

# Generated Documentation



# Contents

<a href="#">Package CodeIgniter Procedural Elements</a>	2
<a href="#">Controller.php</a>	2
<a href="#">Model.php</a>	3
<a href="#">Package CodeIgniter Classes</a>	4
<a href="#">Class CI_Controller</a>	4
<a href="#">Var \$instance</a>	4
<a href="#">Constructor __construct</a>	4
<a href="#">Method get_instance</a>	5
<a href="#">Class CI_Model</a>	5
<a href="#">Constructor __construct</a>	5
<a href="#">Method __get</a>	5
<a href="#">Package SMAQC Procedural Elements</a>	8
<a href="#">smaq.php</a>	8
<a href="#">Package SMAQC Classes</a>	9
<a href="#">Class Smaq</a>	9
<a href="#">Var \$defaultenddate</a>	9
<a href="#">Var \$defaultstartdate</a>	9
<a href="#">Constructor __construct</a>	9
<a href="#">Method index</a>	9
<a href="#">Method instrument</a>	9
<a href="#">Method invaliditem</a>	10
<a href="#">instrumentmodel.php</a>	11
<a href="#">metricmodel.php</a>	12
<a href="#">Class Instrumentmodel</a>	12
<a href="#">Var \$averagedmetrics</a>	12
<a href="#">Var \$definition</a>	13
<a href="#">Var \$enddate</a>	13
<a href="#">Var \$instrument</a>	13
<a href="#">Var \$latestmetrics</a>	14
<a href="#">Var \$metricnames</a>	14
<a href="#">Var \$startdate</a>	14
<a href="#">Var \$status</a>	14
<a href="#">Constructor __construct</a>	15
<a href="#">Method initialize</a>	15
<a href="#">Method __get</a>	15
<a href="#">Class Metricmodel</a>	16
<a href="#">Var \$average</a>	16
<a href="#">Var \$data</a>	16
<a href="#">Var \$definition</a>	17
<a href="#">Var \$enddate</a>	17
<a href="#">Var \$instrument</a>	17
<a href="#">Var \$metric</a>	17

<a href="#">Var \$plotdata</a>	18
<a href="#">Var \$startdate</a>	18
<a href="#">Constructor <code>__construct</code></a>	18
<a href="#">Method <code>initialize</code></a>	19
<a href="#">Method <code>__get</code></a>	19
<a href="#">footView.php</a>	20
<a href="#">headView.php</a>	21
<a href="#">instrumentView.php</a>	22
<a href="#">invaliditemView.php</a>	23
<a href="#">leftMenuView.php</a>	24
<a href="#">mainView.php</a>	25
<a href="#">metricView.php</a>	26
<a href="#">topMenuView.php</a>	27
<b><a href="#">Appendices</a></b>	28
<a href="#">Appendix A - Class Trees</a>	29
<a href="#">SMAQC</a>	29
<a href="#">CodeIgniter</a>	29
<a href="#">Appendix C - Source Code</a>	31
<a href="#">Package SMAQC</a>	32
<a href="#">source code: Controller.php</a>	33
<a href="#">source code: Model.php</a>	34
<a href="#">Appendix D - Todo List</a>	35



# Package CodeIgniter Procedural Elements

## Controller.php

### CodeIgniter

An open source application development framework for PHP 5.1.6 or newer

- **Package** CodeIgniter
- **Author** ExpressionEngine Dev Team
- **Copyright** Copyright (c) 2008 - 2011, EllisLab, Inc.
- **Link** <http://codeigniter.com>
- **Since** Version 1.0
- **Filesource** [Source Code for this file](#)
- **License** [http://codeigniter.com/user\\_guide/license.html](http://codeigniter.com/user_guide/license.html)

# Model.php

## CodeIgniter

An open source application development framework for PHP 5.1.6 or newer

- **Package** CodeIgniter
- **Author** ExpressionEngine Dev Team
- **Copyright** Copyright (c) 2008 - 2011, EllisLab, Inc.
- **Link** <http://codeigniter.com>
- **Since** Version 1.0
- **Filesource** [Source Code for this file](#)
- **License** [http://codeigniter.com/user\\_guide/license.html](http://codeigniter.com/user_guide/license.html)

# Package CodeIgniter Classes

## Class CI\_Controller

[line 30]

### CodeIgniter Application Controller Class

This class object is the super class that every library in CodeIgniter will be assigned to.

- **Package** CodeIgniter
- **Sub-Package** Libraries
- **Author** ExpressionEngine Dev Team
- **Link** [http://codeigniter.com/user\\_guide/general/controllers.html](http://codeigniter.com/user_guide/general/controllers.html)

### CI\_Controller::\$instance

*mixed* = [line 32]

- **Static**
- **Access** private

Constructor *void* function CI\_Controller::\_\_construct() [line 37]

### Constructor

- **Access** public

*void* function CI\_Controller::get\_instance() [*line 56*]

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

## Class CI\_Model

[*line 27*]

### CodeIgniter Model Class

- **Package** CodeIgniter
- **Sub-Package** Libraries
- **Author** ExpressionEngine Dev Team
- **Link** [http://codeigniter.com/user\\_guide/libraries/config.html](http://codeigniter.com/user_guide/libraries/config.html)

Constructor *void* function CI\_Model::\_\_construct() [*line 34*]

### Constructor

- **Access** public

*void* function CI\_Model::\_\_get(\$key) [*line 48*]

### Function Parameters:

- *string* **\$key**



## **\_\_get**

Allows models to access CI's loaded classes using the same syntax as controllers.

- **Access** private



# Package SMAQC Procedural Elements

## smaq.php

### smaq.php

File containing the default CodeIgniter controller for SMAQC.

- **Package** SMAQC
- **Sub-Package** controllers
- **Author** Trevor Owen < [trevor.owen@email.wsu.edu](mailto:trevor.owen@email.wsu.edu)>
- **Author** Aaron Cain
- **Version** 1.0
- **Copyright** TODO
- **License** TODO

# Package SMAQC Classes

## Class SmaqC

[line 26]

CodeIgniter model for a SMAQC metric

- **Package** SMAQC
- **Sub-Package** controllers
- **Author** Aaron Cain
- **Author** Trevor Owen < [trevor.owen@email.wsu.edu](mailto:trevor.owen@email.wsu.edu) >
- **Version** 1.0

**SmaqC::\$defaultenddate**

*mixed = [line 29]*

**SmaqC::\$defaultstartdate**

*mixed = [line 28]*

Constructor *void* function SmaqC::\_\_construct() [line 31]

*void* function SmaqC::index() [line 76]

*void* function SmaqC::instrument(\$name, [\$metric = NULL], [\$start = NULL], [\$end = NULL]) [line 92]

**Function Parameters:**

- **\$name**
- **\$metric**
- **\$start**
- **\$end**

- **Access** public

*void* function Smaq::invaliditem([\$requesteditemtype = NULL], [\$name = NULL]) [*line 220*]

***Function Parameters:***

- **\$requesteditemtype**
- **\$name**

- **Access** public

# instrumentmodel.php

## instrumentmodel.php

File containing a CodeIgniter model for an instrument.

- **Package** SMAQC
- **Sub-Package** models
- **Author** Trevor Owen < [trevor.owen@email.wsu.edu](mailto:trevor.owen@email.wsu.edu)>
- **Version** 1.0
- **Copyright** TODO
- **License** TODO

# metricmodel.php

## metricmodel.php

File containing a CodeIgniter model for a SMAQC metric.

- **Package** SMAQC
- **Sub-Package** models
- **Author** Trevor Owen < [trevor.owen@email.wsu.edu](mailto:trevor.owen@email.wsu.edu)>
- **Version** 1.0
- **Copyright** TODO
- **License** TODO

## Class Instrumentmodel

[line 24]

CodeIgniter model for an instrument

- **Package** SMAQC
- **Sub-Package** models
- **Author** Trevor Owen < [trevor.owen@email.wsu.edu](mailto:trevor.owen@email.wsu.edu)>
- **Version** 1.0

**Instrumentmodel::\$averagedmetrics**

*object = [line 83]*

**The averaged metrics for the instrument over the provided date range.**

The type is what is returned by a call to CI's Active Record db->get().

- **Access** private

#### Instrumentmodel::\$definition

*string = [line 40]*

#### The definition of the instrument.

A string that is retrieved from a database.

- **TODO** Need/want this? Also needs implementing (the string is just set to lorem ipsum).
- **Access** private

#### Instrumentmodel::\$enddate

*string = [line 56]*

#### The end date for grabbing metrics.

This should be a human readable string of the format m-d-Y. (Example: 12-12-2012)

- **Access** private

#### Instrumentmodel::\$instrument

*string = [line 30]*

#### The name of the instrument.

- **Access** private

#### Instrumentmodel::\$latestmetrics



*object = [line 76]*

### **The latest metrics for the instrument.**

The type is what is returned by a call to CI's Active Record db->get().

- **Access** private

### **Instrumentmodel::\$metricnames**

*array = [line 69]*

**A list (php array) of the metrics available in the database.**

- **Access** private

### **Instrumentmodel::\$startdate**

*string = [line 48]*

### **The start date for grabbing metrics.**

This should be a human readable string of the format m-d-Y. (Example: 11-11-2011)

- **Access** private

### **Instrumentmodel::\$status**

*string = [line 63]*

**The status of the instrument (green/yellow/red)**

- **TODO** Actually use/implement this
- **Access** private

Constructor *Instrumentmodel* function *Instrumentmodel::\_\_construct()* [line 95]

### Constructor

The constructor for *Instrumentmodel* simply calls the constructor for the base class (*CI\_Model*). All initialization of the class must be done using the *initialize* function. The reasoning for this has to do with the way *CI* uses loaded models (they cannot take arguments).

*array/boolean* function *Instrumentmodel::initialize(\$instrument, \$start, \$end)* [line 154]

#### Function Parameters:

- *string* **\$instrument** The name of the instrument.
- *string* **\$start** A human readable string for the start of the date range. Assumed to be in m-d-Y format. (Example: 11-11-2011)
- *string* **\$end** A human readable string for the end of the date range. Assumed to be in m-d-Y format. (Example: 12-12-2012)

### Initializer for the Instrument model

Gets all of the needed values for the class/model from the database. Calculates any values that need calculating.

- **Access** public

*mixed* function *Instrumentmodel::\_\_get(\$what)* [line 112]

#### Function Parameters:

- *string* **\$what** The member of the class that we are looking for.

### \_\_get

The custom *\_\_get()* function is essentially a getter for our private members in the class. The use of the variable *\$CI* and *get\_instance()* allows us to access *CI*'s loaded classes using the same syntax as in the controller.

# Class Metricmodel

[line 24]

CodeIgniter model for a SMAQC metric

- **Package** SMAQC
- **Sub-Package** models
- **Author** Trevor Owen < [trevor.owen@email.wsu.edu](mailto:trevor.owen@email.wsu.edu)>
- **Version** 1.0

**Metricmodel::\$average**

*float = 0 [line 67]*

**The average value of the metric over the provided date range.**

- **Access** private

**Metricmodel::\$data**

*object = [line 74]*

**The results of querying the database for the metric values.**

The type is what is returned by a call to CI's Active Record db->get().

- **Access** private

#### **Metricmodel::\$definition**

*string = [line 45]*

#### **The definition of the instrument.**

A string that is retrieved from a database.

- **TODO** Needs implementing (the string is just set to lorem ipsum).
- **Access** private

#### **Metricmodel::\$enddate**

*string = [line 61]*

#### **The end date for grabbing metrics.**

This should be a human readable string of the format m-d-Y. (Example: 12-12-2012)

- **Access** private

#### **Metricmodel::\$instrument**

*string = [line 30]*

#### **The name of the instrument.**

- **Access** private

#### **Metricmodel::\$metric**

*string = [line 36]*

**The name of the metric.**

- **Access** private

**Metricmodel::\$plotdata**

*string = [line 82]*

**A JSON encoded array of (x,y) values for jqplot to use.**

The x value is a time/date in milliseconds. The type is what is returned by a call to CI's Active Record db->get().

- **Access** private

**Metricmodel::\$startdate**

*string = [line 53]*

**The start date for grabbing metrics.**

This should be a human readable string of the format m-d-Y. (Example: 11-11-2011)

- **Access** private

Constructor *Metricmodel* function Metricmodel::\_\_construct() *[line 94]*

### **Constructor**

The constructor for Metricmodel simply calls the constructor for the base class (CI\_Model). All initialization of the class must be done using the initialize function. The reasoning for this has to do with the way CI loads models in the controller (they cannot take arguments).

*array/boolean* function Metricmodel::initialize(\$instrument, \$metric, \$start, \$end) [line 154]

**Function Parameters:**

- *string* **\$instrument** The name of the instrument.
- *string* **\$metric** The name of the metric.
- *string* **\$start** A human readable string for the start of the date range. Assumed to be in m-d-Y format. (Example: 11-11-2011)
- *string* **\$end** A human readable string for the end of the date range. Assumed to be in m-d-Y format. (Example: 12-12-2012)

### Initializer for the Metric model

Gets all of the needed values for the class/model from the database. Calculates any values that need calculating.

- **Access** public

*mixed* function Metricmodel::\_\_get(\$what) [line 111]

**Function Parameters:**

- *string* **\$what** The member of the class that we are looking for.

### \_\_get

The custom `__get()` function is essentially a getter for our private members in the class. The use of the variable `$CI` and `get_instance()` allows us to access CI's loaded classes using the same syntax as in the controller.

# footView.php

## footView.php

File containing the ending code for all views in SMAQC.

- **Package** SMAQC
- **Sub-Package** views
- **Author** Aaron Cain
- **Version** 1.0
- **Copyright** TODO
- **License** TODO

# headView.php

## headView.php

File containing the opening code for all views in SMAQC.

- **Package** SMAQC
- **Sub-Package** views
- **Author** Trevor Owen < [trevor.owen@email.wsu.edu](mailto:trevor.owen@email.wsu.edu)>
- **Author** Aaron Cain
- **Version** 1.0
- **Copyright** TODO
- **License** TODO



# instrumentView.php

## instrumentView.php

File containing the code for the instrumentView.

- **Package** SMAQC
- **Sub-Package** views
- **Author** Trevor Owen < [trevor.owen@email.wsu.edu](mailto:trevor.owen@email.wsu.edu)>
- **Version** 1.0
- **Copyright** TODO
- **License** TODO

# invaliditemView.php

## invaliditemView.php

File containing the code for the invaliditemView.

- **Package** SMAQC
- **Sub-Package** views
- **Author** Trevor Owen < [trevor.owen@email.wsu.edu](mailto:trevor.owen@email.wsu.edu)>
- **Version** 1.0
- **Copyright** TODO
- **License** TODO

# leftMenuView.php

## leftMenuView.php

File containing the code for the leftMenuView loaded on each page.

- **Package** SMAQC
- **Sub-Package** views
- **Author** Trevor Owen < [trevor.owen@email.wsu.edu](mailto:trevor.owen@email.wsu.edu)>
- **Version** 1.0
- **Copyright** TODO
- **License** TODO

# mainView.php

## mainView.php

File containing the code for the default view for SMAQC.

- **Package** SMAQC
- **Sub-Package** views
- **Author** Trevor Owen < [trevor.owen@email.wsu.edu](mailto:trevor.owen@email.wsu.edu)>
- **Author** Aaron Cain
- **Version** 1.0
- **Copyright** TODO
- **License** TODO

# metricView.php

## instrumentView.php

File containing the code for the metricView.

- **Package** SMAQC
- **Sub-Package** views
- **Author** Trevor Owen < [trevor.owen@email.wsu.edu](mailto:trevor.owen@email.wsu.edu)>
- **Version** 1.0
- **Copyright** TODO
- **License** TODO

# topMenuView.php

## topMenuView.php

File containing the code for the topMenuView loaded on each page.

- **Package** SMAQC
- **Sub-Package** views
- **Author** Trevor Owen < [trevor.owen@email.wsu.edu](mailto:trevor.owen@email.wsu.edu)>
- **Author** Aaron Cain
- **Version** 1.0
- **Copyright** TODO
- **License** TODO

# Appendices

# Appendix A - Class Trees

## Package SMAQC

### Instrumentmodel

- [CI\\_Model](#) (Different package)
  - [Instrumentmodel](#)

### Metricmodel

- [CI\\_Model](#) (Different package)
  - [Metricmodel](#)

### Smaq

- [CI\\_Controller](#) (Different package)
  - [Smaq](#)

## Package CodeIgniter

### CI\_Controller

- [CI\\_Controller](#)
  - [Smaq](#)



## CI\_Model

- [CI\\_Model](#)
  - [Instrumentmodel](#)
  - [Metricmodel](#)

# Appendix C - Source Code

## Package SMAQC

# File Source for Controller.php

Documentation for this file is available at [Controller.php](#)

```
1  <?php      if ( ! defined('BASEPATH')) exit('No direct script access allowed');
2  /**
3   * CodeIgniter
4   *
5   * An open source application development framework for PHP 5.1.6 or newer
6   *
7   * @package      CodeIgniter
8   * @author        ExpressionEngine Dev Team
9   * @copyright     Copyright (c) 2008 - 2011, EllisLab, Inc.
10  * @license       http://codeigniter.com/user_guide/license.html
11  * @link          http://codeigniter.com
12  * @since         Version 1.0
13  * @filesource
14  */
15
16  // -----
17
18  /**
19   * CodeIgniter Application Controller Class
20   *
21   * This class object is the super class that every library in
22   * CodeIgniter will be assigned to.
23   *
24   * @package      CodeIgniter
25   * @subpackage   Libraries
26   * @category     Libraries
27   * @author        ExpressionEngine Dev Team
28   * @link          http://codeigniter.com/user_guide/general/controllers.html
29   */
30  class CI_Controller {
31
32      private static $instance;
33
34      /**
35       * Constructor
36       */
37      public function __construct()
38      {
39          self::$instance =& $this;
40
41          // Assign all the class objects that were instantiated by the
42          // bootstrap file (CodeIgniter.php) to local class variables
43          // so that CI can run as one big super object.
44          foreach (is_loaded() as $var => $class)
45          {
46              $this-> $var =& load_class($class);
47          }
48
49          $this-> load =& load_class('Loader', 'core');
50
51          $this-> load-> initialize();
52
53          log_message('debug', "Controller Class Initialized" );
54      }
55
56      public static function & get_instance()
57      {
58          return self::$instance;
59      }
60  }
61  // END Controller class
62
63  /* End of file Controller.php */
64  /* Location: ./system/core/Controller.php */
```

# File Source for Model.php

Documentation for this file is available at [Model.php](#)

```
1  <?php      if ( ! defined('BASEPATH')) exit('No direct script access allowed');
2  /**
3   * CodeIgniter
4   *
5   * An open source application development framework for PHP 5.1.6 or newer
6   *
7   * @package      CodeIgniter
8   * @author        ExpressionEngine Dev Team
9   * @copyright     Copyright (c) 2008 - 2011, EllisLab, Inc.
10  * @license       http://codeigniter.com/user_guide/license.html
11  * @link          http://codeigniter.com
12  * @since         Version 1.0
13  * @filesource
14  */
15
16  // -----
17
18  /**
19   * CodeIgniter Model Class
20   *
21   * @package      CodeIgniter
22   * @subpackage    Libraries
23   * @category     Libraries
24   * @author        ExpressionEngine Dev Team
25   * @link          http://codeigniter.com/user_guide/libraries/config.html
26   */
27  class CI_Model {
28
29      /**
30       * Constructor
31       *
32       * @access public
33       */
34      function __construct()
35      {
36          log_message('debug', "Model Class Initialized" );
37      }
38
39      /**
40       * __get
41       *
42       * Allows models to access CI's loaded classes using the same
43       * syntax as controllers.
44       *
45       * @param    string
46       * @access private
47       */
48      function __get($key)
49      {
50          $CI =& get_instance();
51          return $CI-> $key;
52      }
53  }
54  // END Model Class
55
56  /* End of file Model.php */
57  /* Location: ./system/core/Model.php */
```

# Appendix D - Todo List

## In Package SMAQC

In [Instrumentmodel::\\$definition](#)

- Need/want this? Also needs implementing (the string is just set to lorem ipsum).

In [Metricmodel::\\$definition](#)

- Needs implementing (the string is just set to lorem ipsum).

In [Instrumentmodel::\\$status](#)

- Actually use/implement this

# Index

## C

<a href="#">constructor Smaq:: construct()</a>	9
<a href="#">CI Model:: get()</a>	5
__get	
<a href="#">constructor Instrumentmodel:: construct()</a>	15
Constructor	
<a href="#">constructor Metricmodel:: construct()</a>	18
Constructor	
<a href="#">Controller.php</a>	33
Source code	
<a href="#">constructor CI Model:: construct()</a>	5
Constructor	
<a href="#">CI Model</a>	5
CodeIgniter Model Class	
<a href="#">CI Controller</a>	4
CodeIgniter Application Controller Class	
<a href="#">CI Controller::\$instance</a>	4
<a href="#">constructor CI Controller:: construct()</a>	4
Constructor	
<a href="#">CI Controller::get_instance()</a>	5
<a href="#">Controller.php</a>	2
CodeIgniter	

## F

<a href="#">footView.php</a>	20
footView.php	

## H

<a href="#">headView.php</a>	21
headView.php	

## I

<a href="#">Instrumentmodel::\$status</a>	14
The status of the instrument (green/yellow/red)	
<a href="#">Instrumentmodel::\$startdate</a>	14
The start date for grabbing metrics.	
<a href="#">Instrumentmodel::initialize()</a>	15
Initializer for the Instrument model	
<a href="#">Instrumentmodel:: get()</a>	15

<a href="#">__get</a>	23
<a href="#">invaliditemView.php</a>	22
<a href="#">invaliditemView.php</a>	22
<a href="#">instrumentView.php</a>	14
<a href="#">Instrumentmodel::\$metricnames</a>	14
<i>A list (php array) of the metrics available in the database.</i>	
<a href="#">Instrumentmodel::\$latestmetrics</a>	14
<i>The latest metrics for the instrument.</i>	
<a href="#">Instrumentmodel::\$averagedmetrics</a>	12
<i>The averaged metrics for the instrument over the provided date range.</i>	
<a href="#">Instrumentmodel</a>	12
<i>CodeIgniter model for an instrument</i>	
<a href="#">Instrumentmodel::\$definition</a>	13
<i>The definition of the instrument.</i>	
<a href="#">Instrumentmodel::\$enddate</a>	13
<i>The end date for grabbing metrics.</i>	
<a href="#">Instrumentmodel::\$instrument</a>	13
<i>The name of the instrument.</i>	
<a href="#">instrumentmodel.php</a>	11
<a href="#">instrumentmodel.php</a>	

## L

<a href="#">leftMenuView.php</a>	24
<a href="#">leftMenuView.php</a>	

## M

<a href="#">Metricmodel::initialize()</a>	19
<i>Initializer for the Metric model</i>	
<a href="#">Metricmodel::\$startdate</a>	18
<i>The start date for grabbing metrics.</i>	
<a href="#">Metricmodel::\$plotdata</a>	18
<i>A JSON encoded array of (x,y) values for jqplot to use.</i>	
<a href="#">Metricmodel::__get()</a>	19
<a href="#">__get</a>	
<a href="#">mainView.php</a>	25
<a href="#">mainView.php</a>	
<a href="#">Model.php</a>	34
<i>Source code</i>	
<a href="#">metricView.php</a>	26
<a href="#">metricView.php</a>	
<a href="#">Metricmodel::\$metric</a>	17
<i>The name of the metric.</i>	
<a href="#">Metricmodel::\$instrument</a>	17
<i>The name of the instrument.</i>	
<a href="#">Metricmodel</a>	16
<i>CodeIgniter model for a SMAQC metric</i>	
<a href="#">metricmodel.php</a>	12
<a href="#">metricmodel.php</a>	



<a href="#">Metricmodel::\$average</a>	16
<i>The average value of the metric over the provided date range.</i>	
<a href="#">Metricmodel::\$data</a>	16
<i>The results of querying the database for the metric values.</i>	
<a href="#">Metricmodel::\$enddate</a>	17
<i>The end date for grabbing metrics.</i>	
<a href="#">Metricmodel::\$definition</a>	17
<i>The definition of the instrument.</i>	
<a href="#">Model.php</a>	3
<i>CodeIgniter</i>	

## S

<a href="#">Smaq::instrument()</a>	9
<a href="#">Smaq::invaliditem()</a>	10
<a href="#">Smaq::index()</a>	9
<a href="#">Smaq::\$defaultstartdate</a>	9
<a href="#">Smaq</a>	9
<i>CodeIgniter model for a SMAQC metric</i>	
<a href="#">Smaq::\$defaultenddate</a>	9
<a href="#">smqc.php</a>	8
<i>smqc.php</i>	

## T

<a href="#">topMenuView.php</a>	27
<i>topMenuView.php</i>	