

APPENDIX C



CakePHP 1.2 Functions and Properties

I've included here an overview of Cake's features available in the current release of version 1.2. Because Cake is an open source project and its core is liable to change, be sure to consult the official online API for additional classes, core utilities, and functions at <http://api.cakephp.org>. This appendix is meant to concisely alert you to possibilities in the various resource classes you'll use and build, not to be an exhaustive glossary of the entire Cake core. You will need to check the official API for details on what parameters to use with each function listed here, or in many instances, refer to previous chapters in this book where I've covered the function or property in more detail.

Behavior

Table C-1. *Behavior Properties*

Property	Type	Default Value	Details
<code>\$mapMethods</code>	Array	NULL	Mapping of preg-compatible regular expressions to methods in this class
<code>\$settings</code>	Array	NULL	Configuration settings for the behavior

Table C-2. *Behavior Callbacks*

Function	Details
<code>afterDelete()</code>	Called after the parent model performs a deletion method
<code>afterFind()</code>	Called after the parent model performs a find method
<code>afterSave()</code>	Called after the parent model performs a save method
<code>beforeDelete()</code>	Called before the parent model performs a deletion method
<code>beforeFind()</code>	Called before the parent model performs a find method
<code>beforeSave()</code>	Called before the parent model performs a save method
<code>beforeValidate()</code>	Called before the parent model performs a validation method
<code>cleanup()</code>	Cleans up any initialization this behavior has done on a model
<code>dispatchMethod()</code>	Overrides <code>Object::dispatchMethod()</code> to account for PHP 4's broken reference support

Function	Details
onError()	Called when the DataSource registers an error
setup()	Runtime function to initialize the behavior

Acl

Table C-3. *Acl Behavior Functions*

Function	Details
afterDelete()	Destroys the ARO/ACO node bound to the deleted record
afterSave()	Creates a new ARO/ACO node and binds it to this record
node()	Retrieves the ARO/ACO node for this model
setup()	Runtime function to initialize Acl

Containable

Table C-4. *Containable Behavior Properties and Functions*

Property/Function	Details
\$runtime	Runtime configuration
\$types	Types of relationships available to Containable, i.e., “has one,” “belongs to,” etc.
beforeFind()	Called before a find method in the parent model; allows the contain setting as part of the find call
contain()	Unbinds all relations of a model except given ones
containments()	Processes containments for the parent model
containmentsMap()	Builds the map of containments
fieldDependencies()	Calculates the needed fields to pull the required bindings for the given model
resetBindings()	Restores the original bindings of a given model
setup()	Runtime function to initialize Containable

Translate

Table C-5. *Translate Behavior Property and Functions*

Property/Function	Details
\$runtime	Runtime configuration
afterDelete()	Called after the parent model performs a deletion method
afterFind()	Called after the parent model performs a find method

Property/Function	Details
afterSave()	Called after the parent model performs a save method
beforeFind()	Called before the parent model performs a find method
beforeValidate()	Called before the parent model performs a validation method
bindTranslation()	Binds translation for fields
cleanup()	Cleans up any initialization this behavior has done on a model
setup()	Runtime function to initialize Translate
translateModel()	Gets instance of the parent model to perform translations
unbindTranslation()	Unbinds translation for fields

Tree

Table C-6. *Tree Behavior Property and Functions*

Property/Function	Details
\$errors	Array containing any error messages
afterSave()	Called after the parent model performs a save method
beforeDelete()	Called before the parent model performs a deletion method
beforeSave()	Called before the parent model performs a save method
childCount()	Counts the number of child nodes
children()	Gets the child nodes of the current model
generateTreeList()	Convenience function for returning a hierarchical array used for HTML select boxes
getParentNode()	Gets the parent node
getPath()	Returns the path to a given node
moveDown()	Moves the node down the tree without changing the parent
moveUp()	Moves the node up the tree without changing the parent
recover()	Recovers a corrupted tree
removeFromTree()	Removes the current node from the tree and reparents all children up one level
reorder()	Reorders the nodes and child nodes of the tree based on given conditions
setParent()	Sets the parent node
setup()	Runtime function to initialize Tree
verify()	Checks if the current tree is valid