

# I-SIMPA Scripting Guide

1.1.4

Généré par Doxygen 1.6.1

Tue Jan 19 09 :44 :49 2010



# Table des matières

<b>1</b>	<b>Guide de création de script I-SIMPA</b>	<b>1</b>
<b>2</b>	<b>Ajouter des fonctionnalités</b>	<b>3</b>
2.1	Enregistrer les événements . . . . .	4
2.2	Enregistrer un nouveau gestionnaire de menu . . . . .	4
2.3	Créer la méthode "getmenu" . . . . .	5
2.4	Executer du code lors de la mise à jour d'un élément . . . . .	5
<b>3</b>	<b>Ajouter des données dans les projets</b>	<b>7</b>
3.1	Créer un nouveau type d'élément . . . . .	8
3.1.1	Libellé de l'élément dans l'arbre du projet . . . . .	9
3.1.2	Icone . . . . .	9
3.1.3	Evenement de mise à jour . . . . .	9
3.2	Enregistrer le nouveau module . . . . .	10
<b>4</b>	<b>Index des espaces de nommage</b>	<b>11</b>
4.1	Liste des espaces de nommage . . . . .	11
<b>5</b>	<b>Index des classes</b>	<b>13</b>
5.1	Hiérarchie des classes . . . . .	13
<b>6</b>	<b>Index des classes</b>	<b>15</b>
6.1	Liste des classes . . . . .	15
<b>7</b>	<b>Documentation des espaces de nommage</b>	<b>17</b>
7.1	Référence de l'espace de nommage uictrl . . . . .	17
7.1.1	Description détaillée . . . . .	20
7.1.2	Documentation du type de l'énumération . . . . .	20
7.1.2.1	element_type . . . . .	20
7.1.2.2	graph . . . . .	23

7.1.2.3	idevent	23
<b>8</b>	<b>Documentation des classes</b>	<b>25</b>
8.1	Référence de la classe uictrl : :application	25
8.1.1	Description détaillée	25
8.1.2	Documentation des fonctions membres	26
8.1.2.1	clearlogdata	26
8.1.2.2	clearshellhisto	26
8.1.2.3	getcachedir	26
8.1.2.4	getdataarray	26
8.1.2.5	getlastcalculationpath	26
8.1.2.6	getlocale	26
8.1.2.7	getrootcore	26
8.1.2.8	getrootpreference	26
8.1.2.9	getrootreport	27
8.1.2.10	getrootscene	27
8.1.2.11	getuserinput	27
8.1.2.12	importscene	27
8.1.2.13	loadproject	28
8.1.2.14	newproject	28
8.1.2.15	register_event	28
8.1.2.16	register_menu_manager	28
8.1.2.17	reloadgroupsfrommodel	28
8.1.2.18	savelog	28
8.1.2.19	saveproject	28
8.1.2.20	saveshell	29
8.1.2.21	sendevent	29
8.2	Référence de la classe uictrl : :e_file	30
8.2.1	Description détaillée	30
8.2.2	Documentation des fonctions membres	30
8.2.2.1	buildfullpath	30
8.3	Référence de la structure uictrl : :Element	31
8.3.1	Description détaillée	31
8.4	Référence de la classe uictrl : :element	32
8.4.1	Description détaillée	33
8.4.2	Documentation des fonctions membres	33
8.4.2.1	appendfilsbytype	33

8.4.2.2	appendpropertybool	33
8.4.2.3	appendpropertycolor	34
8.4.2.4	appendpropertydecimal	34
8.4.2.5	appendpropertyentier	34
8.4.2.6	appendpropertyfont	35
8.4.2.7	appendpropertylist	35
8.4.2.8	appendpropertyposition	35
8.4.2.9	appendpropertytext	36
8.4.2.10	appenduserelement	36
8.4.2.11	childs	36
8.4.2.12	deleteallelementbytype	36
8.4.2.13	deleteallelementbytyper	37
8.4.2.14	deleteelementbyxmlid	37
8.4.2.15	getallelementbytype	37
8.4.2.16	getboolconfig	37
8.4.2.17	getcolorconfig	37
8.4.2.18	getdecimalconfig	37
8.4.2.19	getelementbylibelle	38
8.4.2.20	getelementbytype	38
8.4.2.21	getentierconfig	38
8.4.2.22	getid	38
8.4.2.23	getinfos	38
8.4.2.24	getlistconfig	38
8.4.2.25	getmenu	39
8.4.2.26	getpositionconfig	39
8.4.2.27	getstringconfig	39
8.4.2.28	hasproperty	39
8.4.2.29	hide	39
8.4.2.30	modified	39
8.4.2.31	register_update_manager	40
8.4.2.32	setreadonlyallconfig	40
8.4.2.33	setreadonlyconfig	40
8.4.2.34	updateboolconfig	40
8.4.2.35	updatedecimalconfig	40
8.4.2.36	updateentierconfig	41
8.4.2.37	updatelistconfig	41

8.4.2.38	<a href="#">updatepositionconfig</a> . . . . .	41
8.4.2.39	<a href="#">updatestringconfig</a> . . . . .	41

# Chapitre 1

## Guide de création de script I-SIMPA

Il y a deux finalités à la création de script dans I-SIMPA La première vise à ajouter des fonctionnalités au logiciel via des menus supplémentaire lors-ce que l'on fait un clique droit sur un élément. On peut également ajouter des données aux arbres du projet.

- [Ajouter des fonctionnalités](#)
- [Ajouter des données dans les projets](#)





## **Chapitre 2**

# **Ajouter des fonctionnalités**

L'ajout d'une nouvelle fonction se fait en 4 étapes :

- Créer un dossier(module) dans le dossier UserScript// et ajouter un fichier dans ce dossier nommé "\_\_ui\_startup.py" qui va importer le module.
- [Enregistrer les événements](#) lors de l'initialisation du gestionnaire de menu.
- [Enregistrer un nouveau gestionnaire de menu](#) dans le fichier \_\_init\_\_.py
- [Créer la méthode "getmenu"](#) qui va ajouter les nouvelles fonctions dans la structure de données en paramètre.

## 2.1 Enregistrer les événements

Chaque fonction implémenté en python doit être associé à un indice. Cet indice est obtenu grâce à l'enregistrement de cette fonction. Cette indice sera utilisé ultérieurement dans la structure du menu à la dernière étape

Vous devez appeler la méthode `uictrl : :application : :register_event` et stocker l'indice retourné.

### Exemple

```
class manager:
    """
    This class make the user able to enable or disable a group of emitters with one click only
    """
    def __init__(self):
        """
        Constructor. Register the two new menu functions
        """
        self.enable_grp_sourcesid=uictrl.application.register_event(self.enable_grp_sources)
        self.disable_grp_sourcesid=uictrl.application.register_event(self.disable_grp_sources)
    def set_grp_src_activation(self, idgrp, newstate):
        grpsrc=uictrl.element(idgrp)
        all_property=grpsrc.getallementbytype(uictrl.element_type.ELEMENT_TYPE_SCENE_SOURCES_SOURCE_PROPRIETES)
        for prop in all_property:
            uictrl.element(prop).updateboolconfig("enable", newstate)
    def enable_grp_sources(self, idgrp):
        """
        Called by user interface when the user click on the enable menu item
        """
        self.set_grp_src_activation(idgrp, True)
    def disable_grp_sources(self, idgrp):
        """
        Called by user interface when the user click on the disable menu item
        """
        self.set_grp_src_activation(idgrp, False)
```

## 2.2 Enregistrer un nouveau gestionnaire de menu

La méthode `uictrl : :application : :register_menu_manager` permet de déclarer un nouveau gestionnaire de menu.

### Exemple

```
uictrl.application.register_menu_manager(uictrl.element_type.ELEMENT_TYPE_SCENE_SOURCES, manager())
```

- Le premier paramètre `uictrl : :element_type` indique le type d'élément associé avec le gestionnaire.
- Le deuxième paramètre correspond à l'instance du gestionnaire

## 2.3 Créer la méthode "getmenu"

Quand l'utilisateur fait un clic droit sur un élément de l'arbre du projet I-SIMPA appelle la méthode `getmenu` pour chaque gestionnaire lié à ce type d'élément.

Votre méthode doit retourner vrai afin de prendre en compte les modifications.

### Exemple

```
def getmenu(self, typeel, idel, menu) :
    """
        Called by the user interface
        The list menu structure contains the current implemented functions.
    """
    submenu=[(uictrl._("Enable"), self.enable_grp_sourcesid), (uictrl._("Disabl
e"), self.disable_grp_sourcesid)]
    menu.insert(2, (uictrl._("All emitters"), submenu))
    menu.insert(2, ())
    return True
```

## 2.4 Executer du code lors de la mise à jour d'un élément

Vous pouvez lier une méthode python à n'importe quel élément de l'application, quand cet élément sera modifié par le programme ou par l'utilisateur alors cette méthode sera appelé avec en paramètre l'indice de l'élément mis à jour (indice même de l'élément ou d'un de ses fils) La méthode `uictrl : :element : :register_update_manager` vous permet d'effectuer cette opération.



## **Chapitre 3**

# **Ajouter des données dans les projets**

Ajouter des propriétés aux éléments n'est pas suffisant.

Vous devez prendre le contrôle de ses élément afin d'implémenter des contraintes de propriétés et pour ajouter plus de fonctionnalités.

Les étapes pour ajouter un nouvel élément sont :

- Créer un dossier dans UserScript/
- [Créer un nouveau type d'élément](#) dans le dossier UserScript/votre\_module/\_\_\_init\_\_.py
- [Enregistrer le nouveau module](#) dans le script \_\_\_project\_loading\_\_.py

### 3.1 Créer un nouveau type d'élément

L'exemple de notre guide est la création d'un lien entre les projets et un nouveau code de calcul. En premier lieu, il faut créer une classe héritant de `uictrl : :element` et ayant comme type d'élément de base `uictrl : :element_type : :ELEMENT_TYPE_CORE_CORE`.

#### Exemple

```
class mdf(uictrl.element):
    """
    Diffusion model calculation core.
    """
    def __init__(self, idel):
        uictrl.element.__init__(self, idel)

        if not self.hasproperty("exeName"): #Test if this is a new project initialization
            #If this is a new project then we add properties
            #Add tetgen parameters
            self.appendfilsbytype(uictrl.element_type.ELEMENT_TYPE_CORE_CORE_CONF
            MAILLAGE)
            #Add frequencies selection
            self.appendfilsbytype(uictrl.element_type.ELEMENT_TYPE_CORE_CORE_BFRE
            QSELECTION)
            #Add configuration core
            coreconf=uictrl.element(self.appendfilsbytype(uictrl.element_type.ELE
            MENT_TYPE_CORE_CORE_CONFIG))
            #Append hidden config, used by I-SIMPA to find the core files and bin
            aries
            uictrl.element(self.appendpropertytext("modelName", "", "mesh.cbin", Tru
            e, True)).hide()
            uictrl.element(self.appendpropertytext("tetrameshFileName", "", "tetram
            esh.mbin", True, True)).hide()
            uictrl.element(self.appendpropertytext("exeName", "", "md.py")).hide()
            uictrl.element(self.appendpropertytext("corePath", "", "md\\")).hide()

            #User options
            coreconf.appendpropertylist("solver_mode", "Calculation mode", [{"Time"
            , "Static"}, [0,1]], 0, False, 1, True)
            coreconf.appendpropertybool("with_direct_sound", "Use direct sound", Tr
            ue, True)
            _("Calculation mode")
            _("Use direct sound")
            _("Time")
            _("Static")

        else:
            pass #Here in case of loading an existing project
```

Dans le constructeur `__init__` les propriétés sont ajoutées. Toutefois rien n'est fait si elles existent déjà car ce constructeur est appelé également lors-ce que l'utilisateur charge un projet déjà existant.

### 3.1.1 Libellé de l'élément dans l'arbre du projet

Par défaut c'est le nom de la classe qui est affiché dans l'arbre du projet. Pour modifier le libellé vous devez surcharger la fonction `gettreelabel`.

#### Exemple

```
def gettreelabel(self):
    """
        Return label
    """
    return "Mdf"
```

### 3.1.2 Icône

Il y a deux types d'icônes :

- Les indices fixes, visible dans l'énumération `uictrl : :graph` et déclaré par `element : :geticonid(self)` qui doit retourner un membre de l'énumération.
- La déclaration locale des icônes via la fonction `element : :geticonpath(self)` qui doit retourner le chemin d'accès au fichier d'icône (\*.ico)

#### Exemple

```
def geticonid(self, state_open):
    """
        Return tree icon Id
    """
    if state_open:
        return uictrl.graph.GRAPH_FOLDER_OPEN
```

### 3.1.3 Evenement de mise à jour

Dès que l'utilisateur modifie une propriété l'application appelle la méthode `uictrl : :element : :modified` à partir de la propriété jusqu'à l'élément racine du projet.

Cette méthode très utile vous permettra par exemple d'implémenter les contraintes dans les propriétés.

Dans l'exemple, elle est utilisée afin de désactiver l'accès à la modification des propriétés dépendantes du temps si l'utilisateur sélectionne une méthode de résolution statique.

#### Exemple

```
def modified(self, idelmodified):
    #In case of sub element modification this func is call by ui
    #We disable the time dependant parameters in case of static solver mode
    if uictrl.element(idelmodified).getinfos()["name"]=="solver_mode":
        elconf=uictrl.element(self.getelementbytype(uictrl.element_type.ELEME
NT_TYPE_CORE_CORE_CONFIG))
        is_temporal=(elconf.getlistconfig("solver_mode")==0)
        elconf.setreadonlyconfig("duree_simulation",not is_temporal)
        elconf.setreadonlyconfig("pasdetemps",not is_temporal)
        uictrl.element.modified(self, idelmodified)
```

## 3.2 Enregistrer le nouveau module

Lors-ce que l'utilisateur créé ou charge un projet tout les fichiers nommés "\_\_project\_loading\_\_.py" et présent dans le dossier UserScript/ et ses sous dossiers sont exécutés.

Il est nécessaire de créer ce fichier dans votre module afin d'ajouter celui-ci dans les projets.

Il faut au préalable vérifier que l'ajout n'a pas été déjà effectué. Dès lors il faut utiliser la méthode `uicontrol : :element : :appenduserelement`.

### Note:

Vous pouvez utiliser la librairie `uicontrol` déjà importé sous le nom `ui`.

### Exemple

```
rootcore=ui.element(ui.application.getrootcore())
#Check if our mod has been already inserted
if rootcore.getelementbylibelle("mdf")==-1: #Then append our mod
    rootcore.appenduserelement(ui.element_type.ELEMENT_TYPE_CORE_CORE,"mdf","mdf"
    )
```



# Chapitre 4

## Index des espaces de nommage

### 4.1 Liste des espaces de nommage

Liste de tous les espaces de nommage documentés avec une brève description :

<a href="#">uictrl</a> (Portage des fonctions c++ vers python) . . . . .	17
--	----



# Chapitre 5

## Index des classes

### 5.1 Hiérarchie des classes

Cette liste d'héritage est classée approximativement par ordre alphabétique :

uictrl : :application . . . . .	25
uictrl : :Element . . . . .	31
uictrl : :element . . . . .	32
uictrl : :e_file . . . . .	30



# Chapitre 6

## Index des classes

### 6.1 Liste des classes

Liste des classes, structures, unions et interfaces avec une brève description :

<a href="#">uictrl : :application</a> (Classe python de contrôle de I-SIMPA ) . . . . .	25
<a href="#">uictrl : :e_file</a> . . . . .	30
<a href="#">uictrl : :Element</a> (Alias) . . . . .	31
<a href="#">uictrl : :element</a> (Contrôle d'un élément implémenté en python ou c++) . . . . .	32



# Chapitre 7

## Documentation des espaces de nommage

### 7.1 Référence de l'espace de nommage uictrl

Portage des fonctions c++ vers python.

#### Classes

- class `application`  
*Classe python de contrôle de I-SIMPA.*
- class `e_file`
- struct `Element`  
*Alias.*
- class `element`  
*Contrôle d'un élément implémenté en python ou c++.*

#### Énumérations

- enum `element_type` {  
ELEMENT\_TYPE\_CORE\_ROOT, ELEMENT\_TYPE\_RESULT\_ROOT, ELEMENT\_TYPE\_-  
SCENE\_ROOT, ELEMENT\_TYPE\_SCENE\_PROJET\_CONFIGURATION,  
ELEMENT\_TYPE\_SCENE\_GROUPE\_SURFACES, ELEMENT\_TYPE\_SCENE\_-  
GROUPE\_SURFACES\_GROUPE, ELEMENT\_TYPE\_SCENE\_GROUPE\_SURFACES\_GROUPE\_-  
VERTEX, ELEMENT\_TYPE\_SCENE\_RECEPTEURSP,  
ELEMENT\_TYPE\_SCENE\_RECEPTEURSP\_RECEPTEUR, ELEMENT\_TYPE\_SCENE\_-  
RECEPTEURSP\_RECEPTEUR\_PROPRIETES, ELEMENT\_TYPE\_SCENE\_RECEPTEURSP\_-  
RECEPTEUR\_RENDU, ELEMENT\_TYPE\_SCENE\_RECEPTEURSS,  
ELEMENT\_TYPE\_SCENE\_RECEPTEURSS\_RECEPTEUR, ELEMENT\_TYPE\_SCENE\_-  
RECEPTEURSS\_RECEPTEUR\_PROPRIETES, ELEMENT\_TYPE\_SCENE\_RECEPTEURSS\_-  
RECEPTEUR\_RENDU, ELEMENT\_TYPE\_SCENE\_SOURCES,  
ELEMENT\_TYPE\_SCENE\_SOURCES\_SOURCE, ELEMENT\_TYPE\_SCENE\_SOURCES\_-  
SOURCE\_PROPRIETES, ELEMENT\_TYPE\_SCENE\_SOURCES\_SOURCE\_PUISSANCE,  
ELEMENT\_TYPE\_SCENE\_SOURCES\_SOURCE\_RENDU,

```

ELEMENT_TYPE_TEXT,      ELEMENT_TYPE_COLOR,      ELEMENT_TYPE_POSITION,
ELEMENT_TYPE_LIST,
ELEMENT_TYPE_INTEGER, ELEMENT_TYPE_FLOAT, ELEMENT_TYPE_BOOL, ELEMENT_
TYPE_SCENE_PROJET_RENDU,
ELEMENT_TYPE_SCENE_PROJET_RENDU_ORIGINE, ELEMENT_TYPE_SCENE_PROJET_
RENDU_MODEL,      ELEMENT_TYPE_SCENE_BDD,      ELEMENT_TYPE_SCENE_BDD_
SPECTRUMS,
ELEMENT_TYPE_SCENE_BDD_SPECTRUMS_USER,      ELEMENT_TYPE_SCENE_BDD_
SPECTRUMS_APP, ELEMENT_TYPE_SCENE_BDD_MATERIAUX, ELEMENT_TYPE_SCENE_
BDD_MATERIAUX_APP,
ELEMENT_TYPE_SCENE_BDD_MATERIAUX_APP_GROUP,      ELEMENT_TYPE_SCENE_
BDD_MATERIAUX_APP_MATERIAU,      ELEMENT_TYPE_SCENE_BDD_MATERIAUX_USER,
ELEMENT_TYPE_SCENE_BDD_MATERIAUX_USER_GROUP,
ELEMENT_TYPE_SCENE_BDD_MATERIAUX_USER_MATERIAU,      ELEMENT_TYPE_
SCENE_BDD_MATERIAUX_PROP_MATERIAU,      ELEMENT_TYPE_SCENE_BDD_
CATMATERIAL, ELEMENT_TYPE_MATERIAU_APP,
ELEMENT_TYPE_MATERIAU_USER,      ELEMENT_TYPE_GAMMEFREQ_APP,      ELEMENT_
TYPE_GAMMEFREQ_USER, ELEMENT_TYPE_PROPERTY_FREQ,
ELEMENT_TYPE_ROW,      ELEMENT_TYPE_ROW_BFREQ,      ELEMENT_TYPE_ROW_
MATERIAU, ELEMENT_TYPE_SCENE_ENCOMBREMENTS,
ELEMENT_TYPE_SCENE_ENCOMBREMENTS_ENCOMBREMENT,      ELEMENT_
TYPE_SCENE_ENCOMBREMENTS_ENCOMBREMENT_PROPRIETES,      ELEMENT_
TYPE_SCENE_ENCOMBREMENTS_ENCOMBREMENT_CUBOIDE,      ELEMENT_TYPE_
GAMMEABSORPTION,
ELEMENT_TYPE_CORE_SPPS, ELEMENT_TYPE_CORE_CORE_CONFIG, ELEMENT_TYPE_
CORE_CORE_CONFMAILLAGE, ELEMENT_TYPE_SCENE_PROJET,
ELEMENT_TYPE_SCENE_PROJET_USERCONFIGURATION,      ELEMENT_TYPE_SCENE_
PROJET_RENDU_PARTICULES,      ELEMENT_TYPE_SCENE_DONNEES,      ELEMENT_TYPE_
SCENE_ENCOMBREMENTS_ENCOMBREMENT_RENDU,
ELEMENT_TYPE_SCENE_PROJET_ENVIRONNEMENTCONF, ELEMENT_TYPE_DRAWABLE,
ELEMENT_TYPE_CORE_CORE_BFREQSELECTION, ELEMENT_TYPE_BOOL_BFREQ,
ELEMENT_TYPE_REPORT_FOLDER,      ELEMENT_TYPE_REPORT_PARTVISUALISATION,
ELEMENT_TYPE_REPORT_RECEPTEURSSVISUALISATION,      ELEMENT_TYPE_REPORT_
GABE,
ELEMENT_TYPE_REPORT_GABE_RECIP, ELEMENT_TYPE_TREE_LIST, ELEMENT_TYPE_
CORE_TC, ELEMENT_TYPE_SCENE_PROJET_INFORMATION,
ELEMENT_TYPE_SCENE_BDD_MATERIAUX_MATERIAU_RENDER,      ELEMENT_TYPE_
FONT, ELEMENT_TYPE_CORE_TLM, ELEMENT_TYPE_REPORT_GABE_GAP,
ELEMENT_TYPE_REPORT_UNKNOWN,      ELEMENT_TYPE_CORE_SPPS_OCTREE,
ELEMENT_TYPE_REPORT_RPI, ELEMENT_TYPE_SCENE_VOLUMES,
ELEMENT_TYPE_SCENE_VOLUMES_VOLUME,      ELEMENT_TYPE_SCENE_VOLUMES_
VOLUME_RENDU,      ELEMENT_TYPE_SCENE_VOLUMES_VOLUME_PROPRIETES,
ELEMENT_TYPE_PYTHON_EXTENSION,
ELEMENT_TYPE_ELEMENT, ELEMENT_TYPE_CORE_CORE, ELEMENT_TYPE_REPORT_
FILE, ELEMENT_TYPE_REPORT_RECEPTEURSSVISUALISATION_STANDART,
ELEMENT_TYPE_REPORT_RECEPTEURSSVISUALISATION_GAIN,      ELEMENT_
TYPE_REPORT_RECEPTEURSSVISUALISATION_TR,      ELEMENT_TYPE_REPORT_
RECEPTEURSSVISUALISATION_EDT,      ELEMENT_TYPE_SCENE_RECEPTEURSS_
RECEPTEURCOUPE,
ELEMENT_TYPE_SCENE_RECEPTEURSS_RECEPTEURCOUPE_PROPRIETES,      ELEMENT_
TYPE_SCENE_RECEPTEURSS_RECEPTEURCOUPE_RENDU,      ELEMENT_TYPE_USER_
PREFERENCE_NODE, ELEMENT_TYPE_USER_PREFERENCE_ITEM,
ELEMENT_TYPE_USER_PREFERENCE_ITEM_ISOTEMPLATE }
- enum graph {

```



```

GRAPH_FOLDER, GRAPH_ITEM, GRAPH_FOLDER_OPEN, GRAPH_FITTINGS_OPEN,
GRAPH_FITTINGS_CLOSE, GRAPH_FITTING_OPEN, GRAPH_FITTING_CLOSE, GRAPH_
PUNCTUAL_RECEIVERS_OPEN,
GRAPH_PUNCTUAL_RECEIVERS_CLOSE, GRAPH_SURFACE_RECEIVERS_OPEN, GRAPH_
SURFACE_RECEIVERS_CLOSE, GRAPH_SOUND_SOURCES_OPEN,
GRAPH_SOUND_SOURCES_CLOSE, GRAPH_SURFACES_OPEN, GRAPH_SURFACES_
CLOSE, GRAPH_VOLUMES_CLOSE,
GRAPH_VOLUMES_OPEN, GRAPH_PROJECT_OPEN, GRAPH_PROJECT_CLOSE,
GRAPH_DATA_CLOSE,
GRAPH_DATA_OPEN, GRAPH_USER_MATERIALS_CLOSE, GRAPH_USER_
MATERIALS_OPEN, GRAPH_APPLICATION_MATERIALS_CLOSE,
GRAPH_APPLICATION_MATERIALS_OPEN, GRAPH_DATABASE_CLOSE, GRAPH_
DATABASE_OPEN, GRAPH_USER_SPECTRUMS_CLOSE,
GRAPH_USER_SPECTRUMS_OPEN, GRAPH_APPLICATION_SPECTRUMS_CLOSE,
GRAPH_APPLICATION_SPECTRUMS_OPEN, GRAPH_MATERIAL_CLOSE,
GRAPH_MATERIAL_OPEN, GRAPH_STANDARTCORE_CLOSE, GRAPH_
STANDARTCORE_OPEN, GRAPH_CORES_CLOSE,
GRAPH_CORES_OPEN, GRAPH_SPPSCORE_CLOSE, GRAPH_SPPSCORE_OPEN,
GRAPH_DISK_FOLDER_OPEN,
GRAPH_DISK_FOLDER_CLOSE, GRAPH_ENVIRONMENT, GRAPH_PROJECT_
AUTHOR, GRAPH_INFORMATION,
GRAPH_DISK_DEFAULT_FILE, GRAPH_DISK_GABE, GRAPH_DISK_RS, GRAPH_DISK_
PARTICLE,
GRAPH_EL_CONFIGURATION, GRAPH_EL_3D_DISPLAY, GRAPH_EL_POSITION,
GRAPH_EL_TRIANGLE,
GRAPH_SPECTRUM, GRAPH_ORIGIN, GRAPH_TETMESH_PARAMETERS, GRAPH_
RENDERING_FOLDER_CLOSE,
GRAPH_RENDERING_FOLDER_OPEN, GRAPH_ROOT_MATERIALS_OPEN, GRAPH_
ROOT_MATERIALS_CLOSE, GRAPH_ROOT_SPECTRUMS_OPEN,
GRAPH_ROOT_SPECTRUMS_CLOSE, GRAPH_PUNCTUAL_RECEIVER_OPEN,
GRAPH_PUNCTUAL_RECEIVER_CLOSE, GRAPH_SURFACE_RECEIVER_OPEN,
GRAPH_SURFACE_RECEIVER_CLOSE, GRAPH_SOUND_SOURCE_OPEN, GRAPH_
SOUND_SOURCE_CLOSE, GRAPH_VOLUME_OPEN,
GRAPH_VOLUME_CLOSE, GRAPH_PREF_ANIMATION, GRAPH_PREF_GENERAL,
GRAPH_PREF_LEGEND,
GRAPH_PREF_NOISE_MAP, GRAPH_PREF_PARTICLES, GRAPH_USER_PREF_ROOT_
CLOSE, GRAPH_USER_PREF_ROOT_OPEN,
GRAPH_LAST_STATIC_GRAPH }
enum idevent {
IDEVENT_DELETE_ELEMENT, IDEVENT_RENAME_ELEMENT, IDEVENT_COPIER,
IDEVENT_COLLER,
IDEVENT_NEW_SURFACE_GROUP, IDEVENT_GETPROPERTIES, IDEVENT_SELECT_
TREE_ITEM, IDEVENT_NEW_RECEPTEUR_P,
IDEVENT_SELECT_POSITION, IDEVENT_NEW_SOURCE, IDEVENT_NEW_RECEPTEUR_S,
IDEVENT_NEW_USERFREQ,
IDEVENT_NEW_ENCOMBREMENT, IDEVENT_NEW_ENCOMBREMENT_CUBOIDE,
IDEVENT_NEW_USERMAT, IDEVENT_NEW_MATERIAL_GROUP,
IDEVENT_RUN_CALCULATION, IDEVENT_IMPORT_MATERIAL, IDEVENT_LOAD_
PARTICLE_SIMULATION, IDEVENT_LOAD_PARTICLE_SIMULATION_PATH,
IDEVENT_RELOAD_FOLDER, IDEVENT_DELETE_FOLDER, IDEVENT_LOAD_
RECEPTEURSS_SIMULATION_BY_TIMESTEP, IDEVENT_LOAD_RECEPTEURSS_
SIMULATION_BY_TIMESTEP_SUM,
IDEVENT_LOAD_RECEPTEURSS_SIMULATION_SUM, IDEVENT_INVERT_FACE_
ORIENTATION, IDEVENT_EMPTY_POINTER_VERTEX_GROUP, IDEVENT_RECIP_

```

```

COMPUTE_ACOUSTIC_PARAMETERS,
IDEVENT_RECEPTEURS_COMPUTE_TR, IDEVENT_RECEPTEURS_COMPUTE_EDT,
IDEVENT_REPORT_PARTICULES_MAKE_GABE, IDEVENT_BFREQ_PRESELECTION_
NONE,
IDEVENT_BFREQ_PRESELECTION_THIRD_BAND, IDEVENT_BFREQ_PRESELECTION_
BAND, IDEVENT_BFREQ_PRESELECTION_BUILDING_THIRD_BAND, IDEVENT_BFREQ_
PRESELECTION_BUILDING_BAND,
IDEVENT_RECPC_COMPUTE_ADVANCED_ACOUSTIC_PARAMETERS, IDEVENT_NEW_
RECEPTEURP_GROUP, IDEVENT_NEW_SOURCE_GROUP, IDEVENT_NEW_SURFACE_
GROUP_FROM_SELECTION,
IDEVENT_LOAD_RECEPTEURSP_SIMULATION, IDEVENT_BUILD_VOLUMES_FROM_
TRIMESH, IDEVENT_NEW_VOLUME, IDEVENT_OPEN_FOLDER,
IDEVENT_CONVERT_VOL_TO_FITTING, IDEVENT_NEW_RECEPTEUR_S_COUPE,
IDEVENT_LAST_FIXED }

```

### 7.1.1 Description détaillée

Portage des fonctions c++ vers python.

### 7.1.2 Documentation du type de l'énumération

#### 7.1.2.1 enum uictrl : :element\_type

Enumeration de tout les types d'élément "built-in" existant

Valeurs énumérées :

```

ELEMENT_TYPE_CORE_ROOT Type d'élément
ELEMENT_TYPE_RESULT_ROOT Type d'élément
ELEMENT_TYPE_SCENE_ROOT Type d'élément
ELEMENT_TYPE_SCENE_PROJET_CONFIGURATION Type d'élément
ELEMENT_TYPE_SCENE_GROUPESURFACES Type d'élément
ELEMENT_TYPE_SCENE_GROUPESURFACES_GROUPE Type d'élément
ELEMENT_TYPE_SCENE_GROUPESURFACES_GROUPE_VERTEX Type d'élément
ELEMENT_TYPE_SCENE_RECEPTEURSP Type d'élément
ELEMENT_TYPE_SCENE_RECEPTEURSP_RECEPTEUR Type d'élément
ELEMENT_TYPE_SCENE_RECEPTEURSP_RECEPTEUR_PROPRIETES Type d'élément
ELEMENT_TYPE_SCENE_RECEPTEURSP_RECEPTEUR_RENDU Type d'élément
ELEMENT_TYPE_SCENE_RECEPTEURSS Type d'élément
ELEMENT_TYPE_SCENE_RECEPTEURSS_RECEPTEUR Type d'élément
ELEMENT_TYPE_SCENE_RECEPTEURSS_RECEPTEUR_PROPRIETES Type d'élément
ELEMENT_TYPE_SCENE_RECEPTEURSS_RECEPTEUR_RENDU Type d'élément
ELEMENT_TYPE_SCENE_SOURCES Type d'élément
ELEMENT_TYPE_SCENE_SOURCES_SOURCE Type d'élément
ELEMENT_TYPE_SCENE_SOURCES_SOURCE_PROPRIETES Type d'élément
ELEMENT_TYPE_SCENE_SOURCES_SOURCE_PUISSANCE Type d'élément
ELEMENT_TYPE_SCENE_SOURCES_SOURCE_RENDU Type d'élément
ELEMENT_TYPE_TEXT Type d'élément

```

*ELEMENT\_TYPE\_COLOR* Type d'élément  
*ELEMENT\_TYPE\_POSITION* Type d'élément  
*ELEMENT\_TYPE\_LIST* Type d'élément  
*ELEMENT\_TYPE\_INTEGER* Type d'élément  
*ELEMENT\_TYPE\_FLOAT* Type d'élément  
*ELEMENT\_TYPE\_BOOL* Type d'élément  
*ELEMENT\_TYPE\_SCENE\_PROJET\_RENDU* Type d'élément  
*ELEMENT\_TYPE\_SCENE\_PROJET\_RENDU\_ORIGINE* Type d'élément  
*ELEMENT\_TYPE\_SCENE\_PROJET\_RENDU\_MODEL* Type d'élément  
*ELEMENT\_TYPE\_SCENE\_BDD* Type d'élément  
*ELEMENT\_TYPE\_SCENE\_BDD\_SPECTRUMS* Type d'élément  
*ELEMENT\_TYPE\_SCENE\_BDD\_SPECTRUMS\_USER* Type d'élément  
*ELEMENT\_TYPE\_SCENE\_BDD\_SPECTRUMS\_APP* Type d'élément  
*ELEMENT\_TYPE\_SCENE\_BDD\_MATERIAUX* Type d'élément  
*ELEMENT\_TYPE\_SCENE\_BDD\_MATERIAUX\_APP* Type d'élément  
*ELEMENT\_TYPE\_SCENE\_BDD\_MATERIAUX\_APP\_GROUP* Type d'élément  
*ELEMENT\_TYPE\_SCENE\_BDD\_MATERIAUX\_APP\_MATERIAU* Type d'élément  
*ELEMENT\_TYPE\_SCENE\_BDD\_MATERIAUX\_USER* Type d'élément  
*ELEMENT\_TYPE\_SCENE\_BDD\_MATERIAUX\_USER\_GROUP* Type d'élément  
*ELEMENT\_TYPE\_SCENE\_BDD\_MATERIAUX\_USER\_MATERIAU* Type d'élément  
*ELEMENT\_TYPE\_SCENE\_BDD\_MATERIAUX\_PROPMATERIAU* Type d'élément  
*ELEMENT\_TYPE\_SCENE\_BDD\_CATMATERIAL* Type d'élément  
*ELEMENT\_TYPE\_MATERIAU\_APP* Type d'élément  
*ELEMENT\_TYPE\_MATERIAU\_USER* Type d'élément  
*ELEMENT\_TYPE\_GAMMEFREQ\_APP* Type d'élément  
*ELEMENT\_TYPE\_GAMMEFREQ\_USER* Type d'élément  
*ELEMENT\_TYPE\_PROPERTY\_FREQ* Type d'élément  
*ELEMENT\_TYPE\_ROW* Type d'élément  
*ELEMENT\_TYPE\_ROW\_BFREQ* Type d'élément  
*ELEMENT\_TYPE\_ROW\_MATERIAU* Type d'élément  
*ELEMENT\_TYPE\_SCENE\_ENCOMBREMENTS* Type d'élément  
*ELEMENT\_TYPE\_SCENE\_ENCOMBREMENTS\_ENCOMBREMENT* Type d'élément  
*ELEMENT\_TYPE\_SCENE\_ENCOMBREMENTS\_ENCOMBREMENT\_PROPRIETES* Type d'élément  
*ELEMENT\_TYPE\_SCENE\_ENCOMBREMENTS\_ENCOMBREMENT\_CUBOIDE* Type d'élément  
*ELEMENT\_TYPE\_GAMMEABSORPTION* Type d'élément  
*ELEMENT\_TYPE\_CORE\_SPPS* Type d'élément  
*ELEMENT\_TYPE\_CORE\_CORE\_CONFIG* Type d'élément  
*ELEMENT\_TYPE\_CORE\_CORE\_CONFMAILLAGE* Type d'élément  
*ELEMENT\_TYPE\_SCENE\_PROJET* Type d'élément  
*ELEMENT\_TYPE\_SCENE\_PROJET\_USERCONFIGURATION* Type d'élément  
*ELEMENT\_TYPE\_SCENE\_PROJET\_RENDU\_PARTICULES* Type d'élément

***ELEMENT\_TYPE\_SCENE\_DONNEES*** Type d'élément  
***ELEMENT\_TYPE\_SCENE\_ENCOMBREMENTS\_ENCOMBREMENT\_RENDU*** Type d'élément  
***ELEMENT\_TYPE\_SCENE\_PROJET\_ENVIRONNEMENTCONF*** Type d'élément  
***ELEMENT\_TYPE\_DRAWABLE*** **Element** pouvant être représenté dans la vue 3D  
***ELEMENT\_TYPE\_CORE\_CORE\_BFREQSELECTION*** Type d'élément  
***ELEMENT\_TYPE\_BOOL\_BFREQ*** Type d'élément  
***ELEMENT\_TYPE\_REPORT\_FOLDER*** Type d'élément  
***ELEMENT\_TYPE\_REPORT\_PARTVISUALISATION*** Type d'élément  
***ELEMENT\_TYPE\_REPORT\_RECEPTEURSSVISUALISATION*** Type d'élément  
***ELEMENT\_TYPE\_REPORT\_GABE*** Type d'élément  
***ELEMENT\_TYPE\_REPORT\_GABE\_RECP*** Type d'élément  
***ELEMENT\_TYPE\_TREE\_LIST*** Type d'élément  
***ELEMENT\_TYPE\_CORE\_TC*** Type d'élément  
***ELEMENT\_TYPE\_SCENE\_PROJET\_INFORMATION*** Type d'élément  
***ELEMENT\_TYPE\_SCENE\_BDD\_MATERIAUX\_MATERIAU\_RENDER*** Type d'élément  
***ELEMENT\_TYPE\_FONT*** Type d'élément  
***ELEMENT\_TYPE\_CORE\_TLM*** Type d'élément  
***ELEMENT\_TYPE\_REPORT\_GABE\_GAP*** Type d'élément  
***ELEMENT\_TYPE\_REPORT\_UNKNOWN*** Fichier inconnu par PSPS mais connu par le système d'exploitation  
***ELEMENT\_TYPE\_CORE\_SPPS\_OCTREE*** Type d'élément  
***ELEMENT\_TYPE\_REPORT\_RPI*** Type d'élément  
***ELEMENT\_TYPE\_SCENE\_VOLUMES*** Type d'élément  
***ELEMENT\_TYPE\_SCENE\_VOLUMES\_VOLUME*** Type d'élément  
***ELEMENT\_TYPE\_SCENE\_VOLUMES\_VOLUME\_RENDU*** Type d'élément  
***ELEMENT\_TYPE\_SCENE\_VOLUMES\_VOLUME\_PROPRIETES*** Type d'élément  
***ELEMENT\_TYPE\_PYTHON\_EXTENSION*** Type d'élément  
***ELEMENT\_TYPE\_ELEMENT*** Lors de la déclaration d'un élément utilisateur, ce type permet d'exprimer le fait que l'élément hérite directement de l'élément de base  
***ELEMENT\_TYPE\_CORE\_CORE*** Type d'élément  
***ELEMENT\_TYPE\_REPORT\_FILE*** Type d'élément  
***ELEMENT\_TYPE\_REPORT\_RECEPTEURSSVISUALISATION\_STANDART*** Type d'élément  
***ELEMENT\_TYPE\_REPORT\_RECEPTEURSSVISUALISATION\_GAIN*** Type d'élément  
***ELEMENT\_TYPE\_REPORT\_RECEPTEURSSVISUALISATION\_TR*** Type d'élément  
***ELEMENT\_TYPE\_REPORT\_RECEPTEURSSVISUALISATION\_EDT*** Type d'élément  
***ELEMENT\_TYPE\_SCENE\_RECEPTEURSS\_RECEPTEURCOUPE*** Type d'élément  
***ELEMENT\_TYPE\_SCENE\_RECEPTEURSS\_RECEPTEURCOUPE\_PROPRIETES*** Type d'élément  
***ELEMENT\_TYPE\_SCENE\_RECEPTEURSS\_RECEPTEURCOUPE\_RENDU*** Type d'élément  
***ELEMENT\_TYPE\_USER\_PREFERENCE\_NODE*** élément de noeud des préférences utilisateurs  
  
***ELEMENT\_TYPE\_USER\_PREFERENCE\_ITEM*** élément des préférences utilisateurs  
***ELEMENT\_TYPE\_USER\_PREFERENCE\_ITEM\_ISOTEMPLATE*** élément des préférences utilisateurs qui liste les palettes de couleurs iso

### 7.1.2.2 enum uictrl : :graph

Enumeration des icones déjà chargées

Valeurs énumérées :

**GRAPH\_FOLDER** Indice d'une image  
**GRAPH\_ITEM** Indice d'une image  
**GRAPH\_FOLDER\_OPEN** Indice d'une image  
**GRAPH\_FITTINGS\_OPEN** Indice d'une image  
**GRAPH\_FITTINGS\_CLOSE** Indice d'une image  
**GRAPH\_FITTING\_OPEN** Indice d'une image  
**GRAPH\_FITTING\_CLOSE** Indice d'une image  
**GRAPH\_PUNCTUAL\_RECEIVERS\_OPEN** Indice d'une image  
**GRAPH\_PUNCTUAL\_RECEIVERS\_CLOSE** Indice d'une image  
**GRAPH\_SURFACE\_RECEIVERS\_OPEN** Indice d'une image  
**GRAPH\_SURFACE\_RECEIVERS\_CLOSE** Indice d'une image  
**GRAPH\_SOUND\_SOURCES\_OPEN** Indice d'une image  
**GRAPH\_SOUND\_SOURCES\_CLOSE** Indice d'une image  
**GRAPH\_SURFACES\_OPEN** Indice d'une image  
**GRAPH\_SURFACES\_CLOSE** Indice d'une image  
**GRAPH\_LAST\_STATIC\_GRAPH** Dernier icône

### 7.1.2.3 enum uictrl : :idevent

Enumeration des fonctions "built-in" disponibles pour les éléments

Valeurs énumérées :

**IDEVENT\_DELETE\_ELEMENT** Supprime un [element](#)  
**Paramètres:**  
**IDEVENT\_RENAME\_ELEMENT** {"name", : "new label"} Renommer un élément  
**IDEVENT\_COPIER** Evenement  
**IDEVENT\_COLLER** Evenement  
**IDEVENT\_NEW\_SURFACE\_GROUP** Evenement  
**IDEVENT\_GETPROPERTIES** Evenement  
**IDEVENT\_SELECT\_TREE\_ITEM** Evenement  
**IDEVENT\_NEW\_RECEPTEUR\_P** Evenement  
**IDEVENT\_SELECT\_POSITION** Evenement  
**IDEVENT\_NEW\_SOURCE** Evenement  
**IDEVENT\_NEW\_RECEPTEUR\_S** Evenement  
**IDEVENT\_NEW\_USERFREQ** Evenement  
**IDEVENT\_NEW\_ENCOMBREMENT** Evenement  
**IDEVENT\_NEW\_ENCOMBREMENT\_CUBOIDE** Evenement  
**IDEVENT\_NEW\_USERMAT** Evenement  
**IDEVENT\_NEW\_MATERIAL\_GROUP** Evenement  
**IDEVENT\_RUN\_CALCULATION** Evenement

**Paramètres:**

**IDEVENT\_IMPORT\_MATERIAL** {"path", : "material file path"} Importer un fichier de matériaux catt ou odeon

**IDEVENT\_LOAD\_PARTICLE\_SIMULATION** Evenement

**IDEVENT\_LOAD\_PARTICLE\_SIMULATION\_PATH** Evenement

**IDEVENT\_RELOAD\_FOLDER** Evenement

**IDEVENT\_DELETE\_FOLDER** Evenement

**IDEVENT\_LOAD\_RECEPTEURSS\_SIMULATION\_BY\_TIMESTEP** Evenement

**IDEVENT\_LOAD\_RECEPTEURSS\_SIMULATION\_BY\_TIMESTEP\_SUM** Evenement

**IDEVENT\_LOAD\_RECEPTEURSS\_SIMULATION\_SUM** Evenement

**IDEVENT\_INVERT\_FACE\_ORIENTATION** Evenement inverser l'orientation des faces sélectionnées

**IDEVENT\_EMPTY\_POINTER\_VERTEX\_GROUP** Evenement vider un pointeur de groupe de surface

**IDEVENT\_RECPCOMPUTE\_ACOUSTIC\_PARAMETERS** Evenement calculer les paramètres acoustiques d'un récepteur ponctuel

**Paramètres:**

{"C", : "Clarté (ms)", "D" : "Définition (ms)", "TR" : "Décroissance pour le calcul du TR"}

**IDEVENT\_RECEPTEURS\_COMPUTE\_TR** Evenement calcul de la cartographie du tr

**IDEVENT\_RECEPTEURS\_COMPUTE\_EDT** Evenement calcul de la cartographie de l'edt

**IDEVENT\_REPORT\_PARTICULES\_MAKE\_GABE** Evenement création du graph niveau sonore en fonction du temps

**IDEVENT\_BFREQ\_PRESELECTION\_NONE** Sélection automatique des bandes de fréquences

**IDEVENT\_BFREQ\_PRESELECTION\_THIRD\_BAND** Sélection automatique des bandes de fréquences

**IDEVENT\_BFREQ\_PRESELECTION\_BAND** Sélection automatique des bandes de fréquences

**IDEVENT\_BFREQ\_PRESELECTION\_BUILDING\_THIRD\_BAND** Sélection automatique des bandes de fréquences

**IDEVENT\_BFREQ\_PRESELECTION\_BUILDING\_BAND** Sélection automatique des bandes de fréquences

**IDEVENT\_RECPCOMPUTE\_ADVANCED\_ACOUSTIC\_PARAMETERS** Evenement calculer les paramètres acoustiques d'un récepteur ponctuel

**Paramètres:**

{"LF", : "Fraction d'énergie latérale précoce LF (ms)", "LFC" : "Fraction d'énergie latérale précoce LFC (ms)", "LG" : "Tenue acoustique latérale LG(ms)", "G" : "Tenue acoustique G (ms)"}

**IDEVENT\_NEW\_RECEPTEURP\_GROUP** Nouveau groupe de récepteur

**IDEVENT\_NEW\_SOURCE\_GROUP** Nouveau groupe de sources

**IDEVENT\_NEW\_SURFACE\_GROUP\_FROM\_SELECTION** Nouveau groupe de surface à partir de la sélection

**IDEVENT\_LOAD\_RECEPTEURSP\_SIMULATION** Chargement de l'animation des récepteurs ponctuels

**IDEVENT\_BUILD\_VOLUMES\_FROM\_TRIMESH** Déclaration des volumes à partir du maillage

**IDEVENT\_NEW\_VOLUME** Ajout d'un volume

**IDEVENT\_OPEN\_FOLDER** Ouverture d'un dossier

**IDEVENT\_CONVERT\_VOL\_TO\_FITTING** Converti le volume en encombrement scène

**IDEVENT\_NEW\_RECEPTEUR\_S\_COUPE** Ajouter un récepteur en coupe

# Chapitre 8

## Documentation des classes

### 8.1 Référence de la classe uictrl : :application

Classe python de contrôle de I-SIMPA.

#### Fonctions membres publiques statiques

- static void [clearlogdata](#) ()
- static void [clearshellhisto](#) ()
- static std : :string [getcachedir](#) ()
- static boost : :python : :list [getdataarray](#) (const [element](#) &pyel)
- static std : :string [getlastcalculationpath](#) ()
- static std : :string [getlocale](#) ()
- static int [getrootcore](#) ()
- static int [getrootpreference](#) ()
- static int [getrootreport](#) ()
- static int [getrootscene](#) ()
- static boost : :python : :tuple [getuserinput](#) (const std : :string &title, const std : :string &msg, boost : :python : :dict rows)
- static bool [importscene](#) (const std : :string &path, bool keepexistingfacegroup=true, bool docorrection=true, bool domeshsurface=false, const std : :string &paramTetgen="")
- static void [loadproject](#) (const std : :string &path)
- static void [newproject](#) ()
- static int [register\\_event](#) (boost : :python : :object &func)
- static void [register\\_menu\\_manager](#) (const int &element\_typeid, boost : :python : :object &manager)
- static void [reloadgroupsfrommodel](#) ()
- static void [savelog](#) (const std : :string &path)
- static void [saveproject](#) (const std : :string &path="")
- static void [saveshell](#) (const std : :string &path)
- static void [sendevent](#) (const [element](#) &pyel, const int &idevent, boost : :python : :dict parameters=boost : :python : :dict())

#### 8.1.1 Description détaillée

Classe python de contrôle de I-SIMPA.

## 8.1.2 Documentation des fonctions membres

### 8.1.2.1 `static void uictrl : :application : :clearlogdata () [static]`

Efface l'historique de la fenêtre de messages

### 8.1.2.2 `static void uictrl : :application : :clearshellhisto () [static]`

Efface l'historique de la fenêtre de console python

### 8.1.2.3 `static std : :string uictrl : :application : :getcachedir () [static]`

Retourne le chemin d'accès au dossier temporaire du projet

### 8.1.2.4 `static boost : :python : :list uictrl : :application : :getdataarray (const element & pyel) [static]`

Retourne les données sous forme de tableau associé à un élément Pour les éléments ayant des propriétés il est rendu un tableau de propriété Pour les éléments [e\\_file](#) portant des données binaires il est rendu le tableau avec le post-traitement de I-SIMPA.

#### Paramètres:

*pyel* Les données du tableau seront extraites à partir de cet élément.

### 8.1.2.5 `static std : :string uictrl : :application : :getlastcalculationpath () [static]`

Retourne le chemin du dernier calcul effectué.

### 8.1.2.6 `static std : :string uictrl : :application : :getlocale () [static]`

Retourne la langue sélectionné par l'utilisateur dans la forme standart. La forme standart correspond à celle utilisée dans les système UNIX : Il s'agit d'une chaîne de caractères sous forme xx ou xx\_YY, où xx correspond à la norme ISO 639 et YY à la norme ISO 3166 relatif au pays. Il y a par exemple "en", "en\_GB", "en\_US" ou "fr\_FR".

### 8.1.2.7 `static int uictrl : :application : :getrootcore () [static]`

Accéder à l'élément racine d'un arbre

#### Renvoie:

L'indice de l'élément racine de l'arbre des codes de calcul

### 8.1.2.8 `static int uictrl : :application : :getrootpreference () [static]`

Accéder à l'élément racine des préférences utilisateur

#### Renvoie:

L'indice de l'élément racine de l'arbre propre à l'utilisateur (et non au projet)



**8.1.2.9 static int uictrl : :application : :getrootreport () [static]**

Accéder à l'élément racine d'un arbre

**Renvoie:**

L'indice de l'élément racine de l'arbre des résultats de calculs

**8.1.2.10 static int uictrl : :application : :getrootscene () [static]**

Accéder à l'élément racine d'un arbre

**Renvoie:**

L'indice de l'élément racine de l'arbre de la scène

**8.1.2.11 static boost : :python : :tuple uictrl : :application : :getuserinput (const std : :string & title, const std : :string & msg, boost : :python : :dict rows) [static]**

Demande à l'utilisateur d'entrer des valeurs dans des champs via l'interface

**Paramètres:**

*title* Titre de la fenêtre

*msg* Message affiché à l'utilisateur

*rows* Champs en entrée {"champ1" : "valeur par défaut","champ2" : "valeur par défaut","liste champ3" : ["choix 1","choix 2","choix 3"],..}

**Renvoie:**

Tuple (Booléen, Liste) Le booléen indique le choix de l'utilisateur (ok, annuler) et la liste contient les valeurs des champs dans le même ordre que le paramètre rows.

**8.1.2.12 static bool uictrl : :application : :importscene (const std : :string & path, bool keepexistingfacegroup = true, bool docorrection = true, bool domeshsurface = false, const std : :string & paramTetgen = "") [static]**

Importe un modèle 3D qui remplacera le modèle du projet courant. Les formats de modèle supportés sont \*.3ds;\*.ply;\*.bin;\*.poly

**Paramètres:**

*path* Chemin du fichier de modèle

*keepexistingfacegroup* A vrai, Les groupes seront conservés grâce aux coordonnées similaires entre les anciennes et nouvelles faces.

*docorrection* Exécute preprocess.exe pour éclater ou supprimer les faces afin de rendre possible le maillage.

*domeshsurface* Tente de mailler la surface du modèle. Cette opération peut augmenter le nombre de faces et elle détruira les informations de couleurs et de textures.

*paramTetgen* Paramètres de maillage. Inutile si domeshsurface à faux.

**Renvoie:**

Vrai en cas de succès. Faux si l'importation a échoué, se référer à la fenêtre de message pour plus d'information.

**Paramètres:**

*path* Chemin du fichier à créer

**8.1.2.13 static void uictrl : :application : :loadproject (const std : :string & path) [static]**

Charge un fichier de projet

**Paramètres:**

*path* Chemin du fichier de projet à charger

**8.1.2.14 static void uictrl : :application : :newproject () [static]**

Ferme le projet courant et en crée un nouveau

**8.1.2.15 static int uictrl : :application : :register\_event (boost : :python : :object & func) [static]**

Voir [Ajouter des fonctionnalités](#). Permet de lier un indice de menu à une fonction python.

**Paramètres:**

*func* Méthode python à appeler en cas d'exécution d'un événement.

**Renvoie:**

Indice du nouveau type d'événement de I-SIMPA.

**8.1.2.16 static void uictrl : :application : :register\_menu\_manager (const int & element\_typeid, boost : :python : :object & manager) [static]**

Voir [Ajouter des fonctionnalités](#). Enregistre un nouveau gestionnaire de menu pour ce type d'élément.

**Paramètres:**

*element\_typeid* `uictrl : :element_type` Type d'élément lié à ce gestionnaire de menu.

*manager* Instance de l'objet du manager.

**8.1.2.17 static void uictrl : :application : :reloadgroupsfrommodel () [static]**

Recharge les faces à partir du modèle

**8.1.2.18 static void uictrl : :application : :savelog (const std : :string & path) [static]**

Sauvegarde le contenu de la fenêtre de messages.

**Paramètres:**

*path* Chemin du fichier à créer

**8.1.2.19 static void uictrl : :application : :saveproject (const std : :string & path = "") [static]**

Sauvegarde le projet ouvert

**Paramètres:**

*path* Dossier de sauvegarde du projet

**8.1.2.20** `static void uictrl : :application : :saveshell (const std : :string & path) [static]`

Sauvegarde le contenu de la fenêtre de console python.

**Paramètres:**

*path* Chemin du fichier à créer

**8.1.2.21** `static void uictrl : :application : :sendevent (const element & pyel, const int & idevent, boost : :python : :dict parameters = boost : :python : :dict ()) [static]`

Execution d'un événement.

**Paramètres:**

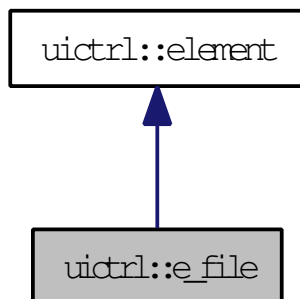
*pyel* [Element](#) de l'arbre concerné

*idevent* Événement interne [uictrl : :idevent](#) ou événement d'un module python déclaré via [application : :register\\_event](#)

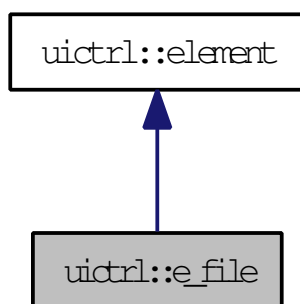
*parameters* Voir les membres de [uictrl : :idevent](#) pour plus de détails.

## 8.2 Référence de la classe uictrl : :e\_file

Graphe d'héritage de uictrl : :e\_file :



Graphe de collaboration de uictrl : :e\_file :



### Fonctions membres publiques

- `std::string buildfullpath ()`
- `e_file (const element &cpyFrom)`
- `e_file (const wxInt32 &_xmlId)`

#### 8.2.1 Description détaillée

Dérive d'un élément, il représente un fichier ou un dossier

#### 8.2.2 Documentation des fonctions membres

##### 8.2.2.1 `std::string uictrl : :e_file : :buildfullpath ()`

Retourne le chemin relatif de cet élément de fichier.

## 8.3 Référence de la structure uictrl : :Element

Alias.

### Types publics

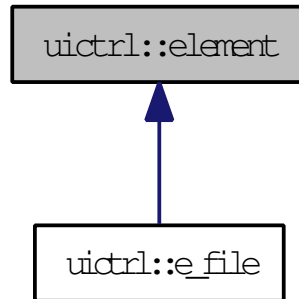
- typedef [element\\_type](#) ELEMENT\_TYPE
- typedef [graph](#) GRAPH
- typedef [idevent](#) IDEVENT

### 8.3.1 Description détaillée

Alias.

## 8.4 Référence de la classe uictrl : :element

Contrôle d'un élément implémenté en python ou c++. Graphe d'héritage de uictrl : :element :



### Fonctions membres publiques

- int [appendfilsbytype](#) (Element : :ELEMENT\_TYPE etypefils, const std : :string &libelle="")
- int [appendpropertybool](#) (const std : :string &propertyName, const std : :string &propertyLabel, bool propertyDefaultValue, bool exportToCore=false)
- int [appendpropertycolor](#) (const std : :string &propertyName, const std : :string &propertyLabel, long defaultRed=0, long defaultGreen=0, long defaultBlue=0)
- int [appendpropertydecimal](#) (const std : :string &propertyName, const std : :string &propertyLabel, float propertyDefaultValue=0.f, bool readOnly=false, int precision=4, bool isMaxValue=false, bool isMinValue=false, float maxValue=0, float minValue=0, bool exportToCore=false)
- int [appendpropertyentier](#) (const std : :string &propertyName, const std : :string &propertyLabel, int propertyDefaultValue=0, bool exportToCore=false, bool isMaxValue=false, bool isMinValue=false, int maxValue=0, int minValue=0)
- int [appendpropertyfont](#) (const std : :string &propertyName, const std : :string &propertyLabel, const std : :string &propertyDefaultValue="")
- int [appendpropertylist](#) (const std : :string &propertyName, const std : :string &propertyLabel, const boost : :python : :list &values, long defaultValue, bool asTitle=false, int hSize=1, bool exportToCore=false)
- int [appendpropertyposition](#) (const std : :string &propertyName, const std : :string &propertyLabel, const boost : :python : :list &propertyDefaultValue, bool exportToCore=false)
- int [appendpropertytext](#) (const std : :string &propertyName, const std : :string &propertyLabel, const std : :string &propertyDefaultValue, bool readOnly=false, bool exportToCore=false)
- boost : :python : :object [appenduserelement](#) (const Element : :ELEMENT\_TYPE &baseType, const std : :string &moduleName, const std : :string &className)
- boost : :python : :list [childs](#) ()
- void [deleteallelementbytype](#) (Element : :ELEMENT\_TYPE typeElement)
- void [deleteallelementbytype](#) (Element : :ELEMENT\_TYPE typeElementToDelete)
- bool [deleteelementbyxmlid](#) (int xmlIdElement, bool setModification=true)
- **element** (const [element](#) &cpyFrom)
- **element** (const wxInt32 &\_xmlId)
- boost : :python : :list [getallelementbytype](#) (Element : :ELEMENT\_TYPE typeElement)
- bool [getboolconfig](#) (const std : :string &name)
- boost : :python : :list [getcolorconfig](#) (const std : :string &name)
- float [getdecimalconfig](#) (const std : :string &name)
- int [getelementbylibelle](#) (std : :string libelle)
- int [getelementbytype](#) (Element : :ELEMENT\_TYPE typeElement)
- int [getentierconfig](#) (const std : :string &name)
- wxInt32 [getid](#) ()
- wxInt32 [getindice](#) () const
- boost : :python : :dict [getinfos](#) ()
- int [getlistconfig](#) (const std : :string &name)
- boost : :python : :list [getmenu](#) ()
- boost : :python : :list [getpositionconfig](#) (const std : :string &name)
- std : :string [getstringconfig](#) (const std : :string &name)

- bool `hasproperty` (const std : :string &name)
- void `hide` (bool visible=false)
- void `modified` (int elementUpdated)
- void `register_update_manager` (boost : :python : :object &pymethod)
- void `setreadonlyallconfig` (bool readOnly=true, int col=0)
- void `setreadonlyconfig` (const std : :string &name, bool readOnly=true, int col=0)
- bool `updateboolconfig` (const std : :string &name, bool newValue)
- bool `updatedecimalconfig` (const std : :string &name, float newValue)
- bool `updateentierconfig` (const std : :string &name, int newValue)
- bool `updatelistconfig` (const std : :string &name, int newIndex)
- bool `updatepositionconfig` (const std : :string &name, const boost : :python : :list &newValue)
- bool `updatestringconfig` (const std : :string &name, const std : :string &newValue)

### Attributs protégés

- wxInt32 `xmlId`

### 8.4.1 Description détaillée

Contrôle d'un élément implémenté en python ou c++. Toute les données du projet dérivent de cette classe. De la simple propriété chaîne de caractère, booléen et nombre à décimal jusqu'à un noeud de l'arbre du projet.

### 8.4.2 Documentation des fonctions membres

#### 8.4.2.1 int uictrl : :element : :appendfilsbytype (Element : :ELEMENT\_TYPE *etypefils*, const std : :string & *libelle* = "")

Ajoute un nouvel enfant de type *etypefils* et retourne son indice xml. Retourne -1 en cas d'échec.

##### Paramètres:

*etypefils* Type de l'élément à ajouter

*libelle* Libellé de l'élément (n'est pas disponible pour tout les types d'éléments)

#### 8.4.2.2 int uictrl : :element : :appendpropertybool (const std : :string & *propertyName*, const std : :string & *propertyLabel*, bool *propertyDefaultValue*, bool *exportToCore* = false)

Ajoute une nouvelle propriété à un élément.

##### Paramètres:

*propertyName* Nom de la propriété. Ce nom doit être unique pour chaque propriété.

*propertyLabel* Libellé non traduite de la propriété.

*propertyDefaultValue* Valeur par défaut de la propriété

*exportToCore* Spécifie l'exportation ou non de cette propriété vers les code de calcul.

##### Renvoie:

L'indice xml de l'élément ajouté, ou l'indice de l'élément portant déjà ce nom. -1 si la procédure a échoué.

**8.4.2.3** `int uictrl : :element : :appendpropertycolor (const std : :string & propertyName, const std : :string & propertyLabel, long defaultRed = 0, long defaultGreen = 0, long defaultBlue = 0)`

Ajoute une nouvelle propriété à un élément.

**Paramètres:**

*propertyName* Nom de la propriété. Ce nom doit être unique pour chaque propriété.

*propertyLabel* Libellé non traduite de la propriété.

*defaultRed* Niveau de rouge par défaut. [0-255]

*defaultGreen* Niveau de vert par défaut. [0-255]

*defaultBlue* Niveau de bleu par défaut. [0-255]

**Renvoie:**

L'indice xml de l'élément ajouté, ou l'indice de l'élément portant déjà ce nom. -1 si la procédure a échoué.

**8.4.2.4** `int uictrl : :element : :appendpropertydecimal (const std : :string & propertyName, const std : :string & propertyLabel, float propertyDefaultValue = 0.f, bool readOnly = false, int precision = 4, bool isMaxValue = false, bool isMinValue = false, float maxValue = 0, float minValue = 0, bool exportToCore = false)`

Ajoute une nouvelle propriété à un élément.

**Paramètres:**

*propertyName* Nom de la propriété. Ce nom doit être unique pour chaque propriété.

*propertyLabel* Libellé non traduite de la propriété.

*propertyDefaultValue* Valeur par défaut de la propriété

*readOnly* A vrai l'utilisateur ne pourra pas modifier la valeur du champ. Sauf via la fenêtre python.

*precision* Nombre de chiffre affiché après la virgule.

*isMaxValue* Vrai pour activer les contraintes de limite maximale.

*isMinValue* Vrai pour activer les contraintes de limite minimale.

*maxValue* Valeur maximum autorisée.

*minValue* Valeur minimale autorisée.

*exportToCore* Spécifie l'exportation ou non de cette propriété vers les code de calcul.

**Renvoie:**

L'indice xml de l'élément ajouté, ou l'indice de l'élément portant déjà ce nom. -1 si la procédure a échoué.

**8.4.2.5** `int uictrl : :element : :appendpropertyentier (const std : :string & propertyName, const std : :string & propertyLabel, int propertyDefaultValue = 0, bool exportToCore = false, bool isMaxValue = false, bool isMinValue = false, int maxValue = 0, int minValue = 0)`

Ajoute une nouvelle propriété à un élément.

**Paramètres:**

*propertyName* Nom de la propriété. Ce nom doit être unique pour chaque propriété.



*propertyLabel* Libellé non traduite de la propriété.

*propertyDefaultValue* Valeur par défaut de la propriété

*exportToCore* Spécifie l'exportation ou non de cette propriété vers les code de calcul.

*isMaxValue* Vrai pour activer les contraintes de limite maximale.

*isMinValue* Vrai pour activer les contraintes de limite minimale.

*maxValue* Valeur maximum autorisée.

*minValue* Valeur minimale autorisée.

#### Renvoi:

L'indice xml de l'élément ajouté, ou l'indice de l'élément portant déjà ce nom. -1 si la procédure a échoué.

#### 8.4.2.6 `int uictrl :element :appendpropertyfont (const std :string &propertyName, const std :string &propertyLabel, const std :string &propertyDefaultValue = "")`

Non implémenté.

#### 8.4.2.7 `int uictrl :element :appendpropertylist (const std :string &propertyName, const std :string &propertyLabel, const boost :python :list &values, long defaultValue, bool asTitle = false, int hSize = 1, bool exportToCore = false)`

Ajoute une nouvelle propriété à un élément.

#### Paramètres:

*propertyName* Nom de la propriété. Ce nom doit être unique pour chaque propriété.

*propertyLabel* Libellé non traduite de la propriété.

*values* Elements de la liste (non traduit) et leurs indices. [["première option","deuxième option"],[0,1]]

*defaultValue* Valeur par défaut de la propriété

*asTitle* Vrai si cette propriété doit être la première affichée

*hSize* Nombre de colonne occupée par cette propriété.

*exportToCore* Spécifie l'exportation ou non de cette propriété vers les code de calcul.

#### Renvoi:

L'indice xml de l'élément ajouté, ou l'indice de l'élément portant déjà ce nom. -1 si la procédure a échoué.

#### 8.4.2.8 `int uictrl :element :appendpropertyposition (const std :string &propertyName, const std :string &propertyLabel, const boost :python :list &propertyDefaultValue, bool exportToCore = false)`

Ajoute une nouvelle propriété à un élément.

#### Paramètres:

*propertyName* Nom de la propriété. Ce nom doit être unique pour chaque propriété.

*propertyLabel* Libellé non traduite de la propriété.

*propertyDefaultValue* Valeur par défaut de la propriété [x,y,z]

*exportToCore* Spécifie l'exportation ou non de cette propriété vers les code de calcul.

**Renvoie:**

L'indice xml de l'élément ajouté, ou l'indice de l'élément portant déjà ce nom. -1 si la procédure a échoué.

**8.4.2.9** `int uictrl : :element : :appendpropertytext (const std : :string & propertyName, const std : :string & propertyLabel, const std : :string & propertyDefaultValue, bool readOnly = false, bool exportToCore = false)`

Ajoute une nouvelle propriété à un élément.

**Paramètres:**

*propertyName* Nom de la propriété. Ce nom doit être unique pour chaque propriété.

*propertyLabel* Libellé non traduite de la propriété.

*propertyDefaultValue* Valeur par défaut de la propriété

*readOnly* A vrai l'utilisateur ne pourra pas modifier la valeur du champ. Sauf via la fenêtre python.

*exportToCore* Spécifie l'exportation ou non de cette propriété vers les code de calcul.

**Renvoie:**

L'indice xml de l'élément ajouté, ou l'indice de l'élément portant déjà ce nom. -1 si la procédure a échoué.

**8.4.2.10** `boost : :python : :object uictrl : :element : :appenduserelement (const Element : :ELEMENT_TYPE & baseType, const std : :string & moduleName, const std : :string & className)`

Affecte un module python en tant qu'enfant de se module.

**Paramètres:**

*baseType* Type de base de l'élément

*moduleName* Nom du module python (nom du dossier dans UserScript/)

*className* Nom de la classe du nouvel élément.

**8.4.2.11** `boost : :python : :list uictrl : :element : :childs ()`

Retourne l'indice, le type et le nom de chaque enfants

**8.4.2.12** `void uictrl : :element : :deleteallelementbytype (Element : :ELEMENT_TYPE typeElement)`

Supprime les enfants immédiat ayant ce type d'élément.

**Paramètres:**

*typeElement* Type d'élément

**8.4.2.13 void uictrl : :element : :deleteallelementbytype (Element : :ELEMENT\_TYPE typeElementToDelete)**

Supprime les enfants ayant ce type d'élément de façon récursive.

**Paramètres:**

*typeElement* Type d'élément

**8.4.2.14 bool uictrl : :element : :deleteelementbyxmlid (int xmlIdElement, bool setModification = true)**

Supprime l'enfant immédiat ayant cet indice.

**Paramètres:**

*xmlIdElement* Indice de l'élément fils

*setModification* Cette méthode appellera automatiquement [element : :modified](#).

**8.4.2.15 boost : :python : :list uictrl : :element : :getallelementbytype (Element : :ELEMENT\_TYPE typeElement)**

Parcourt récursivement tout les enfants de l'élément et retourne la liste de tout les éléments ayant ce type.

**8.4.2.16 bool uictrl : :element : :getboolconfig (const std : :string & name)**

Retourne la valeur d'une propriété

**Paramètres:**

*name* Nom de la propriété. En cas de doute de l'existence de cette propriété merci de vérifier celle-ci via [element : :hasproperty](#)

**Renvoie:**

Valeur de la propriété

**8.4.2.17 boost : :python : :list uictrl : :element : :getcolorconfig (const std : :string & name)**

Retourne la valeur d'une propriété

**Paramètres:**

*name* Nom de la propriété. En cas de doute de l'existence de cette propriété merci de vérifier celle-ci via [element : :hasproperty](#)

**Renvoie:**

Valeur de la propriété [rouge,vert,bleu] [0-255]

**8.4.2.18 float uictrl : :element : :getdecimalconfig (const std : :string & name)**

Retourne la valeur d'une propriété

**Paramètres:**

*name* Nom de la propriété. En cas de doute de l'existence de cette propriété merci de vérifier celle-ci via [element : :hasproperty](#)

**Renvoie:**

Valeur de la propriété

**8.4.2.19 int uictrl : :element : :getelementbylibelle (std : :string libelle)**

Retourne le premier enfant ayant le même libellé

**8.4.2.20 int uictrl : :element : :getelementbytype (Element : :ELEMENT\_TYPE typeElement)**

Retourne le premier enfant ayant le même type que le paramètre de la méthode.

**8.4.2.21 int uictrl : :element : :getentierconfig (const std : :string & name)**

Retourne la valeur d'une propriété

**Paramètres:**

*name* Nom de la propriété. En cas de doute de l'existence de cette propriété merci de vérifier celle-ci via [element : :hasproperty](#)

**Renvoie:**

Valeur de la propriété

**8.4.2.22 wxInt32 uictrl : :element : :getid () [inline]****Renvoie:**

[Element](#) index

**8.4.2.23 boost : :python : :dict uictrl : :element : :getinfos ()**

Retourne un dictionnaire comportant les informations suivantes :

- typeElement : Type de l'élément ([uictrl.element\\_type](#))
- xmlIdElement : Indice de l'élément (

**Voir également:**

GetIndice )

- expanded : Indique si l'élément est déplié ou non
- userDestroyable : Indique si l'élément peut être supprimé par l'utilisateur
- label : Libellé d'origine de l'élément
- label\_located : Libellé traduit dans la langue de l'utilisateur
- name : Nom de l'élément
- parentid : Indice de l'élément parent. -1 si aucun

**8.4.2.24 int uictrl : :element : :getlistconfig (const std : :string & name)**

Retourne la valeur d'une propriété

**Paramètres:**

*name* Nom de la propriété. En cas de doute de l'existence de cette propriété merci de vérifier celle-ci via [element : :hasproperty](#)

**Renvoie:**

Valeur de la propriété

**8.4.2.25 boost : :python : :list uictrl : :element : :getmenu ()**

Retourne le menu final (logiciel+python menu manager) sous forme de liste en arbre contenant pour chaque élément de menu le tuple ( libellé traduit, indice event\_id )

**8.4.2.26 boost : :python : :list uictrl : :element : :getpositionconfig (const std : :string & name)**

Retourne la valeur d'une propriété

**Paramètres:**

*name* Nom de la propriété. En cas de doute de l'existence de cette propriété merci de vérifier celle-ci via `element : :hasproperty`

**Renvoie:**

Valeur de la propriété [x,y,z] (m)

**8.4.2.27 std : :string uictrl : :element : :getstringconfig (const std : :string & name)**

Retourne la valeur d'une propriété

**Paramètres:**

*name* Nom de la propriété. En cas de doute de l'existence de cette propriété merci de vérifier celle-ci via `element : :hasproperty`

**Renvoie:**

Valeur de la propriété

**8.4.2.28 bool uictrl : :element : :hasproperty (const std : :string & name)**

Retourne Vrai si le nom correspond à une des propriétés de l'élément.

**Paramètres:**

*hasproperty* Nom de la propriété

**8.4.2.29 void uictrl : :element : :hide (bool visible = false)**

Cache cet élément de sorte qu'il ne puisse être visible pour l'utilisateur.

**Paramètres:**

*visible* Nouvel état de l'élément

**8.4.2.30 void uictrl : :element : :modified (int elementUpdated)**

Affecte lui et ses parent comme modifié, et il sera sauvegardé à la prochaine sauvegarde.

**Paramètres:**

*elementUpdated* Indice xml de l'élément modifié

#### 8.4.2.31 void uictrl : :element : :register\_update\_manager (boost : :python : :object & pymethod)

Lie une fonction python à l'événement de mise à jour d'un élément. Vous pouvez ajouter plusieurs liens, ces liens restent en place jusqu'à la fermeture du projet.

**Paramètres:**

*func* Classe python avec une fonction nommé OnUpdate avec un paramètre correspondant à l'indice de l'élément mis à jour (l'élément courant ou un de ses fils)

#### 8.4.2.32 void uictrl : :element : :setreadonlyallconfig (bool readOnly = true, int col = 0)

Active/Désactive la possibilité de modifier tout les champs pour l'utilisateur pour une colonne donnée

**Paramètres:**

*readOnly* Nouvel état des champs

*col* Pour les champ ayant plusieurs colonnes tel que E\_Data\_Row indique quel colonne sera affectée par la lecture seule.

#### 8.4.2.33 void uictrl : :element : :setreadonlyconfig (const std : :string & name, bool readOnly = true, int col = 0)

Active/Désactive la possibilité de modifier le champ pour l'utilisateur pour un champ donné

**Paramètres:**

*name* Nom du champ

*readOnly* Nouvel état du champ

*col* Pour les champs ayant plusieurs colonnes tel que E\_Data\_Row indique quel colonne sera affecté par la lecture seule.

#### 8.4.2.34 bool uictrl : :element : :updateboolconfig (const std : :string & name, bool newValue)

Met à jour la valeur d'une propriété

**Paramètres:**

*name* Nom de la propriété

*newValue* Nouvelle valeur de la propriété

**Renvoie:**

Vrai si la propriété a été trouvé et mis à jour

#### 8.4.2.35 bool uictrl : :element : :updatedecimalconfig (const std : :string & name, float newValue)

Met à jour la valeur d'une propriété

**Paramètres:**

*name* Nom de la propriété

*newValue* Nouvelle valeur de la propriété

**Renvoie:**

Vrai si la propriété a été trouvé et mis à jour

**8.4.2.36 bool uictrl : :element : :updateentierconfig (const std : :string & name, int newValue)**

Met à jour la valeur d'une propriété

**Paramètres:**

*name* Nom de la propriété

*newValue* Nouvelle valeur de la propriété

**Renvoie:**

Vrai si la propriété a été trouvé et mis à jour

**8.4.2.37 bool uictrl : :element : :updatelistconfig (const std : :string & name, int newIndex)**

Met à jour la valeur d'une propriété

**Paramètres:**

*name* Nom de la propriété

*newIndex* Nouvelle valeur de la propriété

**Renvoie:**

Vrai si la propriété a été trouvé et mis à jour

**8.4.2.38 bool uictrl : :element : :updatepositionconfig (const std : :string & name, const boost : :python : :list & newValue)**

Met à jour la valeur d'une propriété

**Paramètres:**

*name* Nom de la propriété

*newValue* Nouvelle valeur de la propriété [x,y,z]

**Renvoie:**

Vrai si la propriété a été trouvé et mis à jour

**8.4.2.39 bool uictrl : :element : :updatestringconfig (const std : :string & name, const std : :string & newValue)**

Met à jour la valeur d'une propriété

**Paramètres:**

*name* Nom de la propriété

*newValue* Nouvelle valeur de la propriété

**Renvoie:**

Vrai si la propriété a été trouvé et mis à jour

# Index

- appendfilsbytype
  - uictrl : :element, 33
- appendpropertybool
  - uictrl : :element, 33
- appendpropertycolor
  - uictrl : :element, 33
- appendpropertydecimal
  - uictrl : :element, 34
- appendpropertyentier
  - uictrl : :element, 34
- appendpropertyfont
  - uictrl : :element, 35
- appendpropertylist
  - uictrl : :element, 35
- appendpropertyposition
  - uictrl : :element, 35
- appendpropertytext
  - uictrl : :element, 36
- appenduserelement
  - uictrl : :element, 36
  
- buildfullpath
  - uictrl : :e\_file, 30
  
- childs
  - uictrl : :element, 36
- clearlogdata
  - uictrl : :application, 26
- clearshellhisto
  - uictrl : :application, 26
  
- deleteallelementbytype
  - uictrl : :element, 36
- deleteallelementbytyper
  - uictrl : :element, 36
- deleteelementbyxmlid
  - uictrl : :element, 37
  
- ELEMENT\_TYPE\_BOOL
  - uictrl, 21
- ELEMENT\_TYPE\_BOOL\_BFREQ
  - uictrl, 22
- ELEMENT\_TYPE\_COLOR
  - uictrl, 20
- ELEMENT\_TYPE\_CORE\_CORE\_
  - uictrl, 22
- ELEMENT\_TYPE\_CORE\_CORE\_
  - BFREQSELECTION
  - uictrl, 22
- ELEMENT\_TYPE\_CORE\_CORE\_CONFIG
  - uictrl, 21
- ELEMENT\_TYPE\_CORE\_CORE\_
  - CONFMAILLAGE
  - uictrl, 21
- ELEMENT\_TYPE\_CORE\_ROOT
  - uictrl, 20
- ELEMENT\_TYPE\_CORE\_SPPS
  - uictrl, 21
- ELEMENT\_TYPE\_CORE\_SPPS\_OCTREE
  - uictrl, 22
- ELEMENT\_TYPE\_CORE\_TC
  - uictrl, 22
- ELEMENT\_TYPE\_CORE\_TLM
  - uictrl, 22
- ELEMENT\_TYPE\_DRAWABLE
  - uictrl, 22
- ELEMENT\_TYPE\_ELEMENT
  - uictrl, 22
- ELEMENT\_TYPE\_FLOAT
  - uictrl, 21
- ELEMENT\_TYPE\_FONT
  - uictrl, 22
- ELEMENT\_TYPE\_GAMMEABSORPTION
  - uictrl, 21
- ELEMENT\_TYPE\_GAMMEFREQ\_APP
  - uictrl, 21
- ELEMENT\_TYPE\_GAMMEFREQ\_USER
  - uictrl, 21
- ELEMENT\_TYPE\_INTEGER
  - uictrl, 21
- ELEMENT\_TYPE\_LIST
  - uictrl, 21
- ELEMENT\_TYPE\_MATERIAU\_APP
  - uictrl, 21
- ELEMENT\_TYPE\_MATERIAU\_USER
  - uictrl, 21
- ELEMENT\_TYPE\_POSITION
  - uictrl, 21
- ELEMENT\_TYPE\_PROPERTY\_FREQ
  - uictrl, 21
- ELEMENT\_TYPE\_PYTHON\_EXTENSION



- uictrl, 22
- ELEMENT\_TYPE\_REPORT\_FILE
  - uictrl, 22
- ELEMENT\_TYPE\_REPORT\_FOLDER
  - uictrl, 22
- ELEMENT\_TYPE\_REPORT\_GABE
  - uictrl, 22
- ELEMENT\_TYPE\_REPORT\_GABE\_GAP
  - uictrl, 22
- ELEMENT\_TYPE\_REPORT\_GABE\_RECIP
  - uictrl, 22
- ELEMENT\_TYPE\_REPORT\_-
  - PARTVISUALISATION
    - uictrl, 22
- ELEMENT\_TYPE\_REPORT\_-
  - RECEPTEURSSVISUALISATION
    - uictrl, 22
- ELEMENT\_TYPE\_REPORT\_-
  - RECEPTEURSSVISUALISATION\_-
    - EDT
      - uictrl, 22
- ELEMENT\_TYPE\_REPORT\_-
  - RECEPTEURSSVISUALISATION\_-
    - GAIN
      - uictrl, 22
- ELEMENT\_TYPE\_REPORT\_-
  - RECEPTEURSSVISUALISATION\_-
    - STANDART
      - uictrl, 22
- ELEMENT\_TYPE\_REPORT\_-
  - RECEPTEURSSVISUALISATION\_TR
    - uictrl, 22
- ELEMENT\_TYPE\_REPORT\_RPI
  - uictrl, 22
- ELEMENT\_TYPE\_REPORT\_UNKNOWN
  - uictrl, 22
- ELEMENT\_TYPE\_RESULT\_ROOT
  - uictrl, 20
- ELEMENT\_TYPE\_ROW
  - uictrl, 21
- ELEMENT\_TYPE\_ROW\_BFREQ
  - uictrl, 21
- ELEMENT\_TYPE\_ROW\_MATERIAU
  - uictrl, 21
- ELEMENT\_TYPE\_SCENE\_BDD
  - uictrl, 21
- ELEMENT\_TYPE\_SCENE\_BDD\_-
  - CATMATERIAL
    - uictrl, 21
- ELEMENT\_TYPE\_SCENE\_BDD\_MATERIAUX
  - uictrl, 21
- ELEMENT\_TYPE\_SCENE\_BDD\_-
  - MATERIAUX\_APP
    - uictrl, 21
- ELEMENT\_TYPE\_SCENE\_BDD\_-
  - MATERIAUX\_APP\_GROUP
    - uictrl, 21
- ELEMENT\_TYPE\_SCENE\_BDD\_-
  - MATERIAUX\_APP\_MATERIAU
    - uictrl, 21
- ELEMENT\_TYPE\_SCENE\_BDD\_-
  - MATERIAUX\_MATERIAU\_RENDER
    - uictrl, 22
- ELEMENT\_TYPE\_SCENE\_BDD\_-
  - MATERIAUX\_PROPMATERIAU
    - uictrl, 21
- ELEMENT\_TYPE\_SCENE\_BDD\_-
  - MATERIAUX\_USER
    - uictrl, 21
- ELEMENT\_TYPE\_SCENE\_BDD\_-
  - MATERIAUX\_USER\_GROUP
    - uictrl, 21
- ELEMENT\_TYPE\_SCENE\_BDD\_-
  - MATERIAUX\_USER\_MATERIAU
    - uictrl, 21
- ELEMENT\_TYPE\_SCENE\_BDD\_SPECTRUMS
  - uictrl, 21
- ELEMENT\_TYPE\_SCENE\_BDD\_-
  - SPECTRUMS\_APP
    - uictrl, 21
- ELEMENT\_TYPE\_SCENE\_BDD\_-
  - SPECTRUMS\_USER
    - uictrl, 21
- ELEMENT\_TYPE\_SCENE\_DONNEES
  - uictrl, 21
- ELEMENT\_TYPE\_SCENE\_-
  - ENCOMBREMENTS
    - uictrl, 21
- ELEMENT\_TYPE\_SCENE\_-
  - ENCOMBREMENTS\_-
    - ENCOMBREMENT
      - uictrl, 21
- ELEMENT\_TYPE\_SCENE\_-
  - ENCOMBREMENTS\_-
    - ENCOMBREMENT\_CUBOIDE
      - uictrl, 21
- ELEMENT\_TYPE\_SCENE\_-
  - ENCOMBREMENTS\_-
    - ENCOMBREMENT\_PROPRIETES
      - uictrl, 21
- ELEMENT\_TYPE\_SCENE\_-
  - ENCOMBREMENTS\_-
    - ENCOMBREMENT\_RENDU
      - uictrl, 22
- ELEMENT\_TYPE\_SCENE\_-
  - GROUPESURFACES
    - uictrl, 20

- ELEMENT\_TYPE\_SCENE\_-  
  GROUPESURFACES\_GROUPE  
  uictrl, 20
- ELEMENT\_TYPE\_SCENE\_-  
  GROUPESURFACES\_GROUPE\_-  
  VERTEX  
  uictrl, 20
- ELEMENT\_TYPE\_SCENE\_PROJET  
  uictrl, 21
- ELEMENT\_TYPE\_SCENE\_PROJET\_-  
  CONFIGURATION  
  uictrl, 20
- ELEMENT\_TYPE\_SCENE\_PROJET\_-  
  ENVIRONNEMENTCONF  
  uictrl, 22
- ELEMENT\_TYPE\_SCENE\_PROJET\_-  
  INFORMATION  
  uictrl, 22
- ELEMENT\_TYPE\_SCENE\_PROJET\_RENDU  
  uictrl, 21
- ELEMENT\_TYPE\_SCENE\_PROJET\_RENDU\_-  
  MODEL  
  uictrl, 21
- ELEMENT\_TYPE\_SCENE\_PROJET\_RENDU\_-  
  ORIGINE  
  uictrl, 21
- ELEMENT\_TYPE\_SCENE\_PROJET\_RENDU\_-  
  PARTICULES  
  uictrl, 21
- ELEMENT\_TYPE\_SCENE\_PROJET\_-  
  USERCONFIGURATION  
  uictrl, 21
- ELEMENT\_TYPE\_SCENE\_RECEPTEURSP  
  uictrl, 20
- ELEMENT\_TYPE\_SCENE\_RECEPTEURSP\_-  
  RECEPTEUR  
  uictrl, 20
- ELEMENT\_TYPE\_SCENE\_RECEPTEURSP\_-  
  RECEPTEUR\_PROPRIETES  
  uictrl, 20
- ELEMENT\_TYPE\_SCENE\_RECEPTEURSP\_-  
  RECEPTEUR\_RENDU  
  uictrl, 20
- ELEMENT\_TYPE\_SCENE\_RECEPTEURSS  
  uictrl, 20
- ELEMENT\_TYPE\_SCENE\_RECEPTEURSS\_-  
  RECEPTEUR  
  uictrl, 20
- ELEMENT\_TYPE\_SCENE\_RECEPTEURSS\_-  
  RECEPTEUR\_PROPRIETES  
  uictrl, 20
- ELEMENT\_TYPE\_SCENE\_RECEPTEURSS\_-  
  RECEPTEUR\_RENDU  
  uictrl, 20
- ELEMENT\_TYPE\_SCENE\_RECEPTEURSS\_-  
  RECEPTEURCOUPE  
  uictrl, 22
- ELEMENT\_TYPE\_SCENE\_RECEPTEURSS\_-  
  RECEPTEURCOUPE\_PROPRIETES  
  uictrl, 22
- ELEMENT\_TYPE\_SCENE\_RECEPTEURSS\_-  
  RECEPTEURCOUPE\_RENDU  
  uictrl, 22
- ELEMENT\_TYPE\_SCENE\_ROOT  
  uictrl, 20
- ELEMENT\_TYPE\_SCENE\_SOURCES  
  uictrl, 20
- ELEMENT\_TYPE\_SCENE\_SOURCES\_-  
  SOURCE  
  uictrl, 20
- ELEMENT\_TYPE\_SCENE\_SOURCES\_-  
  SOURCE\_PROPRIETES  
  uictrl, 20
- ELEMENT\_TYPE\_SCENE\_SOURCES\_-  
  SOURCE\_PUISSANCE  
  uictrl, 20
- ELEMENT\_TYPE\_SCENE\_SOURCES\_-  
  SOURCE\_RENDU  
  uictrl, 20
- ELEMENT\_TYPE\_SCENE\_VOLUMES  
  uictrl, 22
- ELEMENT\_TYPE\_SCENE\_VOLUMES\_-  
  VOLUME  
  uictrl, 22
- ELEMENT\_TYPE\_SCENE\_VOLUMES\_-  
  VOLUME\_PROPRIETES  
  uictrl, 22
- ELEMENT\_TYPE\_SCENE\_VOLUMES\_-  
  VOLUME\_RENDU  
  uictrl, 22
- ELEMENT\_TYPE\_TEXT  
  uictrl, 20
- ELEMENT\_TYPE\_TREE\_LIST  
  uictrl, 22
- ELEMENT\_TYPE\_USER\_PREFERENCE\_ITEM  
  uictrl, 22
- ELEMENT\_TYPE\_USER\_PREFERENCE\_-  
  ITEM\_ISOTEMPLATE  
  uictrl, 22
- ELEMENT\_TYPE\_USER\_PREFERENCE\_-  
  NODE  
  uictrl, 22
- element\_type  
  uictrl, 20
- getallementbytype  
  uictrl : :element, 37
- getboolconfig

- uictrl : :element, 37
- getcachedir
  - uictrl : :application, 26
- getcolorconfig
  - uictrl : :element, 37
- getdataarray
  - uictrl : :application, 26
- getdecimalconfig
  - uictrl : :element, 37
- getelementbylibelle
  - uictrl : :element, 38
- getelementbytype
  - uictrl : :element, 38
- getentierconfig
  - uictrl : :element, 38
- getid
  - uictrl : :element, 38
- getinfos
  - uictrl : :element, 38
- getlastcalculationpath
  - uictrl : :application, 26
- getlistconfig
  - uictrl : :element, 38
- getlocale
  - uictrl : :application, 26
- getmenu
  - uictrl : :element, 39
- getpositionconfig
  - uictrl : :element, 39
- getrootcore
  - uictrl : :application, 26
- getrootpreference
  - uictrl : :application, 26
- getrootreport
  - uictrl : :application, 26
- getrootscene
  - uictrl : :application, 27
- getstringconfig
  - uictrl : :element, 39
- getuserinput
  - uictrl : :application, 27
- graph
  - uictrl, 22
- GRAPH\_FITTING\_CLOSE
  - uictrl, 23
- GRAPH\_FITTING\_OPEN
  - uictrl, 23
- GRAPH\_FITTINGS\_CLOSE
  - uictrl, 23
- GRAPH\_FITTINGS\_OPEN
  - uictrl, 23
- GRAPH\_FOLDER
  - uictrl, 23
- GRAPH\_FOLDER\_OPEN
  - uictrl, 23
- GRAPH\_ITEM
  - uictrl, 23
- GRAPH\_LAST\_STATIC\_GRAPH
  - uictrl, 23
- GRAPH\_PUNCTUAL\_RECEIVERS\_CLOSE
  - uictrl, 23
- GRAPH\_PUNCTUAL\_RECEIVERS\_OPEN
  - uictrl, 23
- GRAPH\_SOUND\_SOURCES\_CLOSE
  - uictrl, 23
- GRAPH\_SOUND\_SOURCES\_OPEN
  - uictrl, 23
- GRAPH\_SURFACE\_RECEIVERS\_CLOSE
  - uictrl, 23
- GRAPH\_SURFACE\_RECEIVERS\_OPEN
  - uictrl, 23
- GRAPH\_SURFACES\_CLOSE
  - uictrl, 23
- GRAPH\_SURFACES\_OPEN
  - uictrl, 23
- hasproperty
  - uictrl : :element, 39
- hide
  - uictrl : :element, 39
- idevent
  - uictrl, 23
- IDEVENT\_BFREQ\_PRESELECTION\_BAND
  - uictrl, 24
- IDEVENT\_BFREQ\_PRESELECTION\_-  
BUILDING\_BAND
  - uictrl, 24
- IDEVENT\_BFREQ\_PRESELECTION\_-  
BUILDING\_THIRD\_BAND
  - uictrl, 24
- IDEVENT\_BFREQ\_PRESELECTION\_NONE
  - uictrl, 24
- IDEVENT\_BFREQ\_PRESELECTION\_THIRD\_-  
BAND
  - uictrl, 24
- IDEVENT\_BUILD\_VOLUMES\_FROM\_-  
TRIMESH
  - uictrl, 24
- IDEVENT\_COLLER
  - uictrl, 23
- IDEVENT\_CONVERT\_VOL\_TO\_FITTING
  - uictrl, 24
- IDEVENT\_COPIER
  - uictrl, 23
- IDEVENT\_DELETE\_ELEMENT
  - uictrl, 23
- IDEVENT\_DELETE\_FOLDER
  - uictrl, 23

- uictrl, 24
- IDEVENT\_EMPTY\_POINTER\_VERTEX\_-  
GROUP  
uictrl, 24
- IDEVENT\_GETPROPERTIES  
uictrl, 23
- IDEVENT\_IMPORT\_MATERIAL  
uictrl, 23
- IDEVENT\_INVERT\_FACE\_ORIENTATION  
uictrl, 24
- IDEVENT\_LOAD\_PARTICLE\_SIMULATION  
uictrl, 24
- IDEVENT\_LOAD\_PARTICLE\_SIMULATION\_-  
PATH  
uictrl, 24
- IDEVENT\_LOAD\_RECEPTEURSP\_-  
SIMULATION  
uictrl, 24
- IDEVENT\_LOAD\_RECEPTEURSS\_-  
SIMULATION\_BY\_TIMESTEP  
uictrl, 24
- IDEVENT\_LOAD\_RECEPTEURSS\_-  
SIMULATION\_BY\_TIMESTEP\_SUM  
uictrl, 24
- IDEVENT\_LOAD\_RECEPTEURSS\_-  
SIMULATION\_SUM  
uictrl, 24
- IDEVENT\_NEW\_ENCOMBREMENT  
uictrl, 23
- IDEVENT\_NEW\_ENCOMBREMENT\_-  
CUBOIDE  
uictrl, 23
- IDEVENT\_NEW\_MATERIAL\_GROUP  
uictrl, 23
- IDEVENT\_NEW\_RECEPTEUR\_P  
uictrl, 23
- IDEVENT\_NEW\_RECEPTEUR\_S  
uictrl, 23
- IDEVENT\_NEW\_RECEPTEUR\_S\_COUPE  
uictrl, 24
- IDEVENT\_NEW\_RECEPTEURP\_GROUP  
uictrl, 24
- IDEVENT\_NEW\_SOURCE  
uictrl, 23
- IDEVENT\_NEW\_SOURCE\_GROUP  
uictrl, 24
- IDEVENT\_NEW\_SURFACE\_GROUP  
uictrl, 23
- IDEVENT\_NEW\_SURFACE\_GROUP\_FROM\_-  
SELECTION  
uictrl, 24
- IDEVENT\_NEW\_USERFREQ  
uictrl, 23
- IDEVENT\_NEW\_USERMAT  
uictrl, 23
- IDEVENT\_NEW\_VOLUME  
uictrl, 24
- IDEVENT\_OPEN\_FOLDER  
uictrl, 24
- IDEVENT\_RECEPTEURS\_COMPUTE\_EDT  
uictrl, 24
- IDEVENT\_RECEPTEURS\_COMPUTE\_TR  
uictrl, 24
- IDEVENT\_RECP\_COMPUTE\_ACOUSTIC\_-  
PARAMETERS  
uictrl, 24
- IDEVENT\_RECP\_COMPUTE\_ADVANCED\_-  
ACOUSTIC\_PARAMETERS  
uictrl, 24
- IDEVENT\_RELOAD\_FOLDER  
uictrl, 24
- IDEVENT\_RENAME\_ELEMENT  
uictrl, 23
- IDEVENT\_REPORT\_PARTICULES\_MAKE\_-  
GABE  
uictrl, 24
- IDEVENT\_RUN\_CALCULATION  
uictrl, 23
- IDEVENT\_SELECT\_POSITION  
uictrl, 23
- IDEVENT\_SELECT\_TREE\_ITEM  
uictrl, 23
- importscene  
uictrl : :application, 27
- loadproject  
uictrl : :application, 27
- modified  
uictrl : :element, 39
- newproject  
uictrl : :application, 28
- register\_event  
uictrl : :application, 28
- register\_menu\_manager  
uictrl : :application, 28
- register\_update\_manager  
uictrl : :element, 39
- reloadgroupsfrommodel  
uictrl : :application, 28
- savelog  
uictrl : :application, 28
- saveproject  
uictrl : :application, 28
- saveshell  
uictrl : :application, 28

- sendevent
  - uictrl : :application, 29
- setreadonlyallconfig
  - uictrl : :element, 40
- setreadonlyconfig
  - uictrl : :element, 40
- uictrl, 17
  - ELEMENT\_TYPE\_BOOL, 21
  - ELEMENT\_TYPE\_BOOL\_BFREQ, 22
  - ELEMENT\_TYPE\_COLOR, 20
  - ELEMENT\_TYPE\_CORE\_CORE, 22
  - ELEMENT\_TYPE\_CORE\_CORE\_-  
BFREQSELECTION, 22
  - ELEMENT\_TYPE\_CORE\_CORE\_CONFIG,  
21
  - ELEMENT\_TYPE\_CORE\_CORE\_-  
CONFMAILLAGE, 21
  - ELEMENT\_TYPE\_CORE\_ROOT, 20
  - ELEMENT\_TYPE\_CORE\_SPPS, 21
  - ELEMENT\_TYPE\_CORE\_SPPS\_OCTREE,  
22
  - ELEMENT\_TYPE\_CORE\_TC, 22
  - ELEMENT\_TYPE\_CORE\_TLM, 22
  - ELEMENT\_TYPE\_DRAWABLE, 22
  - ELEMENT\_TYPE\_ELEMENT, 22
  - ELEMENT\_TYPE\_FLOAT, 21
  - ELEMENT\_TYPE\_FONT, 22
  - ELEMENT\_TYPE\_GAMMEABSORPTION,  
21
  - ELEMENT\_TYPE\_GAMMEFREQ\_APP, 21
  - ELEMENT\_TYPE\_GAMMEFREQ\_USER,  
21
  - ELEMENT\_TYPE\_INTEGER, 21
  - ELEMENT\_TYPE\_LIST, 21
  - ELEMENT\_TYPE\_MATERIAU\_APP, 21
  - ELEMENT\_TYPE\_MATERIAU\_USER, 21
  - ELEMENT\_TYPE\_POSITION, 21
  - ELEMENT\_TYPE\_PROPERTY\_FREQ, 21
  - ELEMENT\_TYPE\_PYTHON\_EXTENSION,  
22
  - ELEMENT\_TYPE\_REPORT\_FILE, 22
  - ELEMENT\_TYPE\_REPORT\_FOLDER, 22
  - ELEMENT\_TYPE\_REPORT\_GABE, 22
  - ELEMENT\_TYPE\_REPORT\_GABE\_GAP,  
22
  - ELEMENT\_TYPE\_REPORT\_GABE\_RECIP,  
22
  - ELEMENT\_TYPE\_REPORT\_-  
PARTVISUALISATION, 22
  - ELEMENT\_TYPE\_REPORT\_-  
RECEPTEURSSVISUALISATION,  
22
  - ELEMENT\_TYPE\_REPORT\_-  
RECEPTEURSSVISUALISATION\_-  
EDT, 22
  - ELEMENT\_TYPE\_REPORT\_-  
RECEPTEURSSVISUALISATION\_-  
GAIN, 22
  - ELEMENT\_TYPE\_REPORT\_-  
RECEPTEURSSVISUALISATION\_-  
STANDART, 22
  - ELEMENT\_TYPE\_REPORT\_-  
RECEPTEURSSVISUALISATION\_TR,  
22
  - ELEMENT\_TYPE\_REPORT\_RPI, 22
  - ELEMENT\_TYPE\_REPORT\_UNKNOWN,  
22
  - ELEMENT\_TYPE\_RESULT\_ROOT, 20
  - ELEMENT\_TYPE\_ROW, 21
  - ELEMENT\_TYPE\_ROW\_BFREQ, 21
  - ELEMENT\_TYPE\_ROW\_MATERIAU, 21
  - ELEMENT\_TYPE\_SCENE\_BDD, 21
  - ELEMENT\_TYPE\_SCENE\_BDD\_-  
CATMATERIAL, 21
  - ELEMENT\_TYPE\_SCENE\_BDD\_-  
MATERIAUX, 21
  - ELEMENT\_TYPE\_SCENE\_BDD\_-  
MATERIAUX\_APP, 21
  - ELEMENT\_TYPE\_SCENE\_BDD\_-  
MATERIAUX\_APP\_GROUP, 21
  - ELEMENT\_TYPE\_SCENE\_BDD\_-  
MATERIAUX\_APP\_MATERIAU,  
21
  - ELEMENT\_TYPE\_SCENE\_BDD\_-  
MATERIAUX\_MATERIAU\_RENDER,  
22
  - ELEMENT\_TYPE\_SCENE\_BDD\_-  
MATERIAUX\_PROPMATERIAU,  
21
  - ELEMENT\_TYPE\_SCENE\_BDD\_-  
MATERIAUX\_USER, 21
  - ELEMENT\_TYPE\_SCENE\_BDD\_-  
MATERIAUX\_USER\_GROUP, 21
  - ELEMENT\_TYPE\_SCENE\_BDD\_-  
MATERIAUX\_USER\_MATERIAU,  
21
  - ELEMENT\_TYPE\_SCENE\_BDD\_-  
SPECTRUMS, 21
  - ELEMENT\_TYPE\_SCENE\_BDD\_-  
SPECTRUMS\_APP, 21
  - ELEMENT\_TYPE\_SCENE\_BDD\_-  
SPECTRUMS\_USER, 21
  - ELEMENT\_TYPE\_SCENE\_DONNEES, 21
  - ELEMENT\_TYPE\_SCENE\_-  
ENCOMBREMENTS, 21

- ELEMENT\_TYPE\_SCENE\_-
  - ENCOMBREMENTS\_-
  - ENCOMBREMENT, 21
- ELEMENT\_TYPE\_SCENE\_-
  - ENCOMBREMENTS\_-
  - ENCOMBREMENT\_CUBOIDE, 21
- ELEMENT\_TYPE\_SCENE\_-
  - ENCOMBREMENTS\_-
  - ENCOMBREMENT\_PROPRIETES, 21
- ELEMENT\_TYPE\_SCENE\_-
  - ENCOMBREMENTS\_-
  - ENCOMBREMENT\_RENDU, 22
- ELEMENT\_TYPE\_SCENE\_-
  - GROUPEURFACES, 20
- ELEMENT\_TYPE\_SCENE\_-
  - GROUPEURFACES\_GROUPE, 20
- ELEMENT\_TYPE\_SCENE\_-
  - GROUPEURFACES\_GROUPE\_-
  - VERTEX, 20
- ELEMENT\_TYPE\_SCENE\_PROJET, 21
- ELEMENT\_TYPE\_SCENE\_PROJET\_-
  - CONFIGURATION, 20
- ELEMENT\_TYPE\_SCENE\_PROJET\_-
  - ENVIRONNEMENTCONF, 22
- ELEMENT\_TYPE\_SCENE\_PROJET\_-
  - INFORMATION, 22
- ELEMENT\_TYPE\_SCENE\_PROJET\_-
  - RENDU, 21
- ELEMENT\_TYPE\_SCENE\_PROJET\_-
  - RENDU\_MODEL, 21
- ELEMENT\_TYPE\_SCENE\_PROJET\_-
  - RENDU\_ORIGINE, 21
- ELEMENT\_TYPE\_SCENE\_PROJET\_-
  - RENDU\_PARTICULES, 21
- ELEMENT\_TYPE\_SCENE\_PROJET\_-
  - USERCONFIGURATION, 21
- ELEMENT\_TYPE\_SCENE\_-
  - RECEPTEURSP, 20
- ELEMENT\_TYPE\_SCENE\_-
  - RECEPTEURSP\_RECEPTEUR, 20
- ELEMENT\_TYPE\_SCENE\_-
  - RECEPTEURSP\_RECEPTEUR\_-
  - PROPRIETES, 20
- ELEMENT\_TYPE\_SCENE\_-
  - RECEPTEURSP\_RECEPTEUR\_-
  - RENDU, 20
- ELEMENT\_TYPE\_SCENE\_-
  - RECEPTEURSS, 20
- ELEMENT\_TYPE\_SCENE\_-
  - RECEPTEURSS\_RECEPTEUR, 20
- ELEMENT\_TYPE\_SCENE\_-
  - RECEPTEURSS\_RECEPTEUR\_-
  - PROPRIETES, 20
- ELEMENT\_TYPE\_SCENE\_-
  - RECEPTEURSS\_RECEPTEUR\_-
  - RENDU, 20
- ELEMENT\_TYPE\_SCENE\_-
  - RECEPTEURSS\_RECEPTEURCOUPE, 22
- ELEMENT\_TYPE\_SCENE\_-
  - RECEPTEURSS\_-
  - RECEPTEURCOUPE\_PROPRIETES, 22
- ELEMENT\_TYPE\_SCENE\_-
  - RECEPTEURSS\_-
  - RECEPTEURCOUPE\_RENDU, 22
- ELEMENT\_TYPE\_SCENE\_ROOT, 20
- ELEMENT\_TYPE\_SCENE\_SOURCES, 20
- ELEMENT\_TYPE\_SCENE\_SOURCES\_-
  - SOURCE, 20
- ELEMENT\_TYPE\_SCENE\_SOURCES\_-
  - SOURCE\_PROPRIETES, 20
- ELEMENT\_TYPE\_SCENE\_SOURCES\_-
  - SOURCE\_PUISSANCE, 20
- ELEMENT\_TYPE\_SCENE\_SOURCES\_-
  - SOURCE\_RENDU, 20
- ELEMENT\_TYPE\_SCENE\_VOLUMES, 22
- ELEMENT\_TYPE\_SCENE\_VOLUMES\_-
  - VOLUME, 22
- ELEMENT\_TYPE\_SCENE\_VOLUMES\_-
  - VOLUME\_PROPRIETES, 22
- ELEMENT\_TYPE\_SCENE\_VOLUMES\_-
  - VOLUME\_RENDU, 22
- ELEMENT\_TYPE\_TEXT, 20
- ELEMENT\_TYPE\_TREE\_LIST, 22
- ELEMENT\_TYPE\_USER\_PREFERENCE\_-
  - ITEM, 22
- ELEMENT\_TYPE\_USER\_PREFERENCE\_-
  - ITEM\_ISOTEMPLATE, 22
- ELEMENT\_TYPE\_USER\_PREFERENCE\_-
  - NODE, 22
- element\_type, 20
- graph, 22
- GRAPH\_FITTING\_CLOSE, 23
- GRAPH\_FITTING\_OPEN, 23
- GRAPH\_FITTINGS\_CLOSE, 23
- GRAPH\_FITTINGS\_OPEN, 23
- GRAPH\_FOLDER, 23
- GRAPH\_FOLDER\_OPEN, 23
- GRAPH\_ITEM, 23
- GRAPH\_LAST\_STATIC\_GRAPH, 23
- GRAPH\_PUNCTUAL\_RECEIVERS\_-
  - CLOSE, 23
- GRAPH\_PUNCTUAL\_RECEIVERS\_OPEN, 23
- GRAPH\_SOUND\_SOURCES\_CLOSE, 23
- GRAPH\_SOUND\_SOURCES\_OPEN, 23

- GRAPH\_SURFACE\_RECEIVERS\_CLOSE, 23  
 GRAPH\_SURFACE\_RECEIVERS\_OPEN, 23  
 GRAPH\_SURFACES\_CLOSE, 23  
 GRAPH\_SURFACES\_OPEN, 23  
 idevent, 23  
 IDEVENT\_BFREQ\_PRESELECTION\_-  
   BAND, 24  
 IDEVENT\_BFREQ\_PRESELECTION\_-  
   BUILDING\_BAND, 24  
 IDEVENT\_BFREQ\_PRESELECTION\_-  
   BUILDING\_THIRD\_BAND, 24  
 IDEVENT\_BFREQ\_PRESELECTION\_-  
   NONE, 24  
 IDEVENT\_BFREQ\_PRESELECTION\_-  
   THIRD\_BAND, 24  
 IDEVENT\_BUILD\_VOLUMES\_FROM\_-  
   TRIMESH, 24  
 IDEVENT\_COLLER, 23  
 IDEVENT\_CONVERT\_VOL\_TO\_FITTING,  
   24  
 IDEVENT\_COPIER, 23  
 IDEVENT\_DELETE\_ELEMENT, 23  
 IDEVENT\_DELETE\_FOLDER, 24  
 IDEVENT\_EMPTY\_POINTER\_VERTEX\_-  
   GROUP, 24  
 IDEVENT\_GETPROPERTIES, 23  
 IDEVENT\_IMPORT\_MATERIAL, 23  
 IDEVENT\_INVERT\_FACE\_ORIENTATION,  
   24  
 IDEVENT\_LOAD\_PARTICLE\_-  
   SIMULATION, 24  
 IDEVENT\_LOAD\_PARTICLE\_-  
   SIMULATION\_PATH, 24  
 IDEVENT\_LOAD\_RECEPTEURSP\_-  
   SIMULATION, 24  
 IDEVENT\_LOAD\_RECEPTEURSS\_-  
   SIMULATION\_BY\_TIMESTEP, 24  
 IDEVENT\_LOAD\_RECEPTEURSS\_-  
   SIMULATION\_BY\_TIMESTEP\_SUM,  
   24  
 IDEVENT\_LOAD\_RECEPTEURSS\_-  
   SIMULATION\_SUM, 24  
 IDEVENT\_NEW\_ENCOMBREMENT, 23  
 IDEVENT\_NEW\_ENCOMBREMENT\_-  
   CUBOIDE, 23  
 IDEVENT\_NEW\_MATERIAL\_GROUP, 23  
 IDEVENT\_NEW\_RECEPTEUR\_P, 23  
 IDEVENT\_NEW\_RECEPTEUR\_S, 23  
 IDEVENT\_NEW\_RECEPTEUR\_S\_COUPE,  
   24  
 IDEVENT\_NEW\_RECEPTEURP\_GROUP,  
   24  
 IDEVENT\_NEW\_SOURCE, 23  
 IDEVENT\_NEW\_SOURCE\_GROUP, 24  
 IDEVENT\_NEW\_SURFACE\_GROUP, 23  
 IDEVENT\_NEW\_SURFACE\_GROUP\_-  
   FROM\_SELECTION, 24  
 IDEVENT\_NEW\_USERFREQ, 23  
 IDEVENT\_NEW\_USERMAT, 23  
 IDEVENT\_NEW\_VOLUME, 24  
 IDEVENT\_OPEN\_FOLDER, 24  
 IDEVENT\_RECEPTEURS\_COMPUTE\_-  
   EDT, 24  
 IDEVENT\_RECEPTEURS\_COMPUTE\_TR,  
   24  
 IDEVENT\_RECPCOMPUTE\_-  
   ACOUSTIC\_PARAMETERS, 24  
 IDEVENT\_RECPCOMPUTE\_-  
   ADVANCED\_ACOUSTIC\_-  
   PARAMETERS, 24  
 IDEVENT\_RELOAD\_FOLDER, 24  
 IDEVENT\_RENAME\_ELEMENT, 23  
 IDEVENT\_REPORT\_PARTICULES\_-  
   MAKE\_GABE, 24  
 IDEVENT\_RUN\_CALCULATION, 23  
 IDEVENT\_SELECT\_POSITION, 23  
 IDEVENT\_SELECT\_TREE\_ITEM, 23  
 uictrl : :application, 25  
   clearlogdata, 26  
   clearshellhisto, 26  
   getcachedir, 26  
   getdataarray, 26  
   getlastcalculationpath, 26  
   getlocale, 26  
   getrootcore, 26  
   getrootpreference, 26  
   getrootreport, 26  
   getrootscene, 27  
   getuserinput, 27  
   importscene, 27  
   loadproject, 27  
   newproject, 28  
   register\_event, 28  
   register\_menu\_manager, 28  
   reloadgroupsfrommodel, 28  
   savelog, 28  
   saveproject, 28  
   saveshell, 28  
   sendevent, 29  
 uictrl : :e\_file, 30  
   buildfullpath, 30  
 uictrl : :Element, 31  
 uictrl : :element, 32  
   appendfilsbytype, 33  
   appendpropertybool, 33  
   appendpropertycolor, 33

appendpropertydecimal, 34  
appendpropertyentier, 34  
appendpropertyfont, 35  
appendpropertylist, 35  
appendpropertyposition, 35  
appendpropertytext, 36  
appenduserelement, 36  
childs, 36  
deleteallelementbytype, 36  
deleteallelementbytyper, 36  
deleteelementbyxmlid, 37  
getallelementbytype, 37  
getboolconfig, 37  
getcolorconfig, 37  
getdecimalconfig, 37  
getelementbylibelle, 38  
getelementbytype, 38  
getentierconfig, 38  
getid, 38  
getinfos, 38  
getlistconfig, 38  
getmenu, 39  
getpositionconfig, 39  
getstringconfig, 39  
hasproperty, 39  
hide, 39  
modified, 39  
register\_update\_manager, 39  
setreadonlyallconfig, 40  
setreadonlyconfig, 40  
updateboolconfig, 40  
updatedecimalconfig, 40  
updateentierconfig, 40  
updatelistconfig, 41  
updatepositionconfig, 41  
updatestringconfig, 41  
updateboolconfig  
  uictrl : :element, 40  
updatedecimalconfig  
  uictrl : :element, 40  
updateentierconfig  
  uictrl : :element, 40  
updatelistconfig  
  uictrl : :element, 41  
updatepositionconfig  
  uictrl : :element, 41  
updatestringconfig  
  uictrl : :element, 41