** `Iterator::rewind()`

boolean function `patSession_Storage::set($name, $value)` [*line 332*]

Function Parameters:

- *string* **\$name** name of variable
- *mixed* **\$value** any value to be stored into session

save data into session

- **Access** public

boolean function patSession_Storage::valid() [*line 971*]

implement Iterator interface

- **Access** public
- **See** Iterator::valid()

true function patSession_Storage::_checkExpire() [*line 769*]

expire inactive session

A session will expire after a long time of no further request.

- **Access** protected

true function patSession_Storage::_checkFixBrowser() [*line 644*]

fix session to client' user agent

User MUST NOT change user agent during session

- **Access** protected

true function patSession_Storage::_checkFixIp() [*line 675*]

fix session to client's ip address

See whether fix_ip - feature is in use and check client's ip address. First check client's ip by forwarded address from proxy. In a http-proxy is used, it should send "HTTP_X_FORWARDED_FOR" header and tell the client's ip. We cannot check the proxy's ip, 'cause big providers use proxy pools. :(If there is no proxy, fix ip by "REMOTE_ADDR"

- **Access** protected

true function patSession_Storage::_checkReferer() [*line 719*]

check whether referer stays in limit

- **Access** protected

boolean function patSession_Storage::_checkSecurity([\$restart = false]) [*line 602*]

Function Parameters:

- *boolean* **\$restart** reactivate session

do some checks for security reason

- timeout check (expire)
 - ip-fixiation
 - referer-fixiation
- If one check failed, session data has to be cleaned.

- **Access** protected
- **TODO** add network-mask feature for ip-check
- **TODO** add allowed referer for session-transfers

true function patSession_Storage::_checkToken() [*line 798*]

fix session to client' user agent

User **MUST NOT** send single server token to continue session. Tokens increase security if cookies are used. Passing tokens via GET parameter might be a big mess. Hence cookies should be preferred, because they supports browser navigation like "reload", "back" and "forward"

- **Access** protected

string function patSession_Storage::_createId([\$length = 32]) [*line 847*]

Function Parameters:

- *int* **\$length** length of string

create an id-string

- **Access** protected

boolean function patSession_Storage::_initCounter() [*line 533*]

initialize counter of session usage

- **Access** protected

boolean function patSession_Storage::_setOptions(&\$options, \$options) [*line 872*]

Function Parameters:

- *array* **\$options** list of parameter
- **&\$options**

set additional session options

- **Access** protected

boolean function patSession_Storage::_start() [*line 197*]

start session

- **Access** protected
- **Abstract Element**

boolean function patSession_Storage::_suspend() [*line 206*]

suspend the session values

Make session values persistent

- **Access** protected
- **Abstract Element**

function patSession_Storage::__destruct() [*line 187*]

make sure to suspend session values on shutdown

- **Access** public

mixed function patSession_Storage::__get(\$property) [*line 214*]

Function Parameters:

- *string* **\$property**

Easy-access wrapper

- **Access** public

string function patSession_Storage::__toString() [*line 938*]

debugging output

- **Access** public

Class patSession_Storage_Cli

[*line 38*]

patSession_Storage_Cli

Implements the Cli-Driver for patSession.

The Cli-Driver gets handy, if you want to emulate sessions for PHP-CLI. Of course, sessions in the scope of CLI-programmes are completely different than in HTTP. Nevertheless. You may still want to treat a session in the same way than in web-based scripts. That's where the storage driver CLI comes in.

Please note:

- The session is always active
- All security checks are switched off
- The values stored in this session will not survive multiple instances (the session ends when the programm ends).

- **Package** patSession
- **Sub-Package** Storage

boolean function patSession_Storage_Cli::_setOptions(&\$options, \$options) [*line 90*]

Function Parameters:

- *array* **\$options** list of parameter
- **&\$options**

set additional session options

- **Access** protected

bool function patSession_Storage_Cli::_start() [*line 46*]

Start the session

- **Access** protected

boolean function patSession_Storage_Cli::_suspend() [*line 78*]

suspend the session values

Do nothing! In other words, don't make session values persistent

- **Access** protected

Class patSession_Storage_Joker

[*line 28*]

patSession_Storage_Joker

Implements the Joker-Driver for patSession. Simple test custom drivers.

- **Package** patSession
- **Sub-Package** Storage

patSession_Storage_Joker::STORAGE_DIR

= 'tmp' [line 33]

where to store session file

patSession_Storage_Joker::\$_sessFile

mixed = null [line 45]

store session values in a file

- **Access** protected

patSession_Storage_Joker::\$_sessId

mixed = null [line 39]

session id

- **Access** protected

patSession_Storage_Joker::\$_sessName

mixed = 'joker' [line 51]

session name

- **Access** protected

boolean function patSession_Storage_Joker::_handleFork() [*line 119*]

create a new session and copy variables from the old one

- **Access** protected
- **TODO** think about storage of session values - what happens to \$_SESSION

boolean function patSession_Storage_Joker::_start() [*line 61*]

start session

Simply restore session values from file. Well, this is just for demo purpose. A lot of stuff is missing: Sessions don't expire etc.

- **Access** protected

boolean function patSession_Storage_Joker::_suspend() [*line 95*]

suspend the session values

Make session values persistent

- **Access** protected

Class patSession_Storage_Native

[line 28]

patSession_Storage_Native

Implements the Native-Driver for patSession. Native uses PHP-Session usually stored in \$_SESSION

- **Package** patSession
- **Sub-Package** Storage

patSession_Storage_Native::\$_sessId

mixed = null [line 34]

session id

- **Access** protected

patSession_Storage_Native::\$_sessName

mixed = null [line 40]

session name

- **Access** protected

boolean function patSession_Storage_Native::_handleFork() [line 119]

create a new session and copy variables from the old one

- **Access** protected
- **TODO** think about storage of session values - what happens to \$_SESSION

boolean function patSession_Storage_Native::_setOptions(&\$options, \$options) [*line 162*]

Function Parameters:

- *array* **\$options** list of parameter
- **&\$options**

set additional session options

- **Access** protected

boolean function patSession_Storage_Native::_start() [*line 52*]

start session

- start session (if not already running or if session needs restart)
- select storage container by id
- store id in attributes
- store name in attributes

- **Access** protected

boolean function patSession_Storage_Native::_suspend() [*line 95*]

suspend the session values

Make session values persistent

- **Access** protected

Class patSession_Storage_Null

[line 30]

patSession_Storage_Null

Implements the Null-Driver for patSession. Null does not store any value at all, so this is a quite useless driver. It implements the features of starting and destroying a session. It also keeps track of the session-state.

- **Package** patSession
- **Sub-Package** Storage

boolean function patSession_Storage_Null::_setOptions(&\$options, \$options) *[line 71]*

Function Parameters:

- *array* **\$options** list of parameter
- **&\$options**

set additional session options

- **Access** protected

boolean function patSession_Storage_Null::_start() *[line 37]*

Start the session

- **Access** protected

boolean function patSession_Storage_Null::_suspend() [*line 60*]

suspend the session values

Do nothing! In other words, don't make session values persistent

- **Access** protected

Package patTemplate Procedural Elements

index.php

Frameset for the examples

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

require_once ['index_sections.php'](#)*[line 13]*

require section config

index_main.php

Overview for the examples

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

require_once ['index_sections.php'](#)*[line 13]*

require section config

Navigation for the examples

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

require_once ['index_sections.php'](#)*line 13]*

require section config

autopackage2.php

package.xml generation file for patSession

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

\$Id: autopackage2.php 444 2006-08-28 19:06:22Z gerd \$

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

include **__FILE__** *[line 23]*

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

require_once **'PEAR/PackageFileManager2.php'** *[line 20]*

uses PackageFileManager

Appendices

Appendix A - Class Trees

Package patSession

patSession

- [patSession](#)

patSession_Storage

- [patSession_Storage](#)
 - [patSession_Storage_Cli](#)
 - [patSession_Storage_Joker](#)
 - [patSession_Storage_Native](#)
 - [patSession_Storage_Null](#)

Package patTemplate

Package patSessoin

Appendix D - Todo List

In Package patSession

In [patSession Storage:: checkSecurity\(\)](#)

- add allowed referer for session-transfers
- add network-mask feature for ip-check

In [patSession Storage Native:: handleFork\(\)](#)

- think about storage of session values - what happens to \$_SESSION

In [patSession Storage Joker:: handleFork\(\)](#)

- think about storage of session values - what happens to \$_SESSION

Index

A

autopackage2.php	69
<i>package.xml generation file for patSession</i>	

B

buildNavigation()	35
<i>builds the tab navigation and displays it</i>	

C

constructor patSession Storage:: construct()	45
<i>Start the session</i>	
Cli.php	39
<i>patSession_Storage_Cli</i>	

E

example secure useToken.php	25
<i>example_secure_fix.php</i>	
example spl_array.php	26
<i>example_apl_array.php</i>	
example secure fixreferer.php	24
<i>example_secure_fix.php</i>	
example secure fixip.php	23
<i>example_secure_fix.php</i>	
example secure fixbrowser.php	22
<i>example_secure_fix.php</i>	
example spl iterator.php	27
<i>example_apl_array.php</i>	
example spl property.php	28
<i>example_apl_array.php</i>	
example test toString.php	32
<i>example_apl_array.php</i>	
example test null.php	31
<i>example_basic_first.php</i>	
example test driver.php	30
<i>example_basic_first.php</i>	
example test cli.php	29
<i>example_basic_first.php</i>	
example secure fix.php	21
<i>example_secure_fix.php</i>	

example_secure_expire.php	20
<i>example_secure_expire.php</i>	
example_basic_factory.php	13
<i>example_basic_factory.php</i>	
example_basic_destroy.php	12
<i>example_basic_destroy.php</i>	
example_basic_customdriver.php	11
<i>example_basic</i>	
example_basic_count.php	10
<i>example_basic_counter.php</i>	
example_basic_first.php	14
<i>example_basic_first.php</i>	
example_basic_namespace.php	15
<i>example_basic_options.php</i>	
example_basic_singleton.php	19
<i>example_basic_singleton.php</i>	
example_basic_restart.php	18
<i>example_basic_counter.php</i>	
example_basic_parameter.php	17
<i>example_basic_parameter.php</i>	
example_basic_options.php	16
<i>example_basic_options.php</i>	
example_secure_fork.php	5

G

getExampleSection()	36
<i>get the example section</i>	
getExample()	35
<i>get the example section</i>	

I

index_nav.php	68
<i>Navigation for the examples</i>	
index_main.php	67
<i>Overview for the examples</i>	
index.php	66
<i>Frameset for the examples</i>	
index_sections.php	33
<i>Definitions of all included examples</i>	

J

Joker.php	37
<i>patSession_Storage_Joker</i>	

N

Null.php	<i>patSession_Storage_Null</i>	41
Native.php	<i>patSession_Storage_Native</i>	40

P

patSession_Storage::suspend()	<i>suspend the session values</i>	56
patSession_Storage::start()	<i>start session</i>	56
patSession_Storage::setOptions()	<i>set additional session options</i>	55
patSession_Storage::initCounter()	<i>initialize counter of session usage</i>	55
patSession_Storage::destruct()	<i>make sure to suspend session values on shutdown</i>	56
patSession_Storage::get()	<i>Easy-access wrapper</i>	56
patSession_Storage_Cli::setOptions()	<i>set additional session options</i>	58
patSession_Storage_Cli	<i>patSession_Storage_Cli</i>	57
patSession_Storage::toString()	<i>debugging output</i>	57
patSession_Storage::createId()	<i>create an id-string</i>	55
patSession_Storage::checkToken()	<i>fix session to client' user agent</i>	54
patSession_Storage::valid()	<i>implement Iterator interface</i>	53
patSession_Storage::set()	<i>save data into session</i>	52
patSession_Storage::rewind()	<i>implement Iterator interface</i>	52
patSession_Storage::restart()	<i>restart a destroyed or locked sessionb</i>	52
patSession_Storage::checkExpire()	<i>expire inactive session</i>	53
patSession_Storage::checkFixBrowser()	<i>fix session to client' user agent</i>	53
patSession_Storage::checkSecurity()	<i>do some checks for security reason</i>	54
patSession_Storage::checkReferer()	<i>check whether referer stays in limit</i>	54
patSession_Storage::checkFixIp()	<i>fix session to client's ip address</i>	53
patSession_Storage_Cli::start()	<i>Start the session</i>	58
patSession_Storage_Cli::suspend()	<i>suspend the session values</i>	58
patSession_Storage_Native::start()		62

start session	62
patSession_Storage_Native::setOptions()	62
<i>set additional session options</i>	
patSession_Storage_Native::handleFork()	61
<i>create a new session and copy variables from the old one</i>	
patSession_Storage_Native::\$sessName	61
<i>session name</i>	
patSession_Storage_Native::suspend()	62
<i>suspend the session values</i>	
patSession_Storage_Null	63
<i>patSession_Storage_Null</i>	
patSession_Storage_Null::suspend()	64
<i>suspend the session values</i>	
patSession_Storage_Null::start()	63
<i>Start the session</i>	
patSession_Storage_Null::setOptions()	63
<i>set additional session options</i>	
patSession_Storage_Native::\$sessId	61
<i>session id</i>	
patSession_Storage_Native	61
<i>patSession_Storage_Native</i>	
patSession_Storage_Joker::\$sessFile	59
<i>store session values in a file</i>	
patSession_Storage_Joker::STORAGE_DIR	59
<i>where to store session file</i>	
patSession_Storage_Joker	58
<i>patSession_Storage_Joker</i>	
patSession_Storage_Joker::\$sessId	59
<i>session id</i>	
patSession_Storage_Joker::\$sessName	59
<i>session name</i>	
patSession_Storage_Joker::suspend()	60
<i>suspend the session values</i>	
patSession_Storage_Joker::start()	60
<i>start session</i>	
patSession_Storage_Joker::handleFork()	60
<i>create a new session and copy variables from the old one</i>	
patSession_Storage::offsetUnset()	51
<i>offsetUnset</i>	
patSession_Storage::offsetSet()	51
<i>offsetSet</i>	
patSession_Storage::\$idPrefix	43
<i>string added to session id</i>	
patSession_Storage::\$id	43
<i>id string</i>	
patSession_Storage::\$expire	43
<i>maximum age of unused session</i>	
patSession_Storage::\$emptyReferer	42
<i>allow empty referer</i>	
patSession_Storage::\$security	43
<i>security policy</i>	
patSession_Storage::\$sess	44
<i>session variables</i>	

patSession Storage::\$ tokenMode	45
<i>accept tokens send by</i>	
patSession Storage::\$ state	44
<i>internal state</i>	
patSession Storage::\$ sessAttributes	44
<i>session attributes</i>	
patSession Storage::\$ cookieParams	42
<i>current cookie parameter</i>	
patSession Storage::\$ allowedReferer	42
<i>list of allowed referers</i>	
patSession::ERROR_NOT_ACTIVE	7
<i>error definition: session not active</i>	
patSession::ERROR_DRIVER_NOT_FOUND	7
<i>error definition: could not load driver</i>	
patSession	7
<i>patSession allows easy access to session values</i>	
patSession.php	6
<i>patSession</i>	
patSession::ERROR_NOT_DESTROYED	7
<i>error definition: session not destroyed</i>	
patSession::create()	8
<i>factory</i>	
<i>Allows to create multiple instances of patSession</i>	
<i>Each instance must differ in name of prefix, otherwise this function return a previously created object.</i>	
patSession Storage	41
<i>patSession Storage driver</i>	
patSession::singleton()	8
<i>Singleton</i>	
<i>create a single instance of patSession driver</i>	
patSession::setDriverDir()	8
<i>Custom driver dir to include your own classes</i>	
patSession Storage::\$ tokenName	45
<i>patSession Security Token</i>	
patSession Storage::\$ tokenValue	45
<i>disposable controll value for each request</i>	
patSession Storage::has()	49
<i>check wheter a session value exists</i>	
patSession Storage::getTimer()	49
<i>get session timers</i>	
patSession Storage::getState()	49
<i>get current state of sessoin</i>	
patSession Storage::getSID()	48
<i>recieve url-encoded string</i>	
<i>The query string contains the session name and id</i>	
<i>and can be used the same way as the constant SID</i>	
patSession Storage::isNew()	49
<i>check whether this session is currently created</i>	
patSession Storage::key()	50
<i>implement Iterator interface</i>	
patSession Storage::offsetGet()	51
<i>offsetGet</i>	
patSession Storage::offsetExists()	50

<i>offsetExists</i>	
patSession_Storage::next()	50
<i>implement Iterator interface</i>	
patSession_Storage::getQueryString()	48
<i>recieve url-encoded string</i>	
<i>The query string contains the session name and id and can be used the same way as the constant SID</i>	
patSession_Storage::getName()	48
<i>recieve name of this session</i>	
patSession_Storage::destroy()	46
<i>unset session variables and destroy session</i>	
patSession_Storage::current()	46
<i>implement Iterator interface</i>	
patSession_Storage::clear()	46
<i>unset data from session</i>	
patSession_Storage::fork()	47
<i>create a new session and copy variables from the old one</i>	
patSession_Storage::get()	47
<i>get data from session</i>	
patSession_Storage::getId()	48
<i>recieve id of this session</i>	
patSession_Storage::getCounter()	47
<i>get session usage counter</i>	
patSession_Storage::getContainerId()	47
<i>recieve name of session container</i>	
package.php	2
<i>package.xml generation file for patSession</i>	

S

Storage.php	38
<i>patSession Storage driver</i>	
startsession.php	4
<hr/>	
_viewExample.php	35
<i>helper script that displays the examples including the template and PHP source.</i>	
_error.php	34
<i>helper script that loads initliazies error handler</i>	