

Université Henri Poincaré

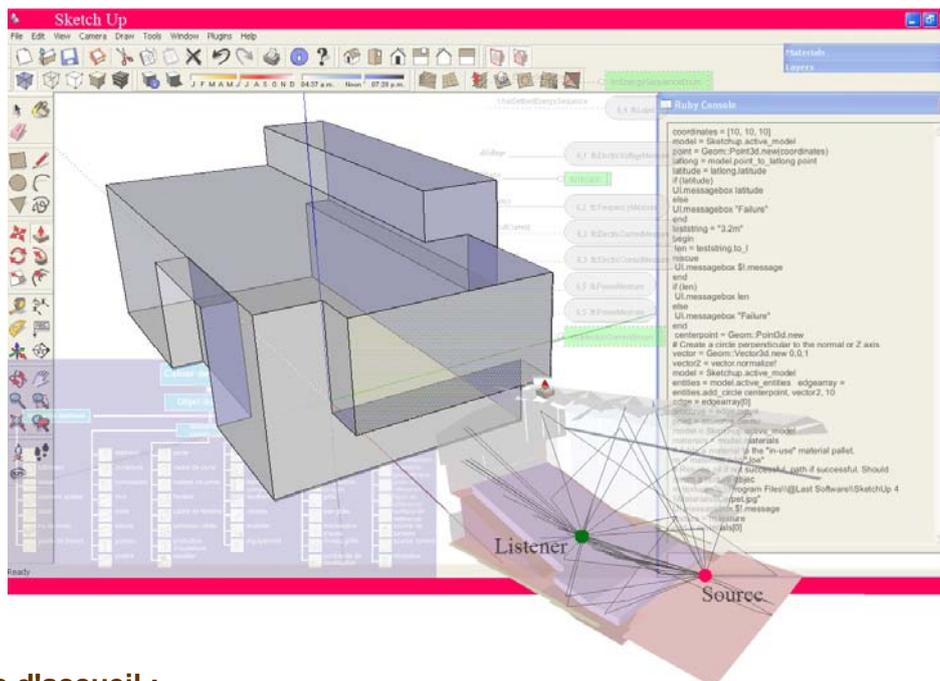
Ecole National Supérieur d'Architecture de Nancy
Ecole National Supérieur d'Architecture de Strasbourg
Institut National des Sciences Appliquée de Strasbourg
Institut National Polytechnique de Lorraine

Mémoire de Master Design Global

" Modélisation et Simulation des Espaces Bâtis "

Modèle 3D et Traitement Acoustique

Définition d'un outil d'aide à la conception acoustique
des espaces architecturaux



Laboratoire d'accueil :

CRAI: Centre de Recherche en Architecture et Ingénierie

Sous la direction de :

Didier BUR

Présenté par :

José Carlos LEON

Octobre 2006

Remerciements

Je remercie tout particulièrement **Didier BUR** pour m'avoir suivi durant cette expérience et pour m'avoir orienté dans le développement de ce sujet.

Je remercie également toute l'équipe du CRAI pour avoir contribué à cette expérience, plus particulièrement :

Jean-Pierre PERRIN, directeur du CRAI.

Jean-Claude BIGNON responsable du Master MSEB.

Annie BOUYER pour sa gentillesse et sa disponibilité.

Mes derniers remerciements vont pour tous mes **collègues du CRAI**.

Sommaire

	Page
1. Introduction	1
2. Problématique	2
2.1 Objectifs	2
2.2 Démarche	2
3. Etat de l'art	4
3.1 Modèles et outils existants	4
3.1.1 CATT-Acoustic®	4
3.1.1.1 Saisie des données	5
3.1.1.2 Module de calcul	5
3.1.1.3 Propriétés acoustiques des matériaux	7
3.1.2 Axys DDA®	7
3.1.2.1 Le contrôle de directivité DDS	8
3.1.3 Bilan	10
3.2 Cahier des charges fonctionnel	10
3.2.1 Plan type d'un cahier des charges fonctionnel	11
3.2.2 Rappeler les objectifs	13
3.2.3 Résumé de la problématique	13
3.2.4 Décrire la solution détaillée	13
3.2.5 Bilan	14
4. Outils de travail	15
4.1 Le format GEO dans CATT-Acoustic®	15
4.2 Méthodes de modélisation dans CATT-Acoustic®	17
4.2.1 Approche par la méthode de Lancer de Rayons	17
4.2.2 Méthode de Réflexions Précoces Détaillées ISM	18
4.2.3 Méthode de Calcule Détaillée RTC	18
4.2.4 Gestion de transparence	19
4.3 Une nouvelle interface de saisie pour CATT-Acoustic®	19
4.3.1 Pourquoi utiliser l'interface SketchUp® ?	20
4.3.2 Langage script propre de SketchUp®	21
4.4 Bilan	21

	Page
5. Lumière et son	23
5.1 L'algorithme Ray Tracing	23
5.1.1 Principe du Lancer de Rayon	23
5.1.2 Cheminement du Rayon	24
5.1.3 Réflexion	25
5.1.4 Transparence et réfraction	26
5.2 Algorithme	28
5.3 Traitement du son et de la lumière	29
5.4 Bilan	30
6. Transfert de données	31
6.1 International Alliance for Interoperability IAI	31
6.1.1 Qu'est-ce que les IFC ?	31
6.1.2 Objets IFC	32
6.1.3 Arborescence des principales classes d'objet du modèle IFC.....	33
6.1.3.1 Projet	33
6.1.3.2 Élément spatiaux	33
6.1.3.3 Les éléments de construction	35
6.1.3.4 Environnement de modélisation	40
6.1.4 Les IFC et STEP (ISO 10303-11).....	42
6.1.5 Dernières spécification IFC2x3.....	43
6.1.5.1 Définition d'une entité IFC	43
6.2 Interfaçage de programmation d'applications de Ruby	46
6.2.1 Programmation orientée objet sur SketchUp®	46
6.2.1.1 Définition d'une classe API Ruby	46
6.3 Bilan	48
7. Conclusions et perspectives	49
7.1 Perspectives	50
7.1.1 Test d'utilisation	51
8. Bibliographie	52
9. Annexes	54
9.1 Cahier des charges fonctionnel <i>Module d'échange de données Skp2Geo</i>	54

1. Introduction

L'acoustique architecturale est le domaine scientifique et technologique qui vise à comprendre et maîtriser la qualité sonore des bâtiments. L'application privilégiée de l'acoustique architecturale est, bien entendu, la construction des salles de spectacle mais cette technique est aussi utilisée dans la conception d'autres bâtiments, comme les lieux de travail ou les maisons d'habitation pour lesquels la qualité acoustique peut avoir d'importantes implications en matière de santé et de bien-être. Le traitement acoustique à l'aide des logiciels informatiques cherche à comprendre et améliorer cette dite qualité sonore des espaces.

Aujourd'hui existent dans le domaine de l'architecture, comme dans d'autres domaines, plusieurs outils informatiques spécialisés qui répondent aux besoins particuliers de la conception des espaces notamment les outils CAO pour le design architectural. En ce qui concerne le domaine de l'acoustique architecturale ce n'est pas l'exception, plusieurs outils informatiques sont à disposition des architectes et ingénieurs pour traiter l'acoustique des espaces bâtis tant réels que virtuels.

Au long de ce travail nous allons essayer d'identifier clairement les différentes entités qui conforment le traitement acoustique d'un espace donné pour pouvoir développer un prototype d'outil d'aide à la conception acoustique des espaces architecturaux. Nous allons aborder une réflexion sur la définition d'un *cahier des charges fonctionnel* de spécification pour un prototype informatique d'échanges de données géométriques et acoustiques entre un modéleur (*SketchUp®*) et un logiciel de simulation acoustique (*CATT-Acoustic®*) où les IFC ont été présentes pour faciliter ce passage.

Notre contribution à l'élaboration d'un outil de transfert de données s'inscrit dans le cadre des travaux de recherche au sein du laboratoire Centre de Recherche en Architecture et Ingénierie (*CRAI*) de l'Ecole Nationale Supérieure d'Architecture de Nancy (*ENSAM*).

2. Problématique

2.1 Objectifs

Cette étude prétend proposer une définition pour l'éventuel développement d'un outil informatique servant à la modélisation de données géométriques et acoustiques ainsi qu'à leur transfert entre logiciels.

2.2 Démarche

Afin de développer notre problématique, il est important de se questionner sur le contenu et la formalisation du problème étudié. Il est donc essentiel de bien connaître l'objet du problème :

- Dans un premier temps le traitement acoustique des espaces bâtis et la similitude qu'il présente avec ce de la lumière.
- Dans un deuxième temps, strictement lié au premier terme (espaces bâtis), on trouve la modélisation tridimensionnelle, qui fait également objet de travail lorsqu'il s'agit de préparer un modèle virtuel en vue de son exploitation par des logiciels dédiés.
- Dans un troisième temps nous nous posons le problème du transfert de données soit géométriques ou non géométriques d'un logiciel à l'autre, où les IFC visent à faciliter ce passage.

A ce niveau-là notre problématique est, alors, celle du paramétrage des objets géométriques et non géométriques d'une part dans un modeleur 3D et leur transfert vers un logiciel de traitement acoustique d'autre part, en visant l'utilisation des IFC comme structure de données sur laquelle se basera la description du transfert. Un projet de mise au point d'une interface entre un modeleur (*SketchUp®*) et un logiciel de simulation de traitement acoustique (*CATT-Acoustic®*) est envisagé.

Outre les objets géométriques du modèle, les entités à manipuler sont les propriétés acoustiques de parois (absorption, diffusion en référence à une base de données ou à partir d'une codification par matériaux-texture), les sources acoustiques (nom en référence à un fichier de définition, directivité, puissance, spectre, vecteur d'orientation), les récepteurs (point de réception orientés, localisation des récepteurs), etc.

3. Etat de l'art

Afin d'élaborer une proposition de réponse vis-à-vis de la problématique énoncée, il est nécessaire d'analyser les différents aspects qu'englobe notre démarche, celle-ci se déroule en deux temps :

- Dans un premier temps, nous allons nous attacher à étudier l'ensemble des ressources et des liens existants sur la thématique du traitement acoustique.
- Dans un second temps, nous allons préciser le concept de cahier des charges et sa définition.

3.1 Modèles et outils existants

L'ensemble des différents logiciels mis sur le marché comme outil d'aide au traitement acoustique, sont basés sur la modélisation des algorithmes qui servent à gérer la simulation des sources et le positionnement du son dans une géométrie 3D définie.

Les outils disponibles sont très semblables entre eux. A travers l'analyse de deux logiciels, *CATT-Acoustic*® et *Axys*® *DDA*, couramment utilisés pour la simulation acoustique, nous tenterons d'en élaborer les avantages et les limites ainsi que leurs apports vis-à-vis du processus de conception.

3.1.1 *CATT-Acoustic*®, développé par Euphonia

Euphonia propose une gamme étendue de logiciels qui couvrent les différents domaines de la simulation acoustique numérique.

Euphonia distribue *CATT-Acoustic*®, un des logiciels de simulation acoustique le plus avancés dans le domaine de l'acoustique virtuelle, également reconnu et largement utilisé dans l'ingénierie de l'acoustique du bâtiment et de la sonorisation. *CATT-Acoustic*® est doté de sa propre interface de saisie de la géométrie.

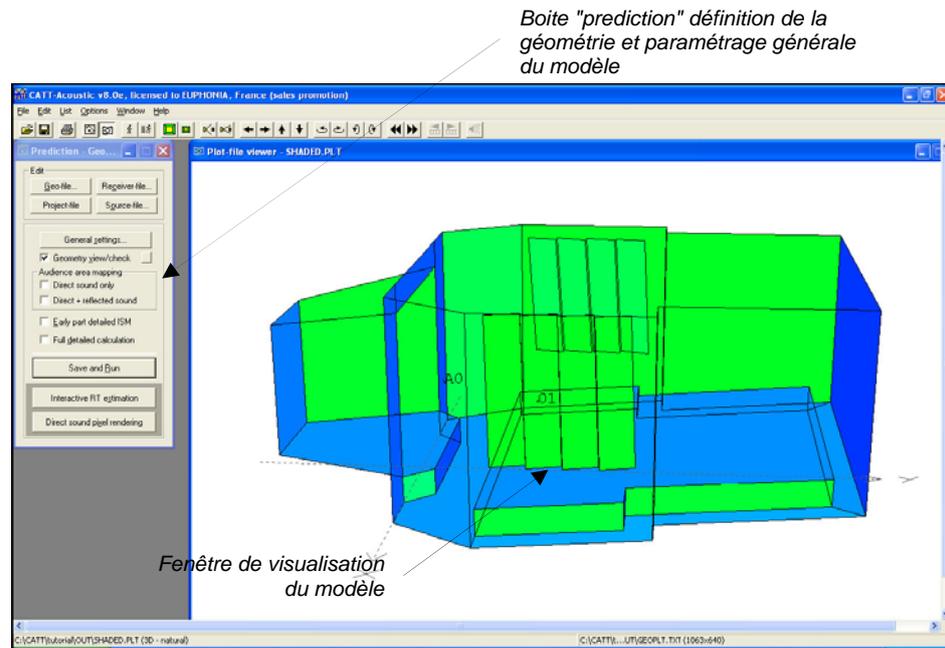


Figure 3.1 Capture d'écran interface d'utilisateur CATT-Acoustic.

3.1.1.1 Saisie des données

La géométrie des salles et les données acoustiques sont définies à l'aide de fichiers texte ASCII¹ hiérarchisés. La définition de constantes, l'interprétation d'expressions arithmétiques et de fonctions mathématiques peuvent être utilisées dans la définition des coordonnées. Ces possibilités permettent les modifications de la géométrie étudiée et autorisent un paramétrage de formes types. Des locaux dessinés à l'aide d'Autocad peuvent être exportés vers le format propre à CATT à l'aide d'un ensemble de fonctions AutoLISP permettant la définition de caractéristiques d'absorption et de la géométrie au sein d'Autocad. CATT peut également importer des données au format DXF à l'aide d'un module DXF2Geo.

3.1.1.2 Module de calcul

Le calcul prévisionnel repose sur une méthode panachée spécifique qui associe une recherche par lancer de cônes dite RTC (Randomized Tail-corrected Cone-tracing), un lancer de rayons ou la méthode des sources images. Le modèle gère :

- 250 sources sonores.
- 100 points de réception individualisés.
- Un facteur de diffusion variable en fonction de la fréquence.

Note 1. La norme **ASCII** (**A**merican **S**tandard **C**ode for **I**nformation **I**nterchange), a longtemps été utilisée pour le codage de caractères en informatique.

- Une diffusion "automatique" en paroi en fonction de la taille des éléments discrétisés.
- Un domaine de fréquences étendu à l'ensemble du spectre audible.

Les cartographies de paramètres sont effectuées à partir d'un lancer de rayons. Elles proposent une visualisation 2D/3D couleurs des résultats. Le paramétrage de la densité de rayon et des temps de trajet sont automatiquement représentés.

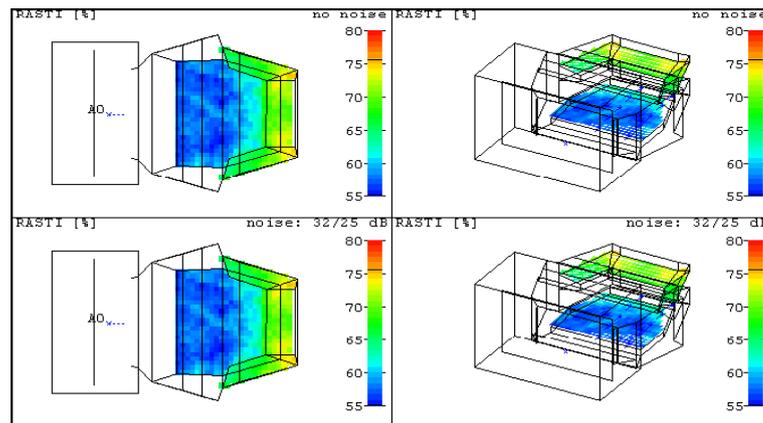


Figure 3.2 Cartographie des champs sonore.

Une fenêtre de calcul interactif de la durée de réverbération permet d'apprécier instantanément l'incidence d'une modification de l'absorption des parois.

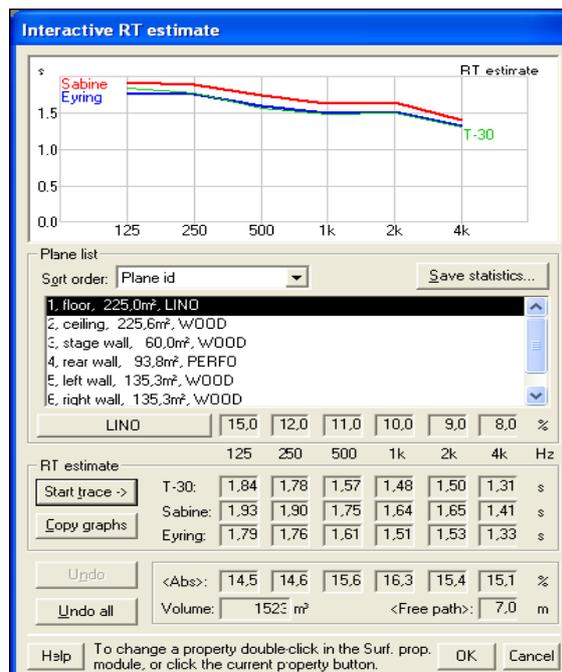


Figure 3.3 Fenêtre de calcul interactif de la réverbération de locaux (Sabine/Eyring).

3.1.1.3 Propriétés acoustiques des matériaux

Ce module se présente sous la forme d'un gestionnaire de données. Il peut importer ou exporter des données au format texte et s'interfacer avec le logiciel *AlphaBase*. Il gère les coefficients d'absorption sur les octaves centrés sur 125 à 16KHz et les facteurs de diffusion. Chaque matériau peut se voir affecté une couleur.

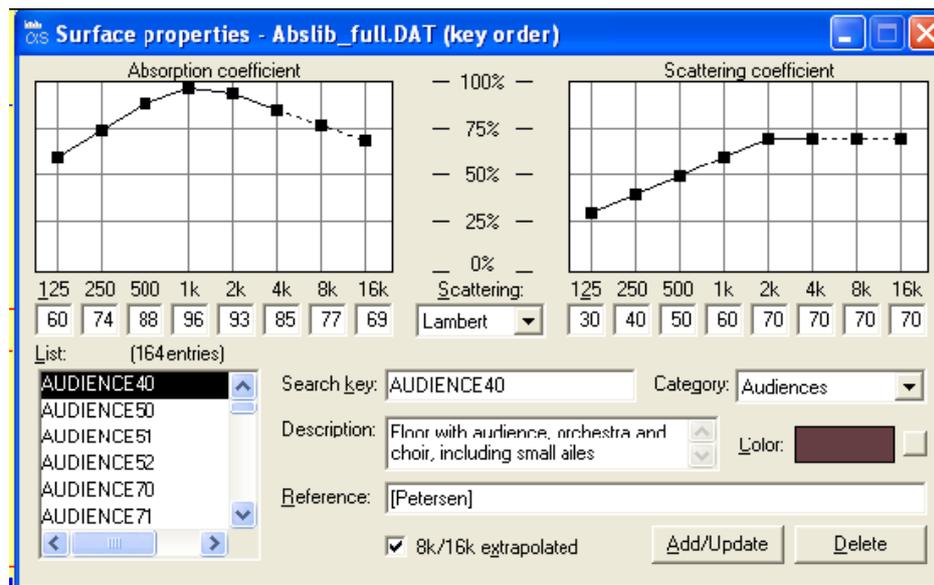


Figure 3.4 Fenêtre de propriétés de matériaux.

Cet outil est très complet grâce aux modules (*plugins*) de gestion et paramétrage acoustique que propose Euphonia.

3.1.2 Axys DDA®, développé par Duran Audio France

Le logiciel Axys DDA a été développé pour la simulation, l'optimisation et la programmation des enceintes *Axys Intellivox* et du système *Axys Target*.

Il s'agit d'un programme 32 bits compatible Windows intégrant les fonctions suivantes :

- Modélisation géométrique.
- Optimisation du rayonnement.
- Visualisations graphiques de principaux indices acoustiques.

Ses principales fonctionnalités sont :

- Définition de la géométrie 3D, grâce à un éditeur textuel tiers ou par importation d'un modèle au format CATT-Acoustic® ou Ease®.
- Simulation et optimisation (mode DDS).
 - Conception modulaire du réseau.
 - Synthèse du rayonnement souhaité.
 - Création de filtres de sortie.
- Simulation et visualisation de la réponse du réseau.
- Exportation des fichiers de directivité au format SD2 pour CATT-Acoustic® et Ease®.
- Export et copies des graphiques sous divers formats.

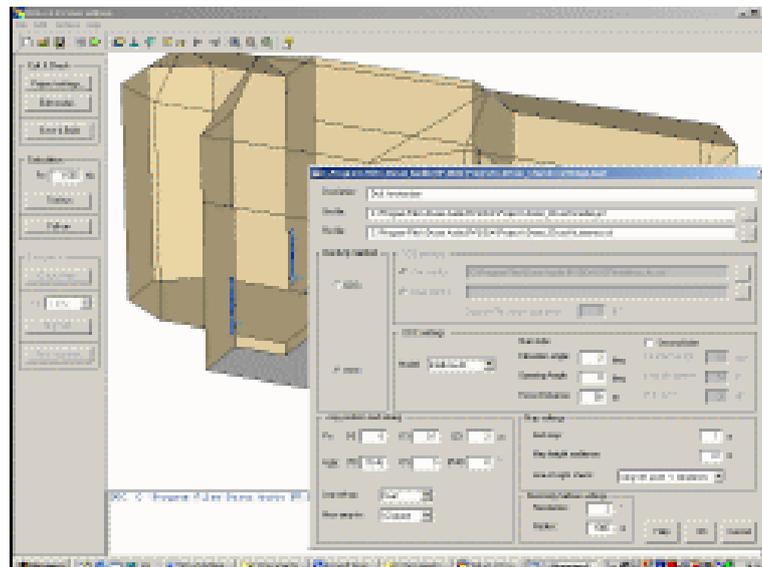


Figure 3.5 Capture d'écran interface d'utilisateur Axys DDA.

3.1.2.1 Le contrôle de directivité DDS

La technologie de contrôle de directivité numérique DDS (Digital Directivity Synthesis) est un concept révolutionnaire de contrôle numérique de la directivité d'un réseau de transducteurs. Cette technologie de deuxième génération a été développée par Duran Audio au début des années 2000.

Associant des algorithmes extrêmement puissants à une nouvelle génération de DSP, elle offre des possibilités jusqu'alors inenvisageables d'optimisation du rayonnement des réseaux de transducteurs.

Grâce à cette technologie, il est possible de synthétiser, en trois dimensions, un modèle de dispersion souhaité en tenant compte des contraintes physiques prédéfinies du réseau (nombre de transducteurs, position, dimensions du réseau etc.)

La technologie DDS est basée sur un algorithme d'optimisation (méthode des moindres carrés pondérés).

A partir de la distribution du niveau de pression souhaitée dans un volume en trois dimensions, les coefficients de filtrage optimums de chaque transducteur du réseau sont calculés. En d'autres termes, "l'illumination" souhaitée du volume est "renvoyée" vers le réseau plutôt que de visualiser la réponse du réseau dans le volume.

Considérons la configuration de réseau de transducteurs 3D dans la figure suivante :

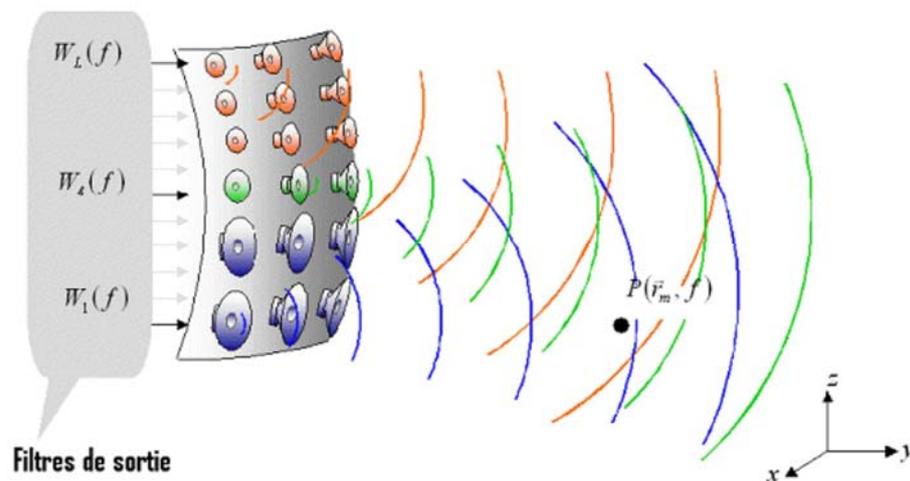


Figure 3.6 Technologie DDS réseau de transducteurs 3D

Comme montré, la géométrie du réseau n'est pas limitée à une disposition linéaire (courbe ou non) mais peut également revêtir une disposition plane (courbe ou non). Le réseau dispose de N canaux de sortie. Chaque canal pilote un ou plusieurs transducteurs.

Cet outil est donc assez similaire au précédent. Il propose la visualisation graphique des champs acoustiques ainsi que l'optimisation du rayonnement.

3.1.3 Bilan

Ces outils sont fondés sur la prédiction, la représentation graphique et le traitement virtuel du son, cependant ils présentent une grande complexité et approfondissement dans le domaine acoustique.

Dans le domaine de l'ingénierie acoustique, ces outils demeurent indispensables pour :

- Le traitement et la simulation acoustique virtuelle.
- La sonorisation de bâtiments construits.

Cependant la création et la gestion de la géométrie 3D sont définies grâce à un éditeur textuel, cela rend difficile l'utilisation de ces outils comme aide à la conception des espaces à sonoriser.

Nous remarquons ainsi une complexité au niveau de premiers pas du processus de conception acoustique, la géométrie même des bâtiments à traiter.

3.2 Cahier des charges fonctionnel CdCF

L' *AFITEP*² définit le *cahier des charges* comme un terme générique pour désigner un document visant à définir exhaustivement les spécifications de bases *d'un produit* à réaliser. Dans le cadre de notre vision, le cahier des charges est un document contractuel qui fait le lien entre les spécifications détaillées du système à développer et les équipes de développeurs.

Le CdCF permet donc par une approche systémique d'apprentissage de la complexité du projet, pour ensuite apporter la rigueur de construction de la solution par une approche analytique.

Le CdCF se compose de trois à quatre parties principales :

- Une présentation générale du problème : elle est destinée à donner toutes les informations générales utiles concernant le produit : marché, contexte du projet, objectifs, énoncé du besoin, environnement du produit.

Note 2. L'**AFITEP** (Association Francophone de Management de Projet) est une association loi 1901, créée en 1982. Elle certifie les compétences dans les métiers du management de projet.

- Une expression fonctionnelle des besoins (partie principale) : elle décrit et définit les fonctions de service du produit, les contraintes, les critères d'appréciation...
- Un appel à des variantes (optionnel) : cette partie demande et fixe des limites à l'étude ou à d'autres solutions possibles.
- Un cadre de réponse : il est destiné à simplifier et à codifier la façon de répondre.

3.2.1 Plan type d'un cahier des charges fonctionnel

Le plan type d'un cahier des charges fonctionnel selon la norme *AFNOR*³ X50-151 :

1. Présentation générale du problème.
 - 1.1. Projet.
 - 1.1.1. Finalités.
 - 1.1.2. Espérances de retour sur investissement.
 - 1.2. Contexte.
 - 1.2.1. Situation du projet par rapport aux autres projets de l'entreprise.
 - 1.2.2. Etudes déjà effectuées.
 - 1.2.3. Etudes menées sur des sujets voisins.
 - 1.2.4. Suites prévues.
 - 1.2.5. Nature des prestations demandées.
 - 1.2.6. Parties concernées par le déroulement du projet et ses résultats (demandeurs, utilisateurs).
 - 1.2.7. Caractère confidentiel, s'il y a lieu.
 - 1.3. Enoncé du besoin (finalités du produit pour le futur utilisateur tel que prévu par le demandeur).
 - 1.4. Environnement du produit recherché.
 - 1.4.1. Listes exhaustives des éléments (personnes, équipements, matières...) et contraintes (environnement).
 - 1.4.2. Caractéristiques pour chaque élément de l'environnement.

Note 3. L'**AFNOR** (Association Française de **NOR**malisation) est un organisme placé sous le contrôle de l'Etat. Il détermine les normes applicables aux techniques, aux sciences et aux commerces.

- 2. Expression fonctionnelle du besoin.**
 - 2.1. Fonctions de service et de contrainte.**
 - 2.1.1. Fonctions de service principales (qui sont la raison d'être du produit).
 - 2.1.2. Fonctions de service complémentaires (qui améliorent, facilitent ou complètent le service rendu).
 - 2.1.3. Contraintes (limitations à la liberté du concepteur réalisateur).
 - 2.2. Critères d'appréciation (en soulignant ceux qui sont déterminants pour l'évaluation des réponses).**
 - 2.3. Niveaux des critères d'appréciation et ce qui les caractérise.**
 - 2.3.1. Niveaux dont l'obtention est imposée.
 - 2.3.2. Niveaux souhaités mais révisables.
- 3. Cadre de réponse.**
 - 3.1. Pour chaque fonction.**
 - 3.1.1. Solution proposée.
 - 3.1.2. Niveau atteint pour chaque critère d'appréciation de cette fonction et modalités de contrôle.
 - 3.1.3. Part du prix attribué à chaque fonction.
 - 3.2. Pour l'ensemble du produit.**
 - 3.2.1. de la réalisation de la version de base.
 - 3.2.2. Options et variantes proposées non retenues au cahier des charges.
 - 3.2.3. Mesures prises pour respecter les contraintes et leurs conséquences économiques.
 - 3.2.4. Outils d'installation, de maintenance ... à prévoir.
 - 3.2.5. Décomposition en modules, sous-ensembles.
 - 3.2.6. Prévisions de fiabilité.
 - 3.2.7. Perspectives d'évolution technologique.
- 4. Annexes.**
 - 4.1. Diagrammes de flux.**
 - 4.2. Dictionnaire de données.**
 - 4.3. Modèle conceptuel de données.**
 - 4.4. Références et contact**

3.2.2 Rappeler les objectifs

Le cahier des charges doit impérativement rappeler les *objectifs*, en effet, le document va être transmis à un certain nombre d'instances qui peuvent devenir prestataires en réalisant le travail demandé.

Le rappel des objectifs va donc, permettre aux prestataires de mieux comprendre le pourquoi de ce travail, c'est-à-dire les *finalités*.

3.2.3 Résumé de la problématique

La *problématique* doit être rappelée dans chaque communication concernant un projet. Elle est indispensable pour comprendre le problème, les solutions, les synthèses. Intellectuellement, elle rappelle à chaque fois à celui qui conçoit pourquoi il fait tous ces efforts.

Il est donc indispensable de ne jamais perdre de vue ce pour quoi on travaille. Les méthodes utilisées s'apparentent dans tous les cas à de la *gestion de problèmes*.

3.2.4 Décrire la solution détaillée

Il s'agit de décrire en détail, tous les aspects de la solution, puisqu'il faut éviter toute ambiguïté.

Ces spécifications détaillées vont permettre aux prestataires d'exploiter ces informations pour les traduire en machine, dans une phase appelée codage ou implémentation. Il est important que la solution détaillée soit complète de manière à ce que la *MOA* reste responsable de ce qu'elle veut faire, et de manière également à ce que la *MOE* puisse se concentrer sur les problèmes de développement et n'ait pas à revenir sur les problèmes de spécification.

3.2.5 Bilan

Bien que cette définition de CdCF soit normalisée, elle ne doit pas être prise comme un absolu, mais comme un outil de travail que peut être modifiée, il faut pas perdre de vue qu'aucun CdCF n'est identique à un autre car les besoins fonctionnels de chaque produit sont uniques.

4. Outils de travail

Dans un premier temps, nous avons étudié deux logiciels de traitement acoustique, l'un d'entre eux, CATT-Acoustic® très répandu parmi les professionnels de l'acoustique architecturale, est objet de notre étude pour répondre à la problématique que nous avons exposée.

Il nous semble important de tenter d'approfondir dans l'étude qui suit les différents composants se conformant à ces outils.

4.1 Le format GEO dans CATT-Acoustic®

CATT-Acoustic® dispose d'un langage propre (*le format GEO*) pour définir la géométrie et les caractéristiques acoustiques des modèles. Ce format de fichier traite les paramètres des contraintes dimensionnelles assurant le contrôle des transformations complexes d'une géométrie et la conduite d'une étude en multipliant hypothèses et scénarios.



Figure 4.1 Capture d'écran fenêtre de paramétrage général de fichiers GEO.

Cependant il peut apparaître complexe à maîtriser dans toute sa puissance.

Voici un exemple de format *GEO* pour la déclaration de la géométrie d'un modèle :

```

;MASTFULL.GEO
;PROJECT:CATT-Acoustic for v8.0

GLOBAL step = 6

;STEP 6 - side wall reflectors
INCLUDE reflect

;general materials defined here or from library
ABS wood = <12 10 8 7 5 4> {213 220 160}
ABS carpet = <10 8 6 4 3 2> {98 209 44}

;STEP 1: audience and floor surfaces
ABS audience = <35 50 55 70 85 80>
L <20 30 40 50 60 70> {128 128 0}
ABS floorabs = carpet

;STEP 2: ceiling, backwall, sidewalls
ABS ceilabs = wood
ABS wallabs = wood

;STEP 3: entrance door
ABS doorabs = wood

;STEP 4: prosc. and stage
ABS sfloorab = wood
ABS swallabs = wood

;STEP 1: floor, audience
GLOBAL ah = 1.0 ; audience height
GLOBAL rh = 0.3 ; receiver height above audience plane
GLOBAL aww = 1.0 ; audience sidewalk widths
GLOBAL fsd = 1 ; front-stage depth
GLOBAL fsw = 6 ; front-stage width
GLOBAL sw = fsw + 2 ; stage width
GLOBAL pw = 2 ; prosc. width
GLOBAL w = 16 ; max hall width
GLOBAL ew = 1.4 ; entrance width
GLOBAL fld = 3 ; flank depth
GLOBAL flt = 1 ; floor tilt
GLOBAL wl1 = 6 ; first side wall length
GLOBAL wl2 = 7 ; second side wall length
GLOBAL hd = fld + wl1 + wl2 ; hall depth from prosc

;STEP 2: ceiling, backwall, sidewalls
GLOBAL h = 8 ; hall height

;STEP 3: entrance door
GLOBAL e = 0.2 ; edge width entrance door
GLOBAL dh = 2 ; entrance door height

;STEP 4: prosc. and stage

```

GLOBAL sh = 1 ; stage height
GLOBAL sd = 5 ; stage depth
GLOBAL sbw = 4 ; stage back width
GLOBAL sfh = h-2 ; stage front height
GLOBAL sbh = sfh-1 ; stage back height
CATT-Acoustic v8.0c Help-file

Géométrie des salles et des données acoustiques réalisées sous le format Geo.

Il n'est pas étonnant que certains préfèrent une approche plus graphique de la saisie des modèles. Pour cela CATT-Acoustic® s'est doté d'une passerelle avec le logiciel AutoCad articulée autour d'un ensemble de fonctions AutoLisp assurant l'import, et l'export structuré de données. De plus, un utilitaire de conversion du format DXF à été proposé aux utilisateurs (*Dxf2Geo*), offrant le transfert avec la plupart des logiciels de CAO 3D du marché.

4.2 Méthodes de modélisation dans CATT-Acoustic®

CATT-Acoustic® intègre trois méthodes indépendantes de calcul prévisionnel intégrées dans les trois modules de calcul prévisionnel. Elles sont décrites ci-après.

4.2.1 Approche par la méthode de Lancer de Rayons

Cette méthode utilise l'approche classique par *Lancer de Rayons* avec des récepteurs sphériques.

La méthode par *Lancer de Rayons Acoustique* propage l'énergie acoustique par des rayons dont l'énergie est collectée sur une grille de récepteurs sphériques de taille constante, ces récepteurs qui sont disposés sur une grille forment un plan.

A la différence de la méthode par Lancer de Cônes Acoustiques, le rayon du récepteur ne change pas en fonction du temps de propagation.

Afin de maintenir une valeur de pression acoustique cohérente en fonction de la résolution des cellules de réception et du temps de parcours

pris en compte, le modèle ajuste la puissance acoustique assignée à chaque rayon pour compenser l'erreur introduite par une densité de rayons insuffisante.

4.2.2 Méthode de Réflexions Précoces Détaillées ISM

La méthode *ISM* (Image Source Model) peut déterminer toutes les réflexions spéculaires entre un point source et un point récepteur. Elle calcule les sources images de premier ordre qui déterminent les sources images de deuxième ordre et ainsi de suite jusqu'à l'ordre maximal souhaité où que le temps de parcours qui ont été fixés soient dépassés.

La diffusion est prise en compte en décomposant les parois en maille rectangulaires qui réémettent l'énergie acoustique répartie comme autant de sources élémentaires selon *la loi de Lambert*.

Cette répartition de l'énergie acoustique diffuse est limitée aux réflexions d'ordre 1. Pour des ordres de réflexion supérieurs, la composante diffuse est négligée (*hypothèse de réflexion spéculaire*)

4.2.3 Méthode de Calcule Détaillée RTC

Cette méthode de calcul dénommée *Randomized Tailcorrected Cone-tracing* (acronyme RTC) a fait l'objet d'amples développements dans la thèse de doctorat Bengt- Inge Dalenbäck soutenue à Chalmers University.

La méthode RTC considère que la densité de réflexions croit de manière quadratique. Pour les salles ouvertes, cette hypothèse n'est pas prise en compte. Elle prend en compte les réflexions diffuses pour l'ordre de réflexions égal à 1 en créant un nombre de sources élémentaires sur chaque surface diffusante. Pour les ordres de réflexion supérieurs à 1, l'approche est similaire à la méthode par lancer de rayons. La dépendance en fonction des fréquences du coefficient de diffusion entraîne le tirage d'un nombre de cônes multiplié par le nombre de bandes fréquences étudiées.

4.2.4. Gestion de transparence Transmission au travers d'un élément avec atténuation

Chacune de ces trois méthodes prend en compte le cas des surfaces *semi-transparentes* de manière légèrement différente. L'approche générale s'organise comme suit :

Le son direct peut transiter selon une solution déterministe au travers d'un maximum d'une surface semi-transparente en subissant une atténuation de $(1-a) t$. Les ordres de réflexion supérieurs sont gérés selon processus aléatoire. Les réflexions de premiers ordres sont atténuées de $(1-a) (1-t)$.

La somme des trois composantes doit évaluer 1 afin de maintenir la conservation de l'énergie :

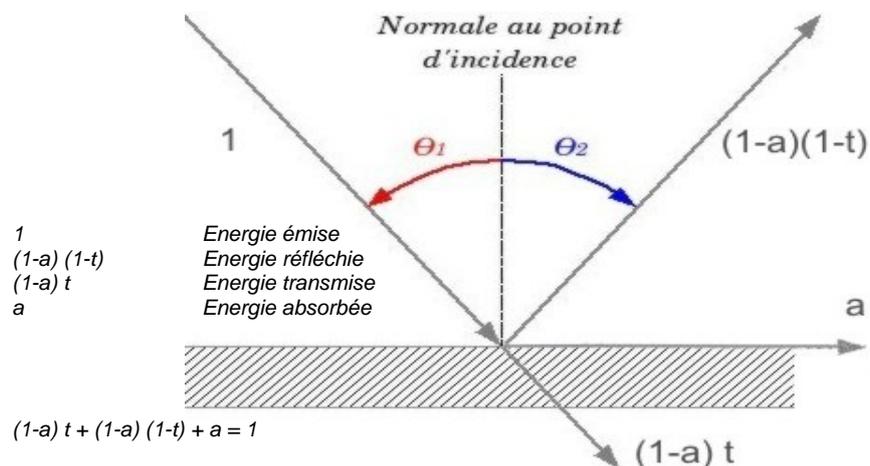


Figure 4.2 Transmission du son au travers d'un élément avec atténuation.

4.3 Une nouvelle interface de saisie pour CATT-Acoustic®

À la suite de différentes analyses nous avons pu faire émerger des pistes pour répondre à notre problématique concernant le transfert des modèles géométriques 3D définis et ses propriétés vers un logiciel de traitement acoustique architecturale.

Ces pistes seront développées à travers un prototype d'outil, un module de conversion du format SketchUp® .skp vers le format CATT-Acoustic® .geo et sa description.

4.3.1 Pourquoi utiliser l'interface SketchUp® ?

Grâce à sa simplicité et son efficacité l'interface avec le logiciel *SketchUp®* augmente le confort de saisie des modèles 3D, grâce à de nombreuses aides au dessin 3D par le biais d'un moteur invisible qui localise, ou infère, des points à partir d'autres points du modèle, comme par exemple, le centre d'un cercle, le milieu d'une ligne, une ligne perpendiculaire au plan du sol, un point sur une surface, un point sur une arête, entre autres.

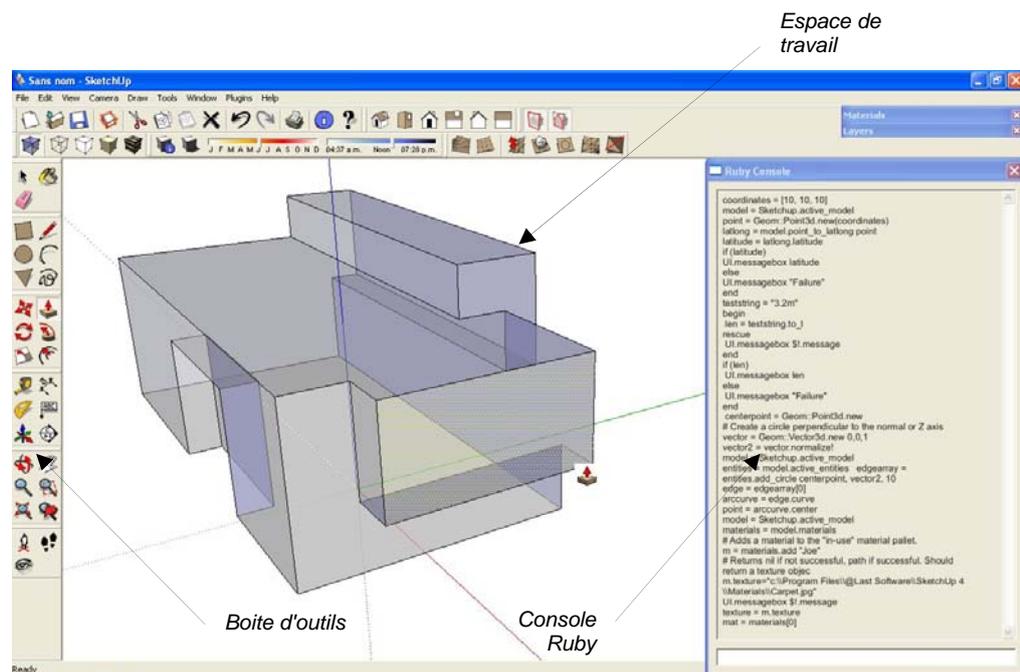


Figure 4.3 Capture d'écran interface d'utilisateur SketchUp®.

L'interface de ce logiciel reste d'une grande aide au regard de poids lourds de la CAO 3D ce qui facilite sa prise en main immédiate. Doté de fonctions de transformations géométriques puissantes et intuitives, SketchUp® fait de la gestion des entités 3D une tâche très souple et rapide. De plus, les entités 3D de SketchUp® se conforment au modèle surfacique (décomposition en facettes co-planaires) exigé par CATT-Acoustic®.

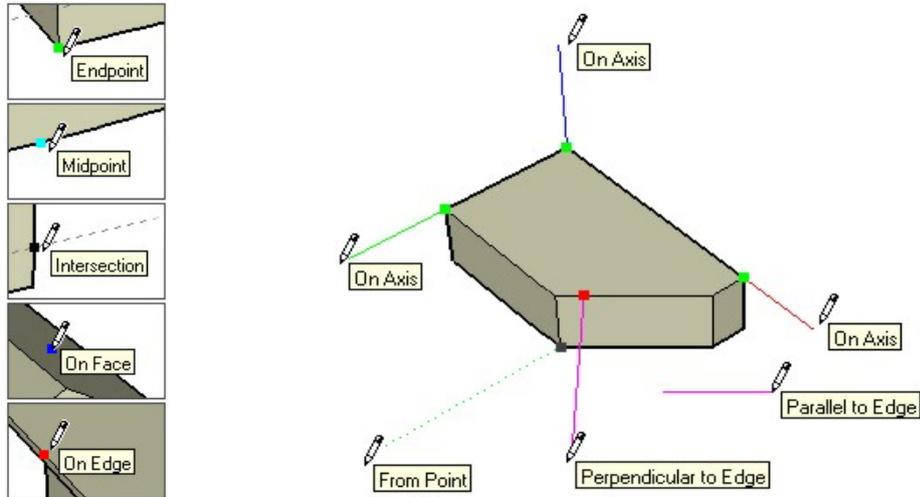


Figure 4.5 Moteur d'inférences ponctuelles et linéaire de SketchUp®.

4.3.2. Langage Script propre de SketchUp®

L'interfaçage avec le logiciel SketchUp® s'appuie sur le langage de script *Ruby*. Il contient un Interfaçage de Programmation d'Applications *API* (Application Programming Interface).

Cette interface permet aux utilisateurs de créer d'outils, icônes et menus, ainsi que d'autres macros comme des générateurs automatiques de composants, pour être inclus aux menus de SketchUp®.

En plus de l'API il est inclus dans SketchUp® une *Console Ruby* qui est un milieu où nous pouvons mettre en pratique les commandes et outils créés sur Ruby.

4.4 Bilan

À travers les analyses précédentes nous avons constaté la problématique que représente, par sa complexité, le paramétrage des données géométriques sous le format *GEO* de CATT-Acoustic®.

En conséquence nous avons établi la nécessité de développer un module d'échange pour pouvoir paramétrer d'une manière plus convenable la géométrie 3D des modèles. A cet effet nous avons choisi le logiciel SketchUp® dont plusieurs avantages ont été mentionné, notamment sa simplicité et flexibilité.

Il ne faut pas perdre de vue qu'outre les objets géométriques du modèle, les entités à manipuler sont les propriétés acoustiques de parois, les sources acoustiques, les récepteurs, entre autres.

5. Son et Lumière

Jusqu'à maintenant nous nous sommes attachés à l'analyse de différents logiciels dont nous aurons besoin pour répondre à notre problématique. Nous pensons convenable de faire à ce stade du travail l'étude comparative des similitudes entre le son et la lumière.

La finalité de cette analyse sera :

- Dans un premier temps, d'identifier les nombreuses similitudes qui présente la propagation du son et la propagation de la lumière.
- Dans un second temps, une fois établie cette similitude, de se baser sur l'existant en matière de logiciels d'illumination pour éventuellement envisager une structure de données commune.

5.1 L'algorithme Ray Tracing

Le *Lancer de Rayon* (*Ray Tracing* en anglais) est un algorithme de synthèse d'images tridimensionnelles. Basé sur *l'algorithme de détermination de surfaces* développé par Arthur Appel nommé *Ray Casting* (1968), il simule le parcours inverse de la lumière de la scène vers l'œil.

5.1.1 Principe du Lancer de Rayon

Le principe de la vision est le suivant : Les sources de lumière (lampes, soleil) émettent des rayons lumineux. Ces rayons viennent frapper les objets qui en réfléchissent une partie, en fonction de leurs propriétés physiques (matériaux). Parmi tous les rayons lumineux, une partie va parvenir, directement ou après réflexion sur les objets, à l'œil et former une image sur la rétine.

Il serait illusoire de vouloir appliquer ce principe à l'informatique. Une infime partie de la lumière émise parvient à l'œil de l'utilisateur. La plupart de la lumière se perd dans la nature.

La technique du lancer de rayon tente de reconstituer le parcours inverse de la lumière, en partant de l'œil en allant vers les sources. L'expérience montre en effet que cette manière de procéder est nettement plus performante. [2]

Pour cela, on place une image virtuelle dans la scène devant l'observateur. Pour chacun des pixels qui constituent l'image, nous lançons un rayon partant du point de vue passant par le pixel, le premier point d'impact du rayon sur un objet définit l'objet concerné par le pixel correspondant (figure 5.1). Des rayons sont ensuite lancés depuis le point d'impact en direction de chaque source de lumière pour déterminer cela. Cette luminosité avec la couleur de l'objet ainsi que d'autres informations éventuelles (angles entre la normale à l'objet et les sources de lumières, réflexions, transparence, etc.) déterminent la couleur finale du pixel.

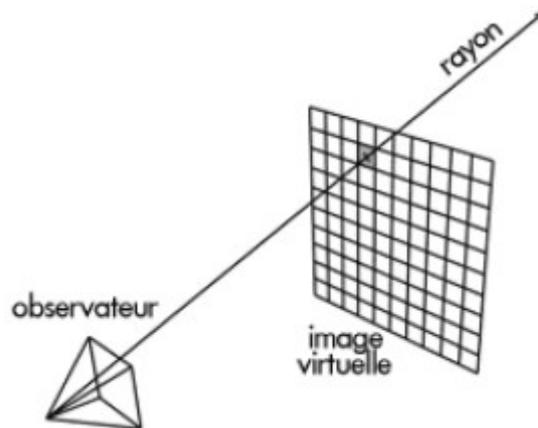


Figure 5.1 Principe du Lancer de Rayon.

5.1.2 Cheminement du rayon

Pour chaque rayon lancé de l'œil, nous calculons le premier objet qu'il intersecte, ce qui implique un parcours de l'ensemble des faces de la scène. Une fois le premier objet trouvé, il faut déterminer la couleur du pixel à afficher. Cette couleur dépend du matériau de la face, et bien sûr de l'éclairage. Aussi faut-il déterminer quelles sont les sources lumineuses qui éclairent le point. Pour cela, nous lançons des rayons issus du point considéré vers chacune des sources lumineuses de la scène. Si ce rayon parvient à la source sans intersecter d'autres objets, alors la source contribue à l'éclairage du point. Sinon, un objet masque la source et un effet d'ombre est produit.

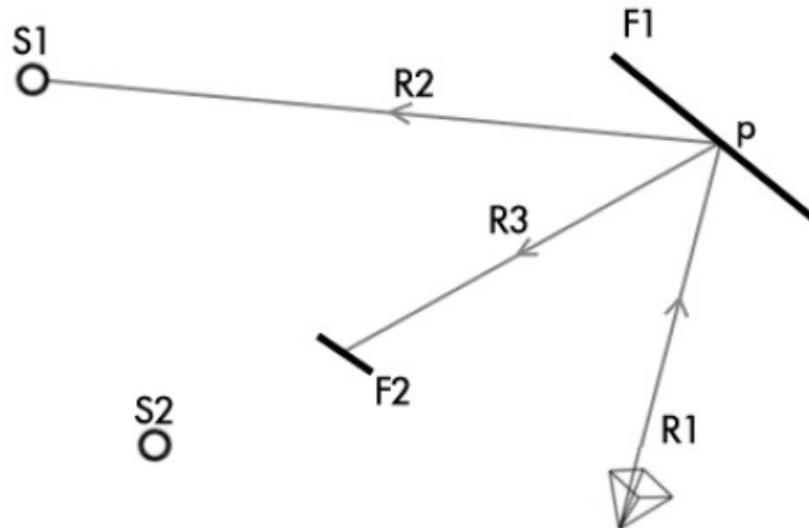


Figure 5.2 Cheminement du rayon.

Le rayon $R1$ provient de l'œil, disons pour le pixel (x, y) . Il heurte la face $F1$ au point p pour déterminer l'éclairage du pixel (x, y) . Ensuite les rayons lancés de p montrent que la source $S1$ va contribuer à l'éclairage de p , mais que la source $S2$ est masquée par $F2$, donc la face $F2$ va porter son ombre sur $F1$.

Dans le cas où le rayon lancé depuis le point de vue n'intersecte aucun objet, le pixel (x, y) considéré prend une valeur dite *de fond*, qui peut être soit une couleur fixe, soit la couleur du pixel (x, y) d'une image.

5.1.3 Réflexion

Supposons à présent qu'on souhaite gérer les effets de réflexion. Lorsqu'un rayon issu du point de vue vient frapper une face définie comme réfléchissante, nous déterminons la couleur du point en lançant un rayon vers les sources, mais également, en lançant un rayon *spéculaire* qui va permettre de prendre en compte la réflexion. La direction du rayon spéculaire est dictée par *les lois de Snell-Descartes*¹ concernant la réflexion.

Le rayon incident $R1$ atteint la surface réfléchissante F au point p , alors un rayon réfléchi $R2$ (rayon spéculaire) est projeté, de façon que l'angle θ_1 formé entre $R1$ et *la normale* au point d'incidence p soit égale à l'angle θ_2 formé entre $R2$ et cette même normale.

Note 1. **Les lois de Snell-Descartes** décrivent le comportement de la lumière à l'interface de deux milieux. Ces lois sont au nombre de deux, une pour la **réflexion** et une autre pour la **réfraction**. Ces lois sont à la base de *l'optique géométrique*.

La face frappé par $R2$ peut elle aussi être réfléchissante, auquel cas on réitère le processus de réflexion en lançant un rayon spéculaire supplémentaire. L'implémentation de ce principe en machine est facile en utilisant des algorithmes récursifs.

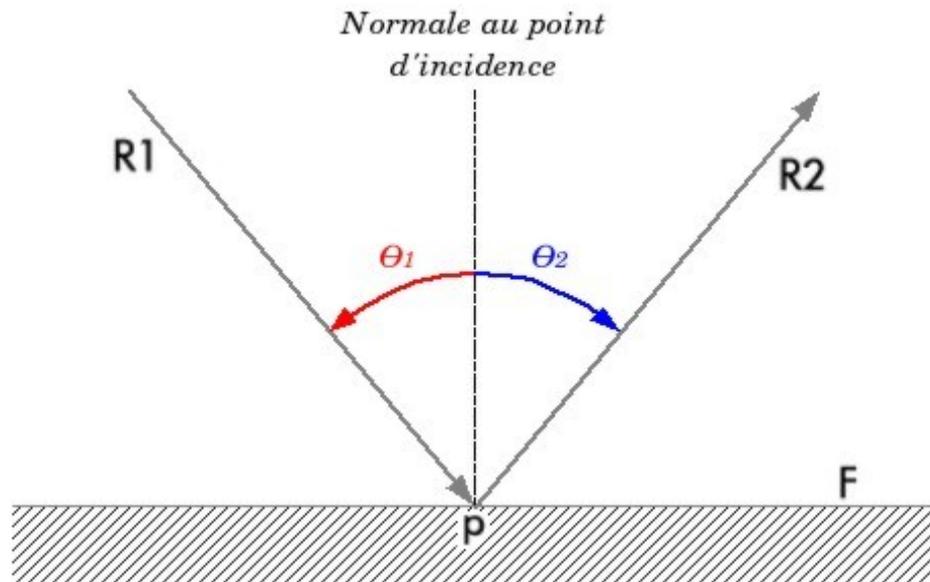


Figure 5.3 Réflexion d'un rayon.

La couleur qui sera finalement attribuée au pixel (x, y) est un mélange de la lumière directement incidente sur la face et de la couleur de réflexion obtenue par le calculé expliqué ci-dessus. Ce mélange dépend évidemment des propriétés de réflexion des objets.

5.1.4 Transparence et réfraction

Pour la transparence, le principe est le même que pour la réflexion, nous déterminons la couleur du point en lançant un rayon vers les sources, en lançant un rayon spéculaire, mais aussi en lançant un rayon supplémentaire, dit rayon *transmis*. La valeur finale de la couleur du pixel est obtenue en mélangeant les couleurs calculées à partir de l'éclairage direct, de la réflexion et de la transparence, en fonction des propriétés du matériau.

Le rayon incident $R1$ atteint la surface transparente F au point p , outre le rayon spéculaire un rayon transmis T est projeté vers les sources lumineuses.

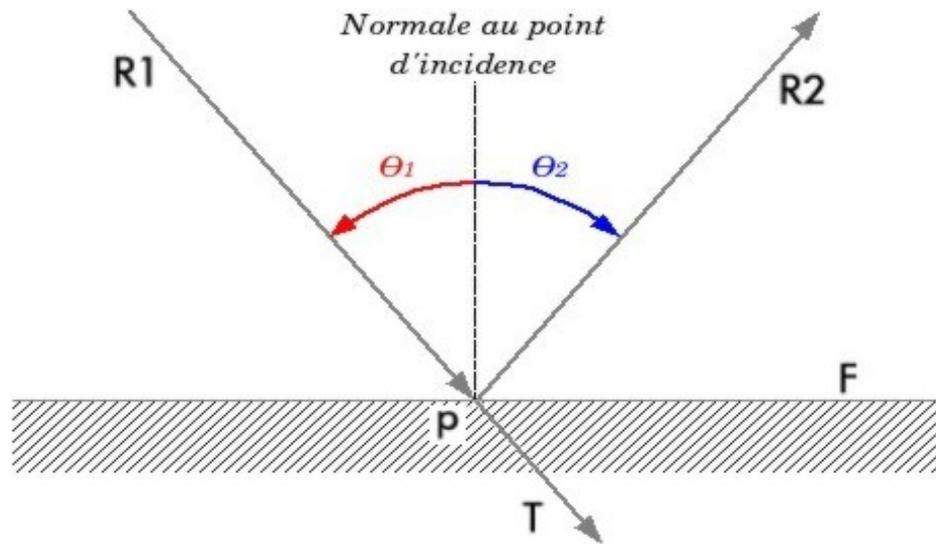


Figure 5.4 Rayon transmis.

Le lancer de rayon permet également pour les objets transparents de reproduire le phénomène de réfraction. L'exemple le plus classique de réfraction est la déformation des objets vus à travers un verre ou une bouteille. La figure 5.5 illustre le cheminement d'un rayon lumineux transmis lors de la traversée d'un matériau transparent tel que le verre.

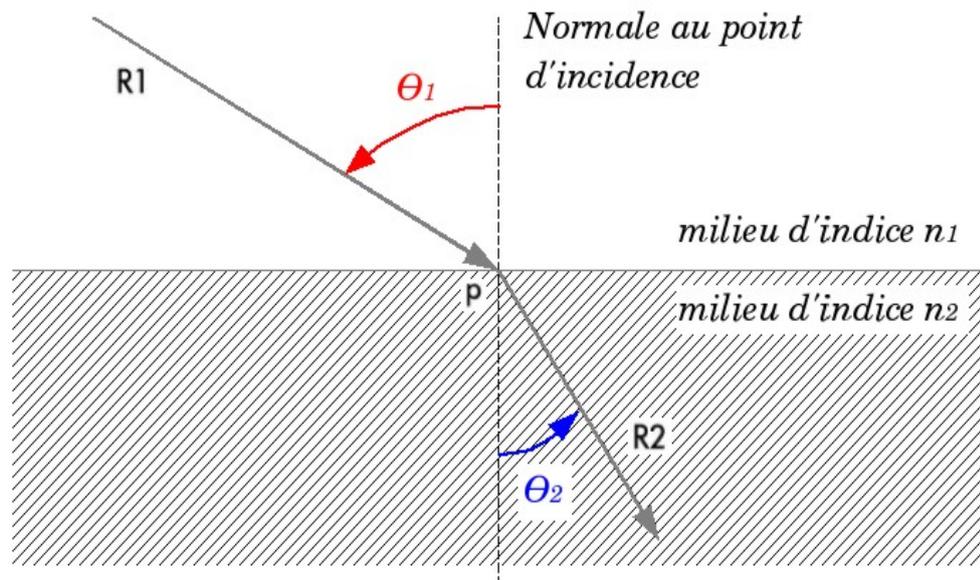


Figure 5.4 Le phénomène de réfraction.

5.2 Algorithmes

Ensuite nous montrons l'algorithme simplifié du programme *Lancer de Rayon* :

```

Programme lancerDeRayon
mémoriser une scène s
pour chaque pixel  $p(x,y)$  de l'écran
    calculer les composantes
    géométriques du rayon  $r$ 
    passant par l'observateur et  $p$ 
    teinte[ $p(x,y)$ ] <- lancerRayon( $s,r$ )
finpour
finprog

couleur lancerRayon(scène  $s$ , rayon  $r$ )
couleur  $cr, ct, cd$ 
rayon  $rr, rt, rd$ 
point  $p$ 
début
déterminer l'objet  $o$  de la scène  $s$ 
dont l'intersection avec  $r$  est
la plus proche de la source de  $r$ 
si  $o$  n'existe pas
    lancerRayon <- noir
sinon
    calculer le point d'intersection  $p$ 
     $rr$  <- rayon réfléchi de  $r$  en  $p$ 
    vis à vis de  $o$ 
     $rt$  <- rayon transmis de  $r$  en  $p$ 
    vis à vis de  $o$ 
     $cr$  <- lancerRayon( $s, rr$ )
     $ct$  <- lancerRayon( $s, rt$ )
     $cd$  <- noir

pour chaque lumière  $sl$  de  $s$ 
    calculer le rayon  $rd$  de  $p$  vers  $sl$ 
    si  $rd$  non intercepté par un objet
     $cd$  <-  $cd$  + diffusion( $sl, o, rd$ )
    finsi
finpour
lancerRayon <-  $cr$  +  $cd$  +  $ct$ 
fin_fonction

```

Algorithme de Lancer de Rayon

5.3 Traitement du son et de la lumière

Nous avons constaté, au travers l'étude en matière de logiciels de traitement acoustique, l'existence d'un parallélisme fonctionnel entre la propagation du son et celle de la lumière².

Si nous prenons comme principe que le son se propage de façon similaire à la lumière, nous pourrions donc parler d'une propagation sous la forme de *rayons acoustiques*, par conséquent, ces derniers pourraient être traités comme des rayons lumineux. Ainsi en appliquant les lois de *Snell-Descartes*¹ concernant *la propagation, la réflexion, la réfraction, la transmission et l'absorption de la lumière*, et en se servant d'un *algorithme de détermination de surfaces visibles*, nous pourrions obtenir des simulations acoustiques à partir de la méthode du lancer de rayons.

Pour finir nous allons énumérer plusieurs éléments communs entre le traitement acoustique et celle de la lumière qu'au long des analyses nous avons constaté :

1. Source.
 - Monodirectionnelle.
 - Bidirectionnelle.
 - Omnidirectionnelle.
2. Récepteurs.
3. Rayonnement.
 - Puissance.
 - Intensité.
 - Densité.
4. Propagation d'énergie.
 - Diffusion.
 - Absorption.
 - Réémission.
 - Dissipation.
 - Réfraction.
 - Réflexion.

Note 2. Voir chapitre 4.2 concernant les *méthodes de modélisation dans CATT-Acoustic*.

- Transmission.
5. Surface de calcul.
 - Hauteur de surface.
 - Matériaux.
 - Textures.
 - Indices de réflexion, réfraction et transmission.
 6. Ambiances d'environnement.

5.4 Bilan

Suite à nos analyses nous avons déterminé que le traitement du son et de la lumière présente de nombreuses similitudes. Notamment les méthodes de modélisation qui gèrent le traitement acoustique dans le logiciel CATT-Acoustic®, qui sont basés sur l'utilisation d'algorithmes de détermination de surfaces visibles.

Après avoir constaté l'existence de plusieurs points communs entre eux, nous pouvons en conclure que l'utilisation des bases de données en matière de traitement lumineux qui servira éventuellement aux paramètres du module de transfert qu'on cherche définir est envisageable.

Outre l'analyse des similitudes entre le son et la lumière, il ne faut pas perdre de vue la gestion du transfert de données géométriques et non géométriques d'un logiciel à l'autre.

6. Transfert de données

Outre le paramétrage de la géométrie 3D et des données acoustiques du modèle, il s'agit maintenant d'étudier le transfert de ces données entre le modeleur et le logiciel de traitement acoustique. Nous avons visé l'utilisation des *IFC* comme structure de données sur laquelle se basera la description du transfert. Puisque ces dernières contiennent la définition de *types d'objets* (*classes*) nécessaires pour la définition et l'éventuel traitement de la géométrie 3D, les caractéristiques des matériaux, l'environnement spatial, entre autres.

6.1 International Alliance for Interoperability IAI

L'IAI a été créé en 1995 avec la finalité principale de développer des nouveaux concepts innovants pour améliorer l'échange d'informations tout au long du cycle de vie d'un projet de construction. Elle s'est fixée comme objectif d'améliorer l'interopérabilité des logiciels utilisés dans le secteur de la construction.

L'IAI ne produit pas des logiciels mais des spécifications destinées à faciliter l'échange et le partage d'informations entre logiciels. Le principal résultat des travaux de l'Alliance est un langage qui rassemble aujourd'hui plus de 600 *classes d'objets* et a pour nom *IFC Industry Foundation Clases* pour ses sigles en anglais. La *plate-forme IFC* a obtenu l'homologation de l'ISO sous la référence ISO/PAS 16739 en novembre 2002 (*Industry Foundation Classes, Release 2x, Platform Specification (IFC2x Platform)*).

6.1.1 Qu'est-ce que les IFC ?

L'IAI a défini les IFC comme le groupe d'éléments qui servent comme langage commun sur lequel sont basés les échanges et le partage d'informations entre logiciels.

L'intention de l'IAI est de spécifier comment les *choses* qui se présentent dans une construction (aussi bien les composants tangibles comme les portes, les murs, les ventilateurs, etc.. que les concepts abstraits

comme les espaces, l'organisation, les processus, etc.) peuvent être représentés sous forme électronique dans un ordinateur.

Ces spécifications représentent une structure de données sur laquelle s'appuie la description électronique du projet utile au partage de données entre applications.

Chaque spécification est appelée *classe*. Le mot classe est employé pour décrire une variété d'éléments qui ont des caractéristiques communes. Par exemple, chaque porte a la caractéristique de s'ouvrir pour permettre l'accès à un espace, chaque fenêtre a une caractéristique de transparence. Les portes et les fenêtres sont des noms de classe.

Le modèle que constitue les IFC est riche et complexe, à l'image de la réalité que l'on veut modéliser. Il est avant tout destiné aux développeurs de logiciels et les utilisateurs ignoreront tout de cette complexité ; pour communiquer en IFC, il leur suffira de choisir le format *IFC* lors des opérations de type *Enregistrer sous* ou *Exporter*.

6.1.2 Objets IFC

Les IFC sont des classes d'objets, près de 600 dans la dernière révision. Ces classes permettent de décrire des objets tangibles comme un plancher ou une poutre, mais aussi des objets plus abstraits comme les espaces ou les organisations ainsi que les nombreuses relations entre objets.

Certaines IFC comme celles qui traitent de la géométrie ont une portée très générale. D'autres sont spécifiques à un domaine particulier. Aujourd'hui, les IFC traitent essentiellement du bâtiment, tout au long du cycle de vie (programmation, conception, construction, gestion) et selon différents points de vue (architecture, structure, thermique, acoustique, estimatif, exploitation, etc.).

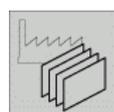
6.1.3 Arborecence des principales classes d'objet du modèle IFC

Il est à noter que dû à la complexité, nous ne sommes pas en mesure de décrire dans sa totalité le modèle de classes d'objet IFC qui en compte plus de 600. Pour cela nous en présentons seulement les principales.

6.1.3.1 Projet

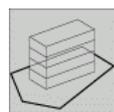
Objet de projet Classe abstraite qui définit tout objet physique, virtuel ou concept abstrait relatif au projet. Déjà ce stade général, un certain nombre d'attributs peuvent être consignés qui qualifient tous les objets de l'arbre d'héritage, mais qui sont l'affaire des développeurs de logiciels. L'utilisateur doit renseigner un seul attribut : la liste des documents externes attachés au projet (par exemple les documents contractuels).

Le projet Permet d'indiquer des informations générales concernant l'opération par les attributs suivants : nom court, nom complet, phase, liste des unités de mesure qui sont utilisées dans le projet, classification utilisée, localisation du projet (qui peut contenir plusieurs bâtiment) dans le système de coordonnées mondial.

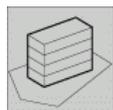


Le produit Classe abstraite pour représenter tout produit manufacturé, découpé ou fabriqué in situ, ou encore résultat de frontières matérielles comme les espaces (local). Attributs : localisation, représentation géométrique ou forme associée, liste de spécifications

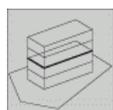
6.1.3.2 Eléments spatiaux



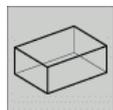
Site Terrain défini (parcelles cadastrales par exemple) sur lequel le bâtiment(s) est construit. Peut inclure une représentation géométrique du terrain (courbes de niveaux) et des équipements extérieurs (parking, VRD, végétation...) Le site peut être le résultat d'un ensemble de sites. Attributs : point de référence définie en latitude, longitude et altitude. Définition géométrique du périmètre du site. Surface.



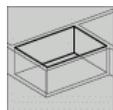
Bâtiment Entité construite indépendante structurellement ou selon l'organisation du projet. Attributs peu nombreux : nom court et long, emprise au sol, volume, hauteur totale, altitude absolue (par rapport au niveau de la mer), altitudes des emprises au sol, services connectés ; Peut appartenir à un groupe de bâtiments.



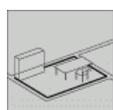
Etage Partie du bâtiment associée à un plan horizontal (même si l'étage comporte des duplex). Attributs : admet une liste similaire de propriétés en plus de celles du Produit : noms courts et longs, altitude de référence, hauteur, surface, volume, description géométrique. Peut éventuellement être associé à une liste de local.



Élément spatial Concept abstrait pour définir un élément spatial composé soit d'un espace mesurable (possédant une surface et un volume, c'est à dire un local), soit des limites (frontières) de l'espace (c'est à dire la face visible des murs, planchers et sous faces de plafonds du local, ou nu de local). Ces deux types d'éléments doivent obligatoirement admettre une représentation géométrique. Pas d'attributs à ce niveau.

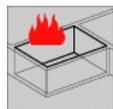


Local En français, le terme local est plus précis que celui d'espace et correspond à la définition sémantique de "IfcSpace". Le local est entièrement délimité par des composants réels (murs, planchers, plafonds) et/ou des frontières virtuelles. Les IFC définissent toute une série de hauteurs et d'altitudes pour un local vu en coupe verticale (voir paragraphe « Le concept de local et son environnement dans les IFC). Attributs : liste des nu de local composant les frontières matérielles ou virtuelles du local (voir plus loin la définition du nu de local). Type d'espace : intérieur ou extérieur (dans ce cas, les frontières virtuelles peuvent être majoritaires), nom court (de référence), nom complet (long), périmètre, surface, volume, hauteur brute sous plafond (revêtement de sol et de plafond, même avec faux plafonds, ignorés). Hauteur d'étage (de sol brut du plancher bas au sol brut du plancher haut), dans les deux cas, une moyenne est admise si les locaux ne sont pas prismatiques, hauteur nette sous plafond (entre les nus finis du sol et du plafond), altitude du nu fini du plancher, liste éventuelle des locaux élémentaires qui composent le présent local, local du niveau supérieur auquel appartient le présent espace, et liste des locaux du même groupe.

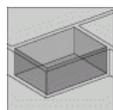


Poste de travail Sorte de local pas obligatoirement clos physiquement abritant un bureau incluant le mobilier et ses

accessoires de travail. Attributs : Type de poste (système global ou détaillé par élément), liste des éléments de catalogue de fabricant, surfaces décomposées en surface totale, emprise du mobilier, emprise des circulations, indication d'appartenance à un groupe de postes.

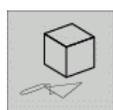


Zone protection contre incendie Réunion des locaux vus pour les dispositifs de protection contre l'incendie, permettant de consigner les listes d'équipements... Attributs : Altitude des bornes d'incendie par rapport aux surfaces de plancher à desservir, classification de résistance au feu des surfaces, indice de risque au feu, caractéristique de la ventilation naturelle, existence de sprinkler (aspersion d'eau).

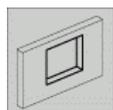


Nu de local Traduction la plus appropriée de « IfcSpaceBoundary ». Définition sémantique des IFC : morceau de frontière, surface de délimitation du local en regard de chaque local adjacent, quelque soit son niveau. Cette définition est lourde de conséquences (voir paragraphe « Le concept de local et son environnement dans les IFC). Attributs : Type de nu (physique ou virtuel, contre l'extérieur ou adjacent à un local extérieur), surface du nu, référence aux deux locaux séparés par le nu, dans le cas où un local adjacent existe (on peut considérer l'extérieur comme une sorte de local).

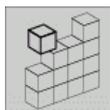
6.1.3.3 Les éléments de construction



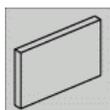
Élément Tout composant matériel du bâtiment, plein ou vide comme une ouverture, à l'exclusion des espaces et local. Attributs : caractérisé par les attributs du produit, avec en plus les relations de voisinage : connexions topologiques ou de contact, appartenances, ces relations pouvant être définies par l'utilisateur selon les performances du logiciel.



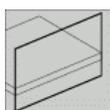
Ouverture Vide créé dans un composant, de forme simple ou complexe. Attributs : Surface (frontale), relation avec tout objet opératoire qui a servi à créer l'ouverture, relation(s) avec les éléments de remplissage s'ils existent. Dans les IFC, il ne peut exister de menuiseries sans ouverture (sauf pour le mobilier modulaire ou non).



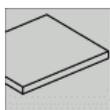
Composant Le composant (plein) est la classe d'objets la plus riche du modèle IFC. Attributs : en plus des propriétés héritées, il admet tout ce qui caractérise les matériaux utilisés pour le réaliser : matériau principal ou d'ossature, couches successives avec épaisseurs, relations avec les ouvertures, et les équipements d'ouvertures, fonction ou non de frontière (nu de local).



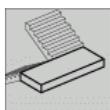
Mur Tout type de mur, toutes fonctions, sauf les cloison visuelle et façade rideaux. Sa géométrie peut être simple (rectangulaire) ou plus complexe (mur à rive supérieure en ligne brisée). Attributs : information des couches du mur (épaisseurs, constituants hormis les revêtements, matériaux), surface, volume, et relations avec les autres murs suivants et précédents. L'utilisateur ne décrit pas directement les relations entre murs, mais doit donner des priorités relatives entre types de murs (pour permettre aux logiciels de redessiner correctement les jonctions).



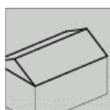
Mur rideau Mur léger de façade s'appuyant exclusivement sur les nez de planchers. Attributs : non définis à ce niveau, mais à travers le parent « Composant ».



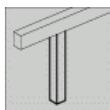
Dalle Structure de plancher ou de toiture horizontale ou inclinée. Attributs : liste de propriétés, épaisseur et informations sur les couches éventuelles, surface, volume, et références aux matériaux. Types définis : dalle de plancher, dalle de toiture, définie par l'utilisateur, non définie.



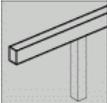
Palier Intermédiaire ou au même niveau qu'un plancher d'étage. Attributs : Hauteur de passage (coup de tête), largeur, longueur.



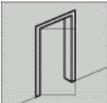
Toiture Décrite dans son ensemble (pour plus de détail, voir dalles de toiture). Seul attribut : surface totale. (Les couches d'étanchéité, de tuiles, d'ardoise... sont considérées comme des revêtements.

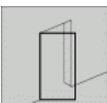


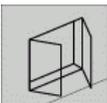
Poteau Attributs : section, volume. Admet une représentation standard (section constante, verticale), ou une représentation *avancée* : section variable, formes inclinées.

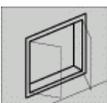
 **Poutre** Attributs : section, volume. Admet une représentation standard (section constante, horizontale), ou une représentation *avancée* : section variable, formes inclinées.

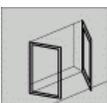
 **Porte** Menuiserie qui assure le passage. Voir les composants constituants.

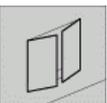
 **Cadre de porte** Ou dormant. Pas d'attributs à ce niveau

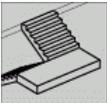
 **Battant de porte** Ou ouvrant. Attributs : liste de propriétés. Types définis : porte pivotante (normale), coulissante, à tambour, défini par l'utilisateur, non défini

 **Fenêtre** Pour les IFC, les menuiseries sont des porte ou des fenêtre. La subtilité française du composant porte-fenêtre n'existe pas. Cet objet regroupe aussi les fenêtres incorporées aux toits : velux et lanterneaux. Une fenêtre IFC laisse obligatoirement passer la lumière naturelle, mais ne s'ouvre pas forcément. Les panneaux vitrés fixes sont également des fenêtres IFC. Pas d'attributs à ce niveau. Voir les composants constituants.

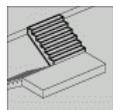
 **Cadre de fenêtre** Ou dormant. Pas d'attributs à ce niveau.

 **Panneau vitré de fenêtre** Attributs : liste de propriétés, et référence aux codifications nationales standard. Types définis : fenêtre fixe, pivotante, coulissante, définie par l'utilisateur, non définie.

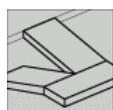
 **Protection d'ouverture** Dispositif recouvrant une ouverture mais non étanche à l'eau et à l'air, comme des volets. Attributs : liste de propriétés. Types définis : grilles, volets et stores, cloisons visuelles (de bureaux paysagers...), défini par l'utilisateur, non défini.

 **Escalier** Décrit dans son ensemble, entre deux étages. Attributs : liste de propriétés, liste des étages desservis. Types définis : Escalier

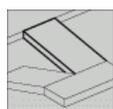
d'incendie, escalier ornemental, escalier standard, défini par l'utilisateur, non défini.



Volée d'escalier Comprise entre deux palier. Reçoit la description des paramètres et attributs de détails : référence aux deux niveaux desservis, matériaux du plat de marche, matériau du nez de marche, hauteur de marche, profondeur de marche, hauteur de passage (coup de tête), hauteur totale à monter, longueur totale de la volée.



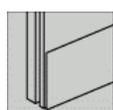
Rampe Accès incliné pour voiture et piétons, fauteuils roulants, représenté globalement dans cette classe. Pour plus de précision, voir les composants constituants. Attributs : liste de propriétés, liste des niveaux desservis. Types définis (relatifs au mode de construction) : conception par élément, bâti sur le sol, solide dans l'espace.



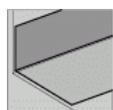
Volée de rampe Tronçon de rampe entre deux niveaux desservis. Attributs : Longueur, largeur, hauteur de montée, pente



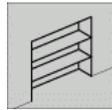
Garde-corps Attributs : liste de propriétés, liste d'accessoires ou de relations (support). Types définis : Main courante, garde-corps, balustrade (en maçonnerie), défini par l'utilisateur, non défini.



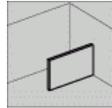
Élément composite Associé à un objet support. Sera plus détaillé dans les révisions ultérieures des IFC. Attributs provisoires : Liste d'éléments inclus, objet support. Types définis : isolation, défini par l'utilisateur, non défini.



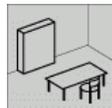
Revêtement Tout objet formant une couche qui n'existe qu'en relation avec un objet support. Revêtement au sens large : aussi bien une finition par peinture, qu'un plaquage, un parquet ou un faux-plafond. Attributs : relation avec l'objet support, dimensions propres (épaisseurs, longueur, largeur, ou bien frontières identiques à celles du support), surface, matériaux. Types définis : revêtement de plafond, revêtement de sol, revêtement de mur, enduits, défini par l'utilisateur, non défini



Meuble incorporé Equipements assemblé in situ et restant incorporé au bâtiment, comme les rangement, penderies, classeurs à dossiers, et leurs rayonnage, tablettes, étagères...



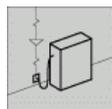
Cloison visuelle Cloison qui n'a d'autre fonction que de réaliser une barrière visuelle (par exemple séparer les postes de travail dans un bureau paysager). Pour une cloison réalisant les autres fonctions (phonique, étanchéité à l'air..., voir le composant *Mur*). Attributs : liste de propriétés. Types définis : Ossature d'assemblage, portes de cloisons, montants de cloisons, panneaux de cloisons, séparation de toilettes, portes de toilettes, définie par l'utilisateur, non défini.



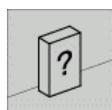
Mobilier Renseigné pour les activités de gestion de local. Attributs : liste de propriétés, référence à l'occupant (acteur, département, organisation), mobilier type. Type définis : table, siège, bureau, poste de travail. Voir aussi mobilier modulaire.



Mobilier modulaire Mobilier composé d'éléments assemblés à partir d'un système modulaire. Attributs : liste de propriétés, liste des éléments composants. Types définis : Panneaux (de cloisons), surface de travail, meubles de rangement. (Pour le mobilier non modulaire, voir *Mobilier*).

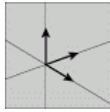


Mobilier électrique Attributs : liste de propriétés. Types définis : ordinateur, photocopieuse, fax, imprimante, téléphone, défini par l'utilisateur, non défini

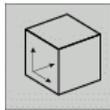


Equipement Classe IFC utilisée pour décrire un équipement n'appartenant pas à un système de distribution (ce terme est ambigu par rapport au sens plus général donné en français. Nous avons ajouté le qualificatif *annexe*). Attributs : liste de propriétés. Types définis (un seul dans la révision 2) : système de nettoyage des fenêtres et façades rideaux, défini par l'utilisateur, non défini.

6.1.3.4 Environnement de modélisation



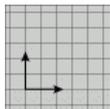
Localisation Tout ce qui permet de localiser un objet du projet dans l'espace en trois dimensions, dans un repère absolu ou relatif, ou par rapport à un autre objet. Cette classe et ses sous-classes n'appartiennent pas aux objets du projet. Ce sont seulement des supports de description. Pas d'attributs à ce niveau générique.



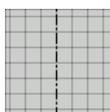
Localisation relative Localisation de tout objet physique, par rapport à un autre objet physique. Certaines règles hiérarchiques existent, mais l'utilisateur n'a pas à s'en préoccuper ; c'est le logiciel qui gère ces contraintes. Par exemple le site est localisé par rapport au projet, les bâtiments par rapport au site, les étages par rapport au bâtiment, un élément peut être localisé par rapport à son objet contenant ou par rapport aux autres éléments avec lesquels il est en relation, ou encore par rapport à la grille (de conception)... etc. Attributs : relation entre les deux objets localisés, coordonnées relatives.



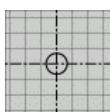
Localisation obligée S'applique pour l'instant seulement à l'intersection de n objets : localisation obéissant à des contraintes. Attributs : liste des objets vérifiant les contraintes.



Grille Système général ou particulier de repérage de localisation en trois dimensions des niveaux, et pour chaque niveau de grille, celle des axes de grille et des intersections d'axes. Attributs : objet de la grille (structure, montage de composants, calage de poteaux...), localisation relative de l'origine, liste des niveaux de grille.

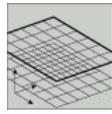


Axe de grille Un axe est un objet indépendant (vecteur) localisé par rapport à l'origine de la grille de conception, sur un niveau de grille donné. Attributs : outre la géométrie du vecteur, son appartenance à un niveau, un symbole de repérage (Numéro ou caractères...), liste des objets intersection d'axes associés.



Intersection d'axes Objet indépendant : intersection de deux (ou plus) axes de grille. Attributs : coordonnées cartésiennes, la liste des axes en cause (l'intersection ne reçoit pas d'identification propre par

l'utilisateur. C'est la liste des noms et numéros d'axes qui permettent de repérer une intersection).



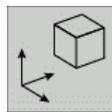
Niveau de grille Plan dans l'espace contenant tous les axes de grille d'un niveau. Attributs : appartenance à une grille de conception, altitude par rapport à l'origine de la grille, nom, liste des axes de grille inclus.



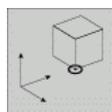
Contrainte de localisation Classe abstraite permettant de décrire des contraintes de nature variées entre objet produit, environnement de conception ou objet utilisateur. Pas d'attributs à ce niveau



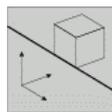
Contrainte d'intersection Permet de définir un point d'intersection commun à plusieurs objets. S'utilise avec la liste des objets définis par le concept localisation obligée (voir plus haut). Les révisions ultérieures étendront les contraintes d'intersection aux lignes et plans. Attributs : référence au point d'intersection, liste des lignes ou courbes concernées, indication de distances pour différencier les solutions multiples d'intersections.



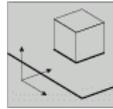
Référence géométrique de localisation Classe abstraite pour décrire le mode de localisation géométrique (point, ligne, plan) d'un objet physique. Cette classe est une localisation relative. Lignes et surfaces peuvent par exemple servir à aligner différents objets. La signification est interprétée par l'utilisateur ou l'application. Pas d'autres attributs à ce niveau que la localisation relative.



Point de référence Point de référence de localisation ayant une signification particulière pour l'utilisateur (résulte de l'action dessinée). Seul attribut : coordonnées.



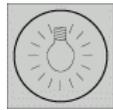
Ligne de référence Ligne ou courbe de référence dans l'espace servant à l'utilisateur pour localiser une liste d'objets physiques, autour d'un axe de symétrie par exemple. Seul attribut : la géométrie de la courbe.



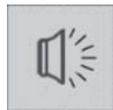
Surface de référence Surface complexe ou plane de référence servant à l'utilisateur pour localiser une liste d'objets physiques. Seul attribut : la géométrie de la surface.



Source de lumière Description des propriétés d'une source lumineuse (la géométrie et la localisation de l'objet sont consignées par le produit appareil électrique, type défini luminaire) Attributs puissance, intensité.



Volume d'éclairage/éclairage Espace en 3D concerné par des propriétés d'éclairage ou d'éclairement induits par une source de lumière. Attributs : Volume d'éclairage décrit par un solide dans l'espace, degré d'intensité associé (plusieurs volumes peuvent qualifier une même source).



Source de son Description des propriétés d'une source sonore (la géométrie et la localisation de l'objet sont consignées par le produit appareil électrique, type défini son) Attributs puissance, intensité.



Environnement acoustique Espace en 3D concerné par des propriétés acoustiques induits par une source de son. Attributs : Volume décrit par un solide dans l'espace, degré d'intensité associé (plusieurs volumes peuvent qualifier une même source). Dernières spécification IFC2x3

6.1.4 Les IFC et STEP (ISO 10303-11)

La *STEP*¹ est une norme internationale qui a pour but de décrire un produit tout au long de son cycle de vie, et indépendamment de tout système informatique, la nature de cette description la rend appropriée non seulement pour l'échange neutre des données mais aussi comme une plate forme pour implémenter et sauvegarder des bases des données.

La norme *STEP* a produit des méthodes et des technologies qui sont utilisables dans des domaines très variés :

- Le langage de description de modèles de données EXPRESS.
- Le format neutre pour l'échange des fichiers.

Note 1. La norme **STEP** (**ST**andard for the **E**xchange of **P**roduct model data) est utilisée pour l'interprétation et représentation des échanges informatiques.

- Les interfaces normalisés pour l'accès aux données.

L'IAI a adopté plusieurs résultats de la norme *STEP* dont le langage *EXPRESS* avec lequel sont décrites les différentes entités des IFC (*figure 6.1*), et le format neutre pour l'échange de fichiers. L'IAI a adapté ces ressources et notamment celles relatives à la géométrie et à la topologie.

6.1.5 Dernières spécifications IFC2x3

Les IFC2x édition 3 (IFC2x3)², est la plus récente édition officielle des IFC recommandée pour la mise en opération des nouveaux modèles. Elle a été publiée en février 2006 et elle contient plusieurs améliorations en par rapport à la dernière édition 2004. Elle a déjà été mise en pratique dans des nouveaux prototypes.

6.1.5.1 Définition d'une entité IFC

Pour continuer nous allons exposer la définition d'une des 653 entités qui conforment le modèle IFC2x3. Nous avons choisi pour cet exemple la définition de l'entité *IfcSoundProperties*, vu qu'elle est le sujet de notre étude. La finalité de cette analyse va nous permettre de connaître de façon détaillée la définition des *classes objet* IFC en vue de faire une comparaison entre *le modèle IFC STEP* et *le modèle* propre à SketchUp® dont l'analyse est abordée plus loin.

IfcSoundProperties

Définition IAI :

C'est la *définition type* utilisée pour décrire les propriétés du *son* existant dans les systèmes de distribution et de circulation d'un bâtiment quelconque. Les propriétés du *son* sont l'intensité et le niveau de pression tout au long des huit bandes d'octaves qui spécifient la quantité de son généré et son intensité.

Note 2. Les **IFC2x Edition 3** est la dernière édition officielle du modèle IFC publiée en février 2006. Pour consulter entièrement le modèle **IFC2x3** vous pouvez accéder au site officiel de l'**International Alliance for Interoperability** dans http://www.iai-international.org/Model/R2x3_final/index.htm

IfcSoundProperties est un groupe de propriétés définies qui doivent être attachées à l'entité *IfcDistributionFlowElement* à travers la relation *IfcRelDefinesByProperties*.

EXPRESSE specification :

```
ENTITY IfcSoundProperties
  SUBTYPE OF (IfcPropertySetDefinition);
  IsAttenuating : IfcBoolean;
  SoundScale : OPTIONAL IfcSoundScaleEnum;
  SoundValues : LIST [1:8] OF IfcSoundValue;
END_ENTITY;
```

Attribute definitions :

IsAttenuating: If TRUE, values represent sound attenuation.
If FALSE, values represent sound generation.

SounScale: Reference sound.

SoudValues: Sound values at a specific frequency. There may be cases where less than eight values are specified.

Inheritance graph :

```
ENTITY IfcSoundProperties;
  ENTITY IfcRoot;
    GlobalId : IfcGloballyUniqueId;
    OwnerHistory : IfcOwnerHistory;
    Name : OPTIONAL IfcLabel;
    Description : OPTIONAL IfcText;
  ENTITY IfcPropertyDefinition;
  INVERSE
    HasAssociations : SET OF IfcRelAssociates FOR RelatedObjects;
  ENTITY IfcPropertySetDefinition;
  INVERSE
    PropertyDefinitionOf : SET [0:1] OF IfcRelDefinesByProperties FOR
      RelatingPropertyDefinition;
    DefinesType : SET [0:1] OF IfcTypeObject FOR HasPropertySets;
  ENTITY IfcSoundProperties;
    IsAttenuating : IfcBoolean;
    SoundScale : OPTIONAL IfcSoundScaleEnum;
    SoundValues : LIST [1:8] OF IfcSoundValue;
END_ENTITY;
```

Schema :

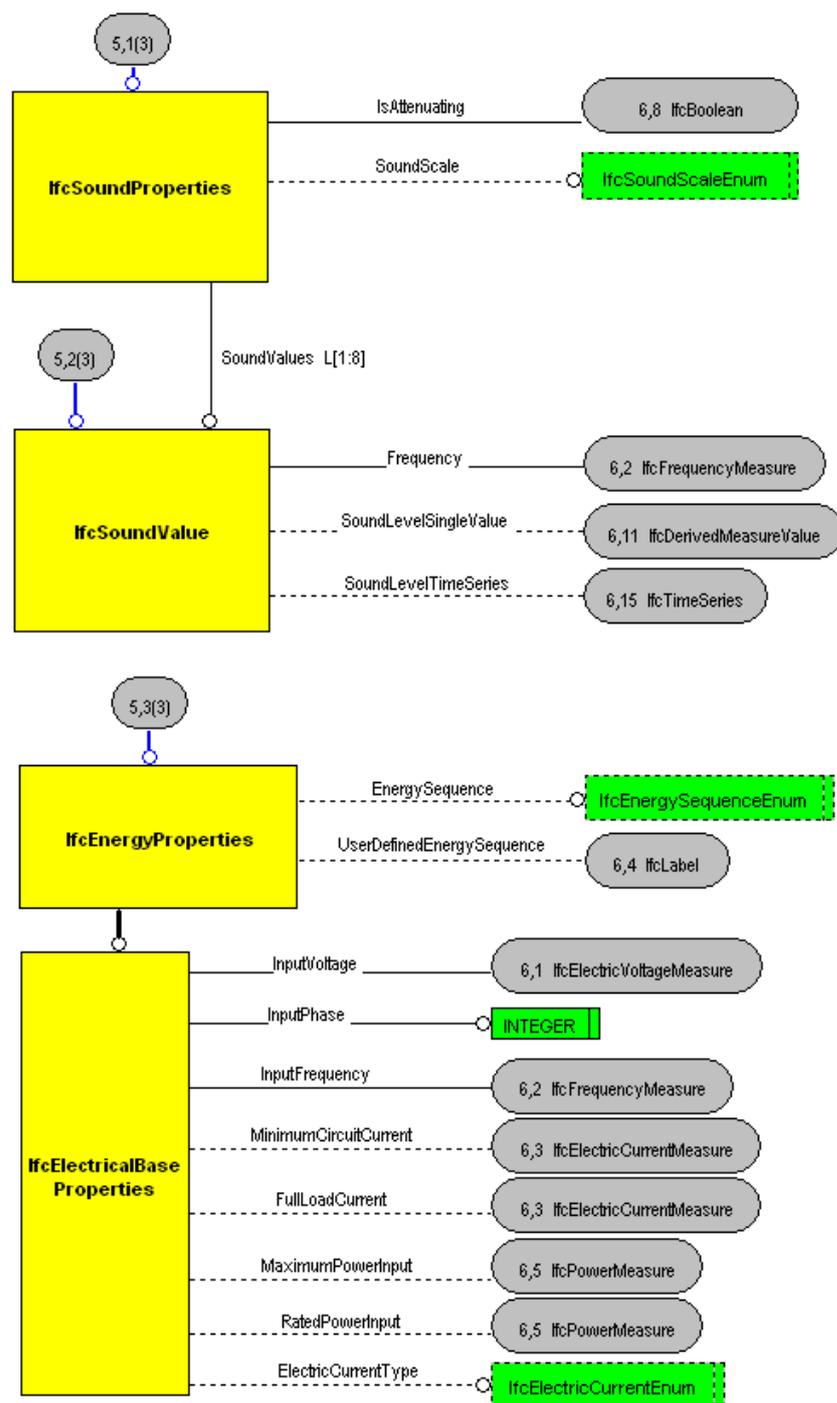


Figure 6.1 Graphique du modèle de données EXPRESS qui définit l'entité IfcSounProperties.

6.2 Interfaçage de programmation d'applications de Ruby

Nous avons aperçu brièvement dans le chapitre 4.3.2 le langage script sur lequel se base la programmation de SketchUp® (interfaçage de programmation d'applications *API* [Application Programming Interface] pour son sigle en anglais).

6.2.1 Programmation orientée objet sur SketchUp®

La *programmation par objet* ou *OOP* (pour son sigle en anglais *Object Oriented Programming*) est une méthode de conception d'un programme informatique dans laquelle celui-ci consiste en un assemblage de *briques logicielles* appelées *objets*, un objet représente un concept, une idée ou une chose du monde réel, de l'entreprise ou des idées.

La modélisation objet consiste à créer une représentation informatique des éléments du monde réel auxquels on s'intéresse, sans se préoccuper de l'implémentation, ce qui signifie indépendamment d'un langage de programmation. Il s'agit donc de déterminer les objets présents et d'isoler leurs données et les fonctions qui les utilisent.

L'API de Ruby comporte une série de modules spécifiques à SketchUp®, appelés *classes* dans le monde d'*OOP*, et les commandes correspondantes à ces classes sont appelées *méthodes*, pour créer les macros et manipuler la géométrie et les entités sur SketchUp®. Les classes peuvent être considérées comme un mécanisme pour le groupement des commandes de Ruby propres de SketchUp®.

6.2.1.1 Définition d'une classe API Ruby

Ci-dessous nous allons montrer une définition utilisée par Ruby de SketchUp pour la *classe Texture*. Nous avons choisi d'exposer cette définition pour montrer la flexibilité que possède l'API de SketchUp® pour définir les entités autres la géométrie propre des modèles.

Class Texture

```
PARENT Entity
METHODS (Average color); (Filename); (Valid?)
SAMPLE CODE Texturetest.rb
```

Instance Methods :

average_color

The `average_color` method retrieves a color object with the average color found in the texture.

Syntax:

```
color = texture.average_color
```

Return Value:

color – a color object (if successful), nil if unsuccessful.

Example:

```
model = Sketchup.active_model
materials=model.materials
# Adds a material to the "in-use" material browser tab
m = materials.add "Test Color"
begin
# Returns nil if not successful, path if successful
m.texture = "c:\Program Files\@Last Software\SketchUp 4\Materials\Carpet.jpg"
rescue
UI.messagebox $!.message
end
texture = m.texture
# Returns a color object
color = texture.average_color
if (color)
UI.messagebox color
else
UI.messagebox "Failure: No average color"
end
```

filename

The `filename` method retrieves the entire path, including the file, for a texture object.

Syntax:

```
filename = texture.filename
```

Return Value:

filename – a string representation of the path and filename used for the texture.

Example:

```
model = Sketchup.active_model
materials=model.materials
# Adds a material to the "in-use" material browser tab
m = materials.add "Test Color"
begin
# Returns nil if not successful, path if successful
m.texture = "c:\Program Files\@Last Software\SketchUp 4\Materials\Carpet.jpg"
rescue
UI.messagebox $!.message
end
texture = m.texture
# Returns a color object
filename = texture.filename
if (filename)
UI.messagebox filename
else
UI.messagebox "Failure"
end
```

valid?

The valid? method ensures that a texture is valid.

Syntax:

```
status = texture.valid?
```

Return Value:

status – true if valid, false if invalid.

Example:

```
status = texture.valid?  
if (valid)  
  UI.messagebox valid  
else  
  UI.messagebox valid  
end
```

6.3 Bilan

Au long de cette analyse nous avons dévoilé l'objectif de l'IAI de faciliter l'interopérabilité des logiciels utilisés par les acteurs de la construction et pour la gestion tout au long de la vie des projets. Elle cherche à améliorer les capacités de communication entre ces logiciels pour une meilleure qualité et une meilleure productivité. L'IAI ne produit pas de logiciels mais des spécifications résultant d'une recherche de consensus au niveau international. Ces spécifications contiennent la description des *IFC* que nous *traduirons* comme les *briques de base d'un langage commun*. Ces spécifications sont publiques, et grâce aux standards d'échange *STEP* et *EXPRESS*, chaque éditeur de logiciel peut les exploiter pour créer une nouvelle génération de logiciels plus aptes à *l'échange et au partage d'informations*.

Nous avons aussi analysé l'interfaçage de programmation de SketchUp® qui s'appuie sur le langage script de programmation par objet, Ruby ; qui permet de déterminer les objets des modèles et d'isoler leurs données et les fonctions qui les utilisent. Ainsi nous pouvons définir et représenter n'importe quelle entité ou objet dans les systèmes informatiques.

Nous pouvons en conclure donc que l'API propre de SketchUp est suffisamment ouverte pour définir également les données géométriques ainsi que les données non géométriques des modèles, notamment les données acoustiques (sources, récepteurs, rayonnement, etc.³). D'autant plus que dans l'API Ruby de SketchUp®, toutes les bibliothèques, classes et méthodes du langage Ruby standard peuvent être utilisées.

Note 3. Voir chapitre 5.3 concernant *le traitement du son et de la lumière*.

7. Conclusion et perspectives

Au cours de ce travail nous avons abordé l'étude pour déterminer la faisabilité d'un module de transfert de données entre logiciels. Nous nous sommes attachés à la réflexion et l'analyse de différents aspects constituant notre problématique. Parmi les problèmes formulés, apparaissent :

- Les données à modéliser.
 - Géométriques.
 - Acoustiques.
- Comment transférer ces données.
- Quelles interfaces adopter.

Nous avons cherché à aboutir à une solution répondant au mieux aux besoins de notre problématique. Pour aider les spécialistes du domaine du traitement acoustique des espaces, il existe aujourd'hui des logiciels qui présentent d'avantages pour cet usage, notamment le logiciel *CATT-Acoustic*® qui a été objet de notre étude, mais dont la complexité pour paramétrer la géométrie des modèles le rend difficile à exploiter. Les outils pour la simulation virtuelle du son sont très puissants, cependant, aucun ne peut adopter une démarche plus flexible au niveau de la saisie de données géométriques.

Ainsi, au travers des réflexions de notre problématique, nous avons déterminé le besoin d'adopter un outil d'aide à la conception de modèles 3D apportant des solutions au traitement des espaces complexes. Basée sur un environnement graphique dynamique et intuitif, le logiciel *SketchUp*® semble plus approprié aux habitudes de travail des concepteurs des espaces.

Le problème du transfert de données du modeleur au logiciel de traitement acoustique restait posée. Nous avons proposé l'usage des IFC comme structure de données sur laquelle se baserait la description du transfert. Ils contiennent la définition de *classes d'objets* nécessaires pour la définition et l'éventuel traitement de la géométrie 3D, les caractéristiques de matériaux, l'environnement spatial, entre autres.

Lors de nos recherches, nous avons entrevu le potentiel des algorithmes de détermination de surfaces visibles employés pour étudier le parcours inverse de la lumière et reconstituer des scènes virtuelles pour la synthèse d'images. Cette étude nous a permis de démontrer les similitudes existant entre la propagation de la lumière et le son, une fois ceci déterminé que la lumière et le son agissent de façon similaire nous pourrions établir une base de données commune pour son traitement.

Ainsi, à la suite de différentes études, il nous a été possible de proposer une réponse à notre problématique sous la forme de la description pour le développement d'un prototype de transfert entre un modeleur et un logiciel de traitement acoustique.

7.1 Perspectives

Il est à noter que, pour des raisons de temps, nous nous sommes arrêtés à la définition du prototype d'échange entre logiciels. Nous pensons qu'il serait intéressant de poursuivre cette expérience avec le développement informatique du prototype d'échange, en prenant en compte divers points à traiter :

- Les similitudes entre le son et la lumière.
 - L'emploi des algorithmes de détermination de surfaces visibles semblerait très prometteur pour la suite de cette étude.
- La définition et le transfert de données non géométriques des modèles.
 - L'exploitation d'un langage *orienté objet* offrirait les éléments pour décrire convenablement les données acoustiques des modèles.
 - L'adoption des IFC pour garantir le transfert neutre des données.
 - Notamment l'exploitation des Classes IFC *Environnement de modélisation* (Voir chapitre 6.1.3.4.).
- Le développement et la mise en opération du module d'échange Skp2Geo.
- La validation du module.

7.1.1 Test d'utilisation

Une fois développé le prototype d'échange, pour qu'il soit validé il faudrait effectuer un certain nombre de tests opérationnels vis-à-vis des utilisateurs. C'est-à-dire, au niveau de l'intérêt de la réalisation concrète de cet outil pour les différents acteurs, concepteurs et spécialistes en acoustique, a fin de constituer un ensemble performant dans le processus de conception des espaces et leur traitement acoustique.

8. Bibliographie

8.1 Publications

- [1] [GLAS2002] GLASSNER (Andrew). An Introduction to Ray Tracing. San Francisco. Morgan Kaufmann Publishers, Inc. 2002.
- [2] [RIS1996] RIS (Philippe). Parallélisation du lancer de rayon par évaluation dynamique de la topologie de la scène. Thèse Université de Marne-la-Vallée, 1996.
- [3] [SAV2000] SAVIOJA (Lauri). Modelling Techniques for Virtual Acoustics. Thesis Laboratory of Acoustics and Audio Signal Processing. Helsinki, 2000.

8.2 Liens de recherche bibliographique

- [4] AFNOR Association Française de Normalisation
<http://www.afnor.fr>
- [5] CATT-Acoustic
<http://www.catt.se>
- [6] Conception Systèmes Informations
http://smyrne.int-evry.fr/dsi/enseignement/csi10/pages/elaborer_cahier_c
- [7] Duran Audio
<http://www.duran-audio.com>
- [8] Entreprises Territoires et Développement
http://www.projetdeterritoire.com/spip/dossier.php3?id_rubrique=52

- [9] Euphonia
<http://www.euphonia.fr>
- [10] IAI France
<http://www.IAI-France.org>
- [11] IAI International
<http://www.iai-international.org>
- [12] ISO International Standards Organization
<http://www.iso.org>
- [13] Médiacconstruct
<http://www.mediaconstruct.org>

Cahier des charges fonctionnel

Module d'échange de données Skp2Geo 1.0

1. Présentation générale.

1.1 Projet.

Ce cahier de charges fonctionnel exprime le besoin et les contraintes d'un outil de transfert de données géométriques d'un modelleur 3D vers un logiciel acoustique pour son exploitation.

Modeleur : SketchUp®

Fichiers : .skp

Logiciel acoustique : CATT-Acoustic®

Fichiers : .geo

1.1.1 Objectifs.

L'outil à réaliser devra permettre à une *Equipe Projet* de préparer, mettre en opération, tester et exploiter un outil de transfert de données à partir des besoins et des contraintes exprimés dans le présent cahier des charges.

La validation sera effectuée sous la responsabilité de La Maîtrise *d'œuvre*, en particulier du *Chef de Projet*, et de La Maîtrise *d'ouvrage*, en particulier du *Responsable de Conduit de Projet*. La validation de responsabilité de La Maîtrise *d'ouvrage* est en général appelée *qualification*.

1.2 Contexte.

1.2.1 Contexte du projet.

Dans le cadre de la recherche de nouveaux outils d'aide à la conception et modélisation des espaces, le *Centre de Recherche en Architecture et Ingénierie* (CRAI) encadre cette réflexion de contraintes et spécifications.

Le prototype d'échange *Skp2Geo 1.0* conformera la première version du développement.

Les acteurs du projet sont les suivants :

- Le *Responsable de Conduit de Projet* (RCP) assurant le rôle de Maître d'Ouvrage vis-à-vis de l'Equipe Projet.

- Les membres de *l'Equipe Projet* assurant le rôle de Maître d'Œuvre et dont un membre joue le rôle de *Chef de Projet* (CP) en tant que responsable du déroulement du projet.

1.2.2 Etudes menées sur des sujets voisins.

La détection de points communs entre la propagation de la lumière et la propagation du son, qu'envisage l'utilisation en matière informatique déjà développé autour de la lumière.

1.2.2 Documents applicables.

- Les normes AFNOR série X50-150, X50-151, X50-152, X50-153 relatives au cahier des charges fonctionnel, à l'analyse fonctionnelle et à l'analyse de la valeur.
- Les normes ISO sous la référence ISO/PAS 16739, qui définissent les spécifications destinées à faciliter l'échange et le partage d'informations entre logiciels.
- Les normes ISO sous la référence ISO 10303-11, qui décrivent un produit au long de son cycle de vie et déterminent un format neutre pour l'échange des fichiers.

1.2.4 Suites prévues.

- Une fois développé le prototype d'échange, pour qu'il soit validé il faudra effectuer un certain nombre de tests opérationnels vis-à-vis des utilisateurs. Une période d'expérimentation sera effectuée à cet effet (voir chapitre 3.2.3 *Mesures prise pour respectes les contraintes*).
- Au lancement du projet, la définition d'un outil d'installation et maintenance n'était pas encore connu. Il faudra donc prévoir une remise à niveau ultérieure une fois le standard établi.

1.2.5 Nature des prestations demandées.

Sans objet.

1.2.6 Parties concernées par le déroulement du projet et ses résultats.

Les acteurs du projet sont les suivants :

- Le *Responsable de Conduit de Projet* (RCP) assurant le rôle de Maître d'Ouvrage vis-à-vis de l'Equipe Projet.

- Les membres de *l'Equipe Projet* assurant le rôle de Maître d'Œuvre et dont un membre joue le rôle de *Chef de Projet* (CP) en tant que responsable du déroulement du projet.

1.2.7 Caractère confidentiel s'il y a lieu.

Sans objet.

1.3 Enoncé du besoin.

L'outil devra satisfaire les besoins exprimés en termes de fonctions principales :

- Définir l'export de données géométriques à partir du format *skp*.
- Définir la description d'un *langage formel* (script)

1.4 Environnement d'utilisation.

L'utilisateur client de l'outil peut être de plusieurs types :

- Représentant du client : le *Responsable de Conduit de Projet*.
- Testeur : membre de *l'Equipe Projet* autre que le Chef de Projet.
- *Chef de Projet*.

2. Expression fonctionnelle du besoin.

2.1 Fonctions de service.

2.1.1 Fonctions de service principales.

Les fonctions antérieures au transfert de données :

- Définition de la géométrie 3D du modèle à exploiter.
- Définitions des éléments acoustiques à modéliser.
 - Sources
 - Récepteurs
 - Surface d'application.

Les fonctions du transfert de données à partir du format *skp* vers *geo* :

- Export de données.
 - Select File > Export > 3D Model > *.geo*

2.1.2 Description des fonctions.

Command *Export 3D Model* :

- Select File > Export > 3D Model. La boîte de dialogue *Export Model* est ouverte.
- Entrez le *nom de fichier* pour l'exportation dans le champ *File name*.
- Sélectionnez le *type de fichier* d'export à partir de la liste déroulante *Export type* (fichier *.geo*).
- Cliquez le bouton *Export*.

Sous commandes optionnelles :

- Cliquez dans le bouton *Options*. La boîte de dialogue *GEO Export Options* est ouverte.
- Ajustez les options dans la boîte de dialogue *GEO Export Options*.
- Cliquez dans le bouton *OK*.
- Cliquez le bouton *Export*.

2.2 Contraintes.

2.2.1 Contrainte d'environnement.

- L'outil devra être développé, installé et s'exécuter sur la plate forme informatique du *CRAI*.
- Le serveur et les postes utilisateurs seront au réseau local du *CRAI*.
- Les postes utilisateurs seront de PC sous environnement Windows.

2.2.2 Contrainte de développement.

- S'agissant d'un module de transfert de données il devra se conformer aux normes ISO sous la référence ISO/PAS 16739 de novembre 2002, qui définissent les spécifications destinées à faciliter l'échange et le partage d'informations entre logiciels.

2.2.3 Contrainte d'interface logicielle.

- SketchUp® peut exporter *des faces, des arêtes, des cotes, des textes, et la géométrie de construction* dans des combinaisons quelconques.

- La commande *Export 3D Model > GEO* sera utilisée pour exporter des fichiers SketchUp® afin d'être exploités dans le logiciel de traitement acoustique CATT-Acoustic®.

2.2.1 Contrainte de qualité.

L'outil devra garantir :

- L'intégralité et la cohérence des données qu'il manipule.
- La confidentialité des informations (non intrusion dans l'outil de personnes non habilitées) s'il y a lieu.
- La performance de réponse aux transferts de données.
- Une ergonomie conforme aux standards en vigueur dans l'environnement Windows.

2.3 Critères d'appréciation.

Hiérarchisation des fonctions.

F1 – Export 3D Model > GEO.

F2 – GEO Export Options.

F3 – Enregistrement de tentatives infructueuses de transfert.

2.4 Niveaux des critères d'appréciation.

F1 niveau 1 (fondamentale)

F3 niveau 2 (important)

F2 niveau 2 (optionnel important)

3. Cadre de réponse.

3.1 Pour chaque fonction.

- Tout transfert de données, soit partiel ou complet, sera journalisé.
- Toute tentative infructueuse de transfert sera obligatoirement journalisée pour son rapport.

3.2 Pour l'ensemble du produit.

3.2.1 Prix de la réalisation de la version de base.

La charge de la version *Skp2Geo 1.0* du module de transfert est estimée à 3 mois-hommes.

3.2.2 Options et variantes proposées non retenues au cahier des charges.

Sans objet.

3.2.3 Mesures prises pour respecter les contraintes et leurs conséquences économiques.

- Pour sa validité le module pilote devra refléter fidèlement l'architecture d'exploitation.
- Aucune mesure de conséquence économique n'a été prise en compte.
- Le module pilote devra passer différents tests de vérification :
 - Test de validation : Ces tests ont pour but de vérifier la conformité du produit développé par rapport à sa spécification, document d'étude décrivant ce que le logiciel doit faire en terme d'exigences fonctionnelles.
 - Test de qualification : Ces tests ont pour but de vérifier la conformité du produit livré au client par rapport au cahier des charges fonctionnel. Cette vérification permet d'obtenir de la part du client l'acceptation du produit.
 - Test fonctionnel : Ce type de test consiste à vérifier l'aptitude d'un produit à fonctionner correctement en ne s'intéressant qu'aux services rendus par le produit, c'est-à-dire à son comportement externe au vu de ses entrées/sorties.

3.2.4 Outils d'installation, de maintenance.

Voir chapitre 1.2.4. *Suites prévues.*

4. Annexes.

4.1 Dictionnaire de données.

- AFNOR** L'association Française de Normalisation est l'organisme officiel français de normalisation. Il est membre de l'Organisation International de Normalisation (ISO) auprès duquel il représente la France.
- BESOIN** Les besoins se situent au niveau de l'interaction entre l'individu et l'environnement. Ils en existent trois types ; *les besoins primaires, les besoins fondamentaux et les besoins secondaires.*
- CHEF DE PROJET** Un *Chef de Projet* est la personne chargée de contrôler le bon déroulement du développement d'un logiciel informatique ou outil informatique.
- CONTRAINTE** Une *contrainte* est une règle obligatoire qui réduit la liberté d'action, est une condition à satisfaire.
- Fonction de contrainte : L'établissement d'un Cahier des Charges qui spécifie les fonctions de service, permet d'inventorier les fonctions de contrainte que le mécanisme doit remplir.
 - Niveau de contrainte : L'étude de contrainte peut à son tour imposer une nouvelle étude à un niveau de considération plus détaillé (analyse descendante).
- ÉQUIPE PROJET** L'Equipe Projet est un groupe d'individus visant à structurer, assurer et optimiser le bon déroulement d'un projet suffisamment complexe pour devoir être planifié dans le temps. Son objectif est d'obtenir un résultat conforme à des normes de qualité et de performance prédéfinies.
- ISO** *L'International Standards Organization* est un organisme de normalisation international composé de représentants d'organisations nationales de normalisation d'environ 150 pays. L'organisme a pour but de produire de normes internationales dans les domaines industriels et commerciaux appelés *Normes ISO*. Le secrétariat central de l'ISO est situé à Genève en Suisse.
- MAÎTRISE D'ŒUVRE** La *Maîtrise d'œuvre* (MOE) est l'instance qui répond au programme fonctionnel fixé par le *Maître d'Ouvrage* et propose

une solution technique et esthétique qui permette de réaliser ce programme, dans l'enveloppe budgétaire et les délais qui lui sont assignés.

MAÎTRISE D'OUVRAGE La *Maîtrise d'ouvrage* (MOA) est responsable de l'efficacité de l'organisation et méthodes de travail d'un projet système d'information. Elle fait appel à un MOE pour obtenir les produits (logiciels, services et solutions) nécessaires à la réalisation de sa mission. Elle fournit au MOE les spécifications fonctionnelles et validité de la recette fonctionnelle des produits.

MODELEUR Un *Modeleur* est un logiciel qui sert à créer des scènes 3D, composées de formes complexes, ou objets, en trois dimensions à partir de *primitives de bases* ou de définitions analytiques.

MOIS-HOMME C'est une unité de coût de développement. Le *Mois-Homme* équivaut au travail d'un homme pendant un mois.

RESPONSABLE DE CONDUIT DE PROJET Ou encore *Coordinateur de Projet* c'est la personne responsabilisée sur l'aboutissement d'un projet, elle est chargée de coordonner les travaux de la MOA et de la MOE. C'est en ce sens qu'il se distingue du *Chef de Projet* qui, en général, n'a en charge que la partie *informatique* (MOE) d'un projet.