

I-SIMPA Scripting Guide

1.1.4

Généré par Doxygen 1.6.1

Tue Jan 19 09 :44 :49 2010

Table des matières

1	Guide de création de script I-SIMPA	1
2	Ajouter des fonctionnalités	3
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
3	Ajouter des données dans les projets	7
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
4	Index des espaces de nommage	11
4.1	Liste des espaces de nommage	11
5	Index des classes	13
5.1	Hiérarchie des classes	13
6	Index des classes	15
6.1	Liste des classes	15
7	Documentation des espaces de nommage	17
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
8	Documentation des classes	25
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	updatepositionconfig	41
8.4.2.39	updatestringconfig	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_sourceid), (uictrl._("Disabl
e"), self.disable_grp_sourceid)]
    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 appelée 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 :

uictrl (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 :

uictrl : :application (Classe python de contrôle de I-SIMPA)	25
uictrl : :e_file	30
uictrl : :Element (Alias)	31
uictrl : :element (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_GROUPEURFACES, ELEMENT_TYPE_SCENE_-
GROUPEURFACES_GROUPE, ELEMENT_TYPE_SCENE_GROUPEURFACES_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_PROPMATERIAU,      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_RECPC, 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_RECP_COMPUTE_ACOUSTIC_PARAMETERS Evenement calculer les paramètres acoustique 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_RECP_COMPUTE_ADVANCED_ACOUSTIC_PARAMETERS Evenement calculer les paramètres acoustique 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 une 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 ¶mTetgen="")
- 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:

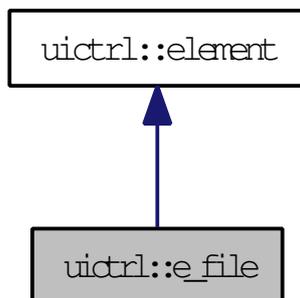
pyel [Element](#) de l'arbre concerné

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

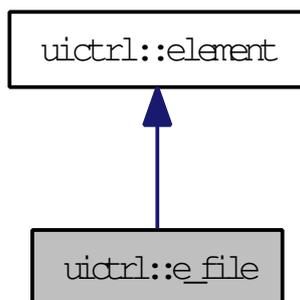
parameters Voir les membres de [uictrl : :idevent](#) pour plus de détails.

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

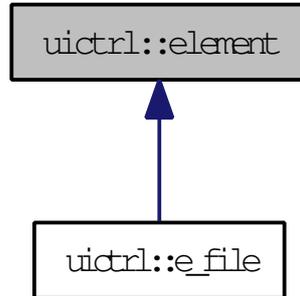
- typedef `element_type` `ELEMENT_TYPE`
- typedef `graph` `GRAPH`
- typedef `idevent` `IDEVENT`

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 [deleteallelementbytypeper](#) (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 `updateconfig` (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 : :deleteallementbytype (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 : :deleteallementbytype (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 : :getallementbytype (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_-
 - GROUPESURFACES, 20
- ELEMENT_TYPE_SCENE_-
 - GROUPESURFACES_GROUPE, 20
- ELEMENT_TYPE_SCENE_-
 - GROUPESURFACES_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
deleteallementbytype, 36
deleteallementbytyper, 36
deleteelementbyxmlid, 37
getallementbytype, 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