

# SimpleCassie

Apache Cassandra PHP Client

## Introduction

[SimpleCassie](#) is entirely stand-alone package which wrap itself around Thrift libs (licensed to Apache Software Foundation) already built into this class.

[SimpleCassie](#) supports all features of TimeUUIDType - examples below.

The latest version is **compatible only with Cassandra 0.7.x** and higher.

Please report all bugs under: <http://code.google.com/p/simpletools-php/issues/list> so will be able to help and comment on them easier.

## Usage

```
<?php

require_once('SimpleCassie.php');

$cassie = new SimpleCassie(HOST, PORT);
if(!$cassie->isConnected())
    throw new Exception('Couldn\'t connect to server');

/*
 * setting working keyspace
 * @return - (false) on failure true on success
 * working keyspace can be change at any point by running ->keyspace() method
 */
$cassie->keyspace('Keyspace1');

/*
 * setting new column (and key if not exist)
 * @return - (false) on failure
 */
$cassie->keyspace('MyApp')->cf('Users')->key('user1')->column('name')->set('Marcin');
$cassie->column('surname')->set('Rosinski');

/*
 * delete column or row/key
 * @return - (false) on failure
 */

//deleting column
$cassie->key('user1')->column('name')->remove();

//deleting row
$cassie->key('user1')->remove();

/*
 * count number of columns in row/key
 * @return - (int) on succes, false on failure
 */
$count = $cassie->key('user2')->count();

/*
 * count number of columns with predicate: from column to column
 * @return - (int) on succes, false on failure
 */
$count = $cassie->key('user2')->column('fromColumn','toColumn')->count();

/*
 * count number of columns with predicate: from column to column for multiple keys
 * @return - (int) on succes, false on failure
 */
$count = $cassie->key('user2','user3')->column('fromColumn','toColumn')->count();

/*
```

```

* getting single column
* @return - object on succes, null on failure
*/
$name = $cassie->keyspace('Keyspace1')->cf('Standard1')->key('user1')->column('name')->get();

/*
* getting multiple columns
* @return - array of objects on success, null on failure
*/
$user = $cassie->column('name','surname')->get();

/*
* getting multiple column values
* @return - array of objects on success, null on failure
*/
$user = $cassie->column('name','surname')->value();

/*
* getting single column from multiple rows/keys
* @return - array of objects on succes, null on failure
*/
$users = $cassie->key('user1','user2')->column('name')->get();

/*
* getting multiple columns from multiple rows/keys
* @return - array of objects on succes, null on failure
*/
$users = $cassie->key('user1','user2')->column('name','username')->get();

/*
* getting slice of columns from single row/key
* @return - array of objects on succes, null on failure
*/
$limit = 10;
$reversed = false;
$from_name = 'Puma';
$to_name = 'Tiger';
$friends = $cassie->key('user1friends')->column($from_name,$to_name)->slice($limit,$reversed);

/*
* getting slice of columns from single supercolumn row
* @return - array of objects on succes, null on failure
*/
$limit = 10;
$reversed = false;
$friends = $cassie->key('user1')->supercolumn('friends')->column($from_name,$to_name)->slice
($limit,$reversed);

//resetting supercolumn for future use
$cassie->key('user1')->supercolumn(null);

/*
* getting slice of columns from multiple rows/keys
* @return - array of objects on succes, null on failure
*/
$limit = 10;
$reversed = true;
$friends = $cassie->key('user1friends','user2friends')->column($from_name,$to_name)->slice($limit,$reversed);

/*
* increment column value
* @return - (int) new value on succes, false on failure
*/
$new_value = $cassie->key('user1')->column('friends')->increment();

/*
* decrement column value
* @return - (int) new value on succes, false on failure
*/
$new_value = $cassie->key('user1')->column('friends')->decrement();

```

```
?>
```

## UUID Usage Examples

```
<?php

//setting new uuid column
$cassie->keyspace('MyApp')->cf('BlogPosts')->key('post')->column($cassie->uuid())->set('I like raw food.');
```

  

```
//getting latest added post (assuming TimeUUIDType)
$post = $cassie->key('post')->slice(1);

//getting post column uuid name in canonical form
$uuid = $cassie->uuid($post->column->name);

$string_form = (string) $uuid; //canonical form
$binary_form = $uuid->uuid;

?>
```

## Development Roadmap

- Connection Load Balancing
- ->get\_range\_slices() - coming soon!
- ->remove() method for multiple rows/keys
- ->increment() method for multiple rows/keys
- ->decrement() method for multiple rows/keys
- ->truncate()
- all describe methods
- ->batch\_mutate()

## Download Link

SimpleCassie.php download - <http://code.google.com/p/simpletools-php/downloads/list>

## Discussion Group

[[http://groups.google.com/group/simpletools/browse\\_thread/thread/b2d5afc83e7acd5b](http://groups.google.com/group/simpletools/browse_thread/thread/b2d5afc83e7acd5b) SimpleCassie related discussion]

<https://c.statcounter.com/9397521/0/fe557aad/1/> | stats