

UNVirtualLab

Generado por Doxygen 1.7.6.1

Sábado, 8 de Febrero de 2014 08:30:52

Índice general

1 Índice de clases	1
1.1 Jerarquía de la clase	1
2 Índice de clases	3
2.1 Lista de clases	3
3 Documentación de las clases	5
3.1 Referencia de la Clase abouttable	5
3.1.1 Descripción detallada	5
3.2 Referencia de la Clase admin2danim	6
3.2.1 Descripción detallada	6
3.3 Referencia de la Clase admin2defect	6
3.3.1 Descripción detallada	7
3.4 Referencia de la Clase adminblock	7
3.4.1 Descripción detallada	9
3.5 Referencia de la Clase admincontrol	9
3.5.1 Descripción detallada	10
3.6 Referencia de la Clase adminlogin	10
3.6.1 Descripción detallada	10
3.7 Referencia de la Clase adminlogout	11
3.7.1 Descripción detallada	11
3.8 Referencia de la Clase adminmodel	11
3.8.1 Descripción detallada	12
3.9 Referencia de la Clase adminmodelfiles	12
3.9.1 Descripción detallada	13
3.10 Referencia de la Clase adminplot	13

3.10.1 Descripción detallada	13
3.11 Referencia de la Clase adminpractice	13
3.11.1 Descripción detallada	14
3.12 Referencia de la Clase adminsection	14
3.12.1 Descripción detallada	14
3.13 Referencia de la Clase anim2d	15
3.13.1 Descripción detallada	15
3.14 Referencia de la Clase animation2dSVG	15
3.14.1 Descripción detallada	16
3.15 Referencia de la Clase block	16
3.15.1 Descripción detallada	18
3.16 Referencia de la Clase configuration	18
3.16.1 Descripción detallada	18
3.17 Referencia de la Clase controls	18
3.17.1 Descripción detallada	19
3.18 Referencia de la Clase credits	19
3.18.1 Descripción detallada	20
3.19 Referencia de la Clase csvreader	20
3.19.1 Descripción detallada	20
3.20 Referencia de la Clase databasemanager	20
3.20.1 Descripción detallada	21
3.21 Referencia de la Clase documentation	21
3.21.1 Descripción detallada	21
3.22 Referencia de la Clase erase	21
3.22.1 Descripción detallada	22
3.23 Referencia de la Clase export	22
3.23.1 Descripción detallada	22
3.24 Referencia de la Clase import	22
3.24.1 Descripción detallada	23
3.25 Referencia de la Clase logo	23
3.25.1 Descripción detallada	23
3.26 Referencia de la Clase menu	23
3.26.1 Descripción detallada	24
3.27 Referencia de la Clase menuadmin	24

3.27.1 Descripción detallada	24
3.28 Referencia de la Clase MyXml	25
3.28.1 Descripción detallada	25
3.29 Referencia de la Clase outputs	25
3.29.1 Descripción detallada	26
3.30 Referencia de la Clase plots	26
3.30.1 Descripción detallada	26
3.31 Referencia de la Clase sessionManager	27
3.31.1 Descripción detallada	27
3.32 Referencia de la Clase simulator	27
3.32.1 Descripción detallada	28
3.33 Referencia de la Clase SVGplot	28
3.33.1 Descripción detallada	29
3.34 Referencia de la Clase tables	30
3.34.1 Descripción detallada	30
3.35 Referencia de la Clase title	30
3.35.1 Descripción detallada	30
3.36 Referencia de la Clase tlistmysql	31
3.36.1 Descripción detallada	31
3.36.2 Documentación de las funciones miembro	31
3.36.2.1 restoreNode	31
3.36.2.2 saveNode	32
3.37 Referencia de la Clase XMLThing	32
3.37.1 Descripción detallada	33
3.37.2 Documentación de las funciones miembro	33
3.37.2.1 parse	33
3.37.2.2 parse_init	34
3.37.2.3 parse_recurse	35
3.37.2.4 XMLThing	37

Capítulo 1

Índice de clases

1.1. Jerarquía de la clase

Esta lista de herencias esta ordenada aproximadamente por orden alfabético:

block	16
abouttable	5
adminblock	7
admin2danim	6
admin2deffect	6
admincontrol	9
adminlogin	10
adminlogout	11
adminmodel	11
adminplot	13
adminpractice	13
adminsection	14
sessionManager	27
adminmodelfiles	12
anim2d	15
controls	18
credits	19
databasemanager	20
documentation	21
erase	21
logo	23
menu	23
menuadmin	24
MyXml	25
animation2dSVG	15
outputs	25
plots	26
simulator	27
SVGplot	28

tables	30
title	30
configuration	18
csvreader	20
export	22
import	22
tlistemysql	31
XMLThing	32

Capítulo 2

Índice de clases

2.1. Lista de clases

Lista de las clases, estructuras, uniones e interfaces con una breve descripción:

abouttable	5
admin2danim	6
admin2defect	6
adminblock	7
admincontrol	9
adminlogin	10
adminlogout	11
adminmodel	11
adminmodelfiles	12
adminplot	13
adminpractice	13
adminsection	14
anim2d	15
animation2dSVG	15
block	16
configuration	18
controls	18
credits	19
csvreader	20
databasemanager	20
documentation	21
erase	21
export	22
import	22
logo	23
menu	23
menuadmin	24
MyXml	25
outputs	25

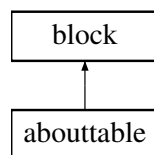
plots	26
sessionManager	27
simulator	27
SVGplot	28
tables	30
title	30
tlistmysql	31
XMLThing	32

Capítulo 3

Documentación de las clases

3.1. Referencia de la Clase abouttable

Diagrama de herencias de abouttable



Métodos públicos

- `display ()`
- `displayModel ()`
- `displayModellers ()`
- `displayFiles ()`
- `displayEmbedLinks ()`
- `displayPractices ()`

3.1.1. Descripción detallada

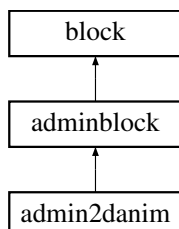
Definición en la línea 4 del archivo `about.php`.

La documentación para esta clase fue generada a partir del siguiente fichero:

- `about.php`

3.2. Referencia de la Clase admin2danim

Diagrama de herencias de admin2danim



Métodos públicos

- **createFields ()**
- **createRelations1N ()**
- **createRelations1N1N ()**
- **findModelId ()**
- **upload ()**
- **displayPostOwn ()**
- **displayAnim ()**

3.2.1. Descripción detallada

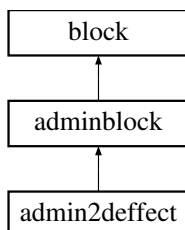
Definición en la línea 5 del archivo admin2danim.php.

La documentación para esta clase fue generada a partir del siguiente fichero:

- admin/admin2danim.php

3.3. Referencia de la Clase admin2defect

Diagrama de herencias de admin2defect



Métodos públicos

- `createFields ()`

- `createRelations1N ()`

- `createRelations1N1N ()`

- `findModelId ()`

3.3.1. Descripción detallada

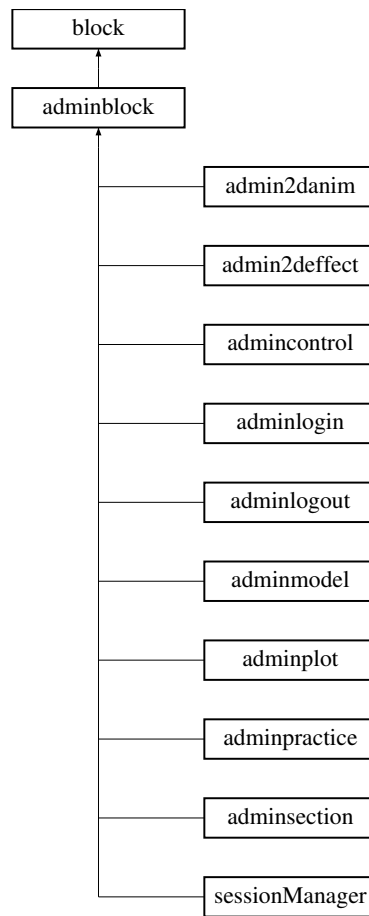
Definición en la línea 5 del archivo `admin2defect.php`.

La documentación para esta clase fue generada a partir del siguiente fichero:

- `admin/admin2defect.php`

3.4. Referencia de la Clase adminblock

Diagrama de herencias de `adminblock`



Métodos públicos

- **selectSection** (\$level, \$section_sql, \$section_id)
- **selectFromTable** (\$table, \$model_id, \$variable_id)
- **selectOptions** (\$F, \$id)
- **displayOneField** (\$data, \$F, \$class)
- **displayFields** (\$data)
- **update** ()
- **update1NN1** ()
- **insert** ()
- **insert1NN1** ()
- **delete** ()
- **delete1NN1** ()
- **createRelations1N** ()
- **createRelations1N1N** ()
- **createRelations1NN1** ()
- **displayOwn** ()

- **displayOneRelation1N** (\$rel)
- **displayRelations1N** ()
- **displayOneRelation1N1N** (\$rel)
- **displayRelations1N1N** ()
- **displayOneRelation1NN1** (\$rel)
- **displayRelations1NN1** ()
- **createFieldsAndRelations** ()
- **displayPostOwn** ()
- **displayAll** ()
- **display** ()

Atributos públicos

- **\$fields**
- **\$table**
- **\$sectionidname**
- **\$idname**
- **\$idvalue**
- **\$blockname**
- **\$relations1N**
- **\$relations1N1N**
- **\$relations1NN1**

3.4.1. Descripción detallada

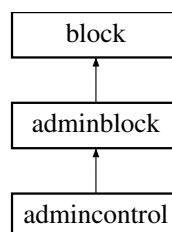
Definición en la línea 4 del archivo adminblock.php.

La documentación para esta clase fue generada a partir del siguiente fichero:

- admin/adminblock.php

3.5. Referencia de la Clase admincontrol

Diagrama de herencias de admincontrol



Métodos públicos

- **createFields ()**
- **createRelations1N1N ()**
- **findModelId ()**
- **update ()**

3.5.1. Descripción detallada

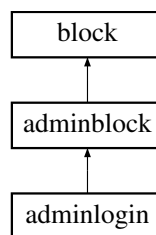
Definición en la línea 4 del archivo admincontrol.php.

La documentación para esta clase fue generada a partir del siguiente fichero:

- admin/admincontrol.php

3.6. Referencia de la Clase adminlogin

Diagrama de herencias de adminlogin



Métodos públicos

- **showMessage ()**
- **showButtons ()**
- **display ()**

3.6.1. Descripción detallada

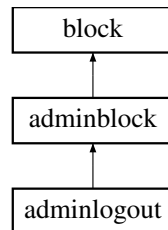
Definición en la línea 4 del archivo adminlogin.php.

La documentación para esta clase fue generada a partir del siguiente fichero:

- admin/adminlogin.php

3.7. Referencia de la Clase adminlogout

Diagrama de herencias de adminlogout



Métodos públicos

- `display ()`

3.7.1. Descripción detallada

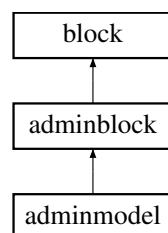
Definición en la línea 4 del archivo adminlogout.php.

La documentación para esta clase fue generada a partir del siguiente fichero:

- admin/adminlogout.php

3.8. Referencia de la Clase adminmodel

Diagrama de herencias de adminmodel



Métodos públicos

- `createFields ()`
- `createRelations1N ()`
- `createRelations1NN1 ()`

3.8.1. Descripción detallada

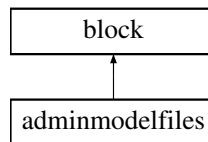
Definición en la línea 4 del archivo adminmodel.php.

La documentación para esta clase fue generada a partir del siguiente fichero:

- admin/adminmodel.php

3.9. Referencia de la Clase adminmodelfiles

Diagrama de herencias de adminmodelfiles



Métodos públicos

- **createdata** ()
- **copyfile** (\$FILE, \$FD)
- **db_modelica** (\$FILE, \$FD)
- **db_documentation** (\$FILE, \$FD)
- **compile** (\$FD)
- **db_executable** (\$FILE, \$FD)
- **readInitXml** (\$fn)
- **varsInInitXml** (\$fn)
- **varsInResCsv** (\$fn)
- **inDB** (\$Table)
- **updateDBvarsAndPars** (\$fn)
- **compareUpdateXMLDB** (\$Xml, \$Table)
- **upload** ()
- **displayFD** (\$FD)
- **display** ()

Atributos públicos

- **\$filedata**

3.9.1. Descripción detallada

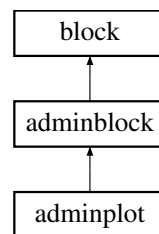
Definición en la línea 4 del archivo adminmodelfiles.php.

La documentación para esta clase fue generada a partir del siguiente fichero:

- admin/adminmodelfiles.php

3.10. Referencia de la Clase adminplot

Diagrama de herencias de adminplot



Métodos públicos

- **createFields** ()
- **createRelations1N1N** ()
- **findModelId** ()
- **findAutoscale** (\$axis)
- **displayPostOwn** ()
- **displayPlot** ()
- **update** ()

3.10.1. Descripción detallada

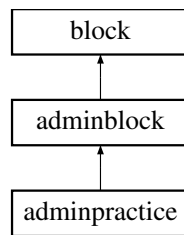
Definición en la línea 5 del archivo adminplot.php.

La documentación para esta clase fue generada a partir del siguiente fichero:

- admin/adminplot.php

3.11. Referencia de la Clase adminpractice

Diagrama de herencias de adminpractice



Métodos públicos

- `createFields ()`

3.11.1. Descripción detallada

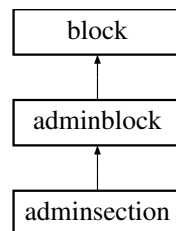
Definición en la línea 4 del archivo adminpractice.php.

La documentación para esta clase fue generada a partir del siguiente fichero:

- admin/adminpractice.php

3.12. Referencia de la Clase adminsection

Diagrama de herencias de adminsection



Métodos públicos

- `createFields ()`
- `createRelations1N ()`

3.12.1. Descripción detallada

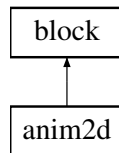
Definición en la línea 4 del archivo adminsection.php.

La documentación para esta clase fue generada a partir del siguiente fichero:

- admin/adminsection.php

3.13. Referencia de la Clase anim2d

Diagrama de herencias de anim2d



Métodos públicos

- **displayAnim2d** (\$animation2d)
- **display** ()

3.13.1. Descripción detallada

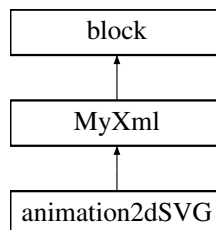
Definición en la línea 6 del archivo anim2d.php.

La documentación para esta clase fue generada a partir del siguiente fichero:

- anim2d.php

3.14. Referencia de la Clase animation2dSVG

Diagrama de herencias de animation2dSVG



Métodos públicos

- **animation2dSVG** ()
- **insertAnimation** (&\$node, \$subtype, \$modification, \$subnode, \$number)
- **modifyXml** (&\$node, \$modification)
- **colorScale** (\$value, \$effect)
- **sync** (\$effect, \$animation)
- **readModifications** (\$animation)

- **modify ()**
- **animateObject** (\$animation, \$results_file)

Atributos públicos

- **\$Modifications**
- **\$Data**

3.14.1. Descripción detallada

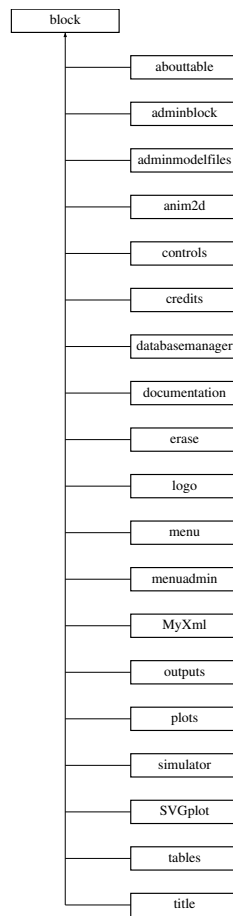
Definición en la línea 5 del archivo `svganim2d.php`.

La documentación para esta clase fue generada a partir del siguiente fichero:

- `svganim2d.php`

3.15. Referencia de la Clase block

Diagrama de herencias de block



Métodos públicos

- **block** ()
- **connect** ()
- **text** (\$str)
- **validateValue** (\$type, \$par_id, \$new_value, \$default_value)
- **opener** ()
- **closer** ()
- **htmlSimpleOpen** (\$Xml, \$nivel)
- **htmlSimpleClose** (\$Xml, \$nivel)
- **htmlBlock** (\$Xml, \$nivel)
- **html** (\$xmlFN)
- **readXml** (\$fn)
- **enabled** (\$table)
- **x_id** (\$key)
- **modeller_id** ()
- **c2deffect_id** ()

- `c2danimation_id ()`
- `control_id ()`
- `practice_id ()`
- `control_group_id ()`
- `curve_id ()`
- `plot_id ()`
- `model_id ()`
- `section_id ()`
- `display ()`

Atributos públicos

- `$Xml`
- `$strings`
- `$configurationSettings`

3.15.1. Descripción detallada

Definición en la línea 5 del archivo `block.php`.

La documentación para esta clase fue generada a partir del siguiente fichero:

- `block.php`

3.16. Referencia de la Clase `configuration`

Métodos públicos

- `readconfig ($configfile)`
- `writeconfig ($configfile, $changes)`

3.16.1. Descripción detallada

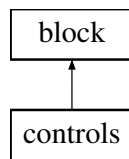
Definición en la línea 2 del archivo `config.class.php`.

La documentación para esta clase fue generada a partir del siguiente fichero:

- `config/config.class.php`

3.17. Referencia de la Clase `controls`

Diagrama de herencias de `controls`



Métodos públicos

- **displayControl** (\$data)
- **displaySelect** (\$data)
- **displayControls** (\$control_group_id)
- **datos** (\$model_id)
- **javascript** ()
- **ecmascript** ()
- **display** ()

Atributos públicos

- **\$datos**

3.17.1. Descripción detallada

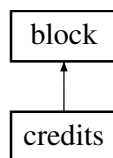
Definición en la línea 4 del archivo controls.php.

La documentación para esta clase fue generada a partir del siguiente fichero:

- controls.php

3.18. Referencia de la Clase credits

Diagrama de herencias de credits



Métodos públicos

- **display** ()

3.18.1. Descripción detallada

Definición en la línea 4 del archivo credits.php.

La documentación para esta clase fue generada a partir del siguiente fichero:

- credits.php

3.19. Referencia de la Clase csvreader

Métodos públicos

- **readCsv** (\$fn)
- **readSync** (\$fn)

3.19.1. Descripción detallada

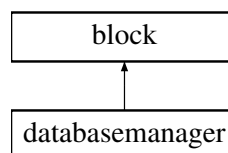
Definición en la línea 3 del archivo csvreader.php.

La documentación para esta clase fue generada a partir del siguiente fichero:

- csvreader.php

3.20. Referencia de la Clase databasemanager

Diagrama de herencias de databasemanager



Métodos públicos

- **update** (\$table, \$idname, \$idvalue)
- **update1NN1** (\$idvalue)
- **insert** ()
- **insert1NN1** (\$idvalue)
- **delete** ()
- **delete1NN1** (\$idvalue)
- **createDirectories** (\$id)
- **deleteDir** (\$dir)
- **deleteDirectories** (\$id)
- **display** ()

3.20.1. Descripción detallada

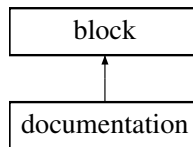
Definición en la línea 4 del archivo databasemanager.php.

La documentación para esta clase fue generada a partir del siguiente fichero:

- admin/databasemanager.php

3.21. Referencia de la Clase documentation

Diagrama de herencias de documentation



Métodos públicos

- `display ()`

3.21.1. Descripción detallada

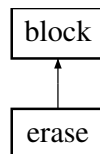
Definición en la línea 4 del archivo documentation.php.

La documentación para esta clase fue generada a partir del siguiente fichero:

- documentation.php

3.22. Referencia de la Clase erase

Diagrama de herencias de erase



Métodos públicos

- `display ()`

3.22.1. Descripción detallada

Definición en la línea 4 del archivo erase.php.

La documentación para esta clase fue generada a partir del siguiente fichero:

- erase.php

3.23. Referencia de la Clase export

Métodos públicos

- **export** ()
- **connect** ()
- **exportLine** (\$table, \$line)
- **exportModellersModels** (\$id)
- **exportTableArray** (\$tn, \$idn, \$id)
- **exportTable** (\$tn, \$idn, \$id)
- **exportTable2** (\$tn, \$idn, \$mid)
- **exportTables** (\$modelID)
- **exportStr** ()

Atributos públicos

- **\$tables**
- **\$configurationSettings**

3.23.1. Descripción detallada

Definición en la línea 4 del archivo exportModel.php.

La documentación para esta clase fue generada a partir del siguiente fichero:

- samples/exportModel.php

3.24. Referencia de la Clase import

Métodos públicos

- **import** ()
- **connect** ()
- **importTable** (\$tn, \$eqs)
- **findRootSection** ()
- **importTables** ()
- **importStr** (\$fn)

Atributos públicos

- `$tables`
- `$id_eq`
- `$configurationSettings`

3.24.1. Descripción detallada

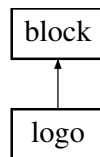
Definición en la línea 4 del archivo `importModel.php`.

La documentación para esta clase fue generada a partir del siguiente fichero:

- `samples/importModel.php`

3.25. Referencia de la Clase logo

Diagrama de herencias de logo



Métodos públicos

- `display ()`

3.25.1. Descripción detallada

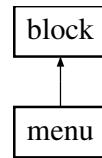
Definición en la línea 4 del archivo `logo.php`.

La documentación para esta clase fue generada a partir del siguiente fichero:

- `logo.php`

3.26. Referencia de la Clase menu

Diagrama de herencias de menu



Métodos públicos

- **display ()**

3.26.1. Descripción detallada

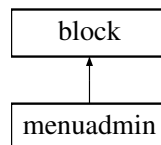
Definición en la línea 5 del archivo menu.php.

La documentación para esta clase fue generada a partir del siguiente fichero:

- menu.php

3.27. Referencia de la Clase menuadmin

Diagrama de herencias de menuadmin



Métodos públicos

- **display ()**

3.27.1. Descripción detallada

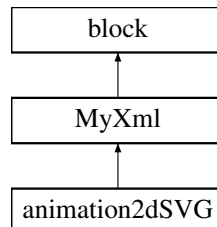
Definición en la línea 5 del archivo menuadmin.php.

La documentación para esta clase fue generada a partir del siguiente fichero:

- admin/menuadmin.php

3.28. Referencia de la Clase MyXml

Diagrama de herencias de MyXml



Métodos públicos

- **readXmlFile** (\$fn)
- **readXmlString** (\$xmlstr)
- **tab** (\$tab)
- **writeXmlFile** (\$fn)
- **writeXmlString** ()
- **cleanString** (\$strin)
- **writeNodeAttributes** (\$node, \$tab)
- **writeNodeValue** (\$node)
- **writeNodeString** (\$type, \$node, \$tab)
- **insertNode** (&\$node, \$type, \$subnode)
- **removeNode** (&\$node, \$type, \$subnode)

Atributos públicos

- **\$tree**

3.28.1. Descripción detallada

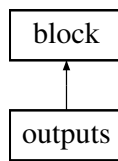
Definición en la línea 4 del archivo MyXml.php.

La documentación para esta clase fue generada a partir del siguiente fichero:

- MyXml.php

3.29. Referencia de la Clase outputs

Diagrama de herencias de outputs



Métodos públicos

- **display ()**

3.29.1. Descripción detallada

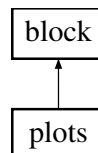
Definición en la línea 4 del archivo outputs.php.

La documentación para esta clase fue generada a partir del siguiente fichero:

- outputs.php

3.30. Referencia de la Clase plots

Diagrama de herencias de plots



Métodos públicos

- **displayPlot (\$plot)**
- **display ()**

3.30.1. Descripción detallada

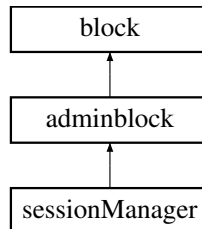
Definición en la línea 6 del archivo plots.php.

La documentación para esta clase fue generada a partir del siguiente fichero:

- plots.php

3.31. Referencia de la Clase sessionManager

Diagrama de herencias de sessionManager



Métodos públicos

- **verify ()**

3.31.1. Descripción detallada

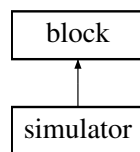
Definición en la línea 4 del archivo session.php.

La documentación para esta clase fue generada a partir del siguiente fichero:

- admin/session.php

3.32. Referencia de la Clase simulator

Diagrama de herencias de simulator



Métodos públicos

- **exeFileName** (\$model_id)
- **readInitXml** (\$fn)
- **findModelicaName** (\$var_id)
- **changeOutputFilter** (\$model_id)
- **changeValues** ()
- **tab** (\$tab)
- **encabezadoXml** ()

- **atributosXml** (\$att, \$tab)
- **cadenaTipo** (\$t, \$xml, \$tab)
- **cadenaScalarXml** (\$xml, \$tab)
- **scalarVarsXml** (\$xml, \$tab)
- **cadenaModelXml** (\$xml, \$tab)
- **cadenaExpXml** (\$xml, \$tab)
- **cadenaFmiXml** (\$xml, \$tab)
- **writelnInitXml** (\$fn, \$tab)
- **runSimulation** (\$model_id, \$exename, \$tplninitFile)
- **simulate** (\$model_id)

Atributos públicos

- **\$XML**

3.32.1. Descripción detallada

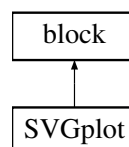
Definición en la línea 5 del archivo simulator.php.

La documentación para esta clase fue generada a partir del siguiente fichero:

- simulator.php

3.33. Referencia de la Clase SVGplot

Diagrama de herencias de SVGplot



Métodos públicos

- **SVGplot** ()
- **settings** (\$settings)
- **title** ()
- **grid** ()
- **curves** ()
- **frame** ()
- **strNum** (\$x)
- **tagH** ()
- **tagV** ()

- **plot ()**
- **legend ()**
- **display ()**
- **data (\$plot, \$resfile)**

Atributos públicos

- **\$svgStr = ""**
- **\$totalWidth = 300**
- **\$totalHeight = 200**
- **\$titleHeight = 10**
- **\$plotHeight = 80**
- **\$legendHeight = 5**
- **\$framePositionX = 20**
- **\$framePositionY = 5**
- **\$frameWidth = 75**
- **\$frameHeight = 85**
- **\$tagHeight = 10**
- **\$tagWidth = 20**
- **\$legendSeparation = 20**
- **\$title = ""**
- **\$description = ""**
- **\$gridX = 5**
- **\$gridY = 3**
- **\$autoscaleX = '1'**
- **\$autoscaleY = '1'**
- **\$minX = 0.0**
- **\$maxX = 1.0**
- **\$minY = 0.0**
- **\$maxY = 1.0**
- **\$data**
- **\$colors**
- **\$legends**
- **\$units**
- **\$descriptions**
- **\$legendH = ""**
- **\$unitsH = ""**

3.33.1. Descripción detallada

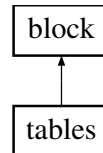
Definición en la línea 5 del archivo `svgplot.php`.

La documentación para esta clase fue generada a partir del siguiente fichero:

- `svgplot.php`

3.34. Referencia de la Clase tables

Diagrama de herencias de tables



Métodos públicos

- **showTable** (\$fn)
- **units** (\$csvLine)
- **display** ()

3.34.1. Descripción detallada

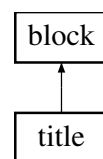
Definición en la línea 5 del archivo tables.php.

La documentación para esta clase fue generada a partir del siguiente fichero:

- tables.php

3.35. Referencia de la Clase title

Diagrama de herencias de title



Métodos públicos

- **display** ()

3.35.1. Descripción detallada

Definición en la línea 4 del archivo title.php.

La documentación para esta clase fue generada a partir del siguiente fichero:

- title.php

3.36. Referencia de la Clase tlistemysql

Métodos públicos

- **tlistemysql** (\$idd="", \$sf="", \$cc="p1", \$sqldata, \$condition="")
- **getParameter** (\$sf)
- **readMysql** (\$sqldata, \$id, \$level)
- **display** ()
- **addElt** (\$xlt)
- **initBar** ()
- **restoreNode** ()
- **saveNode** ()

Atributos públicos

- **\$_id** = ""
- **\$style**
- **\$imagePath**
- **\$sFile**
- **\$sTarget**
- **\$node**
- **\$url**
- **\$param**
- **\$arElt**
- **\$nbElt** = 0
- **\$lnode** = 0
- **\$iOpen** = 0
- **\$condition** = ""

3.36.1. Descripción detallada

Definición en la línea 18 del archivo tlistemysql.php.

3.36.2. Documentación de las funciones miembro

3.36.2.1. tlistemysql::restoreNode ()

restoreNode : restore nodes states

Definición en la línea 273 del archivo tlistemysql.php.

```

{
// global $lnode, $node, $arElt ;

if (is_null($this->node) || trim($this->node) === "") return ;

$this->lnode = doubleval($this->node) ;
$iNo = 0 ;

for ($i=0; $i<count($this->arElt); $i=$i+1)
{
    $elt = $this->arElt[$i] ;
    if ( $elt->iS > -1)
    {
        $l = $this->lnode & (1 << $iNo) ;
        if ($l)
            $elt->open() ;
        else
            $elt->close() ;
        $iNo++ ;
    }
}
}

```

3.36.2.2. tlistemysql::saveNode ()

saveNode : build lnode value

Definición en la línea 300 del archivo tlistemysql.php.

```

{
// global $lnode, $arElt ;

/* for each elements */
$iNo = 0 ;

for ($i=0; $i<count($this->arElt); $i=$i+1)
{
    $elt = $this->arElt[$i] ;
    if ( $elt->iS > -1)
    {
        if ( $elt->iS > 0)
        {
            $this->lnode = $this->lnode | (1 << $iNo) ;
        }
        $iNo++ ;
    }
}
}

```

La documentación para esta clase fue generada a partir del siguiente fichero:

- phptliste/tlistemysql.php

3.37. Referencia de la Clase XMLThing

Métodos públicos

- [XMLThing](#) (\$xml=NULL)
- [parse](#) (\$xml=NULL)
- [parse_recurse](#) ()
- [parse_init](#) ()

Atributos públicos

- **\$rawXML**
- **\$valueArray** = array()
- **\$keyArray** = array()
- **\$parsed** = array()
- **\$index** = 0
- **\$attribKey** = 'attributes'
- **\$valueKey** = 'value'
- **\$cdataKey** = 'cdata'
- **\$isError** = false
- **\$error** = ""

3.37.1. Descripción detallada

Autor

wickedfather at hotmail.com

PHP4 and PHP5 compatible

14th Apr cdataKey index for open tags fixed Class to parse xml into an array the way I like it.

Autor

wickedfather at hotmail.com

Versión

0.0.2

Definición en la línea 17 del archivo xmlthing.class.php.

3.37.2. Documentación de las funciones miembro

3.37.2.1. XMLThing::parse (\$ xml = NULL)

Turns xml into a lovely array

Parámetros

<i>string</i>	option xml to parse. That supplied to constructor used otherwise.
---------------	---

Devuelve

array parsed data

Definición en la línea 61 del archivo xmlthing.class.php.

```

{
    if (!is_null($xml))
    {
        $this->rawXML = $xml;
    }

    $this->isError = false;
    // setup
    if (!$this->parse_init())
    {
        return false;
    }

    $this->index = 0;
    $this->parsed = $this->parse_recurse();
    $this->status = 'parsing complete';

    return $this->parsed;
}

```

3.37.2.2. XMLThing::parse_init()

Turns the xml into the array pairs with xml_parse_into_struct

Devuelve

bool

Definición en la línea 186 del archivo xmlthing.class.php.

```

{
    $this->parser = xml_parser_create();

    $parser = $this->parser;
    xml_parser_set_option($parser, XML_OPTION_CASE_FOLDING, 0); // Dont
mess with my cAsE sEtTings
    xml_parser_set_option($parser, XML_OPTION_SKIP_WHITE, 1);
// Dont bother with empty info
    if (!$res = (bool)xml_parse_into_struct($parser, $this->rawXML, $this->
valueArray, $this->keyArray))
    {
        $this->isError = true;
        $this->error = 'error: '.xml_error_string(xml_get_error_code(
$parser)).' at line '.xml_get_current_line_number($parser);
    }
    xml_parser_free($parser);

    return $res;
}

```


3.37.2.3. XMLThing::parse_recurse ()

Turns the struct array into a nested one

Devuelve

array

Definición en la línea 88 del archivo xmlthing.class.php.

```

    {
        // data found at this level
        $found = array();
        // count of the number of times a tag has been found at
this level
        $tagCount = array();

        // loop through the tags - I use a lazy referencing
thing for where data should go

        while (isset($this->valueArray[$this->index]))
        {
            $tag = $this->valueArray[$this->index];
            $this->index++;

            // if it's a close then return straight away

            if ($tag['type'] == 'close')
            {
                return $found;
            }

            // if it's cdata translate it as a complete tag
named $this->cdataKey
            if ($tag['type'] == 'cdata')
            {
                $tag['tag'] = $this->cdataKey;
                $tag['type'] = 'complete';
            }

            $tagName = $tag['tag'];

            // this bit below creates a reference to where
the data should be going,
            // it saves on some conditions in the switch,
but it probably annoyingly obfuscating
            // has this tag already appeared at this level
? If so we need mods

            if (isset($tagCount[$tagName]))
            {
                // transform to an array IF only one of
the tags been before
                if ($tagCount[$tagName] == 1)
                {
                    $found[$tagName] = array($found[
$tagName]);
                }

                // $found[$tagName][$tagCount[$tagName]]
= ''; // doesn't need to happen, the reference below is enough
                $tagRef =& $found[$tagName][$tagCount[$tagName]]
            ];

            $tagCount[$tagName]++;

```

```

    }
    else // just use the variable
    {
        $tagCount[$tagName] = 1;
        $tagRef =& $found[$tagName];
    }

    // now do the work

    switch ($tag['type'])
    {
        case 'open':
            $tagRef = $this->parse_recurse();

            if (isset($tag['attributes']))
            {
                $tagRef[$this->attribKey] =
$tag['attributes'];
            }
            // open CAN have a value - it
            makes sense to include it as cdata
            if (isset($tag['value']))
            {
                if (isset($tagRef[$this->
cdataKey])) // push it to the start of the cdata array if it's present
                {
                    $tagRef[$this->cdataKey
] = (array)$tagRef[$this->cdataKey]; // <-- needed for php5 compatibility];
                    array_unshift($tagRef[
$this->cdataKey], $tag['value']);
                }
                else
                {
                    $tagRef[$this->cdataKey
] = $tag['value'];
                }
            }
            break;

        case 'complete':
            if (isset($tag['attributes']))
            {
                $tagRef[$this->attribKey] =
$tag['attributes'];
                $tagRef =& $tagRef[$this->
valueKey];
            }
            if (isset($tag['value']))
            {
                $tagRef = $tag['value'];
            }
            break;
    }
}

return $found;
}

```

3.37.2.4. XMLThing::XMLThing (\$xml = NULL)

Constructor

Parámetros

<i>string</i>	xml data to parse - optional
---------------	------------------------------

Definición en la línea 49 del archivo xmlthing.class.php.

```
{
    $this->rawXML = $xml;
}
```

La documentación para esta clase fue generada a partir del siguiente fichero:

- xmlthing.class.php