

## THESIS / THÈSE

### MASTER EN SCIENCES INFORMATIQUES

#### Vers l'informatisation du processus architectural dans son intégralité

Dupuis, Olivier

*Award date:*  
1989

*Awarding institution:*  
Universite de Namur

[Link to publication](#)

#### **General rights**

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

#### **Take down policy**

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Facultés Universitaires Notre-Dame de la Paix  
Institut d'informatique

**VERS L'INFORMATISATION DU  
PROCESSUS ARCHITECTURAL DANS  
SON INTEGRALITE**

Olivier DUPUIS

Mémoire présenté en vue de l'obtention du  
titre de licencié et maître en  
informatique

Directeur : C.CHERTON

Septembre 1989

Facultés Universitaires Notre-Dame de la Paix  
Institut d'informatique  
rue de Bruxelles, 61 B-NAMUR

VERS L'INFORMATISATION  
DU PROCESSUS ARCHITECTURAL DANS SON INTEGRALITE

Olivier DUPUIS

**RESUME**

Les logiciels de Conception Assistée par Ordinateur (CAO) permettent actuellement aux architectes de mettre au propre leurs projets, parfois de les mettre au point. Pourtant, aucun d'entre eux n'est capable d'assister la conception proprement dite, quel que soit son degré de perfectionnement. La solution que nous proposons est sensée combler cette lacune en essayant de tenir compte des caractéristiques propres à ce genre de travail.

**ABSTRACTS**

With Conception Aided Design (CAD), the architects can now make their projects fair and sometimes adjust them. But none of these programs is able to aid the properly so called conception whatever is perfecting degree. The solution that we propose is to fill up this gap by trying to take into account the features specific to this kind of work.

Mémoire de licence et maîtrise en informatique  
Septembre 1989  
Directeur : C.CHERTON

C'est pour moi un plaisir, bien plus qu'un devoir, de remercier toutes les personnes qui ont de près ou de loin contribué à la réalisation de ce travail, et plus particulièrement

- Monsieur C. Cherton, directeur de ce mémoire, et
- Monsieur F. Podevyn pour ses précieux conseils.

## TABLE DES MATIERES

INTRODUCTION .....	1
PREMIERE PARTIE : ETUDE CRITIQUE DE L'EXISTANT	
CHAPITRE I : Etude des logiciels existant .....	5
I.1 Progiciel versus logiciel .....	5
I.2 Version 2.5D ou 3D .....	6
I.2.1 2.5D (ou 2D + 1) .....	6
I.2.2 3D .....	7
I.3 Présentation des trois logiciels sélectionnés .....	7
I.3.1 CADVANCE .....	7
I.3.1.1 Présentation générale .....	7
I.3.1.2 Principes d'utilisation .....	8
I.3.2 ARCHICAD .....	11
I.3.2.1 Présentation générale .....	11
I.3.2.2 Principes d'utilisation .....	11
I.3.3 STAR UX .....	13
I.3.3.1 Présentation générale .....	13
I.3.3.2 Principes d'utilisation .....	14
CHAPITRE II : Critique des logiciels existant .....	16
CHAPITRE III : Analyse de la démarche architecturale .....	19
III.1 Les techniques de représentation du projet .....	19
III.1.1 Première étape .....	20
III.1.2 Deuxième étape .....	20
III.1.3 Troisième étape .....	21
III.1.4 Quatrième étape .....	22
III.2 Les deux types de documents .....	23
III.2.1 Les supports de communication .....	23
III.2.2 Les supports de conception .....	24
III.3 Les propriétés d'un support de conception .....	26
III.4 Conclusions .....	28
DEUXIEME PARTIE : ELABORATION D'UNE SOLUTION	
CHAPITRE IV : La notion de rapidité .....	31
IV.1 la rapidité des tempos de réponses .....	31
IV.2 La convivialité .....	31
IV.3 La rapidité d'utilisation .....	32

CHAPITRE V :	Propriétés de la représentation .....	33
V.1	Le mode de représentation .....	33
V.1.1	L'axonométrie .....	34
V.1.2	La perspective réelle .....	35
V.1.3	La perspective cavalière .....	35
V.2	La représentation des axes .....	36
V.3	La représentation des volumes .....	41
CHAPITRE VI :	Le déplacement dans l'espace .....	43
VI.1	La localisation du curseur dans l'espace ....	43
VI.2	Le plan de référence .....	43
VI.3	Le déplacement du curseur .....	45
VI.3.1	Le déplacement perpendiculaire .....	45
VI.3.2	La translation du plan de référence ....	46
VI.3.3	Le changement de plan de référence ....	47
VI.4	Les représentations du plan de référence ....	48
VI.5	Le curseur et le locateur .....	48
CHAPITRE VII :	Les objets manipulés .....	50
VII.1	Les volumes .....	50
VII.1.1	Le parallélépipède rectangle .....	50
VII.1.2	Le prisme triangulaire .....	52
VII.1.3	Le prisme à bases régulières .....	54
VII.1.4	Le cylindre .....	54
VII.1.5	La pyramide à base régulière .....	55
VII.1.6	Le cône .....	55
VII.1.7	La sphère .....	55
VII.2	Le problème de la non exhaustivité .....	56
VII.3	Les surfaces .....	57
VII.3.1	le rectangle .....	57
VII.3.2	Le polygone et le cercle .....	58
VII.4	Les lignes et le point .....	59
VII.5	Fonctions supplémentaires .....	59
VII.5.1	La décomposition et la recomposition .....	59
VII.5.2	Les objets de travail .....	60
VII.6	Le maniement de la souris .....	61
CHAPITRE VIII :	La notion d'indétermination .....	63
VIII.1	L'indétermination des dimensions .....	63
VIII.1.1	La création sans dimensions .....	63
VIII.1.2	Le dimensionnement a posteriori .....	65
VIII.1.3	La grille .....	71
VIII.1.4	La trame .....	73
VIII.2	L'indétermination des composants constructifs .....	74
VIII.2.1	La sélection du composant .....	74
VIII.2.2	La désignation des éléments .....	75
VIII.2.3	L'affectation partielle .....	76

CHAPITRE IX : La fonction d'archivage .....	78
IX.1 Enregistrer un calque .....	78
IX.2 Rechercher un calque .....	79
IX.3 Comparer des calques .....	80
CHAPITRE X : Les outils .....	83
X.1 Les outils classiques de manipulation .....	83
X.1.1 L'outil de déplacement .....	84
X.1.2 L'outil d'orientation .....	84
X.1.3 L'outil de reproduction .....	84
X.1.4 L'outil de suppression .....	85
X.1.5 L'outil de modification de taille .....	86
X.2 Les outils classiques de présentation .....	86
X.2.1 Le zoom .....	86
X.2.2 L'annulation .....	87
X.2.3 La fonction d'impression .....	87
X.3 Les outils de conception architecturale .....	88
X.4 Les accessoires de conception .....	91
X.4.1 Le générateur de mousse .....	91
X.4.2 Le générateur d'arbres .....	92
X.4.3 Le générateur de personnages .....	92
X.4.4 Le générateur d'automobiles .....	92
X.5 Conclusions .....	92
CONCLUSION .....	94
BIBLIOGRAPHIE	

I N T R O D U C T I O N

Les disciplines touchées par le phénomène informatique sont de plus en plus nombreuses. Cela peut s'expliquer notamment par le fait que les tâches ne doivent plus être strictement algorithmiques pour être informatisées, comme ce fut le cas pendant bien longtemps. L'avènement des traitements de textes en est d'ailleurs une belle illustration. D'autre part, le champ d'action de l'informatique s'élargit encore aujourd'hui par le développement des applications graphiques. Il aura fallu pour cela attendre un peu plus longtemps, afin que la technologie des écrans, des systèmes de pointage, des tables traçantes notamment, soit suffisamment avancée, et que son prix devienne abordable.

L'architecture n'échappe évidemment pas au processus d'informatisation. De nombreux logiciels de CAO sont d'ailleurs vendus actuellement sur le marché. Certains sont spécifiques au domaine architectural, alors que d'autres au contraire se proposent de résoudre les problèmes de CAO de plusieurs disciplines, dont l'architecture.

Ces différents logiciels répondent-ils exactement à tous les besoins des architectes ? Si tel était le cas, notre analyse n'aurait plus beaucoup d'intérêt, si ce n'est celui d'aborder d'une manière différente un problème déjà résolu. Malheureusement, il semblerait qu'il subsiste toujours un certain malaise chez les architectes à l'égard de ces logiciels. Certains sont résolument opposés à l'informatique. Peut-être ont-ils peur que celle-ci ne mette en péril leur rôle créatif ? D'autres pourtant, moins réticents, n'hésitent pas à laisser de côté leur système informatique, aussi perfectionné (et donc aussi cher) soit-il, pour concevoir leur projet, et même parfois pour produire les documents finals ! Nous allons tenter d'apporter une explication à cette situation.

Pour ce faire, nous nous pencherons dans un premier temps sur les logiciels de CAO, en essayant de sélectionner un échantillon le plus représentatif possible. Nous verrons que s'ils permettent de produire des documents d'une qualité jamais égalée auparavant, aucun de ceux-ci ne peut réellement assister l'architecte lors des premiers stades de conception. Nous discuterons ensuite de l'opportunité d'avancer l'intervention de l'ordinateur dans le cycle de vie d'un projet architectural en informatisant sa conception. Nous tâcherons alors d'approcher, à défaut de ne pouvoir

l'analyser de manière plus minutieuse, la démarche sous-jacente à cette conception. Et nous finirons par proposer une solution qui, espérons-le, devrait être plus appropriée aux besoins des architectes.

P R E M I E R E   P A R T I E   :

E T U D E   C R I T I Q U E   D E   L ' E X I S T A N T

## CHAPITRE I : Etude des logiciels existants

---

Le marché de l'informatique pour les ateliers d'architecture est en plein essor. Chaque vendeur vante évidemment les mérites de son produit, et clame haut et fort qu'il est le mieux adapté aux besoins des architectes, tant et si bien que ceux-ci sont souvent déconcertés devant la prolifération des logiciels au moment où ils veulent s'informatiser.

Pour essayer d'y voir un peu plus clair, nous allons étudier les tendances actuelles du marché de la CAO.

### I.1 Progiciel versus logiciel

---

Nous allons commencer par faire une première sélection pour ne tenir compte que des progiciels d'architecture. Ceux-ci sont destinés tout particulièrement aux architectes, et par extension à tous les professionnels du monde de la construction ; ils s'opposent aux logiciels qui sont suffisamment génériques pour assister le travail de conception de diverses catégories d'utilisateurs, dont les architectes. Pourquoi ce choix ?

- \* Si les logiciels génériques proposent toute une panoplie de primitives permettant de résoudre les problèmes spécifiques à chaque domaine d'application, il est plus que probable qu'ils ne peuvent pas offrir toutes les possibilités des progiciels comparables. De plus, le nombre de primitives proposées est sans nul doute un facteur influençant la complexité d'une interface utilisateur. Notons encore qu'il doit être assez désagréable pour un acheteur de devoir payer toute une série de choses qui lui sont inutiles.
- \* A la décharge des logiciels génériques, on pourrait rétorquer qu'il est toujours possible de les spécialiser pour l'architecture. Cette façon de faire est à notre avis contraire à la logique, et risque fort de ne pas apporter de solution adaptée aux besoins précis des architectes.

\* Mais les progiciels d'architecture prennent l'avantage sur les logiciels génériques, quand ils offrent des possibilités autres que celles de dessin proprement dit, et permettent de la sorte le chaînage de différentes tâches informatisées. Dans le monde industriel par exemple, la mémorisation des objets en trois dimensions a entraîné l'adjonction de modules de contrôle de machines à commandes numériques, pour ne perdre aucune information par le passage de la phase de conception à la phase de fabrication. Donc, à partir du moment où le domaine d'application est connu, on peut imaginer toute une série de fonctions qui lui sont spécifiques, et qui font toute la richesse du programme. C'est ainsi qu'un progiciel d'architecture peut proposer la génération automatique de vues en perspective, le calcul et la production de métrés, mais encore le calcul de la résistance des matériaux, le calcul des déperditions calorifiques, le calcul des différentes caractéristiques acoustiques, la génération d'un planning pour la conduite du chantier, la conduite d'une fraiseuse à commandes numériques pour la réalisation rapide de maquettes (pourquoi pas ?), la réalisation de scènes d'images de synthèse (voir STAR UX), le calcul de l'implantation du bâtiment sur un terrain (voir STAR UX), etc... .

Ces différentes considérations nous ont donc poussés à ne retenir que les progiciels de CAO visant exclusivement les ateliers d'architecture. Nous en avons sélectionnés trois de telle manière qu'ils soient suffisamment différents les uns des autres pour avoir un échantillon représentatif. Les trois progiciels retenus sont CADVANCE, ARCHICAD, et STAR UX.

## 1.2 Version 2.5D ou 3D

-----

Mais avant de passer les trois progiciels à la loupe, il faut savoir que CADVANCE est un version 2,5D, et que ARCHICAD et STAR UX sont des versions 3D. Voyons où se situe la différence.

### 1.2.1 2,5D (ou 2D + 1)

L'introduction des différents éléments composant le projet se fait dans une vue en plan. Mais c'est plutôt par le mode de stockage de ces éléments que se marque la différence. En effet, leurs sommets ne sont référencés que par les coordonnées en X et en Y de l'espace. Ils n'ont donc aucune propriété volumétrique.

Ce mode de stockage est celui des logiciels 2D, mais on parle de version 2,5D car une hauteur peut être malgré tout associée au tracé en plan.

Grâce à cette hauteur unique, la génération des perspectives est facilitée, mais n'est jamais entièrement automatique. De plus, ces perspectives sont mémorisées en deux dimensions, ce qui oblige l'utilisateur désirant changer le point de vue, à recommencer toute la procédure de dessin.

### 1.2.2 3D

En règle générale, l'introduction des éléments se fait toujours en plan, et certaines manipulations peuvent être effectuées directement dans une vue en perspective. C'est évidemment le stockage volumétrique qui caractérise la version 3D. En effet, chaque sommet est référencé ici par les trois coordonnées de l'espace représenté.

Ceci rend possible l'obtention directe d'une perspective complète, sans aucune intervention de l'utilisateur. Ce dernier peut même déplacer le point de vue et obtenir très rapidement les perspectives correspondantes, ce qui lui donne l'illusion de se déplacer autour du projet.

Par ailleurs, de par ce type de stockage, le volume des éléments du projet peut être calculé, ce qui rend possible la production automatique du métré d'un projet.

## **1.3 Présentation des trois logiciels sélectionnés**

-----

Nous allons maintenant examiner les trois logiciels, en commençant par celui qui offre le moins de possibilités spatiales, c'est-à-dire CADVANCE, et en terminant par celui qui en offre le plus, c'est-à-dire STAR UX.

### 1.3.1 CADVANCE

#### 1.3.1.1 Présentation générale

CADVANCE est un logiciel de CAO qui reprend le noyau d'un autre logiciel bien connu, puisqu'il s'agit d'AUTOCAD. CADVANCE n'est pourtant pas une version spécialisée d'AUTOCAD, en ce sens que ses développeurs ne se sont pas contentés d'une simple épuration de ce dernier, mais que de nouvelles fonctions ont été implémentées, sans compter la réalisation de macro-instructions préenregistrées. Ce lien de parenté implique d'ailleurs la parfaite comptabilité entre

les données des deux logiciels.

La version actuelle de CADVANCE est une version 2.5D (La version 3D en cours de conception, doit être commercialisée incessamment). Par conséquent, quoique destiné aux architectes, CADVANCE ne permet pas la réalisation des métrés, mais est uniquement consacré au dessin. Certaines données (un nombre d'éléments ou une surface par exemple) peuvent être malgré tout sorties sur fichier ASCII ou même DBASE, ce qui facilite l'emploi de logiciels spécialisés d'aide à la réalisation des métrés, disponibles sur le marché.

Pour bien situer CADVANCE par rapport à ses concurrents, il faut signaler qu'il coûte aux environs de 180.000 FB, et qu'il tourne avec MS-DOS sur les compatibles munis d'un processeur 80286 au moins, d'un coprocesseur numérique, et d'un disque dur.

#### 1.3.1.2 Principes d'utilisation

L'introduction des données, c'est-à-dire des primitives et de leurs arguments, peut se faire de trois manières différents :

##### 1° au clavier

Deux colonnes de menu sont affichées en permanence sur la partie droite de l'écran. L'utilisateur peut donc se déplacer dans ces colonnes avec les flèches directionnelles du clavier pour sélectionner une primitive. Quant aux arguments (numériques ou alphanumériques), ils doivent être introduits au clavier.

##### 2° à l'aide d'une souris

La souris permet de cliquer directement dans les deux colonnes de menu sur la primitive voulue, et de déplacer le curseur sur l'écran pour introduire les arguments numériques. Le clavier reste évidemment utilisable pour l'introduction des arguments alphanumériques, ou même pour celle des arguments numériques lorsqu'une grande précision est requise.

##### 3° avec une tablette à digitaliser

Elle permet comme la souris de se déplacer dans le menu et sur l'écran. Mais elle permet en plus de disposer de menus supplémentaires. Ceux reprennent par exemple un choix de symboles graphiques qu'il faudrait sélectionner avec les deux moyens d'introduction précédents.

Les objets manipulés par CADVANCE sont des éléments architecturaux, dont le principal est le mur. Une primitive existe donc pour dessiner un mur d'un point de départ à un point d'arrivée, après avoir introduit son épaisseur au

clavier. Au point d'arrivée, il est possible de continuer le même mur dans une autre direction, sans devoir changer de primitive. Cette introduction en série permet évidemment de dessiner le contour d'un bâtiment, ce qui pose alors un problème de précision pour la jonction finale (voir figure 1-a). Pour résoudre ce problème rapidement, l'utilisateur peut employer l'outil "coin", qui corrigera automatiquement l'erreur (voir figure 1-c), s'il a défini au préalable la zone où cet outil est sensé agir (voir figure 1-b).

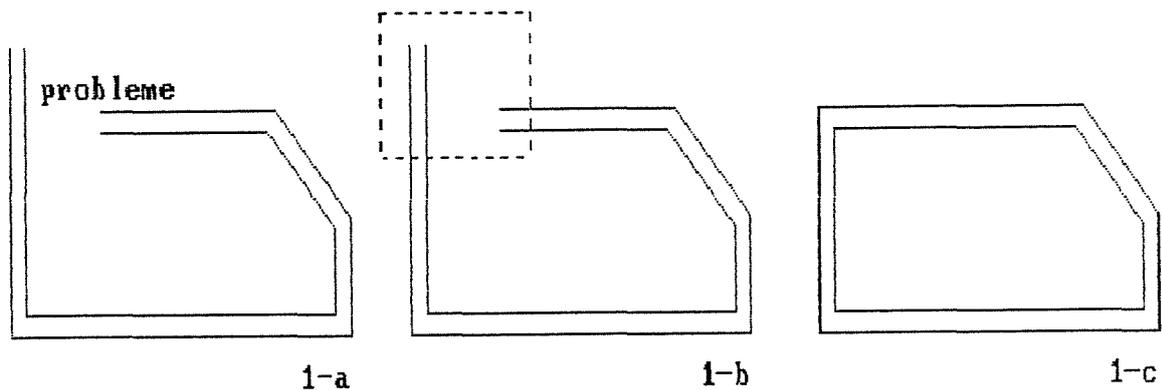


Figure 1

Pour pouvoir se repérer aisément, l'utilisateur a à sa disposition une trame à maille réglable, dont les noeuds peuvent être rendus magnétiques. Dans ce cas, chaque point introduit n'importe où se place automatiquement sur le noeud de la trame le plus proche.

Nous avons vu qu'une primitive permettait de dessiner les murs extérieurs d'un bâtiment, mais elle permet également de dessiner ses cloisons intérieures. La jonction murs/cloisons peut être réalisée par les commandes "té ouvert", "té fermé", ou encore "y" (voir figure 2).

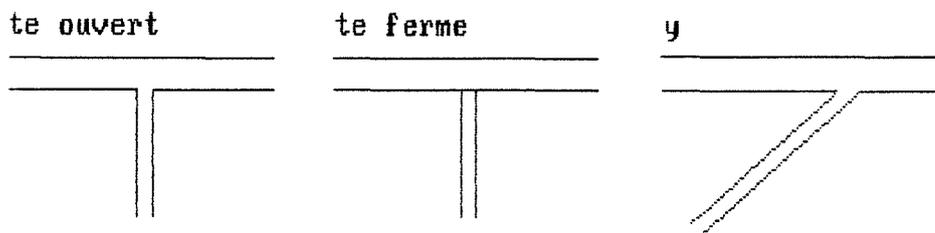


Figure 2

Un fois qu'un mur a été dessiné, il peut toujours être déplacé. Les raccords engendrés par sa nouvelle position se redessinent automatiquement (voir figure 3). D'autres manipulations sont possibles, comme l'effacement ou la rotation par exemple.

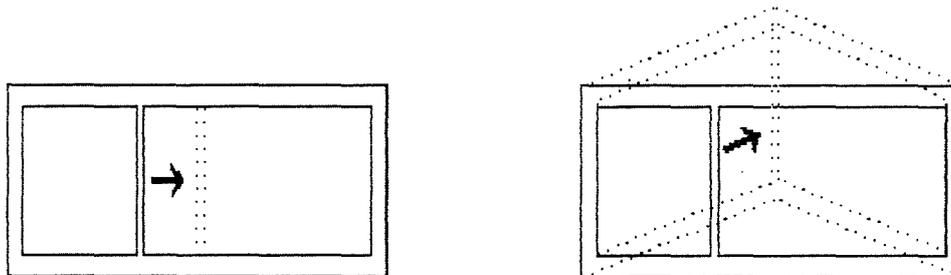


Figure 3

Le mur n'est évidemment pas le seul élément architectural proposé par CADVANCE. Citons-en quelques uns : la porte intérieure, la porte extérieure, ouvrant à droite ou à gauche, la fenêtre de différents types, etc... .

Afin d'étoffer les plans, toute une série de symboles sont à la disposition de l'utilisateur, comme par exemple ceux du salon, de la douche, de l'évier, etc... . Il peut aussi insérer du texte dans son dessin.

Les cotations se calculent bien entendu automatiquement. Pour ce faire, l'utilisateur doit placer la ligne de cotation, c'est-à-dire la ligne sur laquelle viendront s'inscrire les chiffres, et doit soit désigner explicitement ce qu'il veut coter, soit placer une ligne fictive là où les mesures doivent être prises.

Citons en vrac quelques autres fonctions :

- \* zoom out (en arrière, donc qui réduit)
- zoom in (en avant, donc qui agrandit) ;
- \* annulation de la dernière commande (UNDO) ;
- \* fonction miroir (qui n'agit pas sur les textes intégrés au dessin afin qu'ils restent lisibles dans la nouvelle vue) ;
- \* division automatique d'un segment ;
- \* digitalisation en deux dimensions de nouveaux symboles (arbres, personnages, etc...) ;
- \* fonctions d'interfaçage avec la table traçante (mise en page, correspondance couleur/plume, etc...) ;
- \* possibilité de définir de nouvelles macro-commandes.

Signalons enfin que CADVANCE propose une primitive permettant d'obtenir automatiquement la perspective de

n'importe quel volume, pour autant que celui-ci ait une hauteur constante. Si ce n'est pas le cas, l'utilisateur doit achever lui-même la perspective, en dessinant les arêtes adjacentes aux sommets qui sont à une hauteur différente.

### I.3.2 ARCHICAD

#### I.3.2.1 Présentation générale

Comme nous l'avons déjà signalé, ARCHICAD est un logiciel 3D. Quelques restrictions concernant la troisième dimension sont pourtant à épingleter.

ARCHICAD manipule des éléments architecturaux 3D, mais ceux-ci doivent obligatoirement être introduits et manipulés dans une vue en plan, ce qui n'empêche pas l'élaboration simultanée d'un projet volumétrique. De ce fait, l'utilisateur peut demander diverses vues de ce projet (perspectives réelles, axonométries, coupes par exemple), qui sont générées automatiquement. Celles-ci sont alors transformées en images en deux dimensions, que l'utilisateur peut habiller grâce aux outils 2D proposés par le programme. Il peut aussi retoucher une perspective avec un éditeur graphique 2D tel que MAC-PAINT, ou MAC-DRAW, ou l'insérer dans une photo "scannée".

Quant aux métrés produits par ARCHICAD, ils sont quelque peu rudimentaires. Il est toutefois possible de les exporter vers le tableur EXCEL, ou vers ARCHIMAC, qui est un logiciel plus spécifique, donc plus complet, d'aide à l'élaboration des métrés.

ARCHICAD fonctionne exclusivement sur ordinateur MACINTOSH II, avec coprocesseur numérique et disque dur. Il est à noter qu'ARCHICAD préserve l'interface utilisateur du MACINTOSH, bien connue pour sa convivialité, ce qui assure en plus une certaine cohérence avec les autres logiciels du MACINTOSH.

En ce qui concerne le prix d'ARCHICAD, il oscille autour de 175.000 FB.

#### I.3.2.2 Principes d'utilisation

Un point peut être introduit en le désignant sur l'écran avec la souris, ou en entrant numériquement au clavier ses coordonnées absolues ou relatives, exprimées en mètres (ou en degrés). La notion d'échelle n'intervient que pour les sorties sur supports physiques, et la grandeur du dessin à l'écran est contrôlée par la fonction zoom.

Les éléments architecturaux 3D manipulés par ARCHICAD sont les murs, les dalles, les pans de toiture, les fenêtres,

et bien d'autres encore. Examinons le cas du mur. Avant de spécifier l'emplacement d'un mur, l'utilisateur doit éditer ses propriétés. S'il ne le fait pas, il devra se contenter alors des paramètres par défaut. Ces propriétés sont : l'épaisseur, la nature du matériau, le prix (unitaire, au mètre courant, par mètre carré, ou par mètre cube) et la formule de calcul indiquant au programme la manière dont il faut tenir compte de l'élément pour le métré total. Par exemple, un mur porteur pourra avoir les propriétés suivantes : 0.19, bloc béton, 7500/m<sup>3</sup>, add (qui veut dire qu'il faut additionner le volume de tous les mur pour le calcul du prix final).

A ces propriétés vient encore s'ajouter l'emplacement des "points chauds", qui permettent à l'utilisateur d'introduire le mur par son axe, ou par une de ses faces (voir figure 4). Ils permettent également de le manipuler par la suite. Il faut noter que l'aspect du curseur change quand il se trouve sur un tel point, d'où un gain de précision.

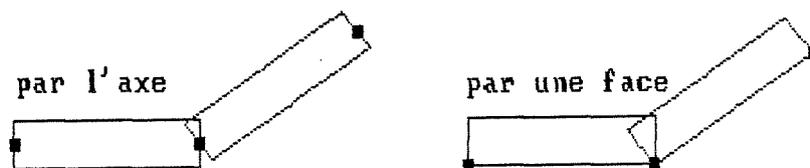


Figure 4.

La jonction entre les différents murs n'est pas calculée en trois dimensions. Par contre, il est possible de demander un dessin 2D soigné du plan, où les raccords sont correctement exécutés.

Si l'utilisateur veut introduire un élément architectural qui n'est pas connu d'ARCHICAD, il a toujours la possibilité de le définir. De plus, grâce au langage GDL (Graphic Description Language), il peut paramétrer cet élément en fonction de ce qu'il en attend.

Comme de nombreux logiciels graphiques, une trame peut être définie, mais elle peut être dissociée ici de la grille magnétique.

Pour clôturer la présentation d'ARCHICAD, voici encore quelques outils 2D et 3D proposés par le logiciel :

- \* calcul automatique des cotations, avec plusieurs présentations, et possibilité de forcer la cote c'est-à-dire d'écrire un chiffre ne correspondant pas tout à fait à la dimension ;

- \* zoom illimité ;
- \* manipulation d'un élément par ses "points chauds", comme par exemple le déplacement, la rotation, l'étirement, etc... ;
- \* fonction "copier à l'étage" ;
- \* fonction "optimiser" qui consiste à déterminer un cadre tel que tout le projet y figure, et à l'afficher sur l'écran ;
- \* édition de hachures, de lignes, ou de couleurs.

### I.3.3 STAR\_UX

#### I.3.3.1 Présentation générale

STAR UX est un logiciel 3D. On peut même affirmer qu'il est un des rares, voire un des seuls, logiciel vraiment 3D. Voyons pourquoi.

Pour commencer, STAR UX manipule des éléments tri-dimensionnels, ce qui autorise la génération des perspectives et la production des métrés. Il permet également à l'utilisateur d'introduire ces éléments directement dans une vue en perspective, ce qui est déjà peu courant.

Ce qui l'est encore moins, c'est que STAR UX manipule des éléments typiquement 2D (des cotations, des hachures, du texte, etc...), qu'il intègre au projet en trois dimensions, plutôt que dans son image bi-dimensionnelle comme le font la plupart des logiciels, même 3D. C'est ainsi que l'utilisateur pourra écrire dans le plan d'une toiture par exemple, ou placer des cotations dans une perspective.

Comment passer sous silence les nombreuses options qui peuvent être ajoutées au module de base, alors que nous avons déjà mis en évidence au début de ce chapitre les avantages que procure le chaînage de plusieurs tâches informatisées. Parmi les différents modules optionnels, nous retiendrons :

- \* le module "bureautique", qui récupère notamment le calcul des métrés pour les intégrer dans des applications plus spécialisées,
- \* le module "terrain", qui peut calculer l'implantation du projet sur son site,
- \* le module "images de synthèse", qui permet de positionner des points lumineux dans l'espace, de sélectionner la couleur de la lumière ambiante, de représenter la texture des différents matériaux, de faire succéder très rapidement des perspectives suivant une séquence déterminée.

STAR UX ne fonctionne que sur les stations de travail HEWLETT PACKARD de la série 9000. Le système d'exploitation utilisé est UNIX, et STAR UX est multitâche, et multifenêtré. Un terminal alphanumérique pour les applications bureautiques peut donc être connecté à une station de travail, et plusieurs fenêtres peuvent apparaître simultanément à

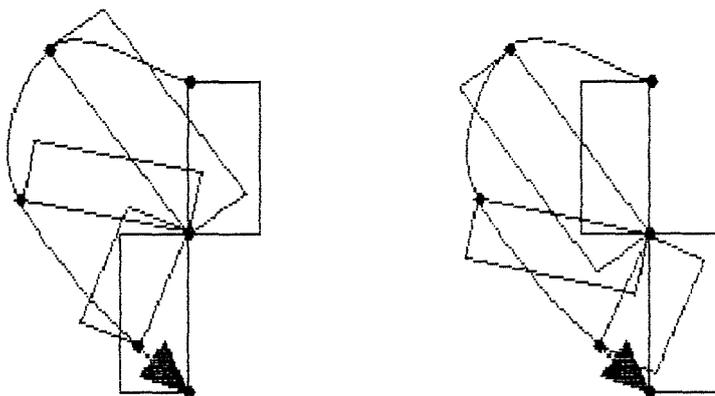
l'écran, chaque fenêtre pouvant contenir une représentation différente du projet.

Etant donné la puissance de STAR UX, l'investissement qu'il représente est évidemment plus conséquent. Le prix du module de base est de 500.000 FB.

### 1.3.3.2 Principes d'utilisation

L'interface utilisateur du logiciel repose sur l'utilisation de menus déroulants, qui peuvent s'afficher à tout moment à l'endroit où se trouve le curseur, en enfonçant le bouton de droite de la souris.

L'élément tri-dimensionnel de base manipulé par STAR UX est le parallélépipède rectangle. Son introduction peut se faire par deux points, en série, en cercle, le long d'une ligne courbe, etc... , et les points spécifiés peuvent appartenir à l'axe, ou à la face de droite ou de gauche par rapport au sens de progression (voir figure 5). L'utilisateur a dû préciser au préalable son épaisseur et sa hauteur.



par la face de gauche

par la face de droite

Figure 5

Ce parallélépipède peut être modifié ensuite en vue d'obtenir n'importe quel volume dont les faces sont planes (voir figure 6). Les opérations de manipulation proposées par STAR UX sont multiples, et possèdent généralement de nombreuses options.

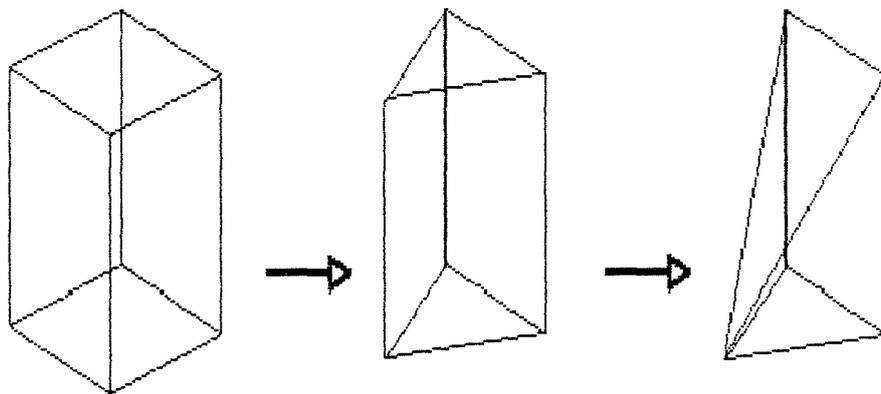


Figure 6

Plusieurs éléments tridimensionnels peuvent être assemblés en un groupe. Ce groupe peut alors être catalogué, ce qui permet à l'utilisateur de le reproduire à d'autres endroits du projet, ou même dans d'autres projets, au même titre qu'un élément tri-dimensionnel de base. Il est à noter qu'un groupe est paramétré en ce sens que son volume de calage, c'est-à-dire le volume circonscrit à tous ses éléments, est défavorable. De plus, si c'est la description cataloguée de ce groupe qui est modifiée, cela entraîne la transformation de toutes ses reproductions.

Aux faces d'un élément tridimensionnel, l'utilisateur a la possibilité d'associer un symbole 2D qu'il a puisé dans la bibliothèque. L'élément devient de ce fait un composant, qui outre une description graphique a également une description technique servant à élaborer les métrés. C'est grâce à cette possibilité qu'un groupe de parallélépipèdes rectangles devient par exemple le composant "mur" (voir figure 7).

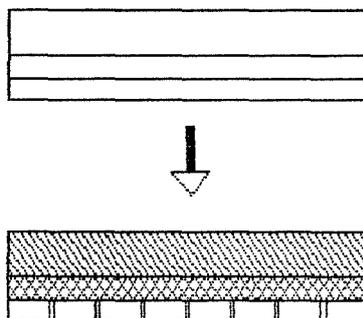


Figure 7

Signalons enfin la présence d'un plan de référence pour le travail en trois dimensions. Le curseur se déplace dans ce plan (on verra plus tard que ce n'est pas tout à fait vrai), et l'utilisateur peut y déposer des éléments 3D, ou y incorporer des éléments 2D. Ce plan peut être horizontal, vertical, ou oblique.

## CHAPITRE II : Critique des logiciels existants

---

Les programmes de CAO pour architectes en vente actuellement sur le marché ne méritent, il faut bien le dire, que partiellement leur nom. En effet, alors qu'ils assistent particulièrement bien l'architecte lors de la phase finale de conception d'un projet, leur aide est faible, voire même inexistante, en ce qui concerne la conception architecturale proprement dite.

Entendons-nous bien sur les termes.

La phase finale de conception correspond en quelque sorte à la mise au point détaillée d'un projet, ou du moins, à sa mise au propre. Cette phase présuppose donc l'existence d'un projet suffisamment étoffé pour ne plus nécessiter que d'éventuelles modifications d'ordre constructif (par exemple : tel type de mur plutôt que tel autre, tel matériau remplaçant tel autre, etc... ).

La conception architecturale proprement dite désigne quant à elle la phase de production d'esquisses précédant la mise au point, et par laquelle le concepteur passe de la "feuille blanche" à un projet tel que décrit ci-dessus.

Les logiciels actuels commencent à intervenir dans le cycle de vie du projet architectural plus ou moins rapidement selon leur puissance. En effet certains logiciels acceptent un degré d'incertitude sur le projet de départ (possibilité de définir après coup quels seront les matériaux employés, modifications dimensionnelles aisées du projet, etc... ), alors que d'autres nécessitent un projet beaucoup plus élaboré, car ils sont totalement dépourvus d'outils facilitant le travail conceptuel.

Ce qui est absolument certain, c'est qu'aucun de ces logiciels ne peut se targuer de pouvoir intervenir dès les premières phases de conception (contrairement à ce que peut laisser sous entendre les documents publicitaires de la plupart de ces programmes !), et ceci malgré la possibilité éventuelle de travailler directement en trois dimensions. Il est assez facile de se convaincre que cette inaptitude à apporter une aide efficace à la conception, réside notamment dans le fait que :

- \* ces programmes ne manipulent bien souvent que des composants architecturaux, qui, dans la majorité des cas, sont loin de constituer la préoccupation majeure du concepteur lors de l'esquisse ;

- \* ils exigent une précision exemplaire, ce qui freine véritablement le concepteur en pleine action. En effet, à ce stade, il importe plus de matérialiser rapidement les idées sortant tout droit de l'imagination que de dessiner soigneusement.

Cette situation nous incite tout naturellement à nous pencher sur le problème que constitue la conception architecturale assistée par ordinateur (CAAO pour la distinguer de la CAO au sens où on l'entend habituellement). Pour ce faire, nous tâcherons de déterminer quels sont les besoins en techniques graphiques des concepteurs de projets architecturaux, afin d'en déduire ensuite les fonctionnalités que devrait présenter un nouveau type de programme pour être efficace dès les premiers stades du projet. Mais avant toute chose, discutons de l'opportunité que représente le développement d'un tel programme. Ce serait effectivement dommage de le développer, pour se rendre compte après coup qu'il n'intéresse pas les utilisateurs potentiels, car il ne leur apporte rien qui puisse vraiment justifier l'investissement qu'il représente.

Que va donc gagner l'architecte par l'informatisation de la conception proprement dite ? Malgré le fait qu'il soit toujours assez hasardeux de répondre à ce genre de question, nous allons mettre en évidence deux points grâce auxquels un système de CAAO prend le pas sur la méthode traditionnelle.

- 1) L'utilisateur va gagner du temps, mais pas sur la phase de conception. En effet, pendant cette phase, c'est l'imagination du concepteur qui règle la vitesse du travail, que les idées soient transcrites par un dessin à main levée sur une feuille de papier, ou par une souris sur un écran. On aurait même tendance à douter que la technique informatique puisse être aussi rapide que la technique manuelle. Par contre, l'absence de transition entre la conception manuelle du projet, et sa mise au point assistée par ordinateur, entraînera un gain de temps considérable sur le cycle de vie complet du projet. Il faut pour cela que le logiciel de CAAO soit couplé à un logiciel de CAO tel qu'on les trouve actuellement. Remarquons que cet avantage n'est pas à négliger, car il arrive régulièrement que des architectes pourtant équipés d'un logiciel de CAO, produisent manuellement un projet jusqu'au bout, et ce afin d'éviter cette étape de transition qui réclame trop de temps, surtout si la maîtrise du logiciel n'est pas parfaite.
- 2) Un autre point positif à mettre à l'actif du système de CAAO est sa capacité à stimuler l'imagination du concepteur. En effet, un logiciel peut proposer certaines choses, alors que la feuille blanche de la méthode traditionnelle est tout à fait inerte. De plus.

le logiciel a un espace de création extensible, ce qui n'est évidemment pas le cas de la feuille de papier. Et enfin, le logiciel permet de créer des objets, mais aussi de les manipuler tout aussi facilement. Dynamisme, extensibilité du support, souplesse : toutes ces caractéristiques ne peuvent que faciliter le travail de conception, du moins le rendre plus agréable.

Nous pouvons passer maintenant à l'analyse de la démarche architecturale.

## CHAPITRE III : Analyse de la démarche architecturale

---

Il faut remarquer tout d'abord qu'il est très difficile d'analyser scientifiquement la conception architecturale. En effet, celle-ci (comme toute démarche artistique d'ailleurs) fait appel à des choses aussi peu rigoureuses que l'expérience personnelle, l'intuition, les circonstances du moment, le hasard même, plutôt qu'à une méthodologie précise. Par conséquent, au lieu d'essayer de modéliser le travail de conception, on se contentera de dégager quelques grandes particularités qui le caractérisent souvent, et qui pourront si pas guider, du moins influencer nos choix en ce qui concerne les fonctions devant être proposées par le nouveau type de logiciel.

### III.1 Les techniques de représentation du projet

---

Avant toute réflexion sur la démarche de conception proprement dite, tâchons de la situer par rapport au cycle de vie complet d'un projet architectural, et ceci en termes de techniques de représentation. Ceci va nous permettre de mieux saisir :

- 1) quels sont les besoins graphiques réellement demandés lors de l'étape de conception architecturale,
- 2) en quoi diffèrent-ils des besoins graphiques des autres étapes,
- 3) pourquoi les techniques de représentation proposées actuellement en CAO sont souvent inadaptées pour cette étape.

Déterminons donc maintenant le procédé de représentation privilégié de chaque étape du cycle de vie d'un projet architectural, c'est-à-dire celui qui semble être le plus approprié pour exprimer les idées à ce stade d'avancement. Cela n'implique en aucune façon que ce procédé est le seul à être utilisé lors de cette étape, ni même qu'il est toujours utilisé.

### III.1.1 Première étape

Lorsque l'architecte entame un projet, c'est bien souvent à l'aide de croquis grossiers (généralement indéchiffrables par d'autres personnes).

Un moyen d'expression parfois utilisé à ce niveau est le texte (eh oui !). Il permet effectivement d'exprimer des idées trop générales pour être dessinées, mais tellement importantes ou influentes pour la suite qu'elles doivent être archivées.

Au croquis et au texte vient encore s'ajouter un autre moyen d'expression, typiquement architectural lui : l'organigramme. Il permet de représenter sous forme symbolique les différentes pièces du bâtiment, leur surface, leur orientation, et surtout leurs inter-relations et l'intensité de celles-ci (voir figure 8).

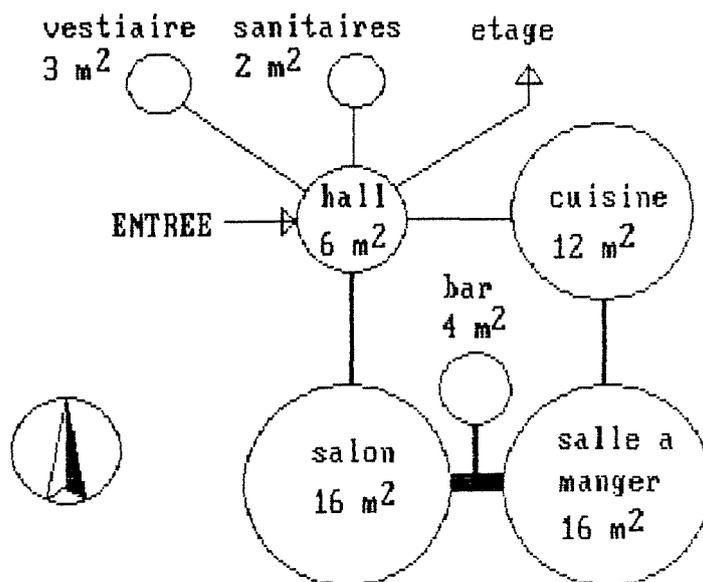


Figure 8

Cette étape correspond à la recherche du "parti", c'est-à-dire à l'énoncé des idées maîtresses qui guideront la conception future.

### III.1.2 Deuxième étape

L'étape suivante (si étape il y a, car en fait l'évolution d'un projet est plutôt continue) permet d'aboutir à une esquisse, c'est-à-dire un projet "dessinable" mais fortement simplifié, dénué de toute information non

significative à ce stade de la conception. Cette étape est réitérée jusqu'à ce que l'architecte arrive à une esquisse (ou même plusieurs) qu'il estime suffisamment intéressante pour être développée plus en détail dans l'étape suivante, ce qui ne veut pas dire que cette esquisse soit définitive, qu'elle ne peut être remise en question, que les choix qu'elle entraîne sont irrévocables.

Ici encore, le croquis est assez volontiers utilisé, sous une forme peut-être moins "brouillon" que dans l'étape précédente.

Et déjà, les techniques classiques de représentation en deux dimensions (plans - coupes - élévations) sont utilisées, mais de manière appauvrie, puisqu'elles font fi de certaines conventions de dessin. Les murs par exemple sont dessinés sans épaisseur, à main levée, à une échelle approximative et avec une précision toute relative. Les documents produits par ces techniques de représentation 2D : le plan découlant directement de l'organigramme de la première étape (d'où son caractère toujours très sommaire), le plan masse ne reprenant que les contours approximatifs du ou des bâtiments, et la coupe de principe qui peut être considérée également comme une évolution verticale de l'organigramme.

Remarque : rappelons une fois encore que s'il apparaît que chaque stade a son moyen d'expression privilégié, il ne faut pas en déduire pour autant que tout le monde utilise "le" moyen de représentation inhérent à un stade déterminé du projet. A vrai dire, chaque concepteur utilise le moyen d'expression avec lequel il est le plus à l'aise.

### III.1.3 Troisième étape

Cette étape peut être située entre la production des esquisses et la production d'un avant-projet (ici encore, l'étape peut être répétée plusieurs fois).

L'avant-projet n'étant qu'une esquisse plus élaborée, les techniques de représentation utilisées dans l'étape précédente restent valables ici. Cependant, une préférence ira aux représentations en deux dimensions, et le souci de la précision sera plus marqué.

Remarque : quand on dit qu'un avant-projet est une esquisse plus élaborée, c'est toujours sur le plan de la conception, et non simplement sur celui des détails techniques. Cette remarque est importante car elle montre bien l'apparition tardive des préoccupations concernant les composants constructifs, et donc l'inadaptation des progiciels actuels qui ne manipulent bien souvent

que ce type d'objet.

Le document final, c'est-à-dire l'avant-projet lui-même, demande un soin tout particulier ainsi qu'un "rendu" (habillage, présentation) éventuel. Par exemple, la végétation sera dessinée, l'échelle humaine sera exprimée par l'ajout de personnages, ou tout simplement des artifices graphiques seront employés pour mettre en valeur le projet. Il faut savoir en effet que c'est à partir de ce document que l'architecte tentera d'obtenir une première confirmation du client.

#### III.1.4 Quatrième étape

Cette dernière étape (sans doute la plus connue car les gens ne pensent souvent qu'à elle quand on leur parle d'architecture) permet d'aboutir au projet final en partant d'un avant-projet.

Les techniques graphiques utilisées ici correspondent, comme on pouvait s'y attendre, aux techniques classiques et académiques de représentation, à savoir le plan, la coupe et l'élévation. Si ces trois modes de représentation ne donnent qu'une idée fort théorique de l'espace, ils ont des avantages non négligeables, ce qui explique probablement leur fréquence d'emploi. Ces avantages sont :

- \* la lecture non ambiguë, pour autant que les conventions soient scrupuleusement respectées et que le lecteur soit rompu à la lecture de ce genre de document ;
- \* la possibilité de représenter l'espace de manière complète (en produisant des plans et des coupes en nombre suffisant), et ceci malgré le caractère bi-dimensionnel de ces techniques ;
- \* la facilité de positionner de manière extrêmement précise un point quelconque dans l'espace, en ne nécessitant que quelques calculs élémentaires, pour la conversion d'échelle notamment. C'est principalement grâce à cette dernière propriété que l'ensemble plan - coupe - élévation l'emporte souvent sur toute autre mode de représentation.

Tous les documents produits dans cette étape font appel aux techniques citées ci-dessus. Ceci reste bien évidemment valable pour le projet final, qui outre ces techniques, mobilise également celles des perspectives. Il s'agit ici de vues en perspective réelle vues axonométriques ou autres perspectives cavalières, qui sont construites, c'est-à-dire calculées selon un modèle mathématique afin d'approcher le plus possible la réalité. L'intérêt majeur de ces perspectives réside dans le fait qu'elles offrent une vue moins théorique du projet, plus réelle, ce qui en fait un moyen d'expression fort attrayant, pas forcément aux yeux des concepteurs mais comme nous allons le voir.

## III.2 Les deux types de documents

---

Que nous révèle le relevé des moyens d'expression du projet ? Principalement qu'ils peuvent engendrer deux types de documents bien distincts : ceux qui servent de support de conception, et ceux qui servent de support de communication. Bien comprendre ce qui les différencie est déjà un premier pas dans la recherche des besoins graphiques des concepteurs de projets architecturaux.

### III.2.1 Les supports de communication

Ils se retrouvent principalement dans la quatrième et dernière étape, bien que l'avant-projet entre également dans cette catégorie de documents. Cette concentration est absolument normale, car les documents produits à l'issue de cette étape sont tous destinés à d'autres personnes que le concepteur même. Il est intéressant de noter ici quelles peuvent être ces différentes personnes, et ce qu'elles attendent exactement des différents documents (limitons-nous aux documents graphiques).

- \* L'entrepreneur figure parmi ces personnes. Il doit sur base des documents qu'il reçoit, proposer tout d'abord un devis, et assurer ensuite l'exécution correcte de la construction. On comprendra qu'il souhaite des documents précis, clairs, complets, qui contiennent toutes les informations nécessaires à sa mission, et qui respectent les conventions graphiques habituelles assurant un déchiffrement sans erreurs. Il est évident que les plans - coupes - élévations, du bâtiment mais aussi de certains détails techniques, sont les documents graphiques qu'il appréciera le plus. Il en va de même pour les bureaux d'études spécialisés, qui sont sollicités pour réaliser une analyse poussée des problèmes relatifs à divers domaines (stabilité, isolation, etc... ). Citons encore les fonctionnaires des administrations qui délivrent les permis de bâtir, même si ceux-ci peuvent avoir le même genre de préoccupation que le client.
- \* Le client (maître de l'ouvrage, futur utilisateur, promoteur, etc... ) est également visé par les documents de l'architecte. Mais il en attend tout autre chose que l'entrepreneur. En effet, le degré de précision, ou le respect des conventions graphiques, qu'il ne connaît peut-être que de manière rudimentaire, lui importe peu.

Par contre, il veut avoir une idée la plus exacte possible du futur bâtiment, afin d'éviter toute surprise lors de sa construction.

Les techniques des perspectives calculées lui sont donc

tout particulièrement destinées, et les logiciels actuellement disponibles sur le marché remplissent remarquablement ce rôle. En effet, outre une qualité accrue du dessin de la perspective, ils permettent d'introduire un facteur primordial, et qui a toujours été laissé pour compte : le facteur temps. Expliquons-nous. Aucune image, même en trois dimensions (perspective) ne donne une vue globale d'un bâtiment. Par conséquent, la compréhension du projet dans sa totalité ne peut se faire que par recoupement des diverses vues que l'on a de ce projet. Etant donné la puissance actuelle des systèmes de CAO et d'images de synthèse, il est possible de faire succéder rapidement sur l'écran différentes perspectives suivant une séquence déterminée, afin de simuler un cheminement dans le projet. Cette évolution virtuelle dans le projet permet à l'utilisateur de comprendre celui-ci d'une manière tout à fait nouvelle, et très proche de la réalité.

Remarque : cette approche scénographique du projet, époustouflante de réalisme, est évidemment appréciée aussi par le concepteur (qui oserait le nier ?). Mais l'intérêt qu'il porte à cette technique est différent de celui du client. En effet, le concepteur est bien la dernière personne qui a besoin d'un tel réalisme pour comprendre son projet. Par contre, on comprend aisément sa réaction d'impatience quand il s'agit de matérialiser son oeuvre. La scénographie pour le concepteur est alors à considérer un peu comme un jouet d'enfant gâté, qui lui propose une matérialisation intermédiaire avant la construction finale.

### III.2.2 Les supports de conception

Les supports de conception se retrouvent dans les quatre étapes, étant donné que la démarche de conception s'étale sur tout le cycle de vie du projet. Pourtant, c'est surtout au cours des deuxième et troisième étapes qu'ils interviennent de manière capitale. En effet, la dernière étape est plus axée sur la conception détaillée et sur la mise au propre, et la première est plus une étape de cogitation, de "digestion" du problème, de préparation à la conception proprement dite.

Remarque : l'assistance par ordinateur de cette étape de préparation peut également être sérieusement

envisagée (il s'agirait de PCAA0 ! Préparation à la Conception Architecturale Assistée par Ordinateur !).

On peut très bien imaginer un éditeur

d'organigramme par exemple, qui proposerait toute une série d'opérations pour manipuler directement les organigrammes. C'est ainsi que l'utilisateur pourrait créer un organigramme, ajouter une pièce, modifier une surface, supprimer une relation, etc... .

Mieux, cet éditeur pourrait être un peu plus intelligent et proposer lui-même les différents organigrammes correspondants aux données introduites sous forme textuelle.

Pris isolément, un tel éditeur serait déjà très intéressant, mais si on le couplait à d'autres outils de conception plus évolués (comme celui que nous nous proposons de réaliser), on obtiendrait un logiciel couvrant alors la quasi totalité du cycle de vie d'un projet, sorte d'"intégré" du domaine architectural.

Cette remarque laisse entrevoir les progrès qui peuvent encore être réalisés. Pour notre part, nous nous limiterons à l'assistance de la conception plus avancée, chose suffisamment nouvelle pour ne pas brûler les étapes.

En quoi les supports de conception diffèrent-ils des supports de communication ? Bien plus que l'aspect ou la technique graphique utilisée, c'est leur rôle qui les distingue fondamentalement. En effet, si les premiers servent de document de base pour la compréhension du projet par des personnes non impliquées dans sa conception, les seconds jouent un rôle absolument différent. Bien sûr, ils assument une certaine communication, mais celle-ci est d'un genre un peu particulier, puisque :

- 1) elle s'établit entre une seule et même personne à des époques différentes,
- 2) les documents épaulant cette communication référencent un projet qui est loin d'être définitif.

Ces deux caractéristiques expliquent sans doute pourquoi les qualités des supports de communication (précision et complétude pour l'entrepreneur, réalisme et lisibilité pour le client) ne sont plus requises ici. En effet, on comprend aisément qu'il est tout à fait inutile que ces supports de conception soient précis et encore moins complets, étant donné le caractère provisoire du projet. De même, un réalisme exemplaire est absolument superflu, puisque ces documents restent entre les mains du concepteur, qui n'éprouve comme nous l'avons déjà dit aucune difficulté particulière pour les comprendre.

Quel est alors le rôle des supports de conception ? On peut affirmer que leur rôle essentiel est l'archivage des idées conceptuelles. Cela signifie que chaque document produit par l'architecte au cours de la conception sera gardé précieusement, même s'il est très sommaire ou très "brouillon". De cette manière, il pourra retrouver facilement les idées rassemblées dans le document, et les comparer avec d'autres. Le résultat de cette comparaison conduira à une nouvelle évolution du projet à partir de ces idées, à l'abandon de toute solution proche de celles-ci, ou tout simplement au report de toute décision les concernant.

Le corollaire de la fonction d'archivage d'un support de conception est la possibilité d'offrir au concepteur une vue du projet moins nébuleuse que celle qui est issue directement de son imagination.

Un support de conception sert donc à archiver une idée conceptuelle. Mais il faut encore remarquer que cet archivage est extrêmement rapide, car le concepteur ne tient pas à freiner sa créativité afin d'élaborer minutieusement chaque support de conception. Cette notion de rapidité est vraiment capitale. En effet, plus un outil de production de supports de conception est rapide, plus le concepteur a de chances d'aboutir à de nombreuses idées, et donc plus les choix qu'il fera seront de qualité. Attention, nous sommes parfaitement conscients qu'il ne suffit pas de doter un concepteur d'un outil de CAAO très rapide, pour que son rendement créatif augmente de manière proportionnelle, mais la rapidité de cet outil nous garantira que toute source de ralentissement ne viendra que du concepteur et que de lui seul.

Signalons enfin que la procédure de classement des divers supports de conception est rapide elle aussi. Elle peut être assimilée à la confection d'une pile sur laquelle viennent se superposer les documents à mesure que le travail de conception avance.

### III.3 Les propriétés d'un support de conception

---

Tâchons maintenant de déterminer les propriétés que doit avoir un support de conception pour remplir correctement son rôle d'archivage rapide. Pour ce faire, nous allons tenter d'affiner encore l'analyse de la démarche de conception, ce qui va nous permettre de dégager deux particularités.

- 1) Pour concevoir ses bâtiments, l'architecte travaille en trois dimensions. Cela ne veut pas dire qu'il travaille

obligatoirement en perspective sur le support de conception, mais plutôt que les images mentales qu'il se fait du projet sont plutôt volumétriques, que ce sont des volumes qui se forment et se succèdent dans son esprit avant d'être matérialisés sous une forme quelconque sur un support physique.

Malheureusement, rares sont les supports physiques qui préservent le caractère volumétrique du projet, ce qui oblige les architectes à se rabattre sur des supports plats (traditionnellement la feuille de papier), et sur des techniques de représentation appropriées à ces supports. Nous avons vu que la technique du croquis est souvent utilisée au stade de la conception. C'est assez logique, car la rapidité de l'archivage est assurée par l'absence de conversion entre la vision mentale du projet et celle offerte par le croquis, ce dernier n'étant qu'un aplatissage direct de l'espace.

Si le croquis est une technique qui permet au support de conception d'assumer son rôle, il a quand même des inconvénients. Le principal est évidemment la difficulté que représente l'aplatissage de l'espace sur une feuille de papier. Or, si un architecte ne maîtrise pas bien la technique du croquis, il est contraint de se replier sur les techniques de représentation 2D, plus faciles mais moins appropriées à la conception.

Il s'agira donc pour nous de développer un outil informatique qui reprendra les avantages du croquis traditionnel, tout en évitant évidemment ses inconvénients.

- 2) Le croquis est encore utilisé pour une autre raison, peut-être moins triviale que la précédente. Il permet en effet de préserver un certain degré d'imprécision. Cette imprécision n'est pas uniquement une conséquence de la rapidité, mais constitue à elle seule un critère de conception primordial. Cette affirmation, assez déroutante peut-être, mérite plus d'explications.

La démarche architecturale n'est pas une évolution constante dans le temps permettant de passer de l'imprécis au précis. Pour employer un langage d'informaticien, elle n'est pas une démarche "top-down". Elle ne se résume donc pas à la production successive de dessins, chaque dessin apportant une solution définitive à des problèmes de plus en plus précis. Au contraire, il s'agit de rechercher et d'explorer les différentes voies qui mèneront au projet final, sans s'attacher à une solution possible. Cette recherche ne peut se faire qu'à l'aide d'esquisses très floues, où les traits traduisent plus les intentions, les pulsions, les

intuitions, que n'importe quelle prise de position concernant le projet final. Comme le dit J-C BURDESE dans [2] : "En fait, il s'agit de retarder "l'arrêt sur image", et de pousser l'abstraction le plus loin possible."

- Remarque : 1) Si l'imprécision est capitale en ce qui concerne les éléments constructifs, elle l'est tout autant en ce qui concerne les dimensions. Par exemple, un trait peut être tracé juste pour signaler qu'il existe une séparation entre deux espaces, mais la position de cette séparation peut varier, son épaisseur également, et elle peut très bien ne pas être rectiligne comme l'indique le trait). Cette remarque augure déjà les difficultés que nous aurons pour coupler un travail imprécis avec une machine de par construction extrêmement précise.
- 2) Il ne suffit pas d'être convaincu de l'importance de l'imprécision lors de la conception, il faut de plus éviter tout penchant pour l'hyper-réalisme, qui aurait tendance à entraîner des esquisses soignées et complètes. Il est effectivement faux de croire que l'adage "qui peut le plus peut le moins" se vérifie ici, que l'imprécision peut être tout à fait théorique, qu'il est possible de dessiner de manière précise, et de supposer que cela ne l'est pas. Il faudra pourtant simuler cette imprécision, ce qui est quand même mieux que de l'ignorer ou de la supposer.

#### III.4 Conclusions

-----

Le but de cette analyse était de dégrossir sommairement le processus de conception architecturale, afin de développer un nouvel outil mieux adapté à ce processus. On s'écarte ainsi des tendances qui auraient la prétention de faire fléchir l'acte architectural aux exigences des outils existants.

Il est évident qu'un processus aussi complexe que la conception architecturale n'a pu être que brossé rapidement, ce qui n'empêche pas de tirer quelques conclusions immédiates, dont il faudra tenir compte lors des spécifications du nouvel outil :

- \* la raison d'être des supports de conception l'archivage des idées conceptuelles,
- \* cet archivage doit être (très) rapide.

- \* la conception proprement dite s'effectue en trois dimensions,
- \* l'imprécision est primordiale afin de rester le plus longtemps possible dans l'abstraction.

Passons maintenant aux spécifications de l'outil d'aide à la conception architecturale, que nous appellerons dorénavant l'éditeur de volumes pour des raisons qui deviendront bien vite évidentes. Quant à l'outil classique de CAO impérativement couplé à l'éditeur de volumes, nous l'appellerons l'éditeur de composants constructifs.

DEUXIEME PARTIE :

ELABORATION D'UNE SOLUTION

## CHAPITRE IV : La notion de rapidité

---

Dans l'analyse précédente, on déduit que le rôle d'un logiciel de CAAO est de produire des supports de conception. Pour que notre éditeur de volumes puisse remplir efficacement ce rôle, il devra être un outil d'archivage rapide des idées conceptuelles, plutôt qu'un outil d'assistance au dessin, aussi performant soit-il. C'est pourquoi il ne peut freiner en aucune façon la productivité créatrice de l'utilisateur. Il convient alors d'attacher une importance toute particulière à la notion de rapidité, ce qui implique la rapidité des temps de réponses, la convivialité, et la rapidité d'utilisation.

### IV.1 La rapidité des temps de réponses

---

Ceux-ci doivent évidemment être extrêmement courts, pour que le concepteur puisse suivre le fil de ses idées sans devoir attendre régulièrement, ce qui nuirait sans aucun doute à son travail de conception.

S'il est évident que les temps de réponses doivent être "fulgurants", il faut remarquer que cela peut influencer le choix des différents algorithmes lors de l'implémentation, mais surtout celui du matériel sur lequel sera installé le logiciel. On peut cependant imaginer que certaines fonctions, gourmandes en calculs, ne soient disponibles que si la configuration est suffisamment puissante.

### IV.2 La convivialité

---

Dans le même ordre d'idée, il faut impérativement que notre éditeur soit très facile à utiliser. En d'autres termes, il faut qu'il soit convivial. Il ne s'agit pas en effet que le concepteur doive s'encombrer l'esprit de procédures à suivre pour l'utiliser, ce qui détournerait à un moment donné son attention du travail de création, pour des problèmes de manipulation. Comme le dit M. LEGLISE dans

[2], "Il ne s'agit plus de maîtriser les outils, il faut au contraire que les outils soient suffisamment souples pour se laisser oublier et que la seule difficulté de la tâche reste ce qui est vraiment difficile : concevoir".

Cette caractéristique nous amènera probablement à concevoir une interface utilisateur basée sur menus et souris (ce qui devient d'ailleurs un standard dans le domaine de la CAO). Une autre conséquence est l'uniformité que doivent avoir les commandes au travers des différentes situations. Cela veut dire que si une commande a une signification dans un cas, l'utilisateur doit pouvoir s'attendre à ce qu'elle ait la même signification, ou qu'elle n'ait éventuellement pas d'effet, dans un autre cas. Le problème de la convivialité a donc de nombreuses implications, qui seront mises en évidence au fur et à mesure de l'élaboration du logiciel.

#### **IV.3 La rapidité d'utilisation**

-----

La convivialité permet d'utiliser facilement toutes les possibilités du logiciel. Mais cela ne suffit malheureusement pas pour éviter les interruptions étrangères au travail de conception. Il faut pour cela qu'elles puissent être mises en oeuvre très rapidement, c'est-à-dire après un nombre restreint d'opérations successives. Ici encore, contentons-nous de mettre en évidence le problème, et tâchons d'en tenir ultérieurement.

## CHAPITRE V : Propriétés de la représentation

---

### V.1 Le mode de représentation

---

Une des toutes premières questions à se poser concerne le mode de représentation qui sera utilisé par l'éditeur pour introduire les éléments. Doit-il proposer une représentation en plan - coupe - élévation, en axonométrie, en perspective réelle, ou bien plusieurs d'entre elles ?

On a vu précédemment que le concepteur imaginait mentalement son projet en trois dimensions. Il est donc tout naturel que l'éditeur propose un mode de représentation approprié à cette caractéristique de conception. Nous retiendrons la technique des perspectives, en précisant toutefois qu'elles doivent jouer le rôle qu'ont les croquis traditionnels. En effet, ceux-ci pourtant très utiles lors de la conception, ne sont pas utilisés de manière unanime, notamment à cause de la difficulté d'aplatir directement l'espace, sans aucun système fiable pour confirmer ou infirmer la justesse de la représentation. Nous ne pouvons donc pas informatiser telle quelle cette technique, au risque d'obtenir un outil où l'écran aurait simplement remplacé le papier, et où les angoisses du mauvais dessinateur seraient maintenues, peut-être même aggravées par l'utilisation d'une machine. Par conséquent, pour offrir une vision spatiale du projet tout en évitant de laisser à l'utilisateur l'initiative de la construction de l'espace, nous ferons appel aux techniques des perspectives automatiquement calculées par ordinateur.

Remarque : le croquis a une deuxième caractéristique non négligeable, qui est la possibilité de préserver l'imprécision inhérente aux premiers stades de conception. Nous solutionnerons ce problème plus tard.

Avant de préciser encore le mode de représentation (puisque la technique des perspectives en rassemble en réalité plusieurs), nous pouvons déjà écarter catégoriquement toutes les techniques 2D. En effet, si notre approche de la conception architecturale est correcte, ou du moins suffisamment proche de la réalité, on peut affirmer que l'emploi des représentations en deux dimensions lors de la

conception est dû à la manipulation difficile du croquis, et à l'absence de toute autre technique 3D plus facile à maîtriser. Comme notre éditeur est sensé combler cette lacune, poussons le raisonnement jusqu'au bout et admettons que les techniques 2D deviennent inutiles. Non seulement elles sont inutiles au sein d'un outil de CAAO, mais elles sont également néfastes ! En effet, on peut très bien imaginer qu'un architecte "intoxiqué" par les représentations planes pendant de nombreuses années de pratique, se raccroche désespérément à celles-ci s'il en a la possibilité. Cet entêtement serait moins dû à une éventuelle inadéquation de l'outil avec le travail de conception, qu'à l'habitude de dessiner en deux dimensions qu'il devrait abandonner, et on sait combien les habitudes sont tenaces (même les mauvaises).

Les techniques 2D ont été définitivement écartées pour les raisons évoquées ci-dessus. Qu'en est-il des différents modes de représentation 3D ?

#### V.1.1 L'axonométrie

Commençons par l'axonométrie, qui malgré son côté théorique semble être une excellente technique de visualisation d'un projet en cours d'élaboration. En effet, les caractéristiques de l'axonométrie jouant en sa faveur sont les suivantes :

- \* deux droites parallèles dans l'espace restent parallèles dans la représentation de cet espace puisque les points de fuite sont inexistant. Cette propriété permet d'estimer beaucoup plus facilement si des droites ou des plans sont parallèles ou non, tandis qu'il est plus hasardeux d'estimer la même chose dans une perspective réelle (voir figure 9). C'est très utile lors de la conception d'un bâtiment qui est bien souvent guidée par un réseau imaginaire de droites et de plans parallèles.

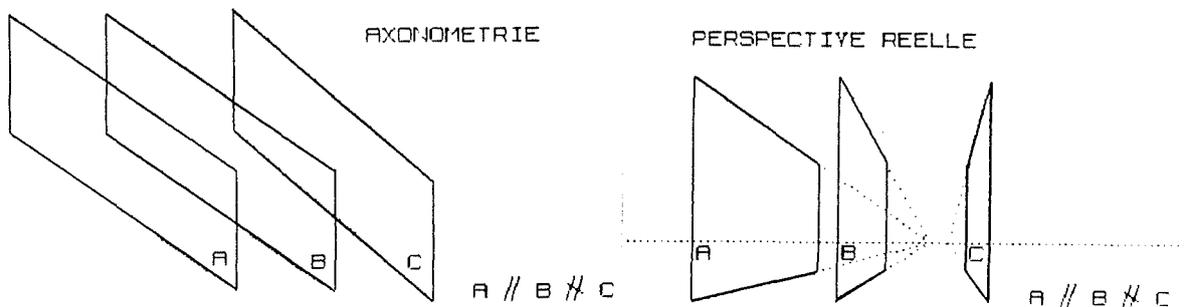


Figure 9

- \* les proportions entre les différents éléments de l'espace sont préservées dans la représentation. C'est ainsi par exemple que la représentation du milieu d'une droite sera positionnée au milieu de la représentation de cette droite. Quand on sait l'importance que les architectes attachent aux proportions, on comprend l'intérêt de cette propriété.

Ces deux caractéristiques réunies font qu'il est plus aisé de se positionner par rapport aux autres éléments dans une axonométrie plutôt que dans une perspective réelle.

### V.1.2 La perspective réelle

La perspective réelle convient parfaitement pour visionner l'espace, mais quand il faut agir sur cet espace, ses avantages (réalisme) s'effacent rapidement devant ses inconvénients (approximation du positionnement). Doit-on pour autant l'éliminer de l'éditeur de volumes ? Pas forcément, car elle ne remet pas en question la cohérence de notre discours concernant la conception architecturale, comme l'aurait fait l'adoption d'une technique 2D.

La perspective réelle sera donc présente dans l'éditeur de volumes. Il faut maintenant décider si elle permettra à l'utilisateur de voir uniquement son projet, ou si elle lui permettra d'y travailler au même titre que l'axonométrie. Ce que l'on peut dire à ce stade de l'élaboration du logiciel, c'est qu'il serait un peu bête de disposer d'une technique de représentation spatiale, calculée par l'ordinateur, mais inopérante lors des modifications de l'espace sous prétexte que l'axonométrie est apparemment plus adaptée pour ce genre d'action. Il se peut d'ailleurs qu'un individu porte sa préférence sur l'axonométrie, alors qu'un autre choisisse lui pour diverses raisons de travailler directement en perspective réelle. Il se peut aussi qu'une seule et même personne, au départ plus à l'aise dans une axonométrie, acquière progressivement une certaine expérience dans le travail en perspective réelle, et ne choisisse plus alors que ce type de représentation encore plus directe par rapport à la vision mentale.

### V.1.3 La perspective cavalière

La perspective cavalière est un mode de représentation en trois dimensions, assez proche de la technique de l'axonométrie, mais qui a la particularité de laisser dans l'image de l'espace deux axes perpendiculaires. Cette représentation fort théorique présente les mêmes avantages que l'axonométrie pour le travail en trois dimensions, qui peut même encore être facilité par la sauvegarde partielle des angles droits. Cette technique sera également proposée

par l'éditeur de volumes.

Nous avons donc sélectionné trois modes de représentation : l'axonométrie, la perspective cavalière, et la perspective réelle. Il faut maintenant spécifier d'avantage la manière de représenter l'espace au travers de ces trois techniques, notamment en ce qui concerne le dessin même des axes et des volumes.

## V.2 La représentation des axes

-----

Voyons ce que peut apporter la représentation du système d'axes. Il est évident qu'au départ, quand encore aucun volume n'est représenté, la visualisation des trois axes permet à l'utilisateur de se faire une idée de l'image qu'il aura d'un volume, avant même qu'il l'introduise. Cet aspect des choses est très important, car l'effet de surprise que peut causer l'apparition de ce volume à l'écran diminue, le temps que prend son ajustement en position et en dimension est réduit également, et par conséquent la rapidité d'utilisation est assurée.

Bien-sûr, il suffit d'introduire quelques volumes pour diminuer l'importance du système d'axes, car ce sont alors les arêtes mêmes de ces nouveaux volumes qui servent de directions de référence, pourvu qu'elles soient de dimension suffisante. Mais même dans ce cas, l'origine peut toujours servir de position de base à partir de laquelle tout le projet s'articule.

Ces différentes remarques nous poussent à opter pour le dessin automatique des axes dès la spécification complète des paramètres de visualisation (mode de représentation, angle que font les axes pour l'axonométrie, position de l'oeil pour la perspective réelle, etc...).

Ce dessin des axes doit pouvoir être annulé sur demande explicite de l'utilisateur, quand ils ne font plus que surcharger la perception du projet. D'autre part, cette annulation ne doit pas être définitive, car on peut très bien imaginer des situations où l'utilisateur désire alternativement voir (pour repérer les directions de référence) et ne plus voir (pour clarifier la vue) les axes.

Si la représentation du système d'axes constitue une aide non négligeable dans une axonométrie ou dans une perspective cavalière, il n'en va pas de même dans une perspective réelle, où les trois axes seuls n'ont plus assez de signification pour guider efficacement l'utilisateur dans la perception de l'espace. En fait, la même représentation

d'un système d'axe peut correspondre à plusieurs perspectives différentes (voir figure 10). C'est pour cette raison que nous ajouterons au dessin des axes celui de la ligne d'horizon. Elle permet effectivement de situer le système d'axes dans l'espace, en facilitant le repérage des deux points de fuite principaux. Ceux-ci sont situés à l'intersection de la ligne d'horizon avec les deux axes horizontaux (voir figure 11). Signalons encore que cette ligne sera toujours représentée dans une perspective, même quand les axes ne le seront pas.

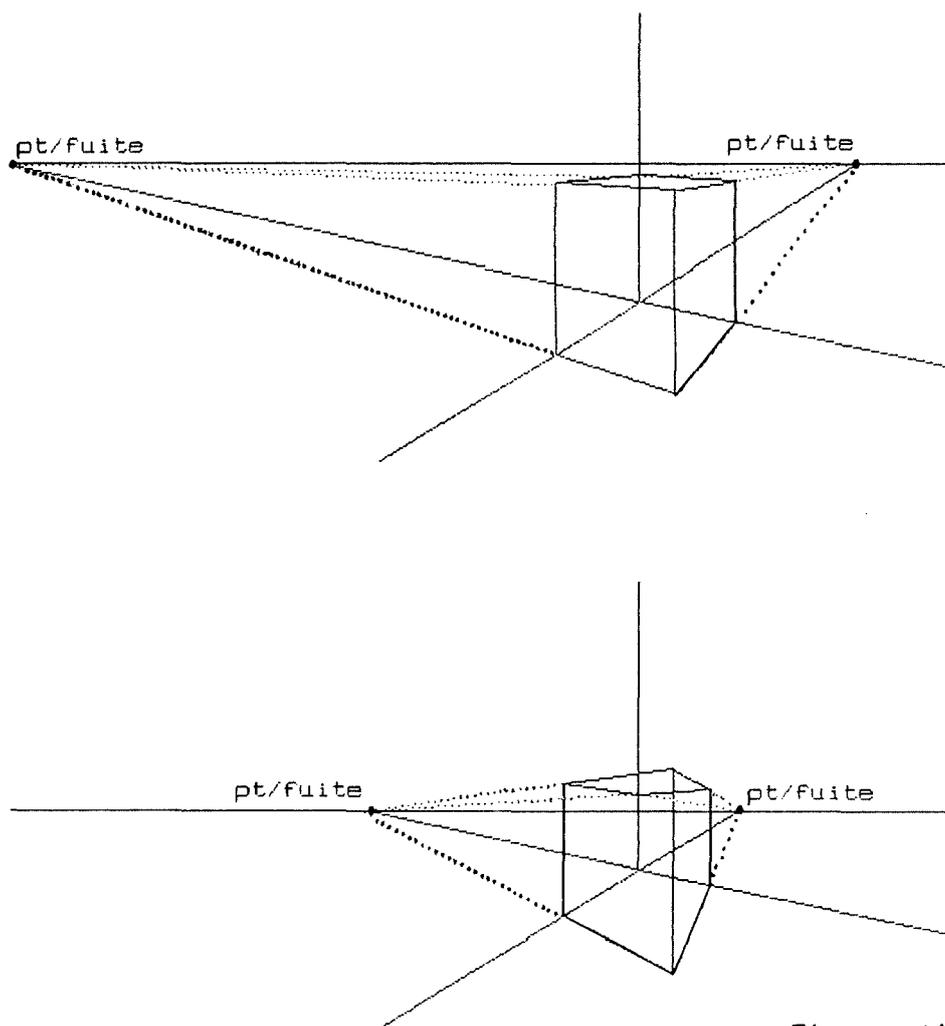
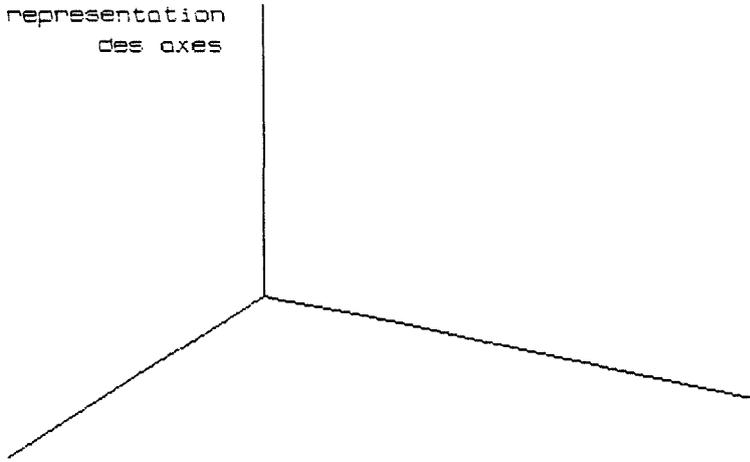


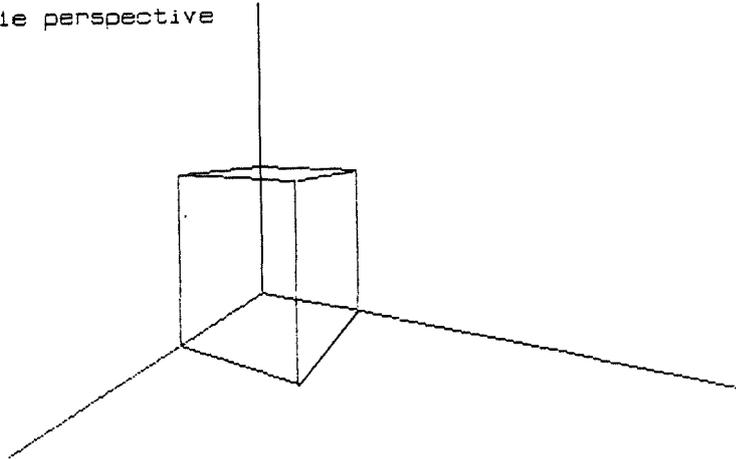
Figure 11

Déplacer le système d'axes, c'est-à-dire déplacer l'origine sans changer les caractéristiques de la vue, peut s'avérer très utile, non pas pour se positionner en fonction des autres volumes, mais pour adapter la position de l'origine à un projet qui se serait développé de manière déséquilibrée par rapport à celle-ci.

representation  
des axes



1e perspective



2e perspective

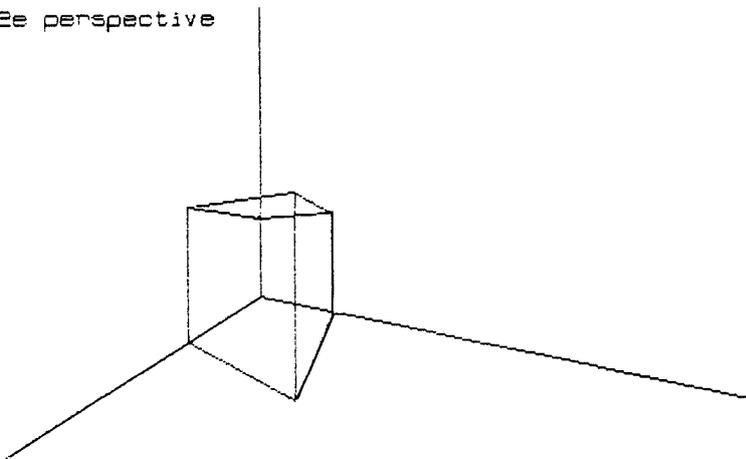


Figure 10

Remarque : il faut bien comprendre que le rôle des axes n'est pas de fournir une mini grille facilitant le positionnement. Les trois axes seraient d'ailleurs tout à fait insuffisants pour apporter efficacement une telle aide. Nous trouverons bien en temps utiles de réels outils d'aide au positionnement, sans devoir y incorporer le dessin du système d'axes.

Une autre caractéristique concernant les axes, et sur laquelle l'utilisateur peut vouloir agir, est la partie représentée. Comme nous avons trois axes pouvant être dessinés de trois manières différentes (+, -, + et -), cela nous donne  $3^3$  c'est-à-dire 27 possibilités parmi lesquelles l'utilisateur choisira celle qui est la plus adaptée à son projet. On peut déjà décider que les parties positives seront dessinées par défaut (voir figure 10).

Nous avons décidé que les axes étaient dessinés automatiquement dès la spécification complète des caractéristiques de visualisation. Mais il serait peut-être préférable que ce soient eux qui aident l'utilisateur à définir interactivement la vue qu'il souhaite avoir de l'espace. Pour ce faire, chaque mode de représentation aura ses paramètres de visualisation par défaut, pour que les axes puissent apparaître à l'écran dès sa sélection. L'utilisateur peut alors modifier ces paramètres en constatant directement sur l'écran les conséquences que cela entraîne. De manière plus précise :

- \* après sélection de l'axonométrie, l'utilisateur peut "draguer" chaque axe avec la souris, et le faire tourner ainsi autour de l'origine. Les angles interdits seront rejetés par le programme. De plus, il pourra également "draguer" un point quelconque de l'écran pour le déplacer sur la représentation de l'espace.
- \* Pour la perspective réelle, le changement de vue est effectué par le déplacement de l'oeil dans l'espace (déplacement similaire à celui du curseur que l'on verra plus tard), et par la rotation de la tête, c'est-à-dire par le déplacement du point visé. Notons que déplacer l'oeil dans l'espace n'équivaut pas à déplacer un point quelconque qui deviendra l'oeil par la suite. En fait, le moindre déplacement de l'oeil doit entraîner le dessin d'une nouvelle perspective, dans laquelle se fera le déplacement suivant. La remarque est bien sûr valable pour la rotation de la tête.
- \* Dans le cas de la perspective cavalière, le problème est simplifié puisqu'il suffit de désigner les deux axes qui restent perpendiculaires par clicks successifs sur la commande adéquate, et de déterminer ensuite l'angle que fait le troisième, en le draguant de la même manière que dans l'axonométrie. Le déplacement de l'écran sur la vue est analogue à celui de l'axonométrie.

Remarque : pour simplifier les choses, les axes perpendiculaires resteront parallèles aux côtés de l'écran (voir figure 12) ce qui nous donne trois cas possibles.

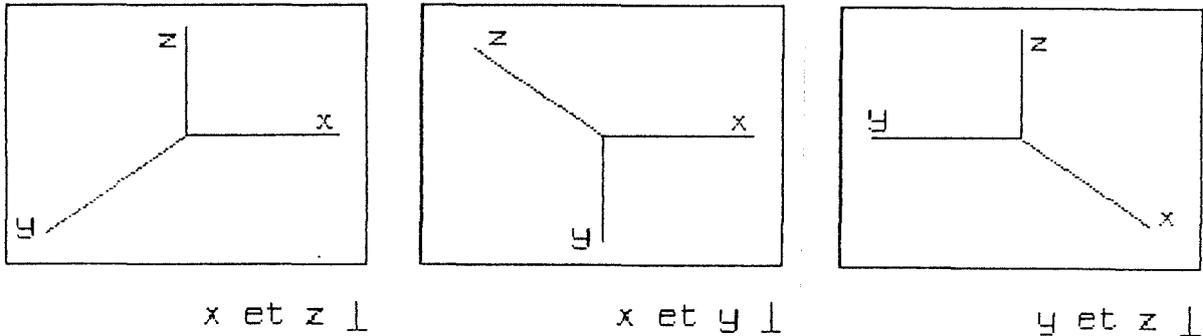


Figure 12

Ajoutons que ces techniques de modification de la vue sont intéressantes lors de sa spécification initiale, mais elles peuvent l'être aussi lors de tout changement ultérieur des paramètres de visualisation. Malheureusement, on se doute bien que plus l'espace est encombré de volumes, plus le temps de calcul que demandera le moindre déplacement sera important, et que plus l'effet du "feedback" sera cassé.

Par conséquent, le "feedback" de la spécification des paramètres de visualisation ne concernera que le dessin des axes, même si ces paramètres sont modifiés en cours de conception. Dans ce cas, le dessin de la nouvelle vue ne sera lancé qu'après confirmation des modifications. On peut tout de même remarquer que cette contrainte technique reste relativement logique, du moins en ce qui concerne la perspective réelle. En effet, si les volumes composant le projet ont été introduits en nombre suffisant, il est assez normal de désigner l'oeil et la cible d'une nouvelle perspective en fonction de tel ou de tel autre point précis du projet, alors que cela n'a aucun sens quand l'espace est vide, c'est-à-dire à la spécification initiale.

Remarque : introduire au clavier les paramètres de représentation sous forme numérique n'a pas de sens, car comme nous le verrons plus tard, les notions de dimensions, de coordonnées, ou encore d'échelle, ne seront pas obligatoirement présentes.

### V.3 La représentation des volumes

---

Sans entrer dans les détails, on peut dire que les méthodes de calcul des perspectives permettent de dessiner les volumes de trois manières différentes ; en filaire, par vues et cachées, et par aplats de couleur. Ici aussi, il faudra choisir une ou plusieurs de ces trois présentations.

Remarquons tout d'abord que c'est le manque d'information qui paradoxalement permet à un individu d'interpréter plus ou moins correctement la vision qu'il a de l'espace. En effet, si dans celle-ci un volume est partiellement caché par un autre, il en déduira que le premier est derrière le deuxième.

Les techniques de dessin par vues et cachées et par aplats de couleur respectent cette caractéristique. Va-t-on pour autant les adopter et rejeter la méthode de dessin en filaire ? Rien n'est moins sûr, car rappelons-le une fois encore, ce qui compte n'est pas le réalisme du dessin, mais sa capacité à matérialiser rapidement les idées en provenance de l'imagination du concepteur, et pour répondre à ce genre d'exigence, c'est justement la représentation en filaire qui plus appropriée. En effet, il y a de fortes chances qu'un nouveau volume soit partiellement caché par d'autres éléments déjà existants du projet. Pourtant, il est bien évident que le contrôle visuel doit être maintenu sur ce nouveau volume lors de son introduction, et ce malgré les parties cachées. C'est pour cette raison que la présentation des volumes en filaire sera retenue pour le travail de conception.

Les deux autres présentations seront-elles abandonnées ? A priori oui, car d'une part, si elles facilitent l'interprétation de l'image, le manque d'information qu'elles entraînent est vraiment un handicap quand il faut agir sur l'espace, et d'autre part, le nombre plus important de calculs qu'elles réclament freinent de manière considérable la vitesse de l'affichage d'un projet. De plus, comme nous l'avons déjà signalé à plusieurs reprises, trop de réalisme peut avoir un effet néfaste sur la démarche architecturale, l'utilisateur étant fortement tenté de peaufiner chaque dessin, donc de le considérer plus comme fin en soi que comme élément de transition.

Cependant, un problème peut surgir avec la présentation des volumes en filaire. Si le projet est complexe et comporte de nombreux volumes imbriqués, il se peut que le "fouillis" de lignes apparaissant à l'écran devienne vite indéchiffrable, même par le concepteur. Par conséquent, nous admettrons quand même un mode de présentation plus réaliste. La préférence ira au dessin en vues et cachées, parce qu'il

ressemble assez au dessin filaire, et parce qu'il s'accorde à un système monochrome.

Remarque : n'oublions pas que l'éditeur de volumes doit faire partie d'un logiciel plus complet comprenant sans doute une fonction de génération de perspectives. Donc, si l'utilisateur désire une vue plus réaliste du bâtiment au cours de la conception, ou même pour l'élaboration de l'avant-projet, il peut toujours faire appel à cette fonction, qui ne lui permettra toutefois pas d'agir sur l'espace.

Pour améliorer encore la lisibilité d'un dessin filaire, il serait encore intéressant que l'utilisateur puisse sélectionner la partie du projet à visualiser. Il ne s'agirait donc plus d'effacer les arêtes cachées, mais une partie entière du projet, sur laquelle le concepteur ne travaille momentanément pas. Pour se faire, il devra dessiner un parallélépipède rectangle un peu particulier (on verra plus tard comment), en dehors duquel tous les traits seront invisibles.

Nous voilà donc prêts à dessiner, mais encore faut-il pouvoir se déplacer aisément dans la représentation de l'espace. C'est ce que nous allons étudier maintenant.

## CHAPITRE VI : Le déplacement dans l'espace

---

### VI.1 La localisation du curseur dans l'espace

---

La haute convivialité d'un instrument comme la souris découle du fait que sa manipulation mimétise l'effet souhaité. C'est ainsi que si l'utilisateur veut bouger le curseur vers la haut et vers la droite par exemple, le mouvement qu'il doit imprimer à la souris est identique.

Tout cela fonctionne à merveille tant que l'on reste dans deux dimensions, mais comment faire évoluer le curseur dans l'espace ? Faut-il imaginer une nouvelle souris de type spatial ? Pas forcément ; en réalité, le curseur n'évolue pas dans l'espace, mais dans une représentation plane de celui-ci. De ce fait, un problème se pose pour localiser avec précision le curseur, car une position du curseur sur l'écran représente une infinité de points de l'espace. Par conséquent, améliorer la souris sans trouver de solution valable au problème de la localisation du curseur ne ferait que diminuer la convivialité, par le déséquilibre engendré entre la manipulation et la visualisation de l'effet. Ces considérations nous poussent donc à penser que malgré le caractère spatial d'une perspective (réelle, axonométrique, ou cavalière), elle ne reste toujours qu'une représentation plane de l'espace, et qu'il est préférable que le déplacement ne se fasse que suivant deux directions simultanées. Cela correspond de plus à la manipulation d'une souris.

### VI.2 Le plan de référence

---

Comment cela peut-il se passer concrètement ? On pourrait considérer l'écran comme un plan frontal, dans lequel le curseur se déplace à la manière d'un éditeur 2D classique. Pour pouvoir se déplacer suivant la troisième direction, on peut envisager deux possibilités. La première consiste à déplacer le plan frontal vers l'avant ou vers l'arrière, afin de couvrir le reste de l'espace. Dans la seconde il ne bouge pas, mais le curseur peut s'en écarter. Attention, pour ne pas retomber sur les problèmes de

localisation dans l'espace, le déplacement du curseur dans la troisième direction ne pourra se faire que perpendiculairement à ce plan frontal. De plus, l'écart entre le curseur et le plan sera matérialisé par une droite qui permettra d'estimer la grandeur de cet écart, de situer la projection orthogonale du curseur sur le plan frontal, et enfin de marquer l'interdépendance du curseur avec le plan. Il est évident que le curseur écarté du plan pourra toujours se déplacer parallèlement à ce dernier avec les mouvements classiques de la souris. Comme ces deux possibilités sont complémentaires, elles seront toutes deux présentes au sein de notre éditeur de volumes.

Si on applique cette solution telle quelle, les déplacements du plan frontal ou du curseur en dehors de ce plan seront fréquents, car on a considéré l'écran comme plan de référence, alors que la plupart des points se positionnent par rapport au système d'axes. Pour éviter ce déphasage, le plan de référence ne sera plus assimilé à l'écran, mais bien à un véritable plan frontal dans l'espace, c'est-à-dire parallèle au plan formé par les axes X et Z du système d'axes (voir figure 13-a).

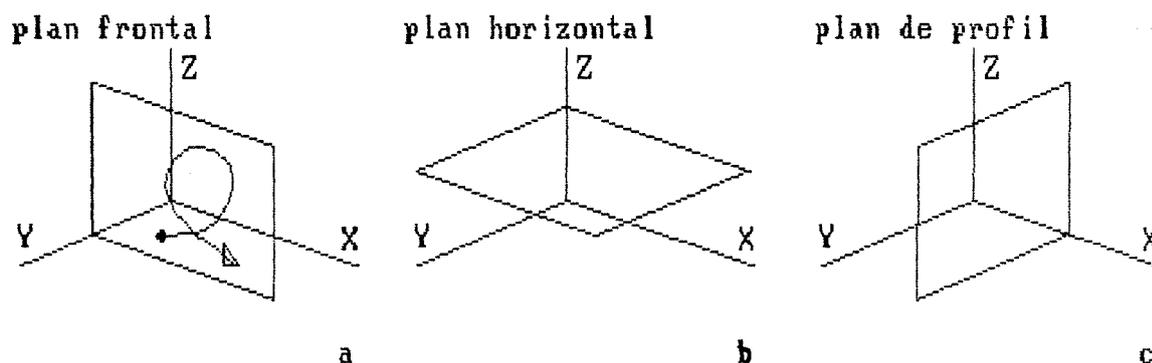


Figure 13

Dans ce cas, la technique de déplacement du curseur peut être généralisée à d'autres plans de l'espace, comme le plan horizontal, c'est-à-dire parallèle à XY (voir figure 13-b), et comme le plan de profil, c'est-à-dire parallèle à YZ (voir figure 13-c).

L'utilisateur ne devra pas forcément décider une fois pour toutes avant la conception d'un bâtiment avec lequel des trois plans de référence il compte travailler. En effet, il aura la possibilité de passer d'un type de plan à un autre, ce qui peut même être assimilé à une troisième méthode de déplacement dans l'espace, si ce passage est très rapide et très facile.

Une commande supplémentaire permettra à l'utilisateur de limiter les déplacements du curseur aux directions des axes. Comme son déplacement perpendiculaire au plan de référence suit déjà une telle direction, la commande n'influencera que ses mouvements au sein de ce plan, et aura pour effet de ne prendre en considération que la direction générale du mouvement de la souris (voir figure 14). Elle pourra aussi être activée par la pression d'une touche pendant la manipulation de la souris, qui pourrait être la touche <CTRL>.

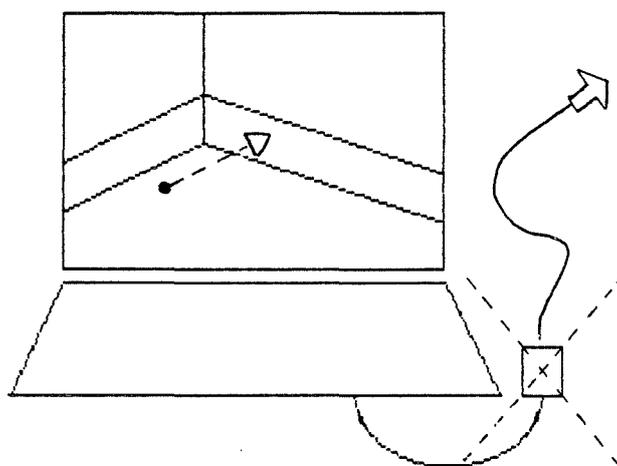


Figure 14

### VI.3 le déplacement du curseur

-----

Il faut peut-être insister maintenant sur l'importance du choix des commandes pour passer d'un plan de référence à l'autre, pour translater ce plan, ou pour en écarter le curseur. Des commandes mal choisies pour ces trois fonctions risquent fort de compromettre toutes nos espérances concernant le travail en trois dimensions, et par la même occasion, l'existence de l'éditeur de volumes tout entier.

#### VI.3.1 Le déplacement perpendiculaire

Commençons par la commande permettant au curseur d'évoluer perpendiculairement au plan de référence. Comme la souris ne monopolise qu'une seule main, il est tout à fait envisageable que son déplacement, combiné avec la pression d'une touche du clavier de l'autre main, entraîne le

déplacement perpendiculaire. Cette façon de faire garantit d'ailleurs une certaine rapidité d'utilisation, puisque le mouvement perpendiculaire est immédiatement accessible. Le tout est ici de bien choisir la touche à presser, car si elle est "noyée" dans le clavier, il est prévisible que l'utilisateur ne prenne pas la peine de la chercher, et choisisse un autre mode de déplacement. Les touches <ALT> ou <SHIFT> pourraient parfaitement convenir, tout comme le deuxième bouton de la souris s'il existe, mais nous verrons plus tard qu'il faudra abandonner cette dernière possibilité.

Il est évident que le mouvement de la souris doit aller dans le même sens que celui désiré pour le curseur. C'est ainsi par exemple que pour se déplacer perpendiculairement au plan horizontal, la souris sera poussée vers l'avant pour monter, et tirée vers l'arrière pour descendre (voir figure 15) ; quant aux déviations latérales, elles seront sans effet sur le curseur.

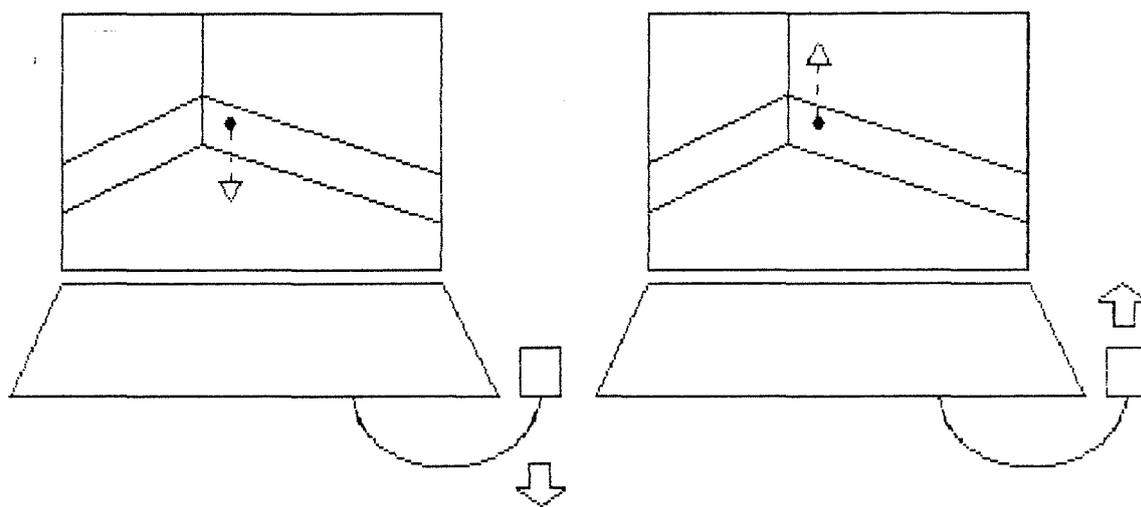


Figure 15

Remarquons qu'une commande peut être ajoutée, pour faire revenir immédiatement le curseur dans le plan de référence, sans obliger l'utilisateur à se servir de la souris.

### VI.3.2 La translation du plan de référence

Penchons-nous maintenant sur la commande qui permet la translation du plan de référence. Celle-ci peut être absolument identique à celle entraînant le déplacement perpendiculaire du curseur. La seule différence réside dans le fait qu'ici, le curseur "colle" au plan de référence, son déplacement provoque donc celui du plan tout entier. Nous allons donc réutiliser cette commande, et en ajouter une nouvelle au menu pour le "collage" ou le "décollage" du

curseur sur le plan de référence.

### VI.3.3 Le changement de plan de référence

Il reste une troisième et dernière technique de déplacement, qui consiste à passer d'un type de plan à un autre. La barre d'espacement, très facilement accessible, peut commander avantageusement cette fonction. Il suffira de la presser répétitivement pour changer de plan suivant une séquence déterminée (par exemple plan parallèle à XY, parallèle à XZ, et parallèle à YZ).

La position du nouveau plan de référence est fonction de la position du curseur lors du changement. Par exemple, admettons que le plan de référence soit un plan horizontal A. La frappe de la barre d'espacement entraînera un plan de référence frontal B, et comprenant le curseur qui n'a pas bougé (voir figure 16-a). Notons que si le curseur avait été à l'extrémité d'une droite perpendiculaire à A, l'effet de la barre d'espacement aurait été inchangé, puisque B est lui aussi perpendiculaire à A (voir figure 16-b).

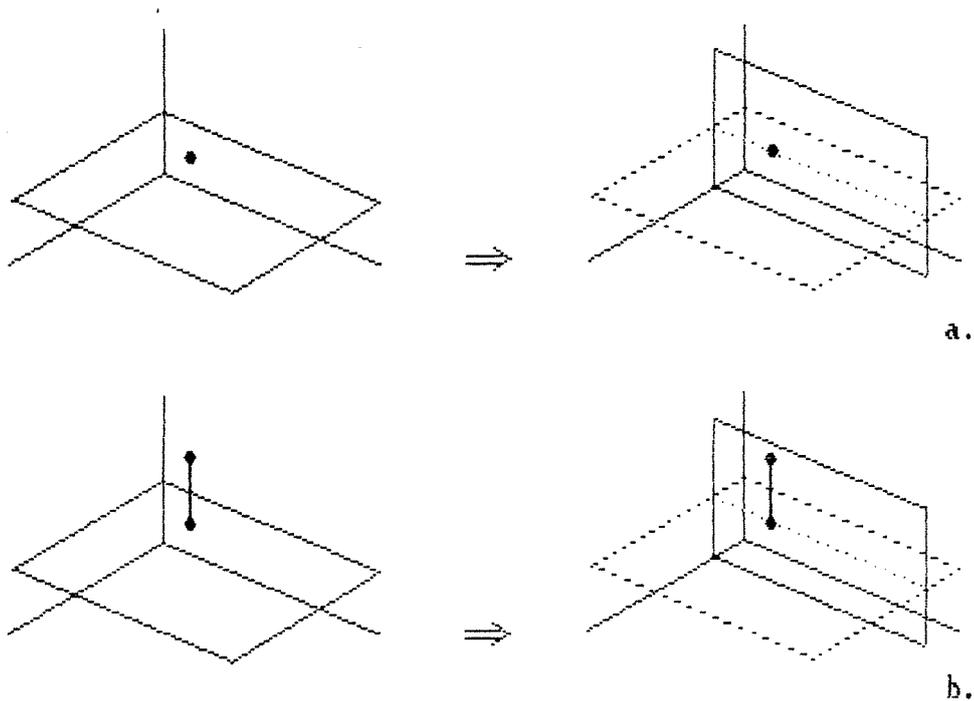


Figure 16

#### VI.4 Les représentations du plan de référence

---

Pour peu que l'utilisateur se serve de ces plans, il se rendra vite compte qu'il doit pouvoir les situer dans l'espace. La trace que laisse le plan de référence sur les parois des volumes qu'il traverse sera donc dessinée. Eventuellement, l'utilisateur peut souhaiter aussi la trace que laisse ce plan sur ceux formés par le système d'axes, afin de le situer encore plus rapidement.

Remarque: il serait assez intéressant de pouvoir distinguer à l'écran les traits de construction comme les traces que peut laisser le plan de référence, ou comme les axes eux-mêmes, des arêtes des volumes, et ce par une couleur différente, ou par une intensité ou une épaisseur différente si on est dans un système monochrome.

Cette visualisation relativement discrète peut s'avérer insuffisante dans certains cas, et doit donc pouvoir être accentuée. Cela peut se faire en tramant (couleur, hachures, pointillés, peu importe pour l'instant) les intersections plan de référence/volumes, ou inversement, en tramant tout le plan sauf ces intersections.

Remarque : si le calcul de ces intersections est activé, il se peut que le déplacement du curseur par translation du plan de référence, ou par passage à un autre, entraîne un temps de réponse trop conséquent pour les petites configurations. Dans ce cas, le premier mode de déplacement pourrait être abandonné, mais pas le deuxième, et ce pour laisser à l'utilisateur la possibilité de changer de plan de référence. Dans ce cas, le passage à un autre plan de référence ne pourrait pas servir de mode de déplacement.

#### VI.5 Le curseur et le locateur

---

Nous avons considéré jusqu'à présent que le curseur se déplaçait dans l'espace, sur le plan de référence précisément, et non sur l'écran. Cette solution peut entraîner quelques anomalies. Dans une perspective réelle par exemple, plus le curseur se rapprocherait de la ligne d'horizon, et plus son déplacement devrait être lent. De plus, comment faire comprendre au programme que le curseur doit quitter l'espace pour revenir sur l'écran pour sélectionner une icône par exemple.

Pour résoudre ce problème STAR UX a distingué deux types de curseur : le curseur, et le locateur. Le premier suit les mouvements de la souris, aussi bien au dessus des menus qu'au dessus du dessin. Le second par contre ne se déplace que sur le plan de référence de l'espace. Sa position est déduite de celle du curseur. Le calcul part des coordonnées de l'écran pour aboutir au coordonnées spatiales, et correspond de ce fait à l'inverse de ce qui se passe pour les éléments du projet. Quand sa position peut être calculée, le locateur suit (avec un petit retard) les mouvements du curseur. Quand celle-ci ne peut plus être calculée (si le curseur se trouve au dessus de la ligne d'horizon dans une perspective réelle par exemple), le locateur reste à sa dernière position.

Nous adopterons cette solution pour notre éditeur de volumes.

Remarque : nous ne distinguerons plus systématiquement les deux types de curseur dans la suite de l'exposé, étant donné qu'il sont intimement liés.

Nous pouvons représenter l'espace, nous pouvons évoluer au sein de cet espace, parlons maintenant des objets que nous manipulerons.

## CHAPITRE VII : Les objets manipulés

---

### VII.1 Les volumes

---

Comme son nom l'indique, notre éditeur manipule principalement des volumes. Ceux-ci prennent la signification que le concepteur leur attribue, c'est-à-dire qu'ils peuvent représenter des éléments constructifs (murs, briques, dalles, escaliers, etc...), les pièces d'un bâtiment, ou encore des bâtiments entiers dans le cas d'une étude urbanistique.

Voyons quels sont les différents volumes qui doivent figurer au menu de l'éditeur, et quel peut être leur mode d'introduction.

#### VII.1.1 Le parallélépipède rectangle

Il s'agit du volume de base. Grâce à lui seul, une bonne partie des projets pourra déjà être dessinée. Pour fixer sa procédure d'introduction, nous allons distinguer deux cas :

- 1) Le cas général est celui où seules les faces horizontales du volume sont parallèles à un plan du système d'axes (voir figure 17-a).

Remarque : l'introduction directe d'un parallélépipède rectangle tout à fait quelconque dans l'espace ne sera pas envisagée. Cette possibilité serait en effet trop rarement utilisée, à cause du caractère orthogonal de l'architecture. D'autre part, l'absence de ce mode d'introduction ne veut pas dire que l'utilisateur n'aura pas la possibilité de spécifier un tel parallélépipède. En effet, il pourra toujours l'introduire normalement, et l'incliner ensuite avec un outil spécialisé qui sera développé plus tard.

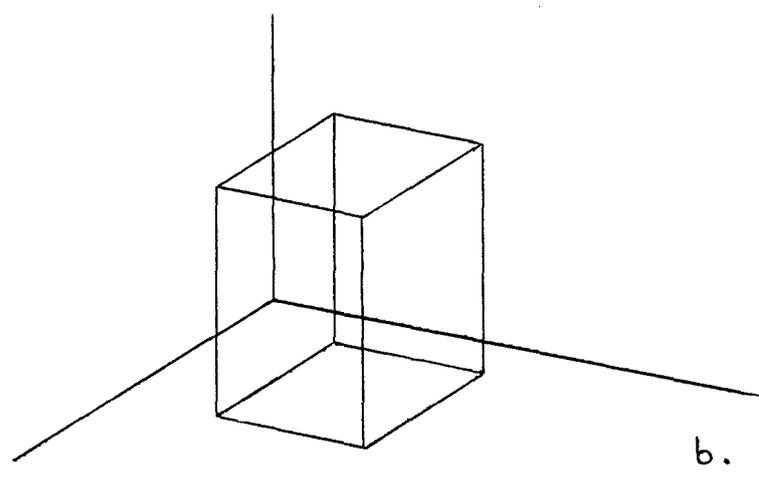
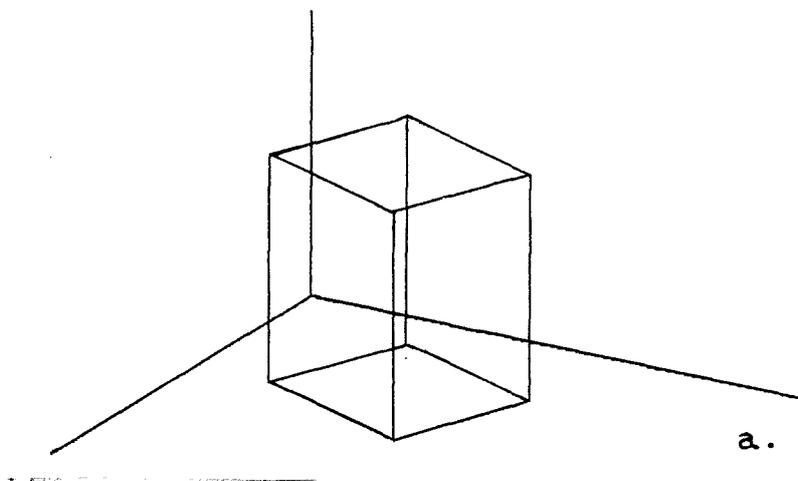


Figure 17

Pour introduire un parallélépipède sur un plan horizontal, il faut spécifier au minimum trois points :

- \* le premier sert à déterminer sa hauteur dans l'espace,
- \* le second, son orientation dans le plan horizontal et la longueur d'un côté de sa base,
- \* et le troisième, la hauteur du volume et la longueur de l'autre côté de sa base.

2) Ce cas est un cas particulier du précédent, puisque toutes les parois du volume sont ici parallèles aux plans du système d'axes (voir figure 17-b). Cette situation particulière permet de simplifier son mode d'introduction, étant donné que deux points diamétralement opposés suffisent pour le spécifier sans ambiguïté.

Notons que dans un cas comme dans l'autre, les deux premiers points doivent appartenir au même plan horizontal. Par conséquent, les déplacements de la souris allant à l'encontre de cette caractéristique lors de la spécification du deuxième point seront ignorés, ou plus exactement n'auront pas d'influence sur la hauteur de ce point.

Notons encore que si nous envisageons un mode d'introduction spécifique à chacun des deux cas, c'est pour préserver la rapidité d'utilisation. En effet, si seule l'introduction par trois points est retenue, l'utilisateur devra spécifier inutilement le troisième point dans la plupart des cas, et s'il ne dispose que de l'introduction par deux points, il devra employer de temps à autre un outil de manipulation pour donner au volume l'orientation voulue. La présence des deux modes d'introduction se justifie donc, d'autant plus que le parallélépipède rectangle sera sans aucun doute le volume le plus employé.

L'intérêt du "feedback" qu'un système peut offrir n'est plus à démontrer. Analysons donc celui de notre système lors de l'introduction d'un parallélépipède rectangle.

Commençons par la spécification par deux points. Pour le premier point, aucune aide particulière n'est attendue, il s'agit en fait de positionner un point quelconque dans l'espace. Par contre, une fois qu'il a été introduit, le volume doit se dessiner (ou plutôt se redessiner) au fur et à mesure que le curseur s'éloigne de ce premier point, exactement comme ce qui se passe dans les éditeurs 2D pour le dessin d'un rectangle.

En ce qui concerne la spécification par trois points, le principe est identique, mis à part que les deux premiers entraînent d'abord le dessin d'une droite horizontale, et ce n'est qu'à partir du troisième que le volume se dessine réellement.

### VII.1.2 Le prisme triangulaire

Le parallélépipède rectangle que nous venons d'analyser n'est qu'un cas particulier de prisme. Un autre cas particulier tout aussi intéressant en conception architecturale est le prisme triangulaire. Il permettra par exemple de dessiner en une seule opération la toiture d'un bâtiment.

Encore une fois, une méthode permettant de l'introduire directement dans une position quelconque dans l'espace ne nous intéresse pas. Nous préférons une méthode plus rapide qui permet d'exploiter les possibilités d'un plan de référence horizontal.

La création d'un prisme triangulaire exige alors la spécification de trois points :

- \* le premier permet de positionner le volume dans son environnement ;
- \* le second est le sommet de la base triangulaire faisant partie du même plan horizontal. Il permet de donner l'orientation du volume dans ce plan, et de tracer la droite joignant les deux points ;
- \* quant au troisième, il rend possible le dessin complet du prisme, puisqu'il détermine l'emplacement du dernier sommet du triangle, ainsi que de la hauteur du prisme (voir figure 18).

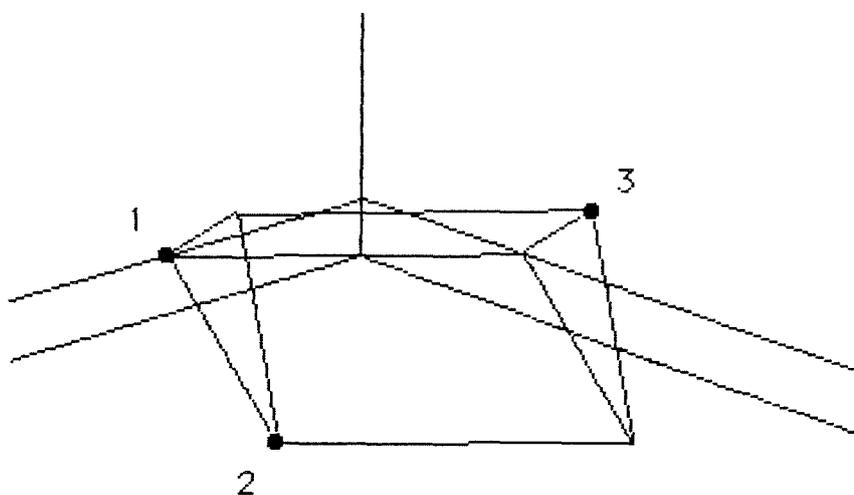


Figure 18

Cette méthode accepte évidemment les triangles quelconques en guise de base, or il serait très utile de pouvoir forcer ce triangle à être équilatéral, isocèle, ou rectangle. Dans ce cas, le troisième point ne doit pas agir sur le triangle de la base s'il est équilatéral, ou ne doit spécifier que sa hauteur s'il est isocèle ou rectangle. Les déplacements de la souris n'allant pas dans ce sens doivent donc être ignorés.

Remarquons que dans le cas du triangle rectangle, trois solutions peuvent être envisagées suivant l'emplacement de l'angle droit. Nous ne retiendrons que celle plaçant l'angle droit sur le premier sommet spécifié (voir figure 19).

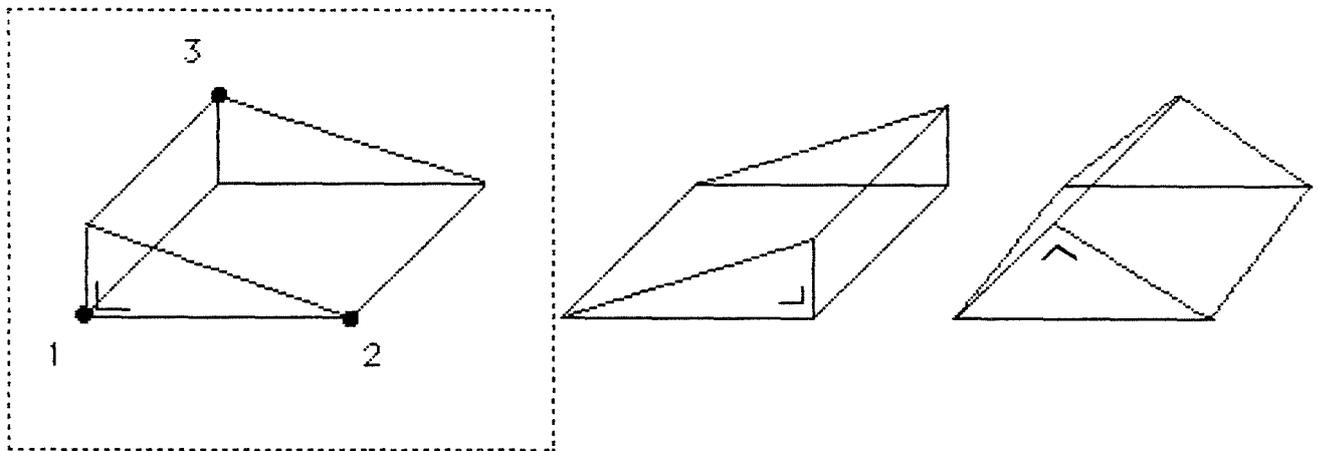


Figure 19

Signalons enfin que le développement du volume par rapport au plan horizontal, s'il peut paraître particulier (c'est en effet la base qu'on aurait tendance à placer dans ce plan), dérive de son orientation architecturale tout à fait particulière puisqu'il est essentiellement destiné au dessin des toitures.

### VII.1.3 Le prisme à bases régulières

Les bases de ce prisme peuvent être formées par n'importe quel polygone régulier, y compris le triangle équilatéral et le carré.

Pour l'introduire, il faudra avant toute chose préciser le type du polygone, c'est-à-dire le nombre de ses côtés. Ceci fait, on pourra alors spécifier deux points :

- \* un sommet de l'une des deux bases, ce qui permet de positionner le volume dans l'espace,
- \* le centre de l'autre base, ce qui achève de spécifier le polygone ainsi que la hauteur du prisme.

### VII.1.4 Le cylindre

Le prisme à bases régulières nous amène inévitablement à penser au cylindre. Celui-ci figurera donc également au menu de l'éditeur, et sa méthode d'introduction sera similaire à celle du prisme, ce qui veut dire que deux points devront être entrés, le premier appartenant à la circonférence de l'une des deux bases, et le second étant le centre de l'autre base.

### VII.1.5 La pyramide à base régulière

Pour préserver un maximum de cohérence dans l'introduction des différents volumes, celle de la pyramide à base régulière sera calquée sur celle du prisme ayant le même type de base.

Remarque : cette recherche de cohérence est importante, car nous avons le sentiment qu'elle peut jouer un rôle considérable dans la convivialité de l'outil, et donc dans sa capacité à s'effacer au profit des préoccupations purement architecturales.

L'introduction d'une pyramide se fera donc de la manière suivante :

- 1) précision du nombre de côtés,
- 2) positionnement d'un sommet de la base,
- 3) positionnement du sommet de la pyramide, ce qui achève de spécifier le polygone et la hauteur de la pyramide.

### VII.1.6 Le cône

La remarque précédente concernant la cohérence reste évidemment d'application dans ce cas-ci. L'introduction d'un cône est alors à rapprocher de celle d'un cylindre et/ou d'une pyramide à base régulière. Elle se fera de la manière suivante :

- 1) positionnement d'un point appartenant à la circonférence de la base,
- 2) positionnement du sommet, ce qui achève de spécifier le cercle formant la base ainsi que la hauteur du cône.

### VII.1.7 La sphère

La présence de ce volume dans un logiciel orienté architecture peut éventuellement étonner, mais s'explique par le fait certains éléments architecturaux classiques comme la coupole, ne peuvent être construits que très difficilement sans ce volume.

En ce qui concerne son introduction, deux points suffisent : un point quelconque de la surface et le centre.

## VII.2 Le problème de la non exhaustivité

-----

Les volumes analysés ci-dessus constituent ce que l'on appellera le "set" de base, c'est-à-dire ceux qui doivent

figurer au menu d'un éditeur de volumes digne de ce nom, sous peine de manquer cruellement tôt ou tard. La liste, on s'en doute, est loin d'être exhaustive.

Ce problème de non exhaustivité peut être résolu en donnant à l'utilisateur la possibilité de manipuler des surfaces pour créer les volumes non proposés. C'est ainsi qu'il pourra créer une pyramide à base irrégulière par exemple, en positionnant directement dans l'espace toutes ses faces.

Les volumes ne seront donc plus les seuls objets à être manipulés par l'éditeur, mais des surfaces le seront également pour la raison évoquée ci-dessus, d'autant plus qu'elles peuvent intervenir dans la conception d'un bâtiment au même titre que les volumes (pour les murs de clôtures, les balcons, etc... ).

Il est évident que l'énumération complète de toutes les surfaces possibles et imaginables est problématique, tout comme celle des volumes l'était. De nouveau, le problème sera résolu en proposant des objets plus élémentaires. Il s'agira des lignes, mais aussi du point. S'il est facilement concevable que les premières interviennent dans un bâtiment sous une forme ou une autre (pensons aux câbles en tout genre), il n'en va pas de même pour le deuxième. Et pourtant, le point permet de représenter une action ponctuelle (un carottage dans le terrain par exemple), ou encore un objet dont la taille réduite n'est pas proportionnelle à son importance (la prise d'un téléphone par exemple).

Remarque : on verra ultérieurement que l'existence du point rend possible la manipulation des différents objets par leurs sommets.

Si l'utilisateur crée un élément qui ne figure pas au menu de l'éditeur, il aimera sans doute conserver sa définition, et l'utiliser éventuellement dans d'autres projets. Pour ce faire, nous reprendrons les possibilités qu'offre la notion de groupe de STAR UX. L'utilisateur pourra donc définir un volume de calage autour de certains éléments existants du projet. Il pourra ensuite stocker le contenu de ce volume, le rappeler à tout moment, et le manipuler comme un vulgaire parallélépipède.

Prenons un exemple : la voûte en berceau est un élément architectural qui ne figure pas au menu de l'éditeur. L'utilisateur doit alors créer un cylindre, le couper en deux, et l'incliner de 90 degrés (on verra plus tard comment). S'il doit effectuer toutes ces opérations à chaque fois qu'il désire une telle voûte, il y a fort à parier qu'il retire à tout jamais cet élément de son répertoire architectural. Par contre, pouvoir enregistrer sa définition

par l'intermédiaire de son volume de calage sera extrêmement payant (voir figure 20).

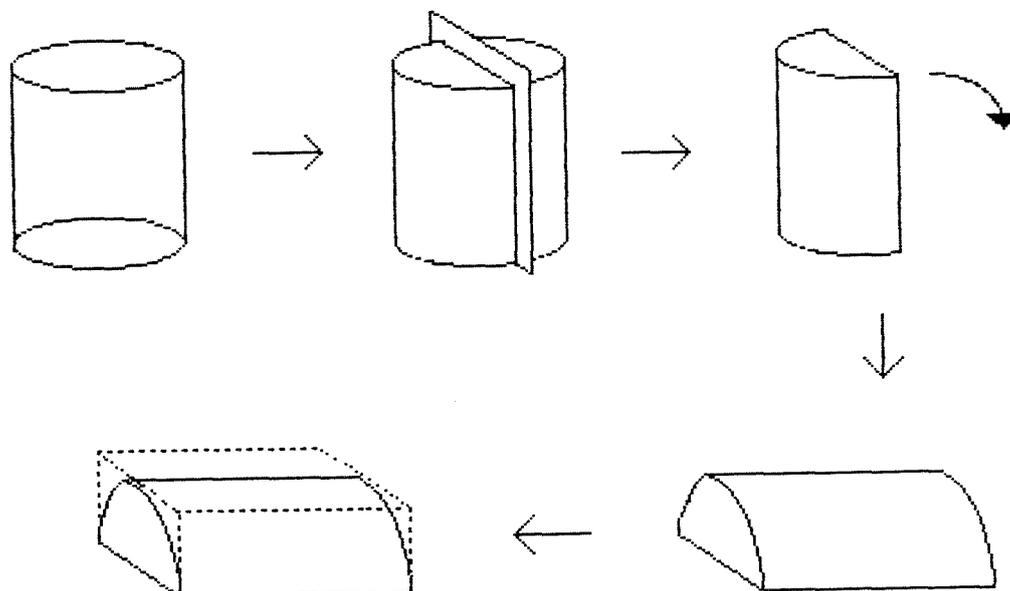


Figure 20

### VII.3 Les surfaces

---

Voyons maintenant quelles sont les surfaces qui doivent être proposées d'emblée par le logiciel, et quel est leur mode d'introduction.

Les surfaces qui constitueront le "set" de base sont les figures géométriques que l'on trouve dans les éditeurs 2D classiques, à savoir les rectangles, les cercles, et les polygones réguliers. Analysons les plus en détails.

#### VII.3.1 Le rectangle

Dans un éditeur graphique 2D, il suffit de spécifier les deux extrémités d'une diagonale pour désigner sans ambiguïté un rectangle dont les côtés sont parallèles aux deux axes. En trois dimensions, les choses ne sont pas aussi simples puisqu'il y a trois axes. On peut imaginer alors que l'introduction d'un rectangle se fasse entièrement dans le plan de référence courant. Dans ce cas, ses côtés sont parallèles aux traces que laisse ce plan sur ceux du système d'axes, et deux points suffisent pour le définir.

Pourtant, nous ne retiendrons que l'introduction directe d'un rectangle dans l'espace par la spécification du nombre suffisants de points, et ce pour les raisons suivantes :

- \* ne pouvoir introduire un rectangle que dans trois types de plan (horizontal, frontal, de profil) est une possibilité trop limitative pour être vraiment intéressante,
- \* le rectangle est la seule surface proposée qui permette d'économiser un point si elle est introduite dans le plan de référence,
- \* nous voulons un éditeur de volumes proposant accessoirement des surfaces, et non un éditeur de surfaces proposant également des volumes. Multiplier les modes d'introduction des surfaces n'est donc pas nécessaire, et risque même d'augmenter la complexité de l'interface utilisateur du logiciel.

Par conséquent, l'introduction d'un rectangle se fera par trois points : les deux premiers permettront de spécifier un de ses côtés, et le troisième, son autre côté ainsi que l'inclinaison du plan dans lequel il est inscrit. Il pourra donc faire apparaître complètement le rectangle créé.

Le programme ignorera les déplacements de la souris visant à écarter le curseur de la droite perpendiculaire à celle reliant les deux premiers points.

### VII.3.2 Le polygone et le cercle

Compte tenu de tout ce qui a été dit précédemment, l'introduction des polygones et celle des cercles se feront directement en trois dimensions, et seront relativement proches l'une de l'autre par souci de cohérence. On retrouvera d'ailleurs ici la même logique de spécification que celle des volumes ayant ces deux figures pour base.

L'introduction d'un polygone se fera de la manière suivante :

- 1) précision du nombre de côtés,
- 2) positionnement d'un des sommets,
- 3) positionnement du centre.

Celle du cercle se fera de la manière suivante :

- 1) positionnement d'un point quelconque de la circonférence,
- 2) positionnement du centre

## VII.4 Les lignes et le point

---

Vu l'orientation spatiale du logiciel, ce type d'objet sera évidemment rudimentaire par rapport aux deux précédents.

Une ligne sera introduite directement dans l'espace. Le seul type de ligne proposé sera la droite, tout naturellement spécifiée par deux points. Le mode d'introduction en série (c'est-à-dire où l'extrémité finale d'un segment correspond automatiquement à l'extrémité initiale du segment suivant) doit-il être proposé ? A priori, rien ne s'y oppose.

Remarque : il s'agit bien ici de lignes positionnées dans l'espace, et non de lignes appartenant à un masque 2D placé sur la vue en perspective afin de corriger ou d'ajouter éventuellement certains éléments qui ne peuvent l'être en trois dimensions.

En ce qui concerne le point, il faut savoir qu'il existe, quant à son mode d'introduction, il est trivial !

## VII.5 Fonctions supplémentaires

---

### VII.5.1 La décomposition et la recomposition

On pourrait se demander pourquoi une commande existe pour introduire un cercle, alors qu'un cône ou un cylindre de hauteur nulle pourrait faire l'affaire. On pourrait même aller plus loin et se dire qu'une droite n'est finalement qu'un parallélépipède rectangle dégénéré.

Il faut remarquer tout d'abord que les mécanismes d'introduction, s'ils se ressemblent, sont malgré tout spécifiques aux différents objets. Pour s'en convaincre, il suffit de comparer ce que donne le déplacement de la souris entre la spécification des deux points, lors de l'introduction d'un cercle et d'un cône de hauteur nulle (voir &figure 21).

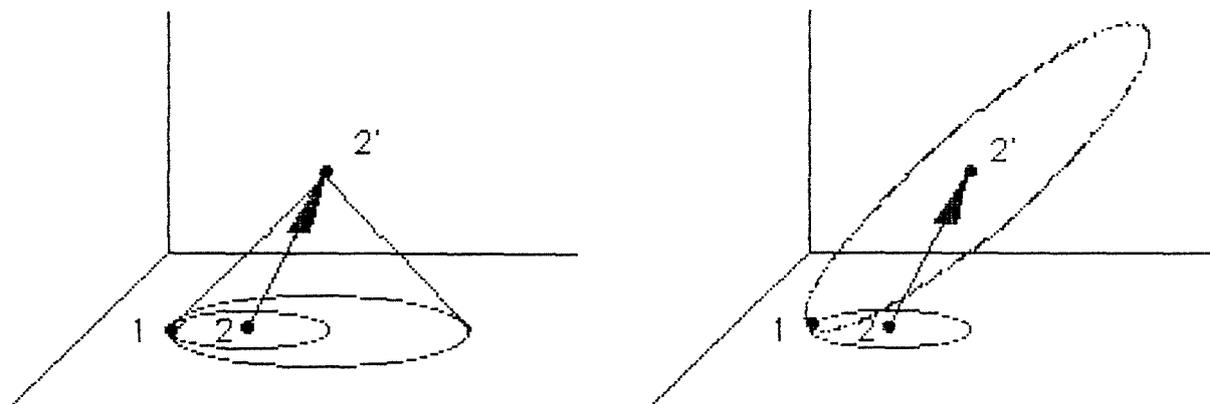


Figure 21

D'autre part, il faut savoir que l'enregistrement d'un volume dans la base de données entraîne la mémorisation d'une plus grande quantité d'informations que celui d'une surface, et est donc volumineux. Par conséquent, ne pas distinguer les différents types d'objet entraîne un gaspillage de place.

Pourtant, il peut s'avérer utile de pouvoir éclater un volume en un ensemble de surfaces, de regrouper un ensemble de surfaces pour en faire un volume, ou encore de pouvoir ajouter une hauteur à une surface afin d'en faire un volume. Par conséquent, l'éditeur proposera un outil de décomposition, ainsi que l'outil inverse de recombinaison. Grâce à ceux-ci, l'utilisateur pourra décomposer un parallélépipède en six faces, chacune de ces faces en quatre arêtes, et chacune de ces arêtes en deux sommets, et pourra ensuite revenir au volume de départ.

remarque : signalons déjà que ces deux outils permettront d'élargir sensiblement les possibilités des outils de manipulation qui seront développés plus tard.

### VII.5.2 Les objets de travail

Certains logiciels 2D un peu plus sophistiqués que les autres permettent de dessiner des traits de construction. Ceux-ci sont affichés dans une couleur particulière ou en pointillé par exemple, et ne s'impriment pas sur les supports physiques.

Nous trouverons l'équivalent de cette fonction dans notre éditeur en donnant à l'utilisateur la possibilité de placer dans l'espace de tels traits susceptibles de faciliter son travail de conception. Cependant, nous étendrons cette notion de tracé provisoire à tous les types d'objet de l'éditeur. C'est ainsi que l'utilisateur pourra créer des

surfaces et des volumes de travail.

Les premières lui permettront par exemple de tracer dans un plan horizontal l'organigramme élaboré au stade de conception précédant, et les seconds, de concevoir un bâtiment dont la silhouette générale est imposée par une étude urbanistique préalable.

## VII.6 Le maniement de la souris

-----

Les modes de maniement de la souris lors de l'introduction d'un volume sont variés et multiples. Par exemple, la spécification d'un volume par deux points peut se faire en utilisant la souris suivant une séquence "down - move - up", ce qui signifie : amener le curseur à l'endroit souhaité pour le premier point, enfoncer la touche de la souris, bouger la souris tout en maintenant la touche enfoncée, et relâcher cette touche à l'endroit souhaité pour le deuxième point. Si trois points doivent être introduits, la séquence peut être : "cliquer" sur le premier point, bouger la souris, "cliquer" sur le deuxième, bouger la souris, et enfin "cliquer" sur le troisième.

Ces types de maniement sont propres aux souris à une seule touche. Or, on trouve actuellement des souris possédant jusqu'à trois touches, ce qui peut influencer de manière considérable leur maniement .

La solution qui sera retenue nécessite un souris à deux touches.

Celle de gauche servira à la validation et sera donc une extension de la touche <RETURN> du clavier. Celle de droite servira à faire défiler les différents paramètres de sélection, s'ils existent.

Illustrons ceci en détaillant la spécification d'un volume par trois points. Elle se fait de la manière suivante :

- \* le premier point se positionne dans l'espace comme n'importe quel autre point, par simple déplacement de la souris ;
- \* après avoir pressé la deuxième touche (sélection), ce point est fixé, et le déplacement de la souris permet le positionnement du second ;
- \* en pressant à nouveau cette touche, c'est au tour du second point d'être fixé, et le déplacement de la souris permet cette fois le positionnement du troisième ;
- \* à ce stade, on peut presser la première touche (validation) pour confirmer la position des trois points, ou alors presser de nouveau la deuxième touche pour reprendre la séquence de positionnement au premier

point ;  
\* et ainsi de suite jusqu'à l'ultime confirmation.

L'avantage de cette méthode réside dans le fait qu'après avoir spécifié les trois points exigés par l'introduction du volume, l'utilisateur peut revenir au premier point pour changer sa position dans l'espace, ce qui entraîne le déplacement de le volume. De même, son orientation ainsi que la longueur d'un des côtés de sa base peuvent être modifiées en revenant au second point ; et sa hauteur ainsi que la longueur de l'autre côté de sa base peuvent être modifiées en revenant au dernier point.

Déplacer le volume, le faire pivoter, corriger ses proportions, sans changer de commande est particulièrement intéressante lors de la phase de conception, où l'introduction d'un nouveau volume se fait plus par essai, ajustement et confirmation, que par spécification de points très précis.

On peut imaginer que le sommet du volume qui guide l'action à un moment donné soit numéroté de manière à permettre à l'utilisateur de savoir où il en est dans l'introduction du volume.

## CHAPITRE VIII : La notion d'indétermination

---

### VIII.1 L'indétermination des dimensions

---

Jusqu'à présent, peu de choses font de notre éditeur de volumes un outil fondamentalement différent des progiciels classiques. Bien sûr on travaille directement en trois dimensions, bien sûr on manipule des volumes et non des composants constructifs, bien sûr l'accent a été mis sur la rapidité plutôt que sur la qualité du dessin, mais est-ce suffisant pour aider efficacement la conception architecturale ? La réponse est évidemment négative, pour la bonne et simple raison que nous avons négligé jusqu'ici un point pourtant capital : l'imprécision que doivent absolument garder les supports de conception pour remplir correctement leur rôle. Toute recherche de précision est en effet un frein considérable au flux des idées créatrices (voir chapitre III).

L'attitude de notre éditeur vis-à-vis de cette caractéristique sera de permettre à l'utilisateur de laisser les dimensions de son projet indéterminées jusqu'à ce qu'il arrive à un stade plus avancé de conception. Il pourra de la sorte matérialiser les intentions du projet, plutôt que ses dimensions. Voyons ce que cela signifie concrètement.

#### VIII.1.1 La création sans dimensions

En tout premier lieu, on peut déjà bannir toute fonction dont l'exécution est liée à la notion de dimension.

L'exemple type d'une telle fonction est l'affichage sur chaque axe d'une règle graduée sur laquelle évolue un repère en fonction des mouvements du curseur. Cela permet de mesurer des éléments du projet, ou encore d'introduire de nouveaux éléments en tenant compte de leurs dimensions. On sent bien que malgré son inutilité flagrante, la présence d'une telle fonction attire énormément un utilisateur quelque peu désabusé par une forme de travail qui lui est nouvelle.

Citons encore quelques autres fonctions du même type : introduction des volumes en entrant au clavier des données numériques (coordonnées, longueur des côtés,...), mesure des

angles, possibilité de donner aux parois des différents volumes une épaisseur mesurable, etc... .

Si les fonctions précédentes peuvent être rejetées sans conteste, il n'en va pas de même pour certaines fonctions dont la référence aux dimensions est beaucoup plus discrète, comme par exemple la division automatique d'un segment en X parties égales. Cette fonction entraîne effectivement une contrainte sur la longueur de chaque segment obtenu après la division, sans pour autant obliger l'utilisateur à manipuler directement ces longueurs.

La question est maintenant de savoir s'il faut ou non proposer cette fonction à l'utilisateur pour son travail de conception.

Ses partisans diront que

- \* la division d'un segment en parties égales peut être désirée par l'utilisateur dès la conception,
- \* la contrainte sur la longueur des segments obtenus reste valable quelle que soit la longueur du segment initial, et par conséquent,
- \* il ne s'agit nullement d'une fonction allant à l'encontre de l'indétermination des dimensions.

Ses détracteurs diront au contraire que

- \* quitter l'espace de création, sélectionner la fonction dans un menu, préciser le nombre de divisions, revenir dans l'espace de création, et désigner le segment, est une procédure trop longue,
- \* s'il est possible de l'écourter, c'est au prix d'un effort de mémorisation plus important de la part de l'utilisateur, et donc
- \* elle l'oblige à rompre momentanément le fil de ses idées pour un problème de manipulation.

Ils ajouteront même que

- \* la division d'un segment en parties égales n'est pas une opération de conception architecturale, et de ce fait, que la fonction n'apporte rien de plus à l'éditeur à part une augmentation de la complexité de son interface utilisateur.

En réaction à ces avis partagés, la solution que nous préconiserons est la suivante :

- \* pendant la phase de conception, l'utilisateur doit placer visuellement les différents objets qui sont sensés diviser un segment en parties égales,
- \* après cette phase, il peut activer la fonction et désigner le segment en question (en "cliquant" sur ses extrémités par exemple), pour que tous les objets qu'il rencontre soient pris en considération pour le calcul de la division. L'utilisateur peut alors les déplacer de manière à respecter la division qui s'affiche, s'il le

désire évidemment.

Il peut être intéressant de pouvoir prolonger fictivement un élément ne traversant pas le segment pour qu'il intervienne dans le calcul de la division, et inversement, d'effacer fictivement un élément traversant le segment pour qu'il n'intervienne pas dans le calcul de la division.

### VIII.1.2 Le dimensionnement à posteriori

Nous venons donc d'éliminer tous les outils qui auraient permis à l'utilisateur de travailler en tenant compte des dimensions. Mais il est assez paradoxal d'affirmer que notre éditeur est plus approprié à la conception architecturale que les progiciels classiques, alors que jusqu'à maintenant, à peu de choses près, on n'a fait qu'élaguer ces progiciels pour définir le nôtre. Ce serait dire que les programmes de CAO classiques conviennent aussi pour la conception architecturale, si l'utilisateur ne se sert de leurs fonctionnalités qu'avec beaucoup de discernement. Ce qui n'est pas exact bien sûr.

Face à cette constatation, nous allons compenser l'absence de ces outils par la possibilité de dimensionner après-coup le projet. Cela va tout d'abord éviter la perte des informations conceptuelles lors du passage à la phase de mise au point (c'est-à-dire avec dimensions et composants constructifs) assistée elle aussi par ordinateur. Mais surtout, cela va certainement encourager l'utilisateur à travailler de manière approximative, car grâce à cette possibilité, il aura toujours l'assurance lors de la conception, de pouvoir s'occuper des problèmes de dimensions par la suite, de pouvoir corriger éventuellement des erreurs de dimensions, de vérifier si son projet approximatif reste valable avec des dimensions précises.

Voyons les services que doit pouvoir offrir l'outil de dimensionnement à posteriori.

- 1) Les dimensions du projet issu de la phase de création sont indéterminées. Cela veut dire qu'elles sont internes au programme, qu'elles sont exprimées dans une unité qui ne signifie rien pour l'utilisateur, et qu'elles sont même ignorées de lui.

La première application évidente de l'outil de mise à dimension est l'établissement d'une correspondance entre l'unité interne, et une unité de longueur plus représentative pour l'utilisateur, c'est-à-dire le mètre.

Pour ce faire, l'utilisateur doit désigner un segment de

l'espace, c'est-à-dire une arête, une portion d'arête, la largeur d'une face, l'espace entre deux faces ou deux volumes, etc..., et entrer au clavier la longueur de ce segment. Ceci fait, toutes les autres dimensions du projet sont automatiquement déduites de cette longueur de référence. Le projet peut alors servir de base pour la phase de mise au point (en passant préalablement par l'affectation des éléments à des composants constructifs), ou peut être re-dimensionné en corrigeant la valeur de la longueur de référence.

- 2) Jusqu'à présent, toutes les dimensions du projet découlent de la seule longueur de référence. Or, bien souvent en architecture, la détermination des hauteurs est beaucoup moins souple que le choix des dimensions du plan. Comme il y a peu de chances d'arriver directement à des hauteurs valables en partant du dimensionnement du plan, il faut que notre outil puisse accepter une seconde longueur de référence pour définir ces hauteurs.

Notons que l'introduction de cette seconde longueur de référence peut entraîner l'aplatissement ou l'étirement vertical du projet, ce qui influence évidemment l'angle que font les plans inclinés avec l'horizontale (voir figure 22).

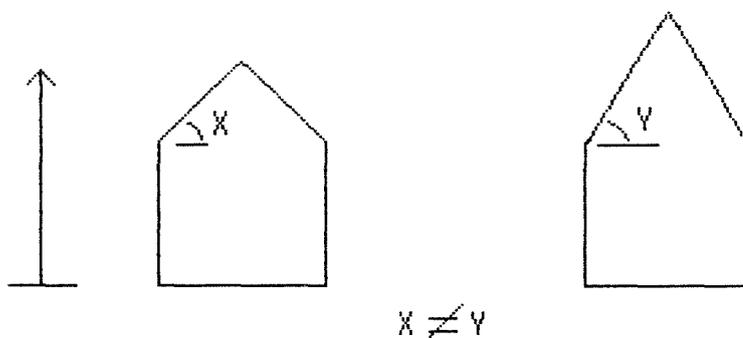


Figure 22

Remarque : pour éviter toute confusion, les modifications de dessin partiront de l'origine, c'est-à-dire que celle-ci ne bouge pas dans l'espace.

Par extension, deux longueurs de référence pourront être introduites pour le plan, une pour les dimensions suivant l'axe des X, l'autre pour celles suivant l'axe des Y.

Le logiciel admettra donc jusqu'à trois longueurs de référence. Si l'une d'elles n'est pas spécifiée, ce

sera évidemment une des deux autres qui la remplacera pour la détermination des dimensions dans sa direction. Celle-ci sera choisie suivant la priorité  $X > Y > Z$ . Par exemple, si la longueur de référence suivant Y n'est pas spécifiée, c'est celle suivant X qui la remplacera.

Par ailleurs, l'utilisateur pourra prendre un segment quelconque de l'espace comme base du dimensionnement. C'est ainsi que si le segment choisi est une horizontale quelconque, deux longueurs de référence seront spécifiées, celle suivant l'axe X et celle suivant l'axe Y, et s'il est tout à fait quelconque dans l'espace, les trois longueurs seront spécifiées en une seule fois.

Remarque : admettons par convention que la longueur de référence suivant l'axe X dimensionne les segments dans cette direction sans modifier leur apparence à l'écran, tandis que les deux autres entraînent la modification du dessin pour l'adapter à ses nouvelles dimensions. Soulignons que cette convention n'influence que la grandeur du dessin à l'écran, et pas ses proportions.

L'utilisateur doit avoir la possibilité d'analyser les effets du dimensionnement sur son projet. Pour cela, il doit disposer d'un outil de mesure, qui pourra être l'affichage de la distance séparant deux points de l'espace pointés par la souris, ou encore la mise en place de lignes de cotation afin d'estimer de manière continue les implications dimensionnelles. Si ces deux fonctions sont utilisées avant le dimensionnement du projet, elles afficheront un caractère d'indétermination (? par exemple) au lieu d'une valeur numérique.

Nous avons parlé de la modification du dessin. Tous les éléments sont-ils concernés par cette modification ? Les cercles, par exemple, risquent-ils de devenir des ellipses, ou les pyramides à base carrée risquent-elles de voir leur base se transformer en rectangle ? Ce serait évidemment dommage, car n'oublions pas que nous définissons ici l'outil de dimensionnement, qui ne modifie le dessin que pour rester cohérent par rapport aux valeurs numériques. Des outils de manipulation qui permettent de réaliser ces différentes choses seront par contre développés ultérieurement.

Par conséquent, afin de limiter au maximum la déformation du dessin par rapport à son état initial, nous déciderons que les objets qui sont créés avec des contraintes sur leur géométrie doivent respecter ces contraintes après leur déformation. C'est ainsi que la base d'un cône sera toujours un cercle, même si la modification du dessin est unidirectionnelle.

Remarque :

- a) au moins une longueur de référence doit être spécifiée avant (ou pendant) l'activation des autres fonctions de l'outil de dimensionnement qui sont proposées ci-après.
  - b) une fois qu'une longueur de référence a été introduite, tout le projet est dimensionné. Par conséquent, le retour à l'éditeur de volumes n'implique plus l'indétermination des dimensions, bien qu'il soit toujours possible de corriger la valeur des longueurs de référence.
- 3) On peut imaginer que les dimensions d'un objet déterminé, ou qu'une partie de celles-ci, ne doivent pas subir l'influence des longueurs de référence, mais qu'elles doivent être spécifiées explicitement par l'utilisateur. Par exemple, ce dernier peut exiger qu'un WC fasse 1.00 m \* 1.60 m, parce qu'il estime que plus serait du gaspillage, et que moins ne serait pas suffisant.

On peut même aller plus loin et décider que les dimensions spécifiées explicitement influencent également celles d'autres objets, ce qui revient en quelque sorte à définir des longueurs de référence locales.

Pour utiliser cette possibilité, l'utilisateur devra appliquer la procédure suivante :

- \* sélection d'un objet, ou d'un groupe d'objets (par les mécanismes qui seront développés plus tard) ;
- \* spécification au sein de ce groupe d'une, de deux, ou de trois longueurs de référence locales. Si l'une d'elles n'est pas spécifiée, la priorité ne joue pas mais c'est la longueur de référence principale suivant la même direction qui la remplacera. La déformation du groupe suivra alors dans cette direction celle du projet tout entier ;
- \* désignation du point qui permettra de rattacher au projet le groupe modifié séparément (voir figure 23).

Remarque :

- a) un groupe peut très bien se réduire à une droite. Il est évident que dans ce cas, la direction de l'unique longueur de référence pouvant être spécifiée est celle de la droite.
- b) l'influence d'une longueur de référence locale sur les objets créés avec des contraintes géométriques (cercles, prismes à bases régulières, etc...) est similaire à celle d'une longueur de référence principale.

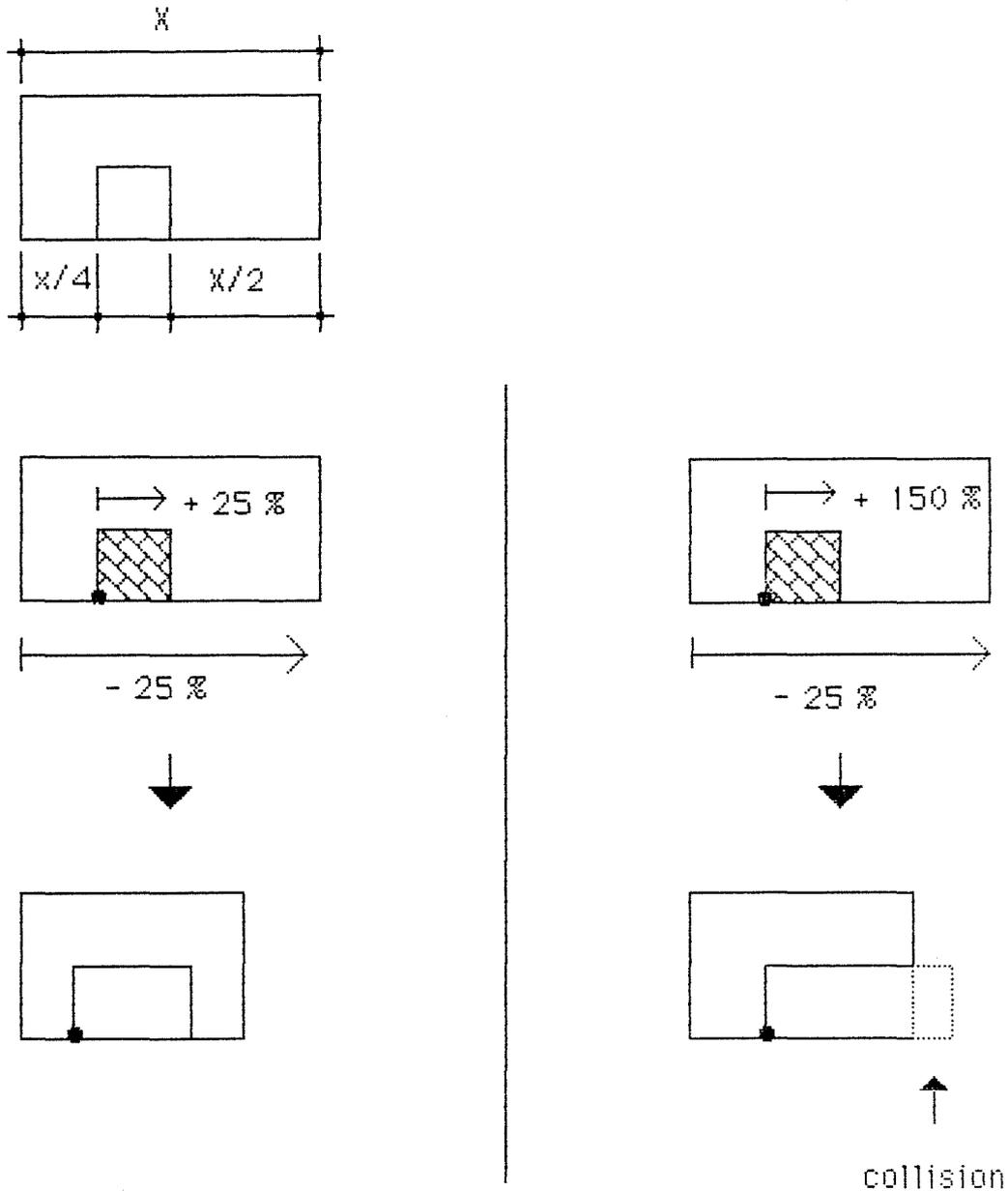


Figure 23

c) une longueur de référence locale peut entraîner la collision entre deux éléments du projet si elle est disproportionnée par rapport à la principale (voir figure 23). Par conséquent, le programme la refusera.

4) Plutôt que d'introduire des longueurs de référence locales, l'utilisateur se contentera des principales, mais celles-ci ne seront acceptées par le programme que

addendum

-----

page 70, ligne 34

On peut étendre de la même manière les possibilités des longueurs de référence locales en admettant non seulement une valeur numérique, mais également une expression similaire à celles décrites ci-dessus.

si les conditions qu'il a imposées au préalable sur la longueur de certains segments sont respectées. Il pourra exiger par exemple que la longueur d'un garage soit comprise entre 4.50 m et 6.00 m.

Si l'utilisateur veut exercer un tel contrôle sur certains segments, il doit tout d'abord les désigner, et exprimer ensuite la condition relative à chacun d'eux. Il dispose pour cela des opérateurs relationnels et logiques comme =, <, >, <=, >= ainsi que ET, OU, NON. Par exemple, la condition imposée sur la longueur du garage sera exprimée de la façon suivante : (longueur-du-segment >= 4.50) ET (longueur-du-segment <= 6.00). Mais l'utilisateur pourra aussi imposer longueur-du-segment > 6.00, ou encore longueur-du-segment NON = 4.00.

Après l'opérateur relationnel, il peut introduire une valeur numérique, la longueur d'un autre segment (spécifiée en "cliquant" sur ses extrémités), ou une expression. Cette dernière est constituée d'une valeur ou de la longueur d'un autre segment comme première opérande, d'un opérateur arithmétique (+, -, \*, /), et d'une autre expression comme deuxième opérande. Par exemple, l'utilisateur peut obliger la longueur d'un segment à être égale à celle d'un autre segment, ou à être deux fois plus grande que celle d'un autre segment (> 2 \* pointage), ou plus petite que la somme des longueurs de deux segments ( = pointage + pointage).

Remarquons que l'utilisateur peut introduire des conditions qui ne seront jamais respectées. A lui de faire attention.

- 5) la fonction suivante permettra de lever l'indétermination concernant les angles du projet qui ne sont pas droits.

Pour ce faire, l'utilisateur doit désigner les objets concernés, et spécifier pour chacun d'eux la charnière autour de laquelle il pivotera, et l'angle, c'est-à-dire sa valeur numérique, trois points de l'espace déterminant sa forme. Les angles entraînant la collision des objets seront refusés par le programme.

Nous en resterons là pour la définition de l'outil de dimensionnement à posteriori. Un des buts de cet outil était d'encourager l'utilisateur à travailler approximativement. Nous allons maintenant essayer de l'assister non plus de manière différée, mais directement dans son travail de création, en diminuant le besoin de faire appel aux dimensions par le recours à deux outils, la grille et la trame.

### VIII.1.3 la grille

La grille, souvent qualifiée de magnétique, est un système qui divise l'espace en cellules plus ou moins grandes, où seuls les noeuds sont accessibles. Cela signifie que le pointage sur un point quelconque de l'espace, correspond à la désignation du noeud de la grille le plus proche. Elle discrétise donc l'espace, et règle ainsi la précision.

La présence de cette grille suscite de nombreuses questions. Peut-elle être désactivée par l'utilisateur ? Doit-elle être visible ? Doit-elle être spatiale (c'est-à-dire définie par les coordonnées de l'espace), ou plane (c'est-à-dire définie par les coordonnées de l'écran) ?

Tâchons de répondre à la dernière question. Avec une méthode traditionnelle de conception, c'est l'épaisseur de la pointe du crayon qui détermine la précision. En effet, le crayon employé dans les premiers stades de conception est souvent très gros, car en donnant des traits épais, il laisse une marge d'incertitude importante quant à leur localisation exacte. L'épaisseur des arêtes ne varie donc pas avec l'éloignement du volume dans l'espace, et le degré de précision reste constant sur toute la surface du papier.

Or, si l'on considère une grille spatiale dans une perspective réelle, les deux caractéristiques précédentes ne sont plus respectées. Pourtant, c'est bien ce type de grille que nous choisirons, et non la grille plane, source de problèmes de correspondance avec l'espace représenté. Nous estimons en effet qu'il n'est pas nécessaire d'informatiser le processus traditionnel en s'efforçant de s'en éloigner le moins possible, et de la sorte, en se fermant peut-être à toute nouvelle possibilité permettant d'améliorer ce processus.

Il importe maintenant de savoir s'il faut dessiner cette grille, et dans l'affirmative, sous quelle forme. Dans certains cas, ne pas voir la grille peut être assez embêtant, car il est plus difficile de prévoir le résultat exact d'une action sur l'espace. Dans d'autres cas, sa représentation peut être gênante, surtout si elle entre en conflit avec d'autres représentations, comme celle de la trame notamment. L'utilisateur aura donc la possibilité de demander ou non le dessin de la grille.

Comme ce sont les noeuds magnétiques qui définissent la grille, ce sont eux, et uniquement eux, qui seront dessinés. La grille sera donc représentée par un ensemble ordonné de points, qui devraient théoriquement occuper tout l'espace, puisque la grille qui a été choisie est spatiale. Comme l'espace est infini, l'écran devrait être tout noir avec une

vue en perspective réelle ! Pour éviter cette situation, nous déciderons que les seuls points de la grille qui seront dessinés sont ceux qui sont compris dans le plan de référence, et ceux qui se trouvent sur la perpendiculaire à ce plan à l'endroit du curseur (voir figure 24). Cette représentation partielle de la grille est acceptable quand on sait qu'elle joue principalement son rôle à l'introduction des points, ce qui veut dire qu'elle est intimement liée au curseur, et par transitivité au plan de référence et à sa perpendiculaire.

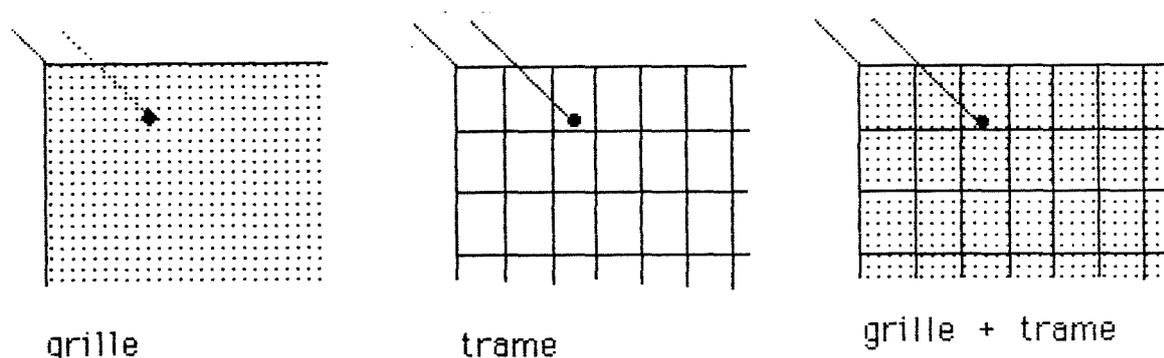


Figure 24

L'utilisateur doit-il avoir la possibilité de désactiver la grille ? Il est évident que l'espacement entre les noeuds de la grille, d'où la précision, doit pouvoir être ajusté. Ce réglage doit avoir une limite inférieure, car s'il était possible de diminuer à volonté l'espacement entre les noeuds, le plan de référence deviendrait graduellement tout noir. L'utilisateur doit donc pouvoir désactiver la grille s'il désire toujours plus de précision après avoir atteint cette limite.

Revenons sur la question du réglage pour éclaircir certains points. La grille est utile dès la création, c'est-à-dire avant tout dimensionnement. Dès lors, l'utilisateur devra ajuster la grille, non pas en entrant numériquement au clavier l'espacement entre les noeuds, mais en exécutant une commande qui augmente ou diminue cet espacement d'un nombre d'unités internes prédéfini.

Ce réglage sera bien sûr limité inférieurement, mais aussi supérieurement, car une grille dont les noeuds sont très espacés rend l'éditeur inutilisable. La grille pourra donc être activée avec une précision par défaut qui peut être le juste milieu entre les deux bornes.

Pour éviter d'alourdir la procédure de réglage, nous déciderons que l'espacement entre les noeuds est identique dans les trois directions.

Ajoutons encore que le zoom qui sera développé plus tard ne doit avoir aucune influence sur la grille. En effet, ce n'est pas parce qu'on agrandit ou diminue le dessin que la précision doit changer.

#### VIII.1.4 La trame

La trame est un outil qui entraîne aussi la division de l'espace en cellules. Mais contrairement à la grille, la trame est passive, car elle n'exerce aucune influence sur la désignation des points. Le but de cette trame n'est donc pas de régler la précision, mais d'offrir à l'utilisateur un système de repérage visuel dans l'espace.

On pourrait penser que l'usage de cette trame s'oppose à l'idée que l'on s'était faite du travail de conception. Il n'en est rien, car :

- \* dans bien des cas, qu'on le veuille ou non, la conception des bâtiments doit, pour des raisons généralement constructives, tenir compte de l'existence d'une trame. Par exemple, l'organisation d'un bâtiment dont la structure est composée de poteaux et de poutres est immanquablement guidée par la trame que forment les poteaux sur le sol.
- \* le travail reste approximatif, principalement à l'intérieur des mailles de la trame (qui, d'ailleurs, peuvent être relativement grandes). De plus, ces mailles peuvent être dimensionnées à posteriori, ce qui préserve l'indétermination des dimensions. Dans l'exemple précédent, on sait très vite que les poteaux forment une trame, mais ses dimensions exactes dépendent de la portée des poutres qui ne sont connues que bien plus tard.

Le problème de savoir si la trame doit être spatiale ou non ne se pose évidemment pas. Une trame définie sur l'écran n'apporte aucune aide au concepteur.

Par contre, il importe de se pencher sur le mode de représentation de cette trame. Tout comme la grille, et pour les mêmes raisons, la trame ne sera dessinée que dans le plan de référence, et sur la perpendiculaire à ce plan, à l'endroit du curseur. Ce sont les mailles de la trame qui seront dessinées, et non plus uniquement les noeuds (voir figure 24), et ce pour éviter toute confusion avec le dessin de la grille.

La trame doit elle aussi être réglable. Ce réglage peut être différent suivant les trois directions. Il doit avoir une limite inférieure, mais pas obligatoirement supérieure, une trame à mailles très larges pouvant s'avérer utile dans certains cas.

Ajoutons enfin que le zoom doit agir sur le projet comme sur la trame.

## VIII.2 L'indétermination des composants constructifs

-----

Nous avons déjà insisté sur le fait que les objets manipulés à la conception doivent être des volumes, et non des composants constructifs qui n'interviennent que bien plus tard. Mais tout comme pour les dimensions où on a dû créer un outil de dimensionnement à posteriori, on va devoir développer ici un outil permettant d'affecter à posteriori des composants constructifs aux éléments du projet. Le but principal de cet outil est d'éviter la perte des informations constituées par le projet volumétrique lors du passage à la mise au point détaillée.

### VIII.2.1 La sélection du composant

Avant toute réflexion sur le fonctionnement de cet outil, il faut se rappeler que notre éditeur de volumes est couplé à un éditeur de composants constructifs calqué sur les progiciels de CAO pour architectes actuellement disponibles. Ce dernier utilise donc une bibliothèque de composants constructifs paramétrés. Le mur, par exemple, en fait partie, et ses paramètres sont : la largeur, la longueur, la hauteur, la position dans l'espace, le matériau employé, le prix au mètre carré, les caractéristiques thermiques, etc...

Par conséquent, le lancement de l'outil d'affectation à posteriori amène l'utilisateur "dans" cette bibliothèque de composants pour qu'il en sélectionne un. Une grille apparaît alors à l'écran pour que l'utilisateur remplisse les champs correspondant aux paramètres connus à ce stade de l'élaboration du projet. Il s'agit bien entendu des paramètres ne pouvant être calculés à partir du dessin. Dans l'exemple du mur, les paramètres déduits du dessin sont la longueur, la hauteur, et la position dans l'espace, alors qu'un paramètre comme l'épaisseur ou comme la nature du matériau doit être spécifié.

Remarquons que si le projet n'est pas dimensionné avant l'activation de l'outil, le programme ne sait pas à quelle échelle les composants doivent être dessinés. Dans ce cas, bien qu'enregistrés, ils ne sont dessinés qu'après le dimensionnement.

## VIII.2.2 La désignation des éléments

Après avoir sélectionné un composant, l'utilisateur doit désigner les éléments du projet (volumes, surfaces, lignes, ou même points) qui correspondent à celui-ci.

La méthode la plus élémentaire oblige l'utilisateur à "cliquer" successivement sur ceux-ci tant qu'il ne désire pas retourner dans la bibliothèque pour changer de composant, ou tout simplement pour modifier un de ses paramètres. Après chaque "click", le dessin de l'élément désigné est remplacé par celui du composant. C'est ainsi par exemple que l'utilisateur peut sélectionner :

- \* le mur dans la bibliothèque, et l'affecter à une paroi ;
- \* la fenêtre (composant "négatif", c'est-à-dire supprimant la matière de celui sur lequel il se superpose) et l'affecter à un rectangle dessiné sur une paroi ;
- \* l'escalier circulaire et l'affecter à un cylindre.

Chaque composant répertorié dans la bibliothèque est associé à un type d'objet bien défini, ce qui empêche l'utilisateur de l'affecter à n'importe quel objet du projet. Par exemple, le programme refusera d'affecter une fenêtre à un cylindre.

On peut imaginer une méthode de désignation plus sophistiquée qui permet d'affecter un composant à plusieurs objets en une seule opération. Cette méthode fait appel à un véritable langage de désignation où le premier terme est un quantificateur universel implicite (pour tout), le deuxième désigne le type des éléments concernés, et le troisième exprime la condition de sélection de ces éléments.

Citons quelques exemples de types d'élément : parallélépipèdes rectangles, cylindres, cercles, mais aussi parois de parallélépipède rectangle, arêtes de parallélépipède rectangle, arêtes de rectangle, etc... .

En ce qui concerne la condition de sélection :

- \* elle peut exprimer une propriété qui doit être respectée par les éléments pour être sélectionnés,
- \* elle peut exprimer une comparaison entre une de leur caractéristique et une valeur numérique, ou
- \* elle peut être une combinaison de conditions de sélection plus élémentaires, reliées par des opérateurs logiques (ET, OU, NON).

Par propriété, on entend : horizontal, de profil, frontal, vertical, de bout, extérieur (par rapport au projet), et intérieur. L'expérimentation du programme risque fort de mettre en évidence d'autres propriétés intéressantes qui doivent être ajoutées à cette liste.

Quant à la comparaison, elle porte sur une caractéristique des éléments, comme la largeur, la longueur, la hauteur, le rayon, le nombre de côté, etc... minimum ou maximum.

Illustrons tout ceci par quelques exemples de mise en oeuvre de l'outil d'affectation.

- \* Sélection dans la bibliothèque du mur creux (c'est-à-dire composé d'un mur porteur intérieur et d'un mur de parement extérieur) et affectation aux éléments désignés par "(toutes les) parois de parallélépipède rectangle, vertical ET extérieur".
- \* Sélection de la dalle, et affectation aux éléments désignés par "(toutes les) parois de parallélépipède rectangle, horizontal ET (hauteur < X)".
- \* Sélection de la toiture plate, et affectation aux éléments désignés par "(toutes les) parois de parallélépipède rectangle, horizontal ET (hauteur > X)".
- \* Sélection du pan de toiture, et affectation aux éléments désignés par "(toutes les) parois de prisme triangulaire, (NON horizontal) ET (NON vertical) ET (hauteur minimum > X)".
- \* Sélection de la volée d'escalier, et affectation aux éléments désignés par "(toutes les) parois de prisme triangulaire, (NON horizontal) ET (NON vertical) (hauteur maximum < X)".

Remarque : la syntaxe exacte du langage reste encore à définir, mais les règles en usage dans d'autres syntaxes du même type (comme les règles de priorité entre les différents opérateurs p ex), seront sans doute respectées.

### VIII.2.3 L'affectation partielle

Il peut être intéressant pour l'utilisateur de pouvoir lever l'indétermination des composants constructifs pour une partie seulement des éléments du projet. C'est le cas s'il a déjà conçu en détail le bâtiment, et qu'il veut revenir dans l'éditeur de volumes pour trouver un aménagement optimal par exemple. Nous accepterons cette possibilité, et pour éviter toute confusion de la part de l'utilisateur, les éléments à qui un composant a déjà été affecté ne seront plus accessibles par les outils de l'éditeur de volumes, mais uniquement par ceux de l'éditeur de composants constructifs. Ils seront d'ailleurs dessinés dans une autre couleur, ou en faible intensité si on est en mode monochrome.

Inversement, l'utilisateur peut très bien vouloir travailler avec l'éditeur de composants constructifs sur un projet partiellement défini au niveau des composants. Dans

ce cas, les éléments à qui aucun composant n'a encore été affecté, ne seront pas accessibles par les outils de l'éditeur de composants constructifs, et seront dessinés différemment.

## CHAPITRE IX : La fonction d'archivage

---

L'éditeur de volumes tel qu'il est défini ci-avant ne remplit que très mal son rôle. En effet, les documents créés ne peuvent être assimilés aux supports de conception décrits précédemment (voir chapitre III), car s'ils permettent de matérialiser les idées conceptuelles, ils ne permettent pas encore de les archiver. Cela s'explique par le fait que leur durée de vie est limitée à la session de travail, sauf peut-être s'ils sont systématiquement imprimés sur papier, ce qui est loin d'être la procédure la plus rapide.

### IX.1 Enregistrer un calque

---

Il est évident qu'une procédure d'archivage plus élaborée doit permettre à l'utilisateur d'enregistrer le projet courant sous forme digitale dans la mémoire de masse. Pour être rapide, elle doit être entièrement automatique, c'est-à-dire ne nécessitant pas l'intervention de l'utilisateur. Ce dernier ne doit donc pas spécifier d'identifiant pour retrouver son projet, mais le programme doit lui attribuer par compostage un numéro identifiant, et peut ajouter la date et l'heure de l'enregistrement à toutes fins utiles.

Remarque : la sauvegarde automatique toutes les X secondes comme le font certains traitements de textes n'a pas de sens ici. En effet, toute l'évolution doit être sauvegardée et pas uniquement la dernière version du projet. De plus, il se peut qu'au moment de la sauvegarde, le projet soit tout à fait inintéressant, parce que l'utilisateur est occupé à le modifier par exemple.

L'écran ne sera pas effacé après l'enregistrement du projet courant, car l'utilisateur doit pouvoir continuer à faire évoluer ce dernier. Le lancement de la commande d'enregistrement équivaut en fin de compte à décalquer le projet courant sur un nouveau calque.

Soulignons que c'est bien le projet volumétrique qui

sera stocké, et non l'image apparaissant à l'écran. En effet, il faut que l'utilisateur puisse non seulement rappeler ce projet, mais aussi le remanier, agir de nouveau sur les objets qui le composent afin de produire une nouvelle version.

Ce stockage sera probablement extrêmement coûteux en place. Par conséquent, la procédure de sauvegarde sera implémentée pour que seules les modifications depuis la dernière sauvegarde soient enregistrées, et non le projet tout entier à chaque fois.

Cette manière de procéder implique un temps d'accès un peu plus conséquent, puisque retrouver une version déterminée du projet revient à retrouver une autre version plus vieille, et toutes les modifications qui permettent d'aboutir à la version demandée. Pour limiter cette perte de temps, l'utilisateur aura la possibilité de créer des calques de synthèse, c'est-à-dire des enregistrements reprenant la dernière version complète du projet, et toutes les nouvelles modifications s'y rapportant. Cette manière de faire est d'ailleurs plus proche de la réalité, où le concepteur ne recopie pas tout le projet quand il superpose un calque vierge, mais ne le fait que quand l'épaisseur de calques devient trop importante, ou quand il arrive à un projet susceptible de servir de base pour de nombreuses autres évolutions.

## IX.2 Rechercher un calque

-----

Si l'utilisateur a la possibilité d'enregistrer un calque d'esquisse, c'est pour pouvoir le rechercher par la suite et y travailler. Mais comment peut-il rechercher un calque bien précis, s'il ne connaît pas son numéro identifiant ?

En fait, on peut considérer que les calques d'esquisse ont été empilés au fur et à mesure que les différentes versions ont été enregistrées. Par conséquent, pour consulter un calque bien précis, il faut parcourir la pile en commençant par le dessus, exactement comme on le ferait avec des calques d'esquisses traditionnels.

Certains calques enregistrés sont aux yeux de l'utilisateur plus importants que d'autres. Dans ce cas, il veut les retrouver facilement et rapidement, c'est-à-dire sans devoir consulter tous les calques qui sont au dessus dans la pile. Pour répondre à ce besoin, le programme offrira à l'utilisateur la possibilité d'attribuer un nom identifiant à un calque existant. Il s'agit donc en quelque

sorte d'une identification après coup.

Remarquons que l'attribution d'un nom à un calque et la création d'un calque de synthèse se font toutes deux pour accélérer le processus de recherche. Face à cette constatation, les deux possibilités seront agrégées en une seule commande. Il faut donc donner obligatoirement un nom à un calque de synthèse, et donner un nom à un calque quelconque revient à en faire un calque de synthèse qui s'intercale dans la pile.

Il est évident que certaines facilités de gestion de fichiers peuvent être appliquées à celle des calques. C'est ainsi que le nom du dernier calque chargé restera mémorisé, que l'utilisateur pourra demander la liste des noms des calques de synthèse et "cliquer" sur celui qui l'intéresse pour le charger ou éventuellement pour le supprimer, etc... .

Si l'utilisateur va rechercher un calque d'esquisse, c'est éventuellement pour le faire évoluer. Mais ce n'est pas le calque retrouvé qui évolue, c'est un autre qui a été superposé au premier.

Notons encore qu'un seul et même calque peut servir de base à plusieurs évolutions différentes. En effet, quand le concepteur explore une possibilité jusqu'au moment où il se rend compte qu'il est dans une impasse, il doit faire marche arrière et reprendre l'évolution à partir d'une version plus ancienne. Donc, un calque peut être le point de départ de plusieurs piles. L'évolution du projet se représente alors par un arbre, qui peut d'ailleurs apparaître à l'écran quand l'utilisateur recherche un calque par désempilement pour l'aider à se situer dans l'évolution du projet.

Remarque : il est préférable pour une question d'efficacité, qu' avant d'entraîner une nouvelle évolution parallèle, ces calques particulièrement importants fassent l'objet d'un calque de synthèse.

Pour en finir avec les possibilités de superposition des calques, signalons encore celle qui consiste à superposer un nouveau calque non plus sur un seul calque, mais sur deux ou même plusieurs calques. La nouvelle version est alors la compilation de versions plus anciennes, ce qui modifie encore le schéma de l'évolution.

### IX.3 Comparer des calques

-----

On vient de voir que l'éditeur propose l'archivage de toute l'évolution du projet, et de ce fait qu'il permet

d'assimiler les documents produits aux supports de conception. Or, nous avons mis en évidence l'importance qu'accorde un concepteur à la comparaison des différents supports de conception dans sa méthode de travail. Il est donc tout naturel que notre éditeur se charge, en plus de l'enregistrement des calques, de la gestion de leur comparaison.

La première facilité qui sera accordée à l'utilisateur est la visualisation simultanée de plusieurs calques d'esquisse. L'écran est alors divisé en neuf cases, dont une seule active. C'est effectivement dans celle-ci que le curseur spatial reste, et que se retrouve tout l'espace de création. Les huit autres cases contiennent chacune un calque d'esquisse créé et enregistré précédemment (voir figure 25).

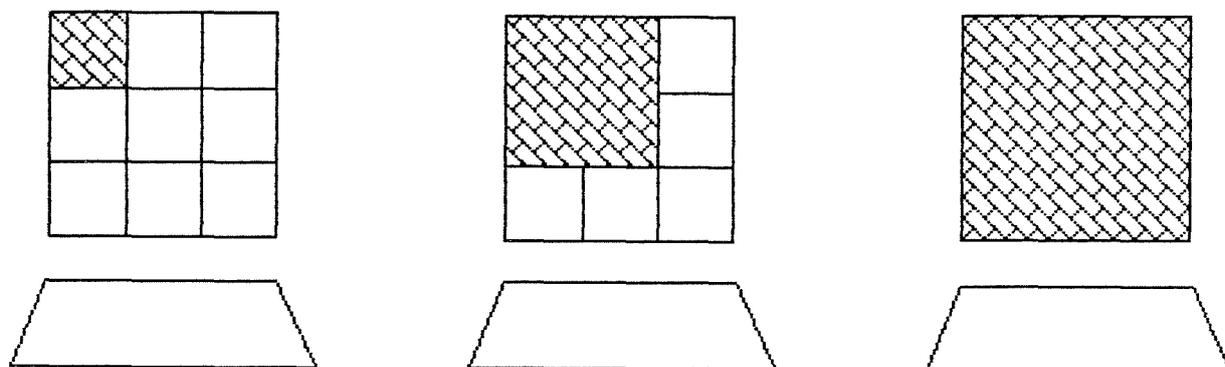


Figure 25

Remarque : l'emplacement de la case active par rapport autres peut être déterminé par l'utilisateur lors de l'initialisation personnalisée du logiciel.

Si l'utilisateur clique sur une case non active, le projet s'y trouvant est copié dans la case active où un nouveau calque est placé. Grâce à cette possibilité, l'utilisateur peut comparer visuellement huit calques, et en choisir un pour le faire évoluer.

Si l'utilisateur estime que l'espace de création est trop petit, il a la possibilité d'augmenter la taille de la case active en se contentant de cinq cases non actives (voir figure 25). S'il estime que l'espace de création est toujours insuffisant, il doit alors renoncer à la possibilité de comparaison, pour que tout l'écran soit de nouveau consacré à la création (voir figure 25).

Une extension intéressante est la possibilité de rendre plus d'une seule case active. Dans ce cas, l'espace de

création est multiplié, le curseur spatial se retrouve dans toutes les cases actives, et toutes les actions sur l'espace (le déplacement du curseur ou l'introduction d'un volume par exemple) sont visualisées dans celles-ci. Cette possibilité permet de faire évoluer simultanément et en parallèle jusqu'à neuf versions d'un même projet, ce qui peut être utile quand elles ne diffèrent que par quelques menus détails, et que le concepteur éprouve des difficultés à les départager.

Remarque : il ne faut pas confondre la notion de calque d'esquisse avec celle de couches, parfois appelées aussi calques par certains programmes de CAO classiques.

Chaque calque d'esquisse d'un même projet correspond à une version différente de celui-ci, et peut éventuellement le contenir en entier. Par contre, toutes les couches d'un projet ne correspondent qu'à une seule version, et ne contiennent qu'une partie du projet. Elles doivent donc être toutes superposées pour voir le projet dans son ensemble.

Généralement, chacune d'elle est destinée à une catégorie bien précise de personnes, comme les différents corps de métiers par exemple. C'est ainsi qu'il existe souvent une couche reprenant le schéma électrique, une couche pour le gros oeuvre, une autre contenant toutes les indications relatives aux menuiseries, etc... .

Cette notion de couche, si elle est particulièrement intéressante dans un éditeur de composants constructifs, n'est d'aucune utilité dans l'éditeur de volumes.

## CHAPITRE X : Les outils

---

L'éditeur de volumes permet à l'utilisateur de créer des volumes directement en trois dimensions, en laissant indéterminés les dimensions et les composants constructifs, et permet encore l'archivage et la comparaison de tous les documents produits. Par conséquent, on peut dire qu'il répond strictement aux exigences de la conception mises en lumière antérieurement.

Pourtant, nous ne disposons que du squelette de l'éditeur, en ce sens que bon nombre de facilités pourraient encore être offertes à l'utilisateur pour son travail de conception. Nous allons en proposer quelques unes dans ce chapitre, tout en admettant qu'elles ne constituent en aucun cas une liste exhaustive. Une expérimentation poussée de notre logiciel risque effectivement de mettre en évidence l'absence de nombreux outils.

Remarquons encore que reprendre systématiquement tous les outils que proposent les logiciels de CAO classiques, ou pire encore ceux des logiciels de DAO (Dessin Assisté par Ordinateur), constitue à nos yeux une démarche qui n'est pas acceptable étant donné l'orientation très précise de notre éditeur, d'autant plus que multiplier les outils ne peut que compliquer l'interface utilisateur.

### X.1 Les outils classiques de manipulation

---

On retrouve sous cette dénomination tous les outils qui permettent d'agir sur un élément du projet (volume, surface, ligne, point), après l'avoir sélectionné. Soulignons le fait que l'activation d'un tel outil implique toujours une mise à jour de la base de données. Ils sont dits "classiques" parce qu'ils sont proposés sous une forme ou sous une autre dans la plupart des logiciels de CAO dignes de ce nom.

La désignation de l'objet à manipuler est une opération commune à tous les outils, et peut donc se faire préalablement au lancement d'un outil particulier. Trois possibilités peuvent être proposées :

- 1) la première consiste à pointer avec la souris sur un

objet. Il apparaît alors dans une autre couleur, ou en surbrillance avec un système monochrome. Remarquons que s'il a été éclaté par l'outil de décomposition décrit précédemment, c'est un des sous-objets qui est désigné par le pointage, et non l'objet tout entier, et l'effet de l'outil est tout autre :

- 2) la deuxième solution est une extension de la première, et permet de sélectionner plusieurs objets en les pointant successivement, afin qu'ils subissent simultanément l'effet de l'outil ;
- 3) la dernière possibilité entraîne la sélection de tous les objets inclus dans un parallélogramme rectangle fictif spécifié à cette fin.

Nous pouvons maintenant passer en revue les principaux outils de manipulation.

#### X.1.1 L'outil de déplacement (effectuant des translations)

Après le lancement de la commande, le point ayant servi à la désignation reste collé au curseur. L'objet tout entier suit donc le déplacement du curseur, jusqu'au moment où l'utilisateur confirme sa nouvelle position par la touche de validation (première touche de la souris).

Pour se convaincre de l'influence de l'éclatement d'un objet sur cet outil, il suffit d'appliquer le principe de déplacement à un volume, et de comparer le résultat avec ce que peut donner l'application stricte du même principe à une des faces de ce volume, ou à une de ses arêtes, ou encore à un de ses sommets (voir figure 26).

#### X.1.2 L'outil d'orientation (effectuant des rotations)

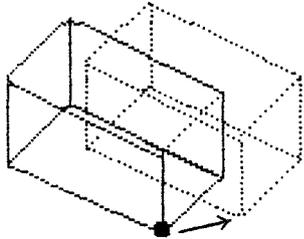
Après le lancement de la commande, le programme demande à l'utilisateur de définir un axe de rotation. Ceci fait, le point ayant servi à la désignation tourne autour de l'axe pour suivre le mouvement du curseur, et entraîne à sa suite l'objet.

Ici aussi, il est intéressant de comparer l'effet de l'outil sur un volume, avec son effet sur les faces ou les arêtes de celui-ci.

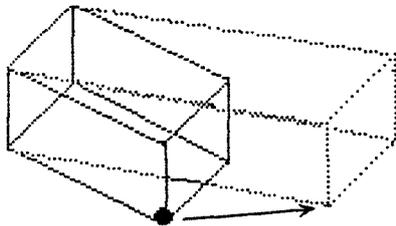
#### X.1.3 L'outil de reproduction

Tout comme l'outil de déplacement, le curseur reste collé au point de désignation après le lancement de la commande. La différence avec cet outil réside dans le fait que c'est une copie de l'objet sélectionné qui suit ici le déplacement du curseur, et non l'objet proprement dit.

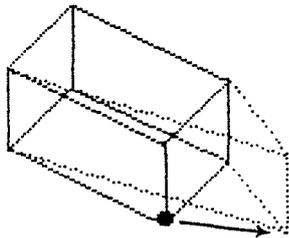
Comme tous les autres outils, son effet sur un objet quelconque varie en fonction de son degré d'éclatement.



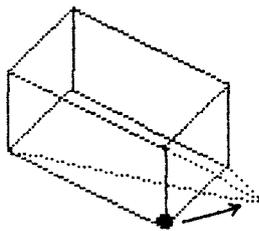
volume non décomposé



volume décomposé en faces



volume décomposé en arêtes



volume décomposé en sommets

Figure 26

#### X.1.4 L'outil de suppression

Comme son nom l'indique, cet outil sert à supprimer du projet l'objet qui a été désigné par le pointage. Une confirmation de la suppression sera exigée. L'utilisateur pourra donc revenir éventuellement sur sa décision après la visualisation de l'élément sélectionné en surbrillance.

Remarque : la gomme, toujours présente en DAO, n'a pas de sens en CAO. En effet, on aurait quelques difficultés pour définir le type d'un objet partiellement effacé, afin de le stocker dans la base de données caractérisant les logiciels 3D.

#### X.1.5 L'outil de modification de taille (effectuant des agrandissements et des réductions)

L'activation de cet outil entraîne, on s'en doute, une modification de la taille de l'objet sélectionné. Mais il est à noter que, contrairement au zoom qui sera développé un peu plus loin, cet objet est agrandi ou diminué séparément par rapport au projet, et remplace son ancienne version dans la base de données.

L'utilisateur doit désigner un point de l'objet qui permet au programme de le rattacher au projet lorsqu'il est modifié, exactement comme pour l'introduction d'une longueur de référence locale (voir chapitre VIII). Pour la spécification de la valeur de l'agrandissement, c'est encore la solution interactive qui sera choisie. Il suffira en effet que l'utilisateur laisse la touche de validation de la souris enfoncée sur l'icône ad hoc (+ et - par exemple), pour agrandir ou diminuer progressivement l'objet d'un nombre d'unités internes prédéterminé.

On peut encore imaginer que cet outil puisse agir séparément dans les trois directions.

Encore une fois, l'éclatement d'un objet joue un rôle prépondérant dans la modification de sa taille.

## X.2 Les outils classiques de présentation

---

Ces outils sont tout aussi classiques que les précédents étant donné leur présence courante dans les logiciels actuels. Par contre, ils n'agissent pas sur les éléments du projet, et donc sur la base de données correspondant à ce dernier, mais facilitent tout simplement le travail de l'utilisateur.

### X.2.1 Le zoom (avant et arrière)

Cet outil, dont nous avons déjà eu l'occasion de parler, permet à l'utilisateur d'agrandir ou de réduire la vue qu'il a du projet. C'est bien la vue du projet qui est modifiée.

et non le projet lui-même, ce qui veut dire qu'il n'y a pas de mise à jour de la base de données.

Rappelons encore que le zoom agit sur la trame, mais non sur la grille.

Pour des raisons évidentes, le zoom peut être regroupé avec les fonctions qui règlent les caractéristiques de la vue.

### X.2.2 L'annulation (UNDO)

L'activation de cette fonction par la touche ESC du clavier ou par un "click" sur une icône entraîne l'annulation de la dernière commande. Cette fonction est donc apparemment très simple à utiliser, encore faut-il préciser ce que l'on entend exactement par dernière commande.

On pourrait effectivement considérer que l'utilisateur change de commande chaque fois que la touche de validation est enfoncée, ou chaque fois qu'il retourne dans le menu pour sélectionner quelque chose. Comme ces deux notions de la dernière commande peuvent être aussi intéressante l'une que l'autre, nous proposerons une fonction d'annulation à deux profondeurs.

C'est ainsi qu'en "cliquant" une première fois sur la commande (ou en pressant ESC), c'est l'action correspondant à la dernière validation qui est annulée. En la "cliquant" une deuxième fois, ce sont toutes les actions correspondant à la même fonction qui sont annulées. Le troisième "click" annule le deuxième, et le quatrième annule le premier, ce qui permet de revenir à la situation initiale. Par exemple, l'utilisateur vient d'introduire en série un certain nombre de cylindres différents. Le premier click annule le dernier cylindre validé, le deuxième annule tous les cylindres relatifs à la dernière série, le troisième les fait tous réapparaître à l'écran sauf le dernier, et le quatrième restaure la situation de départ (voir figure 27).

Cette fonction est extrêmement importante, car en garantissant à l'utilisateur la récupération de ses fausses manoeuvres, elle l'incite à explorer de nouvelles possibilités.

### X.2.3 La fonction d'impression

Il va sans dire que le concepteur désirera à un moment donné une impression sur papier de son projet. Il pourra alors avoir une vue d'ensemble du projet, non plus limitée à l'écran, et de plus, il pourra transmettre le document aux personnes intéressées.

Cette fonction lui permettra donc de lancer l'impression sur la table traçante ou sur l'imprimante, non sans avoir spécifié au préalable tous les paramètres que nécessite leur pilotage.

Remarque : cette fonction est évidemment commune à celle de l'éditeur de composants architecturaux.

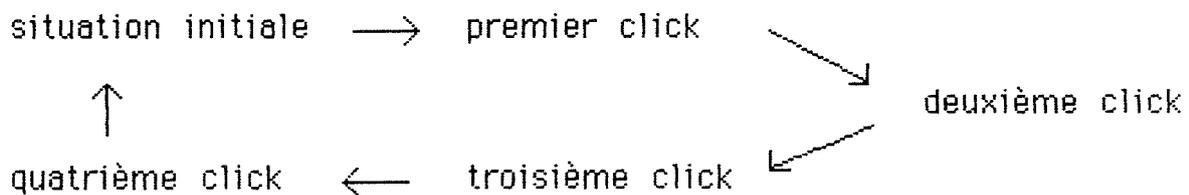
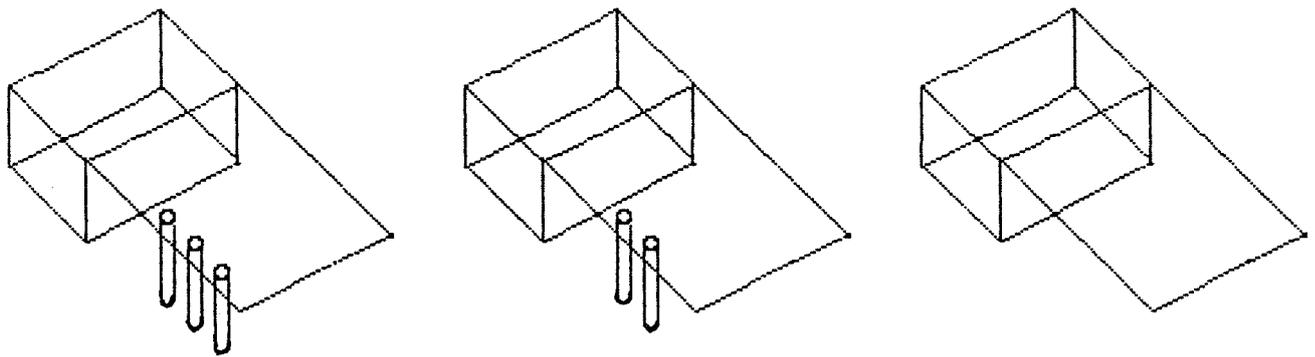


Figure 27

### X.3 Les outils de conception architecturale

---

Alors que la présence de ces outils n'est pas indispensable dans un éditeur de composants constructifs, leur contribution à faciliter le travail de conception proprement dite est telle qu'ils doivent impérativement figurer au menu de l'éditeur de volumes.

Le seul outil que nous proposerons dans cette première approche est la gestion des intersections. Cet outil doit indubitablement être offert par un logiciel d'aide à la conception architecturale, étant donné que bon nombre de volumes architecturaux sont générés, déjà même dans l'esprit du concepteur, par l'intersection de deux ou de plusieurs volumes.

Avant de décrire cet outil, notons qu'il s'applique aux intersections qui apparaissent entre les objets au sens général du terme, et pas entre les volumes uniquement.

Quand l'utilisateur introduit un nouvel objet, le feedback du système consiste à dessiner non seulement cet objet, mais également les intersections qu'il peut éventuellement avoir avec les autres objets du projet. Ces intersections peuvent d'ailleurs être accentuées par des aplats de couleur ou des hachures. A partir de là, pour chaque objet A ayant une partie commune avec le nouvel objet B (voir figure 28), le concepteur peut :

- 1) laisser l'intersection et laisser les deux objets,
- 2) laisser l'intersection, retirer A, et laisser B, c'est-à-dire ne laisser que le nouvel objet,
- 3) laisser l'intersection, laisser A, et retirer B (voir figure 28-4),
- 4) laisser l'intersection et retirer les deux objets (voir figure 28-3),
- 5) retirer l'intersection et laisser les deux objets (voir figure 28-1),
- 6) retirer l'intersection, retirer A, et laisser B (voir figure 28-2),
- 7) retirer l'intersection, laisser A, et retirer B, c'est-à-dire ne laisser que l'objet existant,
- 8) retirer l'intersection et retirer les deux objets, donc tout retirer.

L'utilisateur va donc devoir choisir une de ces stratégies. Pour éviter d'alourdir l'exploitation de la gestion des intersections par la procédure de choix, la stratégie qui consiste à ne rien retirer (cas 1) sera adoptée par défaut. De plus, elle sera valable pour tous les objets A, c'est-à-dire pour tous ceux qui ont une intersection avec le nouvel objet B.

Il faut noter que ces simplifications concernent la mise en oeuvre, et non les possibilités de la fonction. En effet, l'utilisateur aura toujours le loisir de choisir parmi les huit stratégies. Pour ce faire, il devra désigner successivement tous les éléments qui le gênent (intersection, A sans intersection, ou B sans intersection) pour qu'ils soient retirés du projet. S'il ne désigne aucun objet avant de passer à la commande suivante, il accepte alors la stratégie par défaut.

On se doute bien qu'ici encore l'éclatement d'un volume en surfaces, en lignes, ou en points, peut avoir des conséquences non négligeables sur la gestion des intersections. Observons, par exemple, la différence d'intersection si le volume A est décomposé en faces ou non avec le cas 5 (voir figure 29).

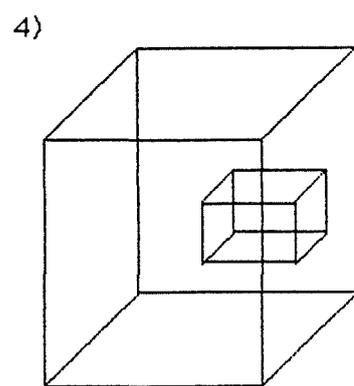
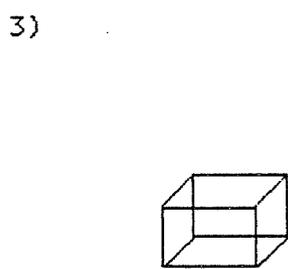
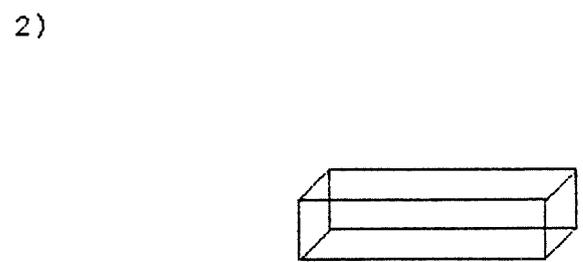
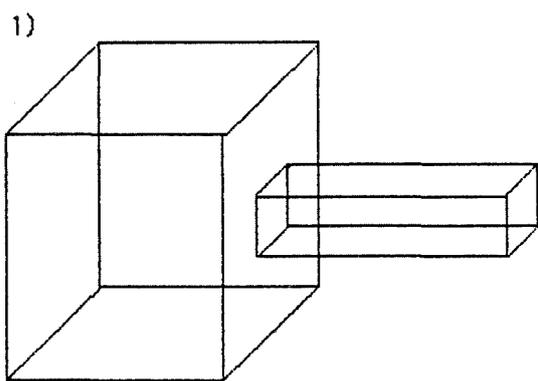
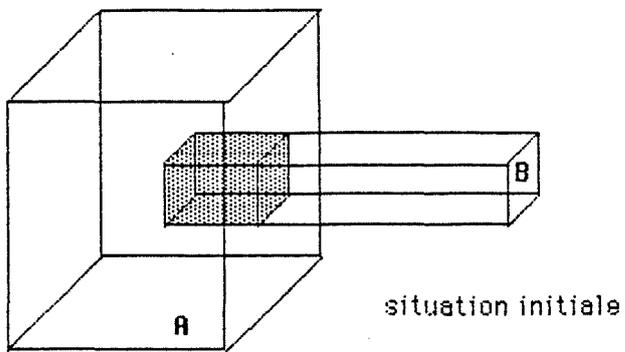
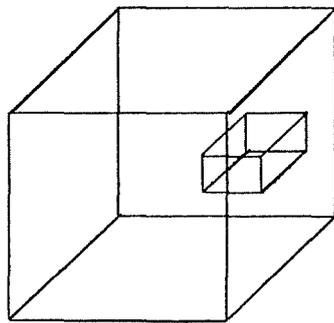
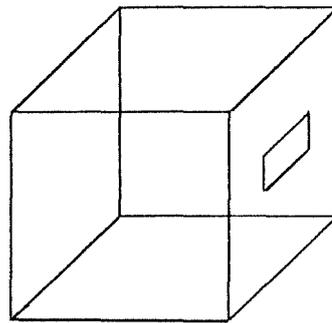


Figure 28



A non décomposé



A décomposé en faces

Figure 29

#### X.4 Les accessoires de conception

---

On a vu que la phase de conception aboutissait à l'élaboration d'un avant projet destiné au client, et que le concepteur n'hésitait pas à le mettre en valeur en employant quelques artifices graphiques (végétation, personnages). D'autre part, l'emploi de ces artifices aide le concepteur à mieux estimer l'ampleur de son projet par rapport à la dimension humaine. Par conséquent, l'éditeur doit peut-être s'écarter un peu des objets strictement architecturaux, et proposer quelques objets permettant d'exprimer leur environnement.

Remarque : il s'agit toujours d'objets spatiaux, et non de dessins 2D superposés sur les diverses vues du projet. Cela signifie que ces objets seront probablement représentés de manière plus symbolique (ce qui n'enlève d'ailleurs rien à leur rôle) mais qu'une fois spécifiés, ils seront visibles quelle que soit la vue du projet.

##### X.4.1 Le générateur de mousse

Il s'agit de l'équivalent 3D de la bombe de peinture que l'on trouve dans beaucoup d'éditeurs 2D. Ce générateur de mousse permet en effet d'introduire dans l'espace une multitude de points en expansion vis-à-vis du curseur spatial, afin de représenter la végétation basse notamment. Il peut même être un peu plus sophistiqué, afin d'éviter que les points projetés ne traversent les parois du projet.

#### X.4.2 Le générateur d'arbres

Si la mousse peut symboliser la végétation basse, cet outil permet d'introduire les arbres sur pied. Pour ce faire, l'utilisateur ne doit spécifier que deux points : l'emplacement de l'arbre, et sa hauteur. Quant à la circonférence du feuillage et celle du tronc, elles sont déduites de la hauteur.

#### X.4.3 Le générateur de personnages

Ce générateur permet d'obtenir la représentation symbolique d'un personnage. Seul son emplacement devra être spécifié, car contrairement aux arbres, la hauteur des gens ne varie que très peu. Par conséquent, elle pourra être calculée automatiquement par le programme, en admettant toutefois quelques variations au sein d'une fourchette. Remarquons que ce calcul automatique n'est possible que si le dimensionnement du projet a été réalisé, et que son résultat change si l'utilisateur modifie les longueurs de référence.

#### X.4.4 Le générateur d'automobiles

Le principe est identique au générateur de personnages décrit ci-dessus. Il suffira que l'utilisateur désigne un point pour qu'une automobile symbolique soit introduite à cet endroit.

### X.5 Conclusions

-----

Nous venons de proposer quelques outils qui peuvent s'avérer fort utiles lors de la phase de conception. Nous sommes bien conscients que beaucoup d'autres outils plus intéressants les uns que les autres peuvent encore être proposés. Cependant, il ne faudrait pas que leur liste s'allonge de manière excessive. En effet, multiplier les outils risque de ralentir l'utilisation de l'éditeur (en obligeant l'utilisateur à traverser une foule de menus en cascade, par exemple), ou de compliquer son interface utilisateur (en obligeant l'utilisateur à retenir une série de commandes permettant des raccourcis, par exemple).

Pour éviter ces désagréments, l'utilisateur pourra puiser dans l'ensemble des outils ceux qui sont susceptibles de l'aider le plus pour le travail qui va suivre. Il se constituera en quelque sorte une boîte à outils, peu fournie

mais accessible très rapidement, en enfonçant simultanément les deux touches de la souris, par exemple. L'utilisateur pourra toujours sélectionner un outil n'appartenant pas à la boîte, par la méthode devenue classique du click sur l'icône adéquate.

Ce chapitre clôture l'élaboration de notre solution. Nous pouvons donc passer maintenant à la conclusion générale.

C O N C L U S I O N

Avant de tirer les conclusions qui s'imposent, passons en revue les différentes étapes qui nous ont permis d'aboutir à la solution proposée.

Nous avons commencé par analyser l'offre du marché de la CAO pour architectes en étudiant de plus près trois logiciels : CADVANCE, ARCHICAD, et STAR UX. Cette étude a permis de mettre en lumière leur incapacité à couvrir l'intégralité du processus architectural. En effet, alors qu'ils assistent admirablement l'architecte dans les derniers stades de ce processus, ils avouent très vite leurs limites lors de la conception proprement dite du projet. Nous avons alors décidé de remédier à cette faiblesse, non sans en avoir discuté de l'opportunité. Pour cela, nous avons analysé plus en profondeur la démarche architecturale afin de dégager ses caractéristiques essentielles. Nous avons alors pu élaborer progressivement notre solution, compte tenu de ces propriétés.

L'éditeur que nous proposons ne permet que de travailler dans une vue en perspective (réelle, axonométrique, cavalière). Les principaux objets manipulés sont des volumes, mais quelques surfaces, la droite, et le point figurent également au menu. Ce qui est plus particulier par rapport aux autres logiciels du même type, c'est la possibilité de laisser indéterminés les dimensions du projet et les composants constructifs. Lorsqu'il est arrivé à maturité, il peut recevoir a posteriori des dimensions, et les éléments qui le composent peuvent être affectés à des composants constructifs.

Maintenant que nous avons proposé une solution, nous n'aurons pas la prétention d'affirmer qu'elle va régler tous les problèmes des architectes. Au contraire, il convient d'être très prudent quant à son applicabilité en milieu réel. Une expérimentation poussée de l'éditeur s'impose donc, et s'il s'avère qu'il est inutilisable, il aura au moins eu le mérite de susciter cette expérimentation. En effet, qu'ils soient bons ou mauvais, les résultats pourront toujours servir de base à une nouvelle solution.

Il importe également de noter que l'éditeur de volumes doit faire partie d'un logiciel plus global intégrant encore bien d'autres modules. Parmi ceux-ci, on pourrait citer le "scanner" de croquis qui permettrait de transformer l'image d'un croquis en un projet volumétrique. Un autre module tout

aussi indispensable serait un éditeur 2D présentant les possibilités d'indétermination que notre éditeur de volumes. C'est avec lui que se réaliserait la conception des bâtiments de volumétrie simple. Un module dont nous avons déjà parlé est l'éditeur d'organigrammes qui avancerait encore la première intervention de l'ordinateur dans le processus architectural. Ce n'est donc qu'avec cette multitude de modules que l'intégralité du processus architectural pourra enfin être informatisée.

Ajoutons enfin que nous avons constamment mis l'accent sur la convivialité de l'éditeur lors de son élaboration. Pourtant, son apprentissage requiert inévitablement un effort de la part de l'utilisateur, tant sur le plan de la mémorisation des différentes commandes, que sur celui de l'habitude à acquérir pour travailler sans dimension dans une perspective, mais quelle satisfaction celui-ci en tirera passé ce cap d'apprentissage !

\*

\*

\*

B I B L I O G R A P H I E



