Institutional Repository - Research Portal
Dépôt Institutionnel - Portail de la Recherche
researchportal.unamur.be
University of Namur

THESIS / THÈSE

MASTER EN SCIENCES INFORMATIQUES

Conception interactive d'objets spatiaux

Podevyn, François

Award date:
1986

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.

Download date: 05. Jul. 2025

C. CHERTON

CONCEPTION INTERACTIVE D'OBJETS SPATIAUX

Mémoire présenté par

François PODEVYN

en vue de l'obtention
du grade de licencié et
maître en informatique

Maître de mémoire : Cl. CHERTON

Facultés universitaires Notre-Dame de la Paix.
Institut d'informatique. Année 1985-1986.

inverse de SAUVER?
inverse d'ELAGUER
comment réutiliser
une déf. d'objet dans
un autre travail?
ou créer un objet à
partir d'un autre
(COPY?) Rem agréable?
ou en parle p 115
mais pas p106 ni 107

134 la fatigue ou le
manque de temps?
ou autre chose...

p3 est-ce réaliste, ou un possible (o-p?) p11
- vérifier assertions s/ Mar Draw
p39 ?
p60 en clos aussi. Remèdes??
p62 dépl. curseur par souris
~~p67~~ ?
p67?
p78
p83!?
p85 ???
p86 ????
p93 ex de ? particulier
p97 ??
p99??

p103 ?!?
p108 !!! ? !!2, !!!
devrait ?14
fig entre !!0 et !!!
p132 !!!!
p141 !!!!
(travailler seul?)
p142 ?!!!

Nous remercions spécialement Monsieur Cherton pour avoir assuré avec confiance la maitrise de ce mémoire.

Nous remercions également tous les membres de l'Institut d'Informatique pour leur amabilité et leur patience.

Nous remercions enfin et surtout nos amis et les membres de notre famille sans lesquels la réalisation de ce document n'eut pas été possible.

1 INTRODUCTION

Introduction

C.I.O.S. signifie conception interactive d'objets spatiaux. Il désigne l'application développée dans le cadre de ce mémoire et le programme qui en résulte.

Nous avons choisi ce nom parce qu'il est joli et "fait sérieux". Mais ne nous méprenons pas : le but de cette étude n'est pas de réaliser un logiciel opérationnel. Nous avons décidé de donner un nom au programme simplement parce que c'est plus pratique que d'écrire chaque fois "conception interactive d'objets spatiaux".

C.I.O.S. est un programme graphique 3D.

Le domaine du graphisme en trois dimensions est immense. Il recouvre des applications dans les domaines les plus variés : scientifiques, techniques, artistiques, médiatiques, etc. Plutôt que d'étudier une application particulière, nous avons choisi de nous intéresser au fait en lui-même d'utiliser le graphisme 3D, et plus précisément à la façon dont il est possible de créer des images 3D.

Nous avons encore limité notre champ d'intérêt aux possibilités de créer interactivement des objets à travers leur image 3D.

Le premier souci de cette étude sera de privilégier une approche générale du problème, en considérant les données en dehors de tout

contexte.

Notre second souci sera d'implémenter un outil capable d'expérimenter le fruit des suppositions que nous serons amenés à formuler.

Notre troisième souci sera de proposer une démarche créative complémentaire à celles qui existent déjà.

A cette fin, nous nous sommes documemtés sur les outils existants. Nous avons étudié les produits destinés à la réalisation de dessins techniques et de plans d'architecture. Cette étude a débouché sur la définition d'une approche complémentaire comprenant la perception spatiale de l'objet, un outil de précision dans l'espace et la décomposition en sous-objets.

Etant donné le temps qui nous était imparti et le matériel didactique mis à notre disposition, il nous a semblé raisonnable de limiter le concept d'objet à ses formes, à son contour.

C.I.O.S. ne traitera donc seulement que des arêtes et pas de surfaces ni de volumes.

Les objets traités seront simples et conçus sans aucune contrainte technique telle que l'aérodynamisme ou la résistance mécanique. Les exemples seront puisés parmi les objets courants de la vie : mobilier, habitation, etc...

En conclusion, le but de C.I.O.S. est de créer interactivement des objets à travers leur image 3D, en respectant les trois objectifs suivants :

conserver une approche générale du problème ;

proposer une approche complémentaire à celles qui existent déjà ;

faire tourner un outil expérimental.

La première partie de ce mémoire contient la description de la situation actuelle et l'étude de quatre logiciels existants.

La deuxième partie contient la définition et l'explication de l'approche complémentaire que propose CIOS.

La troisième partie contient l'analyse fonctionnelle et l'architecture logique de CIOS.

2 PREMIERE PARTIE

2.1 INTRODUCTION

Dans cette partie, nous allons d'abord nous demander qui est concerné par l'expression graphique. Nous essayerons de définir les besoins de chacun et les techniques traditionnelles qui sont mises à sa disposition.

Ensuite, nous verrons en détail comment l'informatique a répondu à ces besoins. Nous étudierons quatre logiciels existants qui sont représentatifs de quatre classes d'utilisateurs différents.

2.2 QUELS SONT LES GENS CONCERNES ?

Ces gens sont en particulier l'ingénieur, le dessinateur, le designer, le décorateur, l'architecte, le paysagiste, l'urbaniste,..., quelqu'un qui crée des espaces, qui conçoit des objets.

Pour ce faire, il utilise principalement trois techniques : le plan, coupe et élévation, la perspective et la maquette. Passons-les en revue.

2.2.1 TECHNIQUES TRADITIONNELLES

La maquette est un modèle réduit de l'objet. C'est le moyen d'expression le plus riche car le plus proche de la réalité, mais c'est aussi de loin le plus onéreux en temps et en matériel. Il ne sera mis en oeuvre que si la dépense est justifiée et souvent pour exprimer le résultat d'un processus de création, non pour aider ce processus lui-même. Retenons deux façons de réaliser une maquette. La première consiste à découper dans une matière rigide de faible épaisseur toutes les faces de l'objet, puis de les assembler dans l'espace. La seconde relève plus de la sculpture : il s'agit de modeler la forme de l'objet en taillant dans un bloc de matière friable, ou en la moulant dans une matière

maléable.

L'expression en plans, coupes et élévations est apparemment le moyen le plus pauvre puisqu'il n'exprime qu'une vision tronquée d'une réalité en trois dimensions. En fait, il a la particularité d'être le plus précis au point d'être indispensable pour la compréhension de celui qui va construire l'objet. Il permet en effet d'adjoindre des commentaires comme par exemple des dimensions ou des recommandations quant aux matériaux ou à la technique de construction, toutes choses qu'une maquette ne peut exprimer. Enfin, cette technique ne nécessite aucune autre dépense qu'une feuille de papier et un crayon. Elle a néanmoins ses limites : il existe des objets qu'elle ne parvient pas à décrire complètement, et en général, elle est assez hermétique pour le non-initié.

La perspective est une solution intermédiaire : elle fournit à peu de frais une image de l'objet tel qu'on le verrait dans l'espace réel sous un angle particulier. Mais sa mise en oeuvre nécessite beaucoup de travail, et surtout, exige que l'objet soit déjà conçu et exprimé au moyen de plans et d'élévations. Il existe beaucoup de techniques de tracé de perspective, mais le principe reste le même : déterminer par une construction géométrique plane l'endroit sur la feuille de papier de chacun des sommets de l'objet.

2.2.2 DIFFERENTS BESOINS

Le choix de telle ou telle technique est lié au type d'application. Les domaines d'application vont du dessin industriel de pièces mécaniques, de circuits intégrés ou de circuits électriques, au design de mobilier, d'étalage ou d'automobile, en passant par la construction : architecture, urbanisme, aménagements intérieurs, jardins, relevés topographiques, tracés de routes...

Le dessinateur industriel par exemple, a besoin d'une grande précision dans son dessin qui sera quasi exclusivement en deux dimensions : plans, profils, coupes...

Le designer, par contre, se contenterait volontiers d'un dessin en trois dimensions, sans une trop grande précision. Il réalisera des maquettes de ses objets pour en étudier l'impact sur l'environnement et les gens.

Quant à l'architecte, il a besoin des deux : s'il conçoit mentalement en trois dimensions, il s'exprime habituellement en deux dimensions avec précision, et quelques fois, pour rendre compte d'une ambiance particulière ou de l'aspect global, en trois dimensions.

En règle générale, plus l'objet est technique, ou moins il est esthétique, plus on se rapproche

d'une conception et d'une expression en deux dimensions, et vice versa. Par exemple, le nouvel Hôtel de ville de Namur n'a certainement jamais été dessiné qu'en plan...

Vous l'avez deviné également, l'expression en trois dimensions est le moyen privilégié de communication avec le client ou l'utilisateur, alors que l'expression en deux dimensions l'est avec le constructeur ou le technicien.

Le cas où le concepteur d'un objet est également responsable de sa construction est particulier. On lui demande en effet d'en fournir ce qu'on appelle un métré qui énumère les quantités des différents matériaux nécessaires à sa réalisation. Cela signifie que lorsqu'un architecte trace sur un plan deux lignes parallèles symbolisant un mur, elles représentent en réalité un certain volume de maçonnerie de tels éléments. Bien que cette quatrième dimension n'entre pas dans la phase plastique de la conception d'un objet, elle a son importance dans le résultat final de cette conception. Nous en tiendrons compte dans l'étude des logiciels existants.

2.3 EVOLUTION DE LA SITUATION

Les informaticiens purs auront reconnu dans l'élaboration du métré une tâche routinière typiquement automatisable. De fait, l'informatique n'a pas tardé à se pencher sur les problèmes liés à la conception graphique, et cela dans le cadre de ce qu'on appelle la Conception Assistée par Ordinateur.

Durant toute une période, on a pu croire que la C.A.O. dans le domaine graphique était réservée à la création d'animations télévisées ou à la recherche spatiale, tant le coût des logiciels était élevé et le matériel utilisé gigantesque...

Puis sont apparus sur le marché divers produits très performants, specialisés pour architectes et ingénieurs en construction. Ils permettent une conception en trois dimensions d'objets spaiaux et en fournissent des images en plans, en élévations, en coupes et en perspectives. A ces objets sont associés des matériaux, ce qui permet au système qui en calcule les volumes d'effectuer le métré quantitatif.

Ces logiciels nécessitent pour tourner des machines d'au moins un mégabyte et sont à la portée de gros bureaux d'étude. En Belgique, le

chef de file de ces produits s'appelle STAR.

Simultanément sont apparus une pléiade de logiciels 2D rivalisant de précision et de fonctions diverses, destinées au dessin technique et même aux plans d'architectes. Ils proposent plusieurs catalogues techniques d'objets prédessinés tels que des composants électriques, du mobilier standard, etc.

Le dernier-né de ces logiciels est le PC-DRAFT (concurrent direct du mieux connu AUTO-CAD) qui tourne comme les autres sur un ordinateur personnel de 512 kilobytes et est accessible à une plus large catégorie de gens.

Enfin, je ne peux omettre de citer les "logiciels intégrés" tournant sur micro ou PC et proposant chacun une application graphique 2D. sans aucune précision et digne du plus pur gadget, mais non sans utilité pour le non professionnel du dessin.

Celles qui sont devenues le modèle du genre sont MAC-DRAW et MAC-PAIN et comme leurs noms l'indiquent, sont disponibles sur un micro-ordinateur bien connu.

Nous allons développer succinctement quelques-uns de ces logiciels, car ils vont nous permettre de

situer la discussion qui va suivre par rapport à des concepts existants.

Nous nous pencherons successivement sur MAC-DRAW, PC-DRAFT, STAR et ARC+, un logiciel israélien qui a fait son apparition sur le marché belge dans le courant du mois de juillet 1986 et qui pourrait être le logiciel de demain de tous les architectes...

2.4 QUELQUES LOGICIELS EXISTANTS...

A partir d'ici, lorsque nous parlerons d'objets, il ne s'agira plus d'objets réels, mais de la notion informatique d'objet, qui sert à symboliser ces objets.

Tous ces logiciels possèdent en commun une série de fonctions destinées à gérer les objets créés. Ces fonctions sont les fonctions traditionnelles qu'un système d'exploitation offre pour les fichiers : créer, rentrer, sortir, recopier, sauver, supprimer et imprimer sur un support particulier. Bien que ces fonctions diffèrent d'un logiciel à l'autre par leur modalité d'exécution, nous ne nous y intéresserons pas car cela sort de l'objectif fixé par ce travail.

Ils se distinguent par contre nettement l'un de l'autre par l'ensemble de fonctions de création qu'ils proposent. Pour chacun de ces logiciels nous étudierons le support qu'il offre pour la création, c'est-à-dire l'espace dans lequel on crée et la façon dont il est repéré. Il sera intéressant de comparer la façon dont est considéré l'objet, la façon dont il se décompose, dont on le construit et dont on le manipule. On se demandera quelle est l'unité de création, c'est-à-dire le plus petit élément que l'on peut créer.

On observera la façon dont on le crée, les facilités offertes pour cette création, et enfin, la façon dont on voit l'objet, dont on contrôle sa conception.

Par contre, nous ne nous arrêterons pas non plus sur les fonctions d'"habillage" offertes par ces logiciels. Elles correspondent à la phase de finition de la conception d'un objet et permettent de le présenter en couleur ou rehaussé de différents rendus, de cotations, de commentaires.

Le but n'étant pas de faire une description complète et objective de ces logiciels, nous prendrons la liberté de choisir parmis leurs caractéristiques celles que nous allons développer, en espérant que ce choix subjectif ne leur cause pas de préjudices.

2.4.1 MAC-DRAW

Bien que ce logiciel ne soit que du Dessin Assisté par Ordinateur en deux dimensions, il présente un intérêt quant à la façon dont il permet de manipuler les objets créés.

Nous étudierons seulement MAC-DRAW car contrairement à MAC-PAIN, c'est d'après le manuel un logiciel de dessin "technique" - profitons de cette intermède pour rire un bon coup. Ceci dit sans méchanceté, reconnaissons que toutes les figures de ce mémoire ont été réalisées grâce à cet outil.

2.4.1.1 SUPPORT

MAC-DRAW propose une grande feuille de papier virtuelle sur laquelle se déplace une fenêtre qui n'est autre que l'écran. Cette feuille est limitée au départ et peut être vue toute entière ou en partie. La taille minimum de la fenêtre correspond à une vue en vraie grandeur de la feuille. La feuille est vaguement tramée tous les vingt-et-un vingt-septième de centimètre. Une trame est une grille carrée que le dessinateur trace pour se repérer plus facilement sur un plan.

2.4.1.2 ELEMENT DE BASE

Les éléments de base de la création sont la ligne droite, la courbe, le rectangle, l'ellipse, le polygone irrégulier et le trait quelconque.

2.4.1.3 CREATION

Le premier attrait de cet outil est la facilité de création de ces éléments : il suffit de désigner deux points au moyen de ce formidable instrument qu'est la souris, celle-ci déplace à l'écran un curseur avec une précision de deux dixièmes de vingt-et-un vingt-septièmes de centimètre (un bon milimètre). Mais surtout, lorsqu'on a choisi le premier point, l'élément se dessine et se transforme d'après la position du curseur jusqu'à ce qu'on ait choisi le deuxième point, ce qui permet de contrôler de visu la position des éléments.

2.4.1.4 MANIPULATIONS

Le deuxième attrait vient du fait que chaque élément créé possède un certain nombre de points caractéristiques appelé "poignées" qui permettent de les déformer avec le même contrôle que lors de la création. On peut par exemple allonger un rectangle jusqu'à un certain endroit en le "saisissant" simplement par une de ses poignées.

2.4.1.5 STRUCTURE DES OBJETS

Une particularité des éléments est qu'ils sont opaques, et que si deux éléments se recouvrent, il

faut choisir celui des deux qui sera devant et qui cachera l'autre. Hormis ce fait, tous les éléments sont visibles.

Enfin, les éléments, que l'on désigne simplement en se plaçant en leur centre, peuvent être associés pour former des groupes. Ces groupes peuvent être déplacés, répétés, supprimés, associés à d'autres groupes ou dissociés. On voit donc qu'il existe une certaine structure hiérarchique de groupes dans la composition d'un dessin. Néanmoins, on ne peut manipuler que le groupe qui se trouve au sommet d'une hiérarchie. Il faut donc dissocier les éléments d'un groupe si l'on veut les manipuler individuellement.

2.4.2 PC-DRAFT

Ce produit est également un logiciel graphique 2D, mais il est intéressant à étudier pour la façon dont il a modélisé la manière de travailler d'un dessinateur et pour la précision des outils qu'il lui fournit.

L'objectif du PC-DRAFT est de procurer aux professionnels un outil performant permettant de réaliser leurs dessins techniques en se passant de latte, d'équerre, de rapporteur, translateur, compas, calque, gomme, etc.

2.4.2.1 SUPPORT

La feuille virtuelle que propose PC-DRAFT est infinie, mais il faut en fixer pour chaque dessin la taille physique qui correspond au format d'impression. Il faut également spécifier l'échelle de travail de chaque dessin. Par exemple: si vous choisissez un format A3 (29x42cm) et une échelle de 10% et que vous demandez de tracer une droite de 1 mètre, l'écran indiquera "100cm" et elle mesurera à l'impression 10cm, et vous serez limité à des objets de 4,20m sur 2,90m. L'écran-fenêtre n'a pas de limite théorique: on peut s'éloigner ou se rapprocher autant qu'on le veut de la feuille et en viser n'importe quel point. Donc, on travaille en vraie grandeur, en

donnant la dimension réelle, mais on ne voit jamais en vraie grandeur, sauf coup de chance.

Il existe également une trame, dont on peut choisir en grandeurs réelles l'écartement horizontal et l'écartement vertical.

2.4.2.2 ELEMENTS DE BASE

Les éléments de base de la création sont le point, la ligne droite, le polygone, le rectangle, le cercle et l'arc de cercle.

2.4.2.3 CREATION

Tous ces éléments se créent en spécifiant deux points, sauf l'arc de cercle et le cercle, dont on peut donner soit le centre et le rayon, le centre et un point, le rayon et deux points, le rayon et deux éléments tangents ou trois points.

On peut, comme dans MAC-DRAW, déterminer un point au moyen de la souris, avec une précision purememt visuelle. On peut même effectuer un zoom puissant afin d'accroître cette précision visuelle. Mais on peut surtout utiliser une des fonctions suivantes:

ABSOLUTE X,Y	permet d'introduire au clavier les coordonnées absolues du point.

RELATIVE X,Y	permet d'introduire au clavier les coordonnées relatives au point courant.
RELATIVE PHI,R	permet d'introduire au clavier les coordonnées polaires.
CURRENT POINT	permet de sélectionner le point courant.
END POINT	détermine l'extrémité d'un élément.
CENTRE	détermine le centre d'un élément.
MID-POINT	détermine le point-milieu entre deux éléments.
INTERSECTION	détermine l'intersection entre deux éléments.
FOOT POINT	détermine la projection orthogonale sur un élément.
TANGENT POINT	détermine l'intersection d'une droite passant par un point et tangente à un élément, avec l'élément lui-même.
POINT AT A DISTANCE	détermine le point à une distance d'un autre sur un élément donné.

NEW CURRENT POINT redéfinit le point courant.

Indépendamment de ces fonctions, il est possible d'activer ou de désactiver la trame. Si la trame est active, tout point introduit sera automatiquement identifié au point de la trame le plus proche. C'est utile lorsque l'on veut travailler avec précision au moyen de la souris, qui reste l'outil d'introduction de point le plus pratique.

2.4.2.4 MANIPULATIONS

Les manipulations d'objets sont classiques: déplacer, pivoter, changer l'échelle, répéter, supprimer, à l'exception du fait qu'elles utilisent les fonctions d'introduction de points vues précédemment et s'effectuent donc avec une grande précision. PC-DRAFT propose deux manipulations supplémentaires très utiles : couper un élément et prolonger un élément.

Couper un élément permet d'en créer deux à partir d'un seul, c'est-à-dire d'en rendre indépendantes deux parties complémentaires.

Prolonger un élément permet de le raccrocher à un autre ou tout simplement en modifier la taille réelle sans en changer l'échelle. En effet, si on change l'échelle d'un carré d'un mètre de côté, il apparaitra plus petit à l'écran et à l'imprimante, mais les lignes de cotation indiqueront toujours

un mètre.

2.4.2.5 STRUCTURE DES OBJETS

Un DESSIN PC-DRAFT est un ensemble d'OBJETS.
Un OBJET PC-DRAFT est soit un ELEMENT de base, soit un GROUPE.
Un GROUPE PC-DRAFT est un ensemble d'OBJETS.

Ce qui signifie que l'on retrouve sur un dessin des éléments de base isolés et d'autres groupés, que ces groupes peuvent être à leur tour groupés à d'autres groupes ou à d'autres éléments pour donner une hiérarchie de groupes similaires à celle de MAC-DRAW. Ici aussi, seul le sommet d'une hiérarchie peut être manipulé et il existe des fonctions de composition et de décomposition de groupe.

La différence entre un dessin et un groupe est que le premier est sauvé sur un fichier propre avec un nom. Quand un dessin est incorporé à un autre, il est considéré comme un groupe de celui-ci et sa structure hiérarchique est conservée.

Parallèlement à cette structure hiérarchique "logique", les dessins ont une structure "physique" de couches. Les différents objets composant un dessin sont situés dans les différentes couches du dessin de telle sorte qu'un objet peut se répartir dans plusieurs couches et qu'une couche peut contenir plusieurs objets...

Le dessin est donc une série de couches à la manière d'une série de calques que l'on superpose. Certaines couches sont invisibles, les autres sont visibles et peuvent servir à la construction du dessin, mais une seule n'est active à la fois, c'est-à-dire que les éléments que l'on rajoute, supprime ou manipule, appartiennent obligatoirement à cette couche.

EDITEUR GRAPHIQUE.

En fait, PC-DRAFT va plus loin, mais ça n'est pas dans le manuel d'utilisation que l'on trouve en Belgique...

Mais avant, nous devons rentrer un peu plus loin dans le détail des modalités de création.

Il faut d'abord savoir que tout objet de PC-DRAFT possède un point de référence. Les éléments de base en ont un par défaut. Celui du cercle par exemple se trouve en son centre. Il existe une fonction qui permet de changer la position de ce point.

Il sert comme son nom l'indique à référencer l'objet lors des manipulations. De plus, une

rotation, un changement d'échelle ou un déplacement déterminé se fera toujours par rapport à ce point de référence.

Ensuite, sachons que lorsque l'on associe plusieurs objets, éléments de base ou groupes d'éléments, pour en former un nouveau groupe, on a le droit de lui attribuer un nom. Par défaut, ce nom est vide, mais il existe. Enfin, le point de référence du nouveau groupe créé est automatiquement celui du premier composant qui a été désigné.

Lors de créations ou de manipulations d'objets, on est amené par le système à devoir introduire des valeurs numériques au clavier : un angle, une coordonnée, une longeur... Il est permis à ce moment-là d'écrire une expression numérique constante à la place de la constante numérique attendue. PC-DRAFT comprend très bien les symboles

'+', '-', '*', '/', 'sin', 'cos', 'tan', 'atn', p', 'log', 'qud', 'sqr'. Rien de surprenant jusqu'ici.

Cela devient intéressant lorsque dans cette expression, on peut remplacer les constantes par des variables, appelées pseudo-variables, qui font référence à des objets existants. Lorsque PC-DRAFT rencontre une telle variable, il demande à l'utilisateur de désigner interactivement un objet ou un point.

Il y a six pseudo-variables :

‹&X› référence la coordonnée X d'un point,
‹&Y› référence la coordonnée Y d'un point,
‹&AX› référence la coordonnée X du point courant,
‹&AY› référence la coordonnée Y du point courant,
‹&WG› référence l'angle de rotation d'un objet,
‹&LG› référence la longeur d'une ligne ou le rayon d'un cercle.

Par exemple, si on veut créer une ligne et si on choisit la fonction "POINT AT A DISTANCE" pour déterminer le second point, PC-DRAFT va demander la distance en question. Si on lui répond "&LG" il demande alors de désigner un élément. Il ne reste plus qu'à désigner la ligne dont nous voulons que la nôtre ait la longueur.

Mais il y a encore plus intéressant : PC-DRAFT possède son propre langage de description

d'objets.

Chaque groupe, à tous les niveaux de la hiérarchie, est décrit par la liste des objets qui le composent, et chacun de ces composants est décrit par une liste de caractéristiques. Par exemple, si ce composant est un groupe, il sera spécifié par son échelle, les coordonnées de son point de référence et un angle de rotation. S'il s'agit d'un élément de base, une ligne par exemple, il sera spécifié par les coordonnées de ses extrémités, et le type de la ligne (couleur, épaisseur,etc.).

Chaque création ou manipulation sur un dessin génère automatiquement une nouvelle définition ou une modification dans le fichier descriptif associé à ce dessin. Non seulement ce fichier est accessible à l'utilisateur, mais celui-ci dispose d'un éditeur qui lui permet de modifier ou de rajouter lui-même une définition d'objet qui sera immédiatement répercutée dans le dessin.

Mieux encore : PC-DRAFT possède un langage de programmation d'objet. La syntaxe de ce langage se compose de toutes les commandes de conception et d'instructions de contrôle (boucle, alternative).

Ces petits programmes constituent des macro-fonctions que l'on active comme n'importe quelle autre fonction. S'ils contiennent des pseudo-variables, ils deviennent purement et simplement

interactifs.

Le programme dessinant une ellipse serait, à titre exemplatif :

```
LINE ABSOLUTE 0,0 RELATIVE 0,5
SCALE 100,60
REFERENCE POINT RELATIVE -20,0
REPEAT (360/ATN(5/20))-1 COPY+ROTATE ATN 5/20
                         SCALE 100,60
```

Sans entrer dans le détail, rappelons que changer l'échelle (ici seulement dans la direction des Y) modifie l'aspect mais non la dimension réelle des objets.

En résumé, PC-DRAFT permet de créer des objets avec une grande précision grâce aux fonctions d'introduction de points et de trois façons: soit en utilisant les fonctions du menu, soit en écrivant la description complète de l'objet, soit en écrivant un programme qui le réalise.

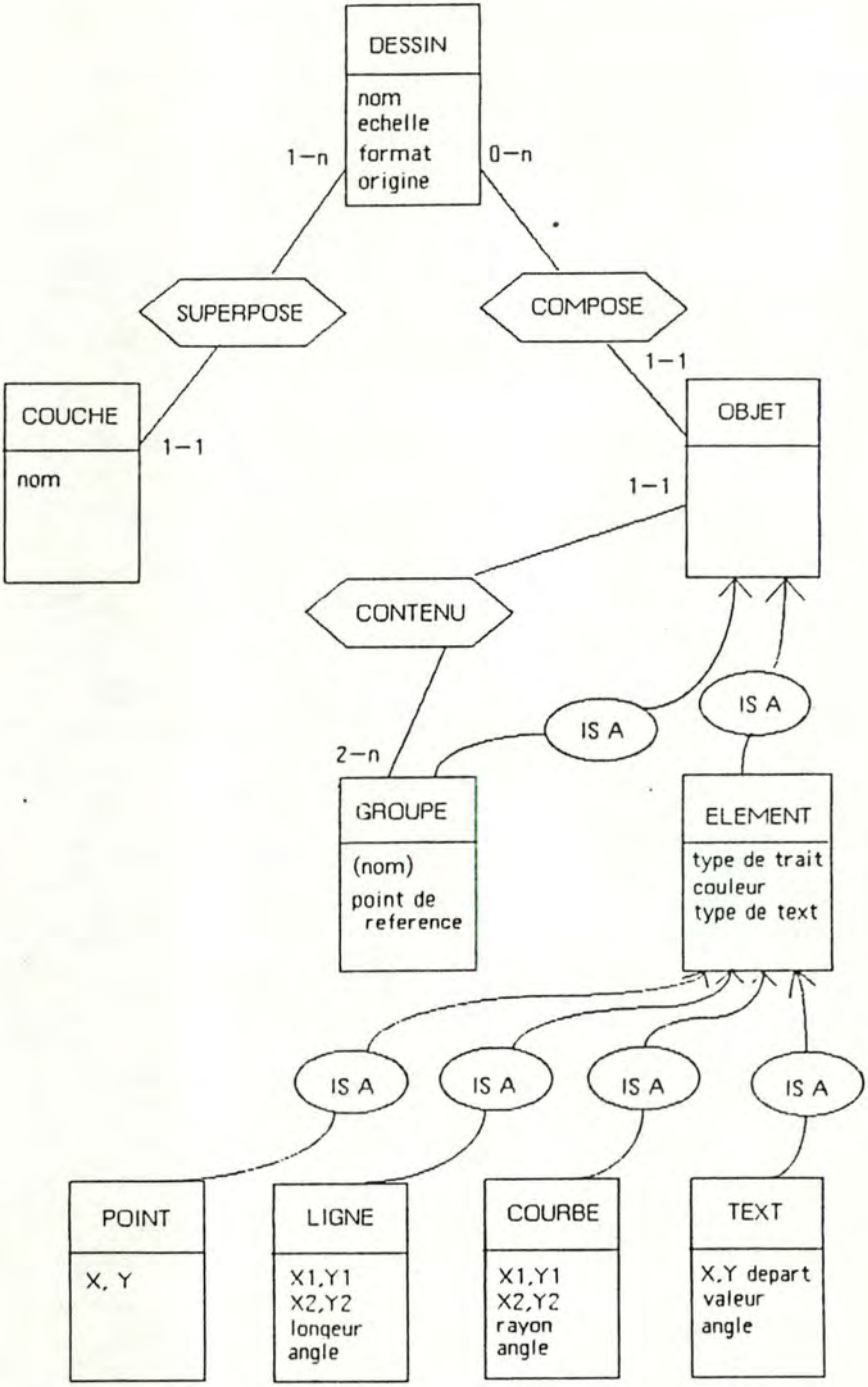


Schéma Entité—Association des informations traitées dans PC—DRAFT

2.4.3 STAR

STAR est un gros logiciel de conception assistée par ordinateur destiné essentiellement aux architectes et aux ingénieurs. Il permet la digitalisation, c'est-à-dire l'introduction de plans existants, la conception en plan, en élévation et en coupe des projets architecturaux, l'habillage des plans (cotations, équipements, titres, rendus,...) et la création de détails techniques 2 dimensions. Il produit automatiquement les vues en plan, en coupe et en élévation, des perspectives avec élimination des arêtes cachées, habillage des faces et tracé des ombres portées. Enfin, il génère automatiquement les mètrés descriptifs, qualitatifs et estimatifs liés au projet.

2.4.3.1 SUPPORT

STAR propose un espace de travail sans limite théorique dont on voit la projection en plan à l'écran. L'utilisateur choisit la précision de cet espace, c'est-à-dire la plus petite distance entre deux points voisins distincts de cet espace. Par exemple, le point de l'espace dont la coordonnée est (100,0) se trouve, pour le système, à un mètre de l'origine si l'on a déclaré une précision de 1cm, à 10m si la précision est de 10cm, etc. A chaque projet est associée une précision.

2.4.3.2 ELEMENT DE BASE

Toute la démarche de STAR tourne autour d'un concept unique: le parallélépipède. Elle est basée sur le fait que toute construction architecturale peut se décomposer en un ensemble structuré d'éléments de base ayant la même caractéristique. Le principe est que, dans notre système de repère spatial à trois axes orthogonaux, le parallélépipède est l'unité de volume. Il est également pour STAR l'unité de conception. Il représente une certaine quantité de volume plein. On conçoit un projet STAR en rajoutant, modifiant ou supprimant des éléments parallélépipédiques.

Ce parallélépipède va s'allonger pour devenir un mur, s'applatir pour devenir une dalle de plancher ou un pan de toiture, ou s'inverser pour devenir une baie de fenêtre ou de porte.

2.4.3.3 CREATION

La conception se fait en trois dimensions, la création se fait en deux dimensions. On a le choix entre travailler en plan, en coupe ou en élévation.

Par exemple, lorsque l'on travaille en plan, l'écran se confond avec un plan horizontal du

projet, plan dont on a préalablement spécifié le niveau. Et lorsque sur ce plan on dessine un mur allant d'un point à un autre, on a dû préalablement en spécifier la largeur, la distance qui sépare son sommet de ce plan, et celle qui sépare son pied du plan. En plan, on peut créer, supprimer et manipuler des murs, des dalles et des baies. En élévation, on peut manipuler les murs et les baies existants.

A tout moment de cette création en deux dimensions, il est possible d'obtenir instantanément une vue en perspective de l'objet, ce qui permet d'exercer un certain contrôle sur la construction. Mais si l'on peut "voir", on ne peut pas "toucher"!

Un mur est caractérisé par
- une épaisseur,
- le niveau relatif de son pied,
- le niveau relatif de son sommet,
- un point de départ et
- un point d'arrivée.

Une baie est caractérisée par
- le niveau relatif de son seuil,
- le niveau relatif de son linteau,
- un point de départ et
- un point d'arrivée.

Son épaisseur est forcément la même que celle du mur dans lequelle elle est pratiquée.

Une dalle est caractérisée par
- le niveau relatif de son pied,
- le niveau relatif de son sommet,
- un point de départ (supérieur gauche par exemple) et
- un point d'arrivée (inférieur droit dans l'exemple)

Ces points sont introduits à l'aide de la souris avec un contrôle visuel à l'écran ou en coordonnées au clavier.

Il est inutile de s'efforcer d'être précis car STAR prend en charge, si on le désire, le raccord entre les murs. L'autre manière de créer et de retrouver par la suite un point avec précision est de déclarer et d'utiliser une trame.

Enfin, il est possible de modifier la position des extrémités des murs, c'est-à-dire en fait de les allonger ou de les raccourcir.

2.4.3.4 MANIPULATIONS

Outre les manipulations classiques qui permettent de déplacer, répéter, faire pivoter ou supprimer un élément, il est possible de transformer cet élément, de le déformer, afin d'obtenir une variété plus grande d'objets. En effet, comment créer un mur pignon pointu? Comment obtenir un pan de toiture incliné, une arcade, etc.?

Trois transformations sont prévues.

Remonter ou descendre une arête horizontale de la face supérieure ou inférieure d'un élément.

Prolonger le bas ou le haut d'un mur jusqu'à un plan donné dont la pente est parallèle à l'axe de ce mur.

Tronquer le bas ou le haut d'un mur par un plan dont la pente est parallèle à l'axe de ce mur.

Un rectangle délimité par deux points définit une zone en plan dont tous les éléments peuvent être manipulés simultanément.

Enfin, il est possible de créer un mur courbe. Ce mur est considéré comme une suite de petits murs parfaitement raccordés et est caractérisé par son rayon de courbure et le nombre de "brisures".

2.4.3.5 STRUCTURE DES OBJETS

Jusqu'à présent, nous avons vu comment il était possible de créer des éléments en les situant par rapport à ce plan.

En fait, le projet est constitué d'un ensemble de plans superposés appelés "vues" et symbolisant chacune une tranche horizontale du bâtiment. On aura par exemple les vues: SEMELLE DE FONDATION, MUR DE FONDATION, DALLE DE SOL, MUR DE REZ et

TOIT. Lorsque l'on travaille dans l'une de ces vues, il est évidemment possible de lui superposer une ou plusieurs autres vues.

Chaque vue possède ses éléments propres. Les vues se situent les unes par rapport aux autres grâce à une hauteur relative à un niveau zéro commun, et nous avons vu comment les éléments se situent par rapport à leur vue. Le tout forme donc un ensemble cohérent.

Enfin, on peut créer ou supprimer une vue et même la répéter plusieurs fois dans le cas par exemple, d'un immeuble à appartements.

2.4.3.6 ASSOCIATION DE COMPOSANTS

L'idée la plus riche de STAR, et qui n'aurait pu s'appliquer sans éléments de base volumiques, est d'attribuer aux éléments que l'on a créés une nature. On dira par exemple que tel mur est une maçonnerie de briques, ou que telle baie est un chassis basculant ouvrant. Cette nature se définit par des caractéristiques techniques et une présentation graphique.

Pour réaliser cela, STAR met à la disposition de l'utilisateur un éditeur de texte et un dessin assisté dans deux dimensions.

Ensuite, il lui permet de gérer des tables de détails et de symbolisme dans lesquelles sont

stockés ses propres plans de détails et les symboles de base utilisés pour représenter les différentes matières, et des tables techniques qui contiennent pour chaque matériau ou article, le descriptif, une gamme de prix, le nom du fournisseur, etc.

Enfin, il permet la gestion d'un catalogue graphique et d'un catalogue technique de composants. Le catalogue graphique contient pour chaque composant créé, sa vue de face et ses coupes horizontale et verticale. Ces vues utilisent les différents symbolismes des tables. Dans le catalogue technique, on stipulera pour chaque composant les informations nécessaires pour le calcul du métré.

Par exemple, le composant MACONNERIE DE BRIQUE DE PAREMENT se représente de face en dessinant les différents lits de briques décalés, et en coupe par des lignes hachurées. Il utilise au mètre carré une quantité x de briques de parement qui est décrite dans les tables techniques.

En résumé, après avoir introduit les caractéristiques et la symbolique des matériaux, et après avoir rempli ses catalogues de composants, l'utilisateur va associer un ou plusieurs composants aux éléments de base qui constituent son projet.

A l'issue de cela, STAR sera à même de fournir les mètrés descriptifs, estimatifs et quantitatifs détaillés ou récapitulatifs du projet.

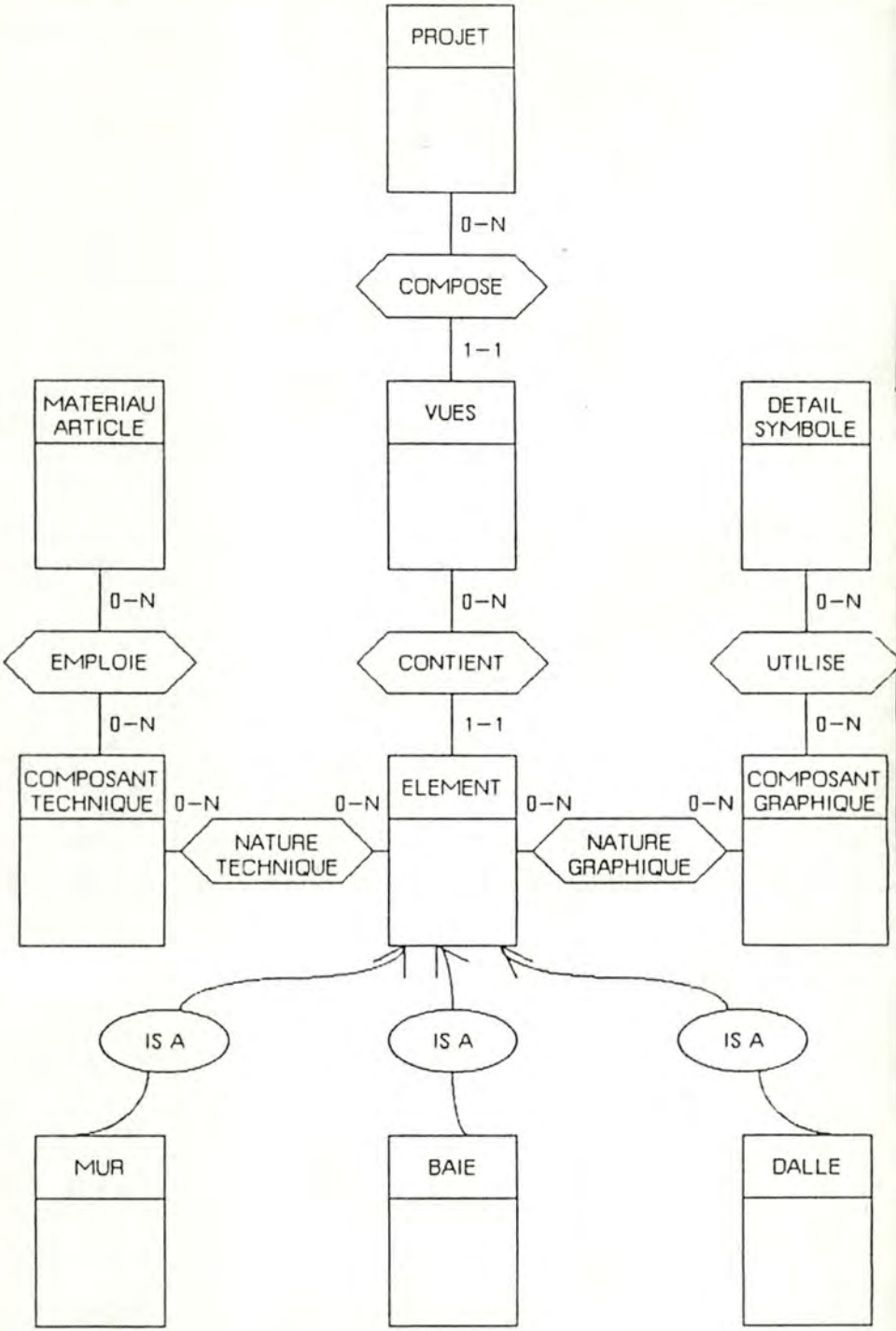


Schéma Entité—Association des informations traitées dans STAR

2.4.4 ARC+

ARC+ est la version sur PC d'un logiciel appelé ARCAID commercialisé depuis 1981 et fonctionnant sur ordinateur "mainframe" de la série 43. Il se dit introduire dans l'univers des micro-ordinateurs les performances 3D jusqu'alors seulement disponibles sur les gros systèmes de CAO.

Il fournit outre les classiques outils 2D propres au bâtiment, les fonctions suivantes : "solid modeling", visualisation 3D, obtention automatique de coupes quelconques et ombrages. Il génère automatiquement des murs et des dalles avec des percements 3D et effectue la fusion murs-dalles-poutres-poteaux avec continuité de matière. Il réalise des coupes avec rendu automatique du plan de coupe et élimination des lignes cachées et des perspectives à un, deux ou trois points de fuite, avec remplissage des surfaces avec ombres propres.

En ce qui concerne les documents écrits, il permet la production de métrés quantitatifs et de nomenclatures par extraction de données du modèle 3D, l'affectation à tout élément d'attributs non graphiques et l'affectation de fonctions aux surfaces avec calcul automatique des surfaces affectées.

2.4.4.1 SUPPORT

ARC+ propose un espace sans limite théorique dont la précision est fixe, le centimètre. Les coordonnées des points dans cet espace s'expriment en mètres et en centimètres. Par exemple: (0.93, 10.20, 0.00) On travaille donc en vraie grandeur. La notion d'échelle représente pour Arc+ le facteur qui détermine la portion d'espace vue à l'écran: plus l'échelle est grande, plus on voit d'espace à l'écran.

2.4.4.2 ELEMENTS DE BASE

L'élément de base de ARC+ s'appelle une ENTITE. Il y a cinq entités différentes: la ligne, le polygone ou "face", le corps solide, le mur et la ligne de construction.

Une ligne est soit une arête droite entre deux points déterminés de l'espace, soit un arc de cercle dont on donne trois points ou le centre, le point de départ et l'angle.

Un polygone est un ensemble d'arêtes situées dans un plan et dessinant une figure fermée.

Un corps solide est soit un prisme formé à partir d'un polygone et d'une hauteur, soit la portion d'espace définie par deux polygones reliés entre eux par des arêtes.

Un mur est un corps solide particulier, dont le polygone de base est un rectangle généré automatiquement par le système à partir de deux points donnés déterminant l'axe de ce mur. Il faut spécifier la largeur et la hauteur d'un mur.

Une ligne de conception est une droite infinie générée par le système. Elle apparait en pointillé. Par exemple, l'axe d'un mur est une ligne de construction.

Chacune de ces entités peut être référencée par l'utilisateur en plaçant le curseur à un endroit déterminé. Par exemple: un solide est référencé en pointant une des arêtes qui relient les deux polygones et un mur en pointant l'axe.

2.4.4.3 CREATION

2.4.4.3.1 PERCEPTION DE L'ESPACE

ARC+ propose trois façons de travailler: en plan, en élévation et en axonométrie (à soixante degrés par défaut). Il permet de passer instantanément d'une vue à une autre par une simple fonction. Les fonctions ZOOM -qui modifie l'échelle- et SHIFT permettent de déplacer la fenêtre-écran sur ces trois images planes.

Comme Star, il permet d'obtenir une vue en

perspective de l'objet, mais pas de modifier l'objet via cette vue. Pour changer l'angle de vue, il permet de changer la position de l'oeil et celle du point visé, soit en formant la nouvelle position, soit en introduisant le déplacement relatif par rapport à sa position actuelle, déplacement qui peut se faire dans une des six directions orthogonales.

2.4.4.3.2 CURSEUR

Le curseur se promène dans l'espace. Ses coordonnées spatiales sont constamment affichées à l'écran. Dans la vue en plan, seules ses coordonnées X et Y sont "actives". La coordonnée Z est fixe et affichée en rouge au lieu de blanc. Dans les vues en élévation, seules ses coordonnées X et Z peuvent varier, et, en axonométrie, les trois sont actives et il se déplace parallèlement à un des trois axes.

Ce curseur se dirige au moyen de quatre touches directionnelles dans la vue en plan ou en élévation et par six touches dans la vue en axonométrie.

Il peut également être déplacé au moyen de la souris qui le fait voyager dans le plan actif, c'est a'-dire le plan horizontal passant par la coordonnée Z du curseur dans la vue en plan, et dans la vue en élévation, dans le plan debout passant par la coordonnée Y du curseur. Une

commande permet de modifier la coordonnée inactive sans sortir de la vue.

Le manuel d'utilisation reste muet quant à l'effet de la souris dans la vue en axonométrie. En fait, cette action est spectaculaire et surprenante. Le curseur (une petite croix) se déplace sur l'écran en suivant exactement les mouvements de la souris, mais il représente en fait la projection d'un point appartenant au plan XY actif. Par exemple, déplaçons la souris de gauche à droite et le curseur se déplace de gauche à droite à l'écran. Ses coordonnées, qui sont affichées en permanence, annoncent un Z constant, un X allant en décroissant alors que Y croit. Déplaçons la souris de bas en haut et les coordonnées X et Y décroissent.

2.4.4.3.3 BOITE

Afin d'éviter qu'un trop grand nombre d'éléments n'encombrent, il est possible de définir par 2 points une "boite" qui détermine la portion de l'espace qui est visible. Tous les éléments situés hors de cette boite sont donc invisibles.

2.4.4.3.4 DEFINIR UNE POSITION

Lorsque le système demande d'identifier un point de l'espace, il y a trois façons de procéder:

- soit en introduire les coordonnées au clavier,
- soit positionner le curseur à l'endroit désiré,
- soit laisser au système le soin de sélectionner le point existant le plus proche de la position actuelle du curseur en activant la fonction "SNAP".

CE point existant le plus proche peut être soit un point appartenant à une entité, soit l'intersection entre deux lignes de construction.

Cette intersection peut s'obtenir de deux façons: elle correspond au point de percée d'une ligne de construction existante
- dans un plan quelconque déterminé par trois points,
- dans un plan debout, frontal ou horizontal coupant le troisième axe à une coordonnée donnée (CUT3P et HCUT).

ARC+ la matérialise en dessinant deux nouvelles lignes de construction appartenant à ce plan et concourant en ce point.

Enfin, on peut créer une ligne de construction de différentes manières: ce peut être une ligne
- passant par deux points donnés,
- passant par un point et formant deux angles avec le repère,
- passant par un point et formant un angle donné avec une autre ligne donnée (le manuel annonce cela froidement sans préciser que cette opération qui génère une infinité de

droites ne peut s'effectuer que dans un plan donné),

- parallèle à une droite donnée et passant par un point donné,
- perpendiculaire à une droite définie par deux points donnés et à une distance donnée du premier de ces deux points,
- parallèle à l'axe des X et passant par un point donné,
- parallèle à l'axe des Y et passant par un point donné,
- parallèle à l'axe des Z et passant par un point donné,
- étant enfin l'intersection de deux plans définis par trois points et deux angles donnés. Explication: Imaginons deux murs formant un coin de maison. De chacun d'eux part un pan de toiture incliné à un certain angle. Ce sont les deux angles donnés. Les trois points donnés sont l'intersection des deux corniches et un point de chacune d'elles. La ligne résultant est l'intersection entre les deux pans de toiture (l'arêtier).

Afin de remédier à l'imprécision de la méthode pour définir une position qui consiste à y placer le curseur, il y a deux façons: soit définir et utiliser la traditionnelle trame, soit donner à l'incrément du curseur une certaine valeur. Si celle-ci est de 10cm, les coordonnées du curseur varient 10cm par 10cm quel que soit son déplacement à l'écran.

Remarquons que si la trame est active sur ces trois vues, elle n'est visible qu'en plan et en élévation, pas en axonométrie.

2.4.4.4 MANIPULATIONS

Les opérations classiques de répétition, déplacement, rotation spatiale et suppression sont applicables à toutes les entités de base ainsi qu'à un groupe sélectionné d'entités de base. Ce groupe est formé en référençant les différentes entités qui le composent ou en délimitant une zone sur le plan par deux points. Ce groupe peut à tout moment être agrandi ou, au contraire, vidé (SENT).

L'entité "MUR" jouit d'un traitement spécial. On peut:

- y pratiquer des ouvertures que l'on peut modifier, déplacer, répéter ou supprimer,
- l'aligner automatiquement horizontalement et verticalement,
- arrondir un angle de mur,
- modifier la position des extrémités d'un mur,
- couper un pan en deux (en faire deux entités),
- en modifier hauteur et largeur.
- créer deux murs alignés et perpendiculaires entre eux en une seule opération.

2.4.4.5 STRUCTURE DES OBJETS

Le résultat d'une création s'appelle un objet. Un objet est donc un ensemble d'entités qui porte un nom et est conservé dans un fichier. On peut créer un nouvel objet, rentrer dans un objet existant (OPEN), le sauver ou le supprimer.

Un objet peut également contenir d'autres objets existants. Tout objet est donc une hiérarchie d'objets et d'entités.

Les seules opérations possibles sur les objets "placés" sont le déplacement et la rotation. Il est donc impossible de manipuler les entités propres à un de ces objets placés. Pour ce faire, il faut sortir de l'objet actuel et rentrer dans l'objet correspondant à l'objet placé. Toute modification d'entités appartenant à un objet qui est placé dans un autre se répercute à tous les niveaux supérieurs de la hiérarchie, et cette répercussion sera visible dès que l'on rentrera dans un des objets supérieurs le contenant.

Un objet peut être placé dans toutes les positions et orientations possibles et à n'importe quelle échelle. Ensuite, il peut être manipulé comme une simple entité à condition d'avoir été introduit dans le groupe sélectionné.

Une des facilités de ARC+ est la reconnaissance et le stockage dans des librairies d'objets répétitifs comme des portes ou des colonnes. De

ces objets, on n'enregistre qu'un modèle et on identifie les paramètres -le manuel ne précise pas comment- qui permettront de les transformer chaque fois qu'on les placera dans un autre objet. Ces librairies contiennent par défaut des dessins en plan de mobilier standard, mais il est possible de créer en trois dimensions son propre mobilier.

Les baies (fenêtres, portes) sont des objets placés particuliers en ce sens qu'ils s'adaptent automatiquement en largeur, en longueur et en hauteur au mur et à l'endroit de ce mur où ils sont placés.

Les fonctions concernant les objets placés sont les suivantes:

- OPLAC qui permet d'insérer dans un mur l'ouverture dont on donne le nom, à une échelle et à un endroit donné, parallèlement à une ligne donnée.
- PL1P qui permet d'insérer l'objet dont on donne le nom à une échelle et un endroit donnés, dans le même sens que celui où il a été créé.
- PL3P qui permet en plus d'orienter l'objet dans un plan horizontal.

2.5 CONCLUSION

Au terme de cette première partie consacrée à la description de la situation actuelle, nous pouvons dire que l'ordinateur apporte des outils d'aide à la conception riches et variés.

Ils sont variés car ils s'adressent à des utilisateurs différents ou, plus précisément, car ils sont orientés vers des types d'utilisation différents.

Ils sont riches car ils se sont spécialisés de plus en plus pour répondre à des besoins précis liés à leur type d'utilisation.

MAC-DRAW permet au mémorant de dessiner des schémas Entité-Association avec facilité, PC-DRAFT permet aux professionnels de réaliser des dessins techniques avec précision, STAR permet aux bureaux d'étude et d'architecture de concevoir en quatre dimensions, et ARC+ permet aux architectes de créer dans l'espace.

D'un point de vue général, en faisant abstraction de ses spécifications propres, il est intéressant de dégager chez chacun de ces logiciels ce qu'il apporte de particulier en tant qu'outil de conception graphique pure.

- Le premier permet de créer et de manipuler les objets avec une aisance diabolique.

- Le second permet de les créer et de les manipuler avec une précision millimétrique.

- Le troisième permet de les considérer non seulement comme des formes mais encore comme des entités solides possédant un poids, une matière et même un prix.

- Le dernier permet de les manipuler en trois dimensions et d'en obtenir une image très réaliste grâce au coloriage et à l'ombrage des faces.

Dans la partie qui va suivre, nous allons réagir vis-à-vis de ces constatations et définir les objectifs de notre étude.

3 DEUXIEME PARTIE

3.1 INTRODUCTION

Après avoir fait un rapide tour d'horizon de la situation et après s'être attardé sur quelques logiciels représentatifs, nous allons maintenant définir les objectifs précis de CIOS.

Ces objectifs découlent de l'étude que nous avons faite des trois premiers logiciels présentés dans la première partie, c'est-à-dire, MAC-DRAW, PC-DRAFT et STAR.

En découvrant le logiciel STAR, nous avons été impressionnés par sa puissance. Le contrôle qu'il permet d'exercer sur la conception au moyen de la perspective a confirmé notre idée qu'il pouvait être intéressant de créer directement dans la vue en perspective.

En utilisant MAC-DRAW et PC-DRAFT, nous avons été sensibilisés à l'utilité de la structure hiérarchique des éléments qui découle d'une démarche constructive naturelle. D'autre part, STAR et PC-DRAFT ont attiré notre attention sur le fait qu'ils incorporent des copies d'objets prédéfinis stockés dans des librairies spéciales. Nous avons alors pensé que ces objets prédéfinis pouvaient être des objets comme les autres dont on n'incorpore pas une copie mais une "image". De

telle sorte que, si l'objet est modifié par la suite, toutes ses "images" le sont aussi. Cela a donné naissance à une approche constructive par décomposition en sous-objets autonomes.

Nous avons été séduits par la rigueur tout allemande que PC-DRAFT met dans la recherche de points caractéristiques dans le plan. Cela assure que le dessin est "juste" quel que soit le zoom avec lequel on le regarde. Nous nous sommes rappelés des problèmes que posent, par exemple, la détermination du point de percée du faite d'une toiture à deux pans dans une autre toiture, plus grande. Nous nous sommes alors demandés pourquoi ne pas inventer des outils de dessin dans l'espace. Une règle, des équerres, un compas, un rapporteur ou un translateur qui travaillent dans trois dimensions...

3.1.1 OBJECTIFS

- Le premier objectif est de rendre la perception et la manipulation de l'espace la plus agréable et naturelle possible.

- Le deuxième objectif est d'expérimenter une approche constructive complémentaire par l'incorporation de sous-objets indépendants.

- Le troisième objectif est de permettre de déterminer avec précision n'importe quel point caractéristique de l'espace.

Ces trois objectifs étaient fixés, l'analyse était réalisée et le programme tournait avant que l'existence d'ARC+ ne nous soit connue. Nous n'avons donc pu en tenir compte lors de la définition des objectifs.

Dans un premier chapitre, nous allons expliquer pourquoi l'élément de base est l'arête.

Dans les trois chapitres suivants, nous allons développer chacun des trois objectifs de CIOS.

Dans le dernier chapitre, nous définirons l'ensemble des fonctions nécessaires en raison des objectifs visés.

3.2 ELEMENT DE BASE

L'élément de base de la création sera l'arête.

Comme nous l'avons vu dans l'introduction, un des soucis de ce mémoire est d'effectuer une approche générale de la conception tridimensionnelle.

L'idée que l'élément de base de l'espace est le parallélépipède rectangle est séduisante, mais nous parait trop orientée. Même dans le domaine de l'architecture, elle limite la création.

On pourrait imaginer que cet élément de base, un cube par exemple, soit maléable et que l'on puisse en déplacer "manuellement" une face, une arête ou un sommet pour le déformer. En rejoignant deux arêtes parallèles, on obtiendrait un prisme à base triangulaire qui conviendrait parfaitement comme toiture à une maison.

On pourrait également rajouter des sommets sur les arêtes, qui, une fois déplacés, modifieraient automatiquement les arêtes supplémentaires partant d'un sommet rajouté à ceux non adjaçants des deux faces dont l'arête commune a reçu ce sommet.

On obtiendrait ainsi une sorte de jeu de construction dont on pourrait dans une certaine mesure modeler les pièces.

De telles applications existent sur les gros systèmes et permettent entre autres de créer de très jolis temples grecs.

Un certain Rudofsky disait à propos du jeu de cubes de l'enfant :

"Ils servent à le divertir, en même temps qu'à imprimer dans son esprit la principale icône de notre credo architectural. Il existe certainement un rapport entre la prolifération de ce genre d'architecture que nous qualifions de "moderne" faute d'épithète plus précis, et le jeu de construction de l'enfance."

S'il est vrai que le parallélépipède est l'unité de volume, il n'est pas forcément l'unité de création dans l'espace. On peut avoir envie de créer seulement un point dans l'espace, une ligne, ou une surface.

Si l'on voulait modéliser les éléments spaciaux d'une façon générale, on dirait qu'il y a le point, l'arête, la face et le volume.

Le point est le plus petit élément de l'espace, l'arête est une portion de droite délimitée par deux points ; la face est une portion de plan délimitée par plus de deux droites et le volume est une portion d'espace délimitée par au moins quatre plans.

Remarquons que tous ces éléments sont définis à partir de trois notions : le point, la droite et

le plan. De plus, une arête définit une droite et une face définit un plan. On peut donc, à partir de points et d'arêtes, définir des faces et des volumes.

Nous nous limiterons à la création d'arêtes et de points en raison de l'objectif général qui est de faire tourner un nouvel outil. De plus, l'arête suffit amplement pour réaliser les objectifs fixés qui sont de tester la perception dans l'espace et d'expérimenter un outil de précision et une approche par décomposition de sous-objets.

3.3 TROIS DIMENSIONS

3.3.1 INTRODUCTION

Tous les objets qui nous entourent sont dans l'espace. Ils occupent un certain volume. Nous-même, nous nous déplaçons, nous vivons, nous pensons dans l'espace.

Comment se fait-il que, depuis toujours, lorsque nous imaginons des objets, nous les exprimons en deux dimensions?

Nous avons vu que la réalisation de maquettes était exceptionnelle, à l'instar de l'utilisation de plans, en raison de son faible coût et de sa simplicité.

Pourtant, l'homme a trouvé le moyen d'exprimer, dans le tracé en perspective, des objets en deux dimensions d'une manière presque aussi riche qu'en trois dimensions. Il lui a fallu des siècles avant d'inventer la perspective. Il a d'abord dû comprendre et modéliser l'espace. Il y est si bien parvenu qu'actuellement, on ne voit plus de différences entre une image générée par ordinateur et une photographie...

Nous pensons que si l'homme exprime en plan les objets qu'il imagine dans l'espace, c'est par tradition et par manque de moyens.

Or, avec l'ordinateur, il dispose aujourd'hui d'un outil capable de gérer des données en trois dimensions et de les exprimer en perspective. Il a donc le moyen de concevoir et d'exprimer directement en trois dimensions les objets qu'il imagine dans l'espace.

Bien plus, cette "maquette virtuelle" pourrait suffire à l'entrepreneur ou au fabriquant puisqu'il pourrait à loisir couper dedans pour la voir sous toutes ses coutures...

Quant au client ou à l'acheteur, il ne devrait plus reconstruire mentalement l'objet qui lui est présenté en plan et en coupe, avec tous les risques de perte d'informations que cela comprend.

Actuellement, les logiciels permettent de concevoir un objet en plan et en élévation. Puis, de le voir en perspective. La création se fait en deux temps: le concepteur crée l'objet sans vraiment le voir, puis, le voit sans pouvoir le manipuler.

Or, toute création est un processus cyclique dans lequel créativité et auto-critique, imagination et choix sont intimement liés.

3.3.2 AXONOMETRIE OU PERSPECTIVE?

L'idée est de réunifier le processus de création, de procurer un outil qui n'aura peut-être pas la richesse d'expression que possède une maquette d'étude, mais qui donnera au créateur d'espace l'illusion de travailler la matière brute en lui permettant de modifier directement l'image en perspective de cet espace.

Cette idée n'est pas neuve. Cependant, il n'existe à l'heure actuelle, aucun outil de ce genre qui soit accessible au concepteur d'espace particulier: architecte, décorateur, architecte d'intérieur,... Pourtant, le besoin existe bel et bien.

Le logiciel ARC+ a fait le premier pas et est toujours le seul, à notre connaissance, à proposer un outil tournant sur PC et permettant de travailler directement sur une image tridimensionnelle des objets.

Or, nous l'avons vu, ARC+ permet de travailler simultanément en plan, en élévation et en axonométrie, mais pas en perspective...

Alors que la perspective -l'ensemble de toutes les vues possibles d'un objet- fournit une image du réel qui correspond quasiment avec la perception que l'on a de ce réel, les plans, l'élévation et l'axonométrie ne donnent que trois images

symboliques différentes de cette même et unique réalité.

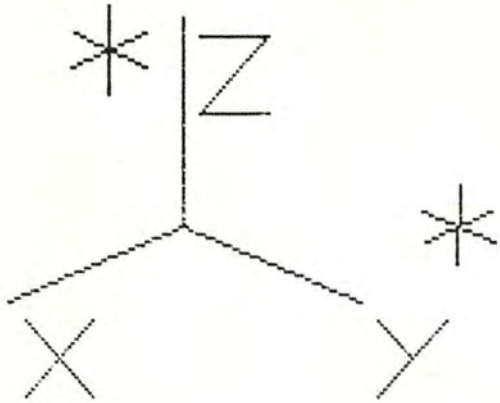
S'il est vrai que l'axonométrie donne une certaine impression d'espace, il reste une différence importante entre l'axonométrie et la perspective.

L'aspect statique de cette différence est le fait que les images de lignes parallèles en perspective convergent en un point appelé point de fuite, alors qu'elles restent parallèles en axonométrie. Mais cette différence a aussi un aspect dynamique.

Lorsqu'un point se déplace dans l'espace à une vitesse constante, l'observateur le verra bouger d'autant plus lentement qu'il sera éloigné. La perspective reproduit fidèlement ce phénomène. En axonométrie, quelle que soit la position de ce point dans l'espace, son image se déplace toujours à la même vitesse.

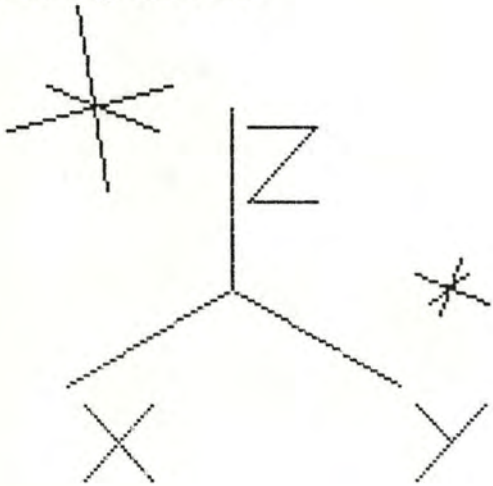
Le phénomène est si surprenant que la première fois que nous avons utilisé le logiciel ARC+ et que nous avons promené le curseur en axonométrie, nous nous sommes tout simplement perdus! Nous étions incapables, à travers l'image qu'on en avait, de dire où se trouvait exactement le curseur dans l'espace, et donc, de trouver la direction du chemin de retour...

:800 y:800 z:600 pas:100



maximum 2 chiffres)
ouverture 10:

:80 y:80 z:90 pas:10



(page 60) L'axonométrie ne rend pas la profondeur...

(maximum 2 chiffres)
ouverture 89:

3.3.3 PROBLEMES DE LA PERSPECTIVE

Le premier objectif de ce mémoire est de se demander pourquoi aucun logiciel existant n'a franchi le cap qui consiste à travailler directement et uniquement sur l'image en perspective des objets.

Il est vrai que cela ne va pas sans problèmes.

3.3.3.1 REPERE

Le premier problème est lié à la perception statique de l'espace. Nous sommes habitués à repérer l'espace au moyen de trois axes qui le graduent dans trois directions principales. Le fait de les voir en projection s'est avéré pratique mais pas indispensable. Car, dès que l'on a créé un objet, il donne lui-même, grâce à ses arêtes propres l'indication des trois directions principales. Voir le repère devient utile lorsque l'on veut modifier l'angle de vue de l'espace.

Pour ces raisons, ARC+ affiche dans le coin supérieur droit de la vue en axonométrie la direction de chacun des trois axes du repère.

CIOS permet, si on le désire, de voir les axes en vraie grandeur à l'endroit où ils se trouvent en réalité dans l'espace, c'est-à-dire, à l'origine du repère. Ce qui a l'avantage de bien montrer où

3.3.3.3 TRAME

Le troisième problème concerne la trame dont nous avons déjà vu l'utilité et la nécessité. Si dans une représentation en deux dimensions, il n'y a pas d'équivoque possible quant à la position du curseur par rapport aux points de la trame, il n'en va pas de même dans une représentation à trois dimensions. En effet, si le curseur ne se trouve plus dans le même plan que la trame, doit-il garder une "trace" dans le plan de celle-ci, ou au contraire, la trame doit-elle automatiquement "suivre" le curseur dans le plan défini par sa position, au risque d'embrouiller le dessin?

D'autre part, la trame doit-elle se trouver en projection dans un plan frontal, horizontal, vertical ou dans chacun d'eux simultanément ou doit-elle trouver trace dans tout l'espace ?

Pour simplifier le problème, Arc+ ne permet pas de voir la trame en axonométrie.

CIOS proposait une, deux, ou les trois trames, au choix, à un endroit au choix, à condition d'avoir limité l'espace couvert par cette trame. En effet, si, dans le plan, l'image de trame est naturellement limitée par les côtés de l'écran, dans une projection en perspective de l'espace, l'infini se trouve bel et bien sur l'écran et une trame infinie brouillerait la vision.

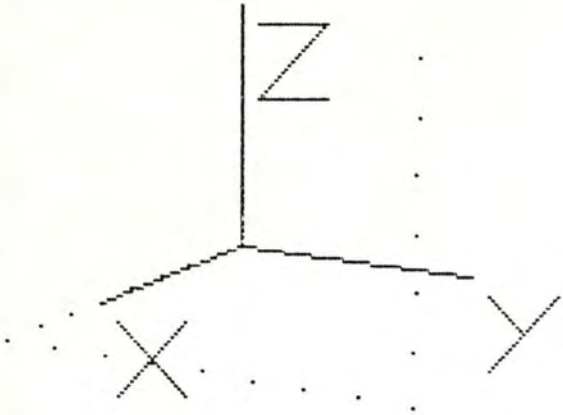
se trouve le point de coordonnée (0,0,0), ainsi que l'endroit où se situe l'observateur: plus il est éloigné de l'origine, plus le repère apparait petit.

3.3.3.2 CURSEUR

Le problème suivant est lié à la perception dynamique de l'espace. Nous avons déjà soulevé le problème lié à la perception de l'endroit où se trouve le curseur dans l'espace. Dans la vue en axonométrie de Arc+, la position du curseur se matérialise par une petite croix à l'écran. CIOS propose une petite étoile à six branches vue en perspective qui laisse une trace sur chacun des trois axes principaux. Comme dans Arc+, sa position dans l'espace est constamment affichée, et elle se déplace soit à un pas donné au moyen de six touches dans six directions (les trois principales et leurs directions opposées), soit au moyen de la souris dans le plan actif correspondant à la coordonnée Z et d'une fonction pour faire varier cette coordonnée Z ,c'est-à-dire, la faire monter ou descendre.

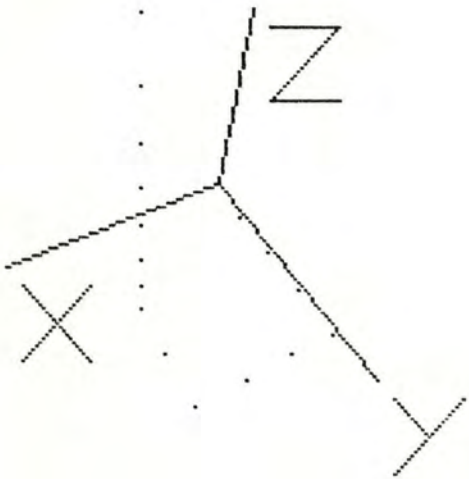
Il y a cependant une différence importante entre Arc+ et CIOS. Le déplacement de la souris dans Arc+ correspond au déplacement de l'image du curseur à l'écran, alors que le déplacement de la souris de CIOS correspond au déplacement du curseur lui même dans un plan horizontal de l'espace virtuel.

:80 y:80 z:60 pas:10



x:30 y:20 z:80 pas:10

:



amera x50,y60,z100 P:

(page 63) La "trace" du curseur suggère sa position.

Le hasard nous fit découvrir une façon bien pratique de nous passer de la trame lorsque l'on déplace le curseur au moyen des touches. En effet, dans le module du déplacement de ce curseur, nous avions oublié de commander l'effacement de sa position précédente et nous nous sommes rendu compte qu'il laissait derrière lui une trace de son passage qui suggérait sans équivoque sa position actuelle dans l'espace.

3.3.3.4 CAMERA

Le quatrième problème lié au fait de n'avoir qu'une (infinité de) vue(s) en perspective des objets créés, concerne précisément le changement de l'angle de vue qui détermine la perspective.

Ici, nous ne comparerons plus CIOS avec l'application d'axonométrie d'ARC+, qui ne permet que de modifier l'angle que forme l'axe des X avec la direction verticale de l'écran (l'axe des Z reste toujours fixe et l'axe des Y suit le mouvement en fonction d'un algorithme obscur), mais avec son application de perspective qui, nous le rappelons, ne permet aucune manipulation de l'objet.

L'idée de déplacer le point visé et l'oeil de l'observateur de la même façon que si l'on déplaçait une caméra dans l'espace, semble être la bonne puisque c'est celle adoptée par ARC+ et par STAR. Le principe est de déterminer la position de

ces deux points soit en en donnant les nouvelles coordonnées, soit en en donnant le déplacement relatif dans une des six directions.

Il y a toutefois une différence entre ARC+ et CIOS au point de vue de l'utilisation de ces fonctions.

Il est important que l'utilisateur puisse déplacer et contrôler facilement la position de la caméra et du point visé. Cet aspect est d'autant plus important dans CIOS que la perspective est la seule image que l'on a de l'objet.

Lorsque l'on veut changer l'angle de vue, CIOS met à notre disposition pour définir ces deux points un outil appellé POINT. Cet outil, dont le but est de déterminer les coordonnées d'un point particulier dans l'espace, sera développé dans le chapitre PRECISION, mais nous en retiendrons dès maintenant deux utilisations. On peut déterminer une position soit en en introduisant les trois coordonnées, soit en y plaçant le curseur.

Il en résulte trois façons de déterminer la position de la caméra et du point visé:

- introduire leurs coordonnées au clavier,
- en s'aidant de la vue actuelle, aller placer le curseur à l'endroit d'où l'on voudrait voir et à l'endroit que l'on voudrait viser, ce qui permet un très bon contrôle "visuel" du déplacement de la caméra,
- en s'aidant d'une fonction qui place automatiquement le curseur à l'endroit de la caméra actuelle.

A ce moment-là, le curseur est bien entendu invisible, mais tout déplacement de ce curseur correspond en fait à déplacer la future position de la caméra. Cela équivaut à donner, au moyen des touches commandant le curseur, le déplacement relatif de la caméra, mais, dans les trois directions d'un coup et d'une manière très "sentie". Cela permet, lorsqu'on l'a compris, un très bon contrôle "tactile" de ce déplacement. De même, une fonction permet de placer automatiquement le curseur à l'endroit du point visé.

3.3.3.5 ELAGUER

Le dernier problème concerne la clarté et la lisibilité de l'image. Lorsque l'objet créé devient complexe, l'image se sature d'autant plus vite en lignes qu'il s'agit d'une image en perspective plutôt qu'en axonométrie, en plan ou en élévation. Ce problème, qui touche tous les logiciels, devient donc crucial pour CIOS.

PC-DRAFT l'a résolu en permettant de transférer les objets d'une couche à l'autre et de rendre visible ou invisible l'une ou l'autre "couche" du dessin.

STAR le résout en permettant de superposer plus ou moins de "vues" à la fois.

ARC+ permet de définir cette fameuse "boite" à

l'extérieur de laquelle les objets et entités ne sont pas visibles...

CIOS fait intervenir ici la notion de "sous-objet" qui sera développée dans le chapitre STRUCTURE. Un objet de CIOS est composé d'éléments propres et d'autres objets appelés sous-objets. Les sous-objets, comme c'est le cas dans ARC+, ne peuvent être modifiés en aucun cas. La solution du problème consiste à pouvoir rendre visible ou invisible tel ou tel sous-objet de l'objet sur lequel on travaille.

Une dernière remarque concerne la clarté et, plus généralement, toute la perception de l'espace. Il est possible d'avoir l'illusion d'une vue en plan ou en élévation, en plaçant la caméra très loin ou très haut, et en visant l'objet avec un angle d'ouverture très petit. Bien que cela aille à l'encontre de l'objectif poursuivi, une fonction permet d'amener le curseur automatiquement très loin sur un des trois axes.

3.3.4 CONCLUSION

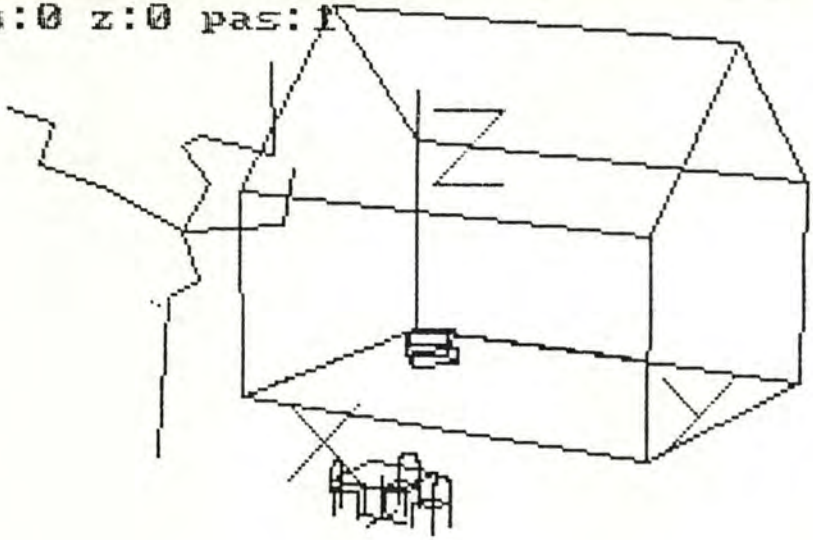
Nous pensons qu'il est possible et même agréable pour un utilisateur de concevoir des objets en les voyant en perspective.

Il ne s'agit pas de remettre en question la conception en plan et en élévation : elle sera

toujours la principale. Le but est de fournir un outil supplémentaire intermédiaire entre le plan et la maquette.

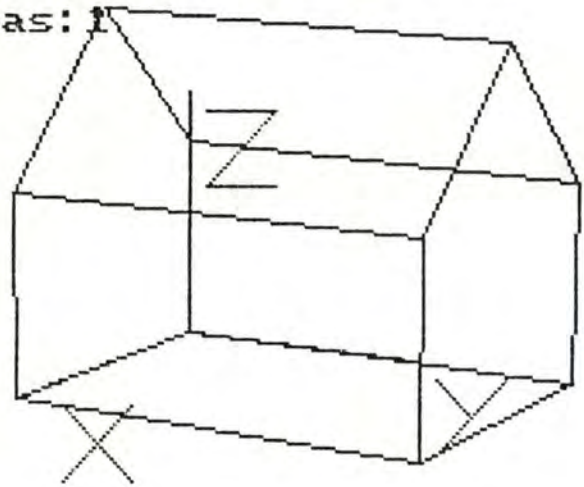
Les logiciels actuels lui permettent de réaliser ce rêve. Mais pourquoi ne pas aller un tout petit peu plus loin et lui permettre également de "toucher" et de transformer les objets qu'il "voit" ?

x:0 y:0 z:0 pas:1



arbre1 :fauteil :table1 ::
liste (O)bjets ou (S)objets ?:

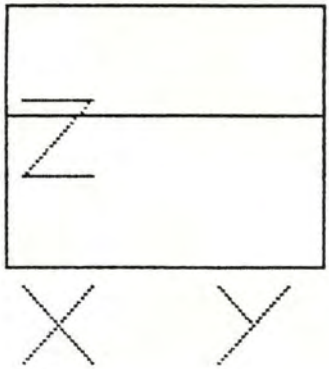
x:0 y:0 z:0 pas:1



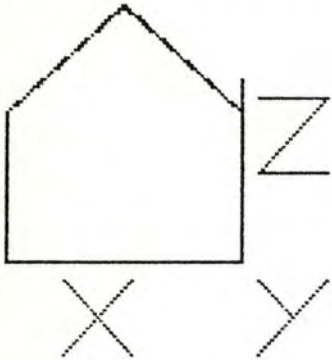
(page 68) Les sous-objets visibles apparaissent en sombre dans la liste.

arbre1 :fauteil :table1 ::
liste (O)bjets ou (S)objets ?:

x:16000 y:0 z:0 pas:1000

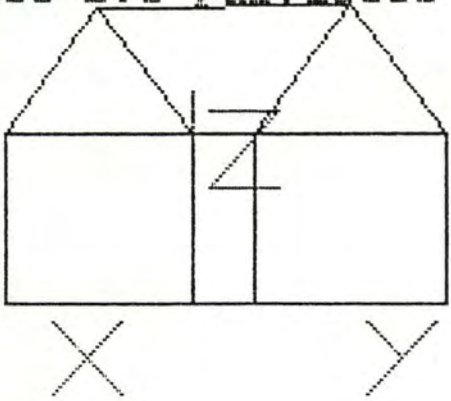


x:0 y:16000 z:0 pas:1000



(page 68) "Illusions" de plan et d'élévations.

x:10000 y:10000 z:0 pas:10000



3.4 DEFINITION DE LA STRUCTURE

3.4.1 INTRODUCTION

Considérons qu'un objet est un ensemble d'arêtes dans l'espace.

La conception d'un objet met en évidence l'existence de relations au sein de ces arêtes. Elles ne sont pas situées au hasard l'une par rapport à l'autre.

Une de ces relations possible fait référence à la notion de structure. Nous envisagerons deux structures possibles, qui, au delà de leur similitude apparente, sont fondamentalement différentes.

Illustrons par un exemple de la vie courante la notion de structure d'éléments.

Lorsque l'on parle d'une chambre-à-coucher, on dit qu'elle contient un lit, une armoire et un lavabo. On dit d'autre part que le lit contient un sommier, un matelas, etc. On dit aussi qu'elle est composée d'un lit, d'une armoire et d'un lavabo, et que le lit est composé d'un sommier et d'un matelas.

Nous avons structuré les éléments de la chambre en

une structure hiérarchique du type "contient" ou "est composé de". Cette structure décrit la façon dont les éléments s'associent pour former la chambre. Cette hiérarchie découle d'une démarche constructive.

Lorsque, dans le cadre d'un outil de CAO, on construit un objet à partir d'un ensemble d'arêtes, on crée également une structure hiérarchique parmi ses éléments.

Or, d'un point de vue purement conceptuel, il y a deux façons très différentes de considérer cette notion de hiérarchie. Il en résulte deux structures différentes correspondant à des approches créatives différentes de l'objet.

Nous allons étudier successivement ces deux approches. Par pure convention, décidons d'apppeler la première structure : "hiérarchie EST COMPOSE DE", et la deuxième : "hiérarchie CONTIENT".

3.4.2 HIERARCHIE EST COMPOSE DE

Tout objet que l'on crée peut se décomposer en sous-objets qui le peuvent également, et ainsi de suite jusqu'à ce que tous les sous-objets soient des objets élémentaires, en l'occurence, des arêtes.

Cette hiérarchie découle d'une démarche constructive "bottom up" : un rectangle est composé de quatre arêtes, le dessin d'une fenêtre est composé de cinq rectangles, le dessin d'une façade est composé de dix fenêtres et d'un rectangle, etc.

Entendons-nous bien : nous avons dessiné quatre arêtes en forme de rectangle, nous les avons regroupées pour en faire un sous-objet qu'on a répété quatre fois dans un autre rectangle que l'on a créé; après avoir groupé le tout, nous l'avons répété dix fois et entouré d'un grand rectangle avant de tout grouper à nouveau, etc.

Chacun de ces sous-objets intermédiaires est appelé GROUPE. Un groupe a une raison d'exister éphémère: celle d'être répété, déplacé, agrandi ou supprimé, avant d'être incorporé dans un autre groupe pour composer un niveau supérieur de la hiérarchie.

La hiérarchie elle-même n'a servi qu'à l'élaboration du dessin, et pourrait très bien être oubliée, si on ne voulait pas à un moment donné modifier ce dessin.

Elle peut même se révéler contraignante, car pour modifier une des arêtes d'un rectangle d'une fenêtre d'une façade d'un bloc d'un ensemble d'immeubles, il faut reparcourir toute la hiérarchie du sommet jusqu'à cette arête...

Ce qui suit constitue une des différences fondamentales avec la deuxième hiérarchie : toutes les arêtes composant l'objet se situent dans un système de coordonnées unique. Une arête ou un groupe d'arêtes déplacées ou allongées changent donc de coordonnées.

Cette structure de groupes existe dans tous les logiciels graphiques, car elle fournit un outil de base pour la conception. Pour certains de ces logiciels, il ne s'agit que d'une pseudo-hiérarchie car pour accéder à un groupe de niveau inférieur, il faut démanteler la hiérarchie niveau par niveau jusqu'à isoler le sous-groupe en question.

Dans MAC-DRAW, il s'agit précisément des fonctions "group object" et "ungroup object" qui permettent de gérer cette hiérarchie.

PC-DRAFT propose tout un "group definition set of functions" qui permet de:

- chaîner des objets individuellement,
- insérer un objet dans un groupe,
- détacher un objet d'un groupe,
- chaîner tous les objets se trouvant dans une zone,
- démanteler le sommet d'une hiérarchie,
- démanteler les groupes du premier niveau d'une hiérarchie,
- démanteler toute une hiérarchie, sauf son sommet.

ARC+ ne propose pas de hiérarchie, mais simplement un "selection group" auquel on peut rajouter ou enlever des entités et des objets placés. C'est seulement sur ce "groupe courant" que peuvent s'appliquer les manipulations.

STAR permet juste de grouper sans distinction les éléments d'une zone délimitée par un rectangle afin de les manipuler simultanément.

Comme nous venons de le voir, cet outil a déjà été largement exploité par certains logiciels. Malgré le fait qu'il soit d'une utilité capitale, nous ne l'implémenterons pas dans CIOS.

3.4.3 INCORPORATION D'OBJETS

Nous venons de voir qu'un objet est un ensemble structuré d'arêtes dans un espace de référence. Chaque objet porte un nom, et il y a une bibliothèque qui contient tous les objets déjà créés.

La gestion de cette bibliothèque va servir pour la conception des objets. Rappelons-nous de la démarche constructive de la chambre-à-coucher. La chambre contient un lit. Si ce lit est un objet qui a déjà été créé et qui existe dans la bibliothèque, il est inutile de le recréer une

nouvelle fois dans la chambre. On va aller chercher le lit dans la bibliothèque et l'apporter dans la chambre...

Il est possible, à tout moment de la conception d'un objet, d'y incorporer la COPIE d'un autre objet existant, c'est-à-dire d'en RECOPIER TOUTES LES ARETES dans l'espace de référence du premier objet. Lors de cette incorporation, les arêtes conservent leur éventuelle structure de groupe.

A la suite d'une telle opération, la copie de l'objet incorporé est devenue un groupe comme les autres de l'objet hôte, qui peut être manipulé et transformé normalement.

Cette application s'avère très pratique dans le cas de finition de plans par exemple : on dispose dans la bibliothèque de catalogues d'objets représentant du mobilier ou des symboles électriques, et on les recopie à volonté dans ses plans.

3.4.4 HIERARCHIE CONTIENT

3.4.4.1 INTRODUCTION

Nous abordons ici l'autre aspect de la notion de hiérarchie. Il s'agit d'une démarche différente

que nous allons essayer de mettre en évidence.

Lorsque l'objet hôte reçoit une copie d'un autre objet, il NE CONTIENT PAS vraiment celui-ci, mais est simplement COMPOSE de ses arêtes qui ont été recopiées dans son propre espace de référence. L'objet recopié n'est plus un objet, mais un groupe d'arêtes composant et appartenant à l'objet hôte. Il fait maintenant partie de la hiérarchie "est composé de" de l'objet hôte.

Pour pouvoir dire qu'un objet en CONTIENT un autre, il faut que ce deuxième objet reste un objet à part entière, c'est-à-dire un ensemble d'arêtes propres dans un espace de référence propre.

3.4.4.2 EXEMPLE DE LA CHAMBRE

Supposons que nous soyons en train de concevoir l'objet "chambre" dans lequel nous avons incorporé l'objet "lit" précédemment créé.

Dans le cas de la hiérarchie "est composé de", nous n'avons aucun mal à aller modifier ce lit puisque toutes ses arêtes font partie de l'espace de la chambre. Il suffit de dégrouper les arêtes le composant jusqu'à accéder à l'arête que l'on veut modifier.

Alors que dans le cas de la hiérarchie "contient", il est impossible de retirer par exemple un pied

du lit sans avoir préalablement quitté l'espace de la chambre, et pénétré celui du lit.

Ceci parce que lorsque nous avons incorporé le lit dans la chambre, nous avons en fait disposé à un endroit particulier de l'espace de la chambre, l'espace du lit lui-même, avec toutes les arêtes qu'il contenait.

Pour modifier ce lit dans la chambre, il n'y a pas d'autre solution que de modifier l'OBJET "lit" lui-même.

Il faut donc bien distinguer deux choses dans le cas de cette hiérarchie : la DEFINITION de l'objet "lit", et une OCCURRENCE de ce "lit". La définition d'un objet est son espace de référence et les arêtes qu'il contient. Une occurrence d'un objet est le fait que sa définition soit incorporée à un endroit précis de celle d'un autre objet.

Remarquons que le fait d'incorporer la copie d'un objet dans un autre, n'est qu'un cas particulier de la structure "est composé de". On ne le fait que lorsque les arêtes de cet objet peuvent servir à la construction de l'objet hôte. Si par contre l'objet à incorporer est considéré comme un tout qu'on ne veut pas modifier, on utilisera la hiérarchie "contient". On dira alors que l'hôte "contient" un objet indépendant qui a été conçu de façon autonome, et qui sert à l'"aménagement" de l'objet hôte. Cette règle d'utilisation est

évidemment subjective. Mais si le choix de l'une ou l'autre structure paraît arbitraire, il faut savoir que leurs effets sont fondamentalement différents.

3.4.4.3 TYPES ET OCCURRENCE

Avant de continuer, convenons d'appeler la définition d'un objet un TYPE D'OBJET, et le fait que cet objet apparaisse dans la définition d'un autre objet, une OCCURRENCE de ce type d'objet.

Un TYPE D'OBJET est défini par un espace de référence, un ensemble d'arêtes structurées dont les coordonnées sont exprimées dans cet espace, et l'ensemble des occurrences de type d'objet qui rentrent dans la définition de ce type.

Une OCCURRENCE est donc un type d'objet, plus un facteur d'échelle, une orientation et une coordonnée dans le repère du type d'objet hôte.

Un type d'objet peut avoir plusieurs occurrences dans des types d'objet différents ou dans un même type.

Lorsque l'on conçoit un type d'objet, on peut agir soit sur ses arêtes propres, soit sur les paramètres des occurrences de type d'objet qu'il

contient. Ces paramètres sont l'échelle, l'orientation et la position.

La rigueur veut que l'on accède à un type d'objet uniquement en l'appelant directement. Or il pourrait arriver que l'on veuille concevoir un type d'objet dans un certain contexte, en fonction d'un autre type, ce qui n'empêche pas ce type d'être autonome et indépendant. Cela veut dire qu'à tout moment de la conception d'un type d'objet "A", on peut rentrer dans la définition d'un autre type d'objet "B" dont une occurrence se trouve dans ce type "A".

3.4.4.4 COMPARAISON

La différence fondamentale entre "contenir des occurrences" et "être composé de groupes", est que si l'on modifie un type d'objet, tous les types en contenant au moins une occurrence sont concernés.

Donc, si nous retirons un pied du lit qui est contenu dans la chambre, tous les objets contenant ce lit le voient instantanément amputé d'un pied. En effet, nous avons retiré un pied à la définition du type d'objet "lit" qui est contenue dans le type d'objet "chambre", et donc dans toutes les occurrences de ce lit.

Dans certains cas, cela peut être gênant. On a peut-être envie de ne modifier le lit que d'une seule chambre-à-coucher. Ce désagrément peut

facilement être évité en créant un nouvel objet du même type mais portant un autre nom, avant de le modifier.

Cela peut par contre représenter un grand avantage. Imaginons une façade contenant une centaine de fenêtres identiques. Supposons que l'on veuille modifier ces fenêtres. Si elles sont des groupes, il faudra les modifier une à une. Ou en supprimer nonante-neuf et les remplacer par des copies de la centième, après l'avoir modifiée. Si par contre la fenêtre est un type d'objet, une fois ce type modifié, toutes les occurrences seront mises à jour automatiquement.

Dans la première approche, la fenêtre a été considérée comme un groupe d'arêtes faisant chacune partie de celles de la façade. Dans la seconde, la fenêtre a été considérée comme un certain type de représentation de fenêtre, ce type n'étant pas forcément définitif. Les démarches sont totalement différentes. Les effets le sont aussi.

3.4.4.5 A APPROFONDIR...

La définition des types et des occurrences peut être exploitée davantage.

Lorsque l'on crée une occurrence, on doit actuellement spécifier l'échelle, la position et l'orientation. On pourrait également devoir

actualiser d'autres paramètres concernant des coordonnées, des longueurs ou des angles. Cela implique que lors de la définition du type de l'objet, on ait laissé ces valeurs en suspens et que lors de l'incorporation, le système s'en rende compte et interroge l'utilisateur.

C'est réalisable en remplaçant ces valeurs par des variables du type des pseudo-variables de PC-DRAFT. L'utilisation de ces variables interactives suppose que le type ne soit plus défini graphiquement mais sous forme de déclarations ou de programmes.

Par exemple, on pourrait créer le type d'objet "mur" ayant comme paramètre son épaisseur, sa hauteur et deux points de l'espace le délimitant en longueur. A chaque "appel" de "mur", c'est-à-dire chaque fois que l'on incorpore une occurrence de "mur" dans la définition d'un autre objet, le système demanderait la valeur de chacun des paramètres.

On pourrait même imaginer que la définition d'un type d'objet soit décrite par un ensemble de propriétés et de règles gérées par un système qui serait capable de construire l'objet en fonction d'un environnement particulier.

Cela voudrait dire que, dans l'exemple précédant, le "mur" serait capable de s'intégrer à un endroit particulier en résolvant les problèmes de raccords avec les autres murs, avec les dalles, les pans de toiture, etc.

3.4.5 CONCLUSION

Notre but n'est pas de confronter ces deux approches. Elles sont complémentaires et interviennent à des moments différents de la conception d'un objet. La première hiérarchie fait appel à des notions ensemblistes. La seconde fait appel aux notions de type et d'occurrence.

3.5 PRECISION

Le mot précision revêt deux concepts indépendants: on travaille avec une précision de autant, on travaille avec précision.

3.5.1 AVEC UNE PRECISION DE

Cette précision spécifie la plus petite distance qui sépare deux points distincts de l'espace. Si elle est infiniment petite, on travaille dans un espace continu et les coordonnées s'expriment au moyen de nombres réels. Si elle n'est pas infiniment petite, on travaille dans un espace discret et les coordonnées s'expriment au moyen de nombres entiers.

Ce raisonnement est théorique. En pratique, les réels ont également une précision limitée par les caractéristiques de la machine.

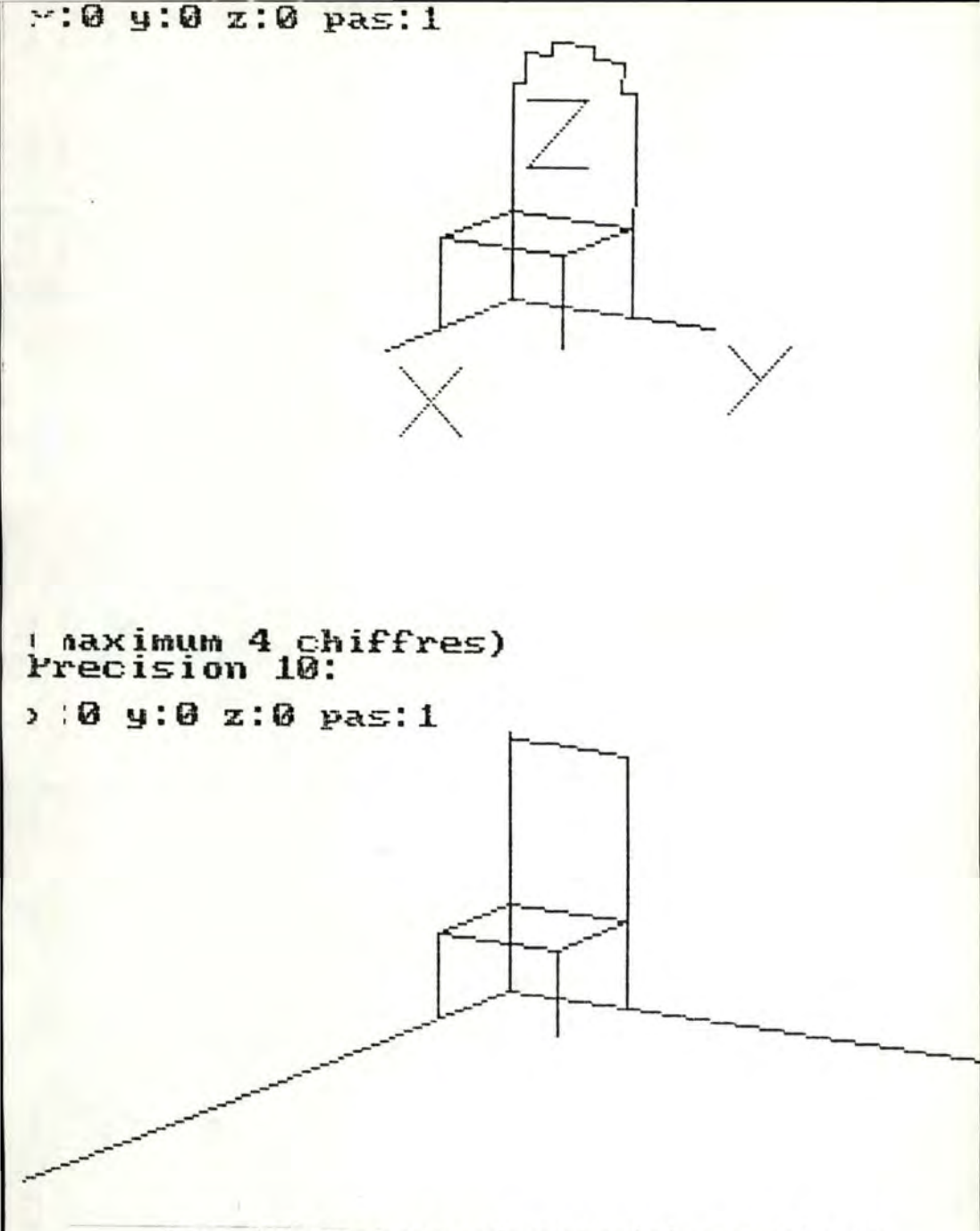
Nous ignorons comment les autres logiciels représentent leurs coordonnées. Nous avons choisi de représenter celles de CIOS par des entiers pour une raison de place en mémoire centrale. Une arête est représentée par deux points et un point par trois nombres, ce qui limite au départ le nombre d'arêtes mémorisables. Or, un réel occupe trois

fois plus de place en mémoire qu'un entier...

L'espace mis à la disposition de l'utilisateur est donc un espace discret dont la précision est limitée. L'utilisateur doit spécifier la précision de chaque objet, en fonction du niveau de détail auquel il veut descendre lors de la conception. Pour une maison, on travaille habituellement avec une précision d'un centimètre, mais pour un meuble par exemple, on peut descendre jusqu'au millimètre.

Etant donné que la valeur des entiers est limitée par la machine, généralement entre -32768 et 32767, cet espace est limité à 65636 points distincts dans chaque dimension. Cela signifie qu'avec une précision d'un centimètre, l'utilisateur pourra créer un objet mesurant jusqu'à 656 mètres, alors qu'avec une précision du dixième de millimètre, cet objet serait limité à quelques 6 métres 5o.

Il est important de spécifier la précision d'un objet car il en sera tenu compte lors de l'incoporation d'un objet dans un autre. Si une maison de dix mètres de côté est conçue avec une précision d'un centimètre, et un fauteuil d'un mètre avec une précision d'un millimètre, pris séparément, ces deux objets ont la même taille: mille unités d'espace chacun. Mais si l'on incorpore ce fauteuil dans la maison, celui-ci apparaitra dix fois plus petit. L'objet fauteuil est toujours défini avec une précision d'un millimètre, mais son occurrence dans l'objet



(page 83) L'occurrence de cette chaise a perdu un niveau de détail par rapport à son type.

(maximum 4 chiffres)
Precision 100:

maison est représentée avec la même précision que celle de la maison, c'est-à-dire le centimètre, et les détails du fauteuil de moins d'un centimètre ont disparu !

3.5.2 AVEC PRECISION

3.5.2.1 INTRODUCTION

Travailler avec précision signifie exprimer le fait qu'un objet, une ligne par exemple, va d'un endroit bien spécifique à tel autre.

Il y a deux sortes d'endroits spécifiques:
- ceux dont on connait la position "effective", c'est-à-dire les coordonnées ou la distance par rapport à un point connu dans une certaine direction,
- ceux dont on connait la position "théorique", c'est-à-dire l'intersection entre deux éléments, la projection d'un point connu sur un élément, le milieu d'un élément, etc, ou tout simplement un point connu dont on a oublié la position effective.

Lorsqu'un architecte trace, à l'aide d'une latte graduée, deux traits au crayon dessinant un angle droit, il ne faut pas une loupe pour s'apercevoir de l'imprécision du raccord entre les deux traits. Mais cela n'a aucune importance: l'architecte qui

dessine ainsi que celui qui lira le plan SAVENT très bien tous les deux qu'il s'agit d'un angle droit situé à un endroit précis. Et cela, parce que ce dessin n'est que le SYMBOLE de quelque chose de précis.

L'ordinateur, lui, est infiniment plus précis. Et comme il ne comprend pas le sens de nos symboles, il faut lui communiquer ce que l'on veut qu'il dessine avec la plus grande précision possible.

C'est précisément ce qui peut être reproché aux logiciels du type de MAC-DRAW. Les points dont on ne connaît que la position théorique sont introduits avec une précision purement visuelle, c'est-à-dire la même que celle avec laquelle on travaille sur une feuille de papier. On se sert donc de la machine comme on se sert d'une feuille de papier, ce qui est dommage car la machine est capable de beaucoup plus. Indépendamment de cet avis, c'est justement l'objectif du MAC de permettre d'effectuer les gestes de tous les jours, et ceux qui affirment que le MAC peut aider un technicien du dessin se trompent.

C'est ce qu'ont compris les concepteurs de PC-DRAFT qui l'ont muni d'un outil très performant pour déterminer les points tangents, les intersections, les milieux, les projections, etc...

STAR et ARC+ de leur côté proposent des outils plus spécifiques aux problèmes architecturaux : prolonger un mur jusqu'à un autre, résoudre le

raccord entre deux murs, tronquer un mur par un plan, etc.

CIOS voudrait traiter ce problème d'une façon beaucoup plus générale, afin de recouvrir le plus de cas particuliers possible.

Nous avons vu quatre classes d'éléments de base de l'espace : les points, les arêtes, les surfaces et les volumes. Nous avons dit qu'ils étaient définis au moyen de trois notions de base : le point, la droite et le plan. Le problème se réduit à pouvoir déterminer théoriquement n'importe quel point, n'importe quelle droite et n'importe quel plan de l'espace.

3.5.2.2 MECANISME DE L'OUTIL

D'une façon générale, un point se définit par l'intersection d'une droite et d'un plan, une droite se définit par l'intersection de deux plans et un plan, par un point et une droite qui lui est perpendiculaire. Ceci donne les trois règles suivantes :

P = INTER1(Pl,D)
D = INTER2(Pl,Pl)
Pl= PERP1(D,P)

Le système n'a pas de solution car chaque règle fait intervenir les deux autres. Pour le résoudre, il faut admettre le fait qu'à un moment donné on

connaisse la position effective d'un point, c'est-à-dire par exemple ses coordonnées.

```
P = COORD
P = INTER1(P1,D)
D = INTER2(P1,P1)
P1= PERP1(D,P)
```

Le système bouclera encore tant que nous n'aurons pas dit qu'une droite est également définie par deux points.

```
(a) P = COORD
(b) P = INTER1(P1,D)
(c) D = PASSE1(P,P)
(d) D = INTER2(P1,P1)
(e) P1= PERP1(D,P)
```

Imaginons un outil capable de calculer les fonctions COORD, INTER1, INTER2, PASSE1 et PERP1, et capable d'interroger l'utilisateur afin de déterminer les paramètres inconnus jusqu'à ce qu'ils soient tous connus. Un paramètre est connu lorsqu'il est défini par une fonction sans paramètre. Lorsqu'un paramètre peut être résolu en appliquant plusieurs règles, c'est l'utilisateur qui choisit la quelle appliquer. La fonction COORD correspond à un traitement interactif qui saisit trois valeurs de coordonnées. Un utilisateur possédant un tel outil est capable de résoudre tous les cas de figures ne faisant pas intervenir de valeurs d'angles et de distances.

Supposons par exemple que l'utilisateur veuille

déterminer un plan dont il connait trois points

1)..... il choisit la règle (e)
2)..... l'outil lui demande une droite
3)..... il choisit la règle (d)
4)..... l'outil lui demande le premier plan de la droite 2)
5)..... il choisit la règle (e)
6)..... l'outil lui demande une droite
7)..... il choisit la règle (c)
8)..... l'outil lui demande le premier point de la droite 6)
9)..... il choisit la règle (a) et donne les coordonnées de son 1er point,
1o)..... l'outil lui demande un 2ème point de la droite 6)
11)..... il choisit la règle (a) et donne les coordonnées de son 2ème point,
12)..... l'outil lui demande alors le point par lequel passe le plan 4)
13)..... il choisit la règle (a) et donne les coordonnées de son 1er point,
14)..... l'outil demande alors le 2ème plan de la droite 2)
15) à 23) même scénario que de 5) à 13), sauf qu'il donne les coordonnées du 3ème point à la place de celles du 2ème point,
24)..... l'outil lui demande enfin le point par lequel passe le plan 1)
25)..... il choisit la règle a) et donne les coordonnées du 1er point.

A ce moment-là, l'outil est capable de calculer le

plan passant par les trois points.

Cet exemple a été choisi délibérément simple afin de montrer la lourdeur de cet outil. Nous allons l'enrichir en y ajoutant des règles et des faits. Nous allons utiliser de nouvelles fonctions interactives, DIST et ANGLE, qui saisissent chacune une valeur. Toutes les règles suivantes qui n'utilisent pas ces deux fonctions sont dérivables à partir du système actuel des cinq règles et ne sont introduites que pour alléger l'utilisation de l'outil.

3.5.2.3 DEFINITION DE L'OUTIL

Il est possible d'imaginer une infinité de règles. Nous avons choisi d'écarter d'une part celles qui sont manifestement redondantes, c'est-à-dire dérivables à partir seulement d'une ou de deux autres règles, d'autre part celles qui sont orientées vers une application spécifique. Une règle orientée serait par exemple " une droite est l'intersection d'un plan quelconque et d'un plan horizontal situé à une hauteur de 1 mètre 2o ".

Nous avons décidé d'ajouter les faits suivants afin de faciliter l'introduction des points effectivement connus.

Un point est défini par la position du curseur.

P = CURSEUR

Un point est défini par la position du point courant, c'est-à-dire le dernier point résolu.

P = LAST

Afin d'enrichir le calcul de points, droites et plans théoriquement connus, nous avons ajouté les règles suivantes :

Un point est défini par la position du point existant le plus proche d'un point donné

P = NEAREST(P)

Un point est défini par la position milieu entre deux points donnés

P = MILIEU(P,P)

Un point est défini par l'intersection de deux droites

P = INTER2(D,D)

Un point est défini par la projection orthogonale d'un point donné sur un plan donné

P = PROJ(Pl,P)

Un point est défini par l'intersection d'une droite donnée avec le plan perpendiculaire à cette droite passant par un point donné

P = INTER4(D,P)

Un point peut être à une distance donnée d'un point donné sur une droite donnée

P = A(DIST,D,P)

On peut désirer la droite passant par un point donné et perpendiculaire à un plan donné

D = PERP2(Pl,P)

On peut désirer la droite passant par un point donné et parallèle à une droite donnée

D = PARALL1(D,P)

On peut désirer la droite passant par un point donné, décrivant un angle donné par rapport au plan horizontal passant par l'origine et située dans un plan debout décrivant un angle donné avec le plan frontal passant par l'origine

D = FORME1(ANGLE,ANGLE,P)

On peut désirer la droite qui dans un plan donné passe par un point donné et décrit un angle donné avec une droite donnée

D = FORME2(ANGLE,D,P1,P)

On pourrait aussi obtenir directement la droite parallèle à l'axe des X, des Y ou des Z, et passant par un point donné.

D = PARALLX(P)
D = PARALLY(P)
D = PARALLZ(P)

Un plan est défini par trois points distincts

P1 = PASSE2(P,P,P)

Un plan est défini par deux droites distinctes et concourantes

P = PASSE3(D,D)

On peut déterminer le plan passant par un point donné et parallèle à un plan donné

P = PARALL2(P1,P)

Un plan est défini par une droite donnée par laquelle il passe et qui matérialise sa pente

```
Pl = PENTE(D)
```

Un plan est défini par deux angles et un point donné

```
Pl = FORME 3(ANGLE,ANGLE,P)
```

On peut déterminer le plan passant par une droite donnée appartenant à un plan donné avec lequel il forme un angle donné

```
P = FORME4(ANGLE,Pl,D)
```

On peut aussi vouloir obtenir directement le plan frontal, horizontal ou debout, passant par un point donné.

```
Pl = FRONTAL(P)
Pl = HORIZONTAL(P)
Pl = DEBOUT(P)
```

Etc...

REMARQUES:

Au lieu de dire "un point est défini par ...", il aurait été plus exact de dire "un point peut être défini par...". Il y a en effet des configurations

impossibles comme l'intersection de deux droites parallèles, etc. L'outil tient compte de cela et le cas échéant, avorte le calcul.

D'autres règles permettent de déduire deux solutions possibles. C'est un problème lié à l'orientation lorsque l'on utilise les angles et les distances. Par exemple, la règle P=A(DIST,D,P) définit deux points différents en fonction du sens de la droite. Etant donné qu'il est difficile de maitriser cette caractéristique, il est envisageable de rajouter un paramètre à ces fonctions.

3.5.2.4 RESUME

```
P = CURSEUR
P = COORD
P = LAST
P = NEAREST(P)
P = MILIEU(P,P)
P = INTER3(D,D)
P = INTER1(PL,P)
P = PROJ(P1,P)
P = INTER4(D,P)
P = A(DIST,D,P)

D = PASSE1(P,P)
D = INTER2(P1,P1)
D = PERP2(P1,P)
D = PARALL1(D,P)
D = PARALLX(P)
```

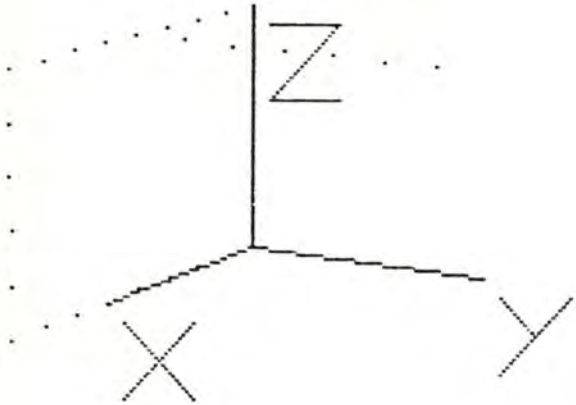
```
D = PARALLY(P)
D = PARALLZ(P)
D = FORME1(ANGLE,ANGLE,P)
D = FORME2(ANGLE,D,P1,P)

P1 = PASSE2(P,P,P)
P1 = PASSE3(D,D)
P1 = PARALL2(P1,P)
P1 = PERP1(D,P)
P1 = PENTE(D)
P1 = FORME3(ANGLE,ANGLE,P)
P1 = FORME4(ANGLE,P1,D)
P1 = FRONTAL(P)
P1 = DEBOUT(P)
PL = HORIZONTAL(P)
```

3.5.2.5 CONCLUSION

Cet outil permet théoriquement à l'utilisateur de connaître la position de n'importe quel point de l'espace, à condition qu'il soit capable de décrire le problème en termes de rapports entre points, droites et plans.

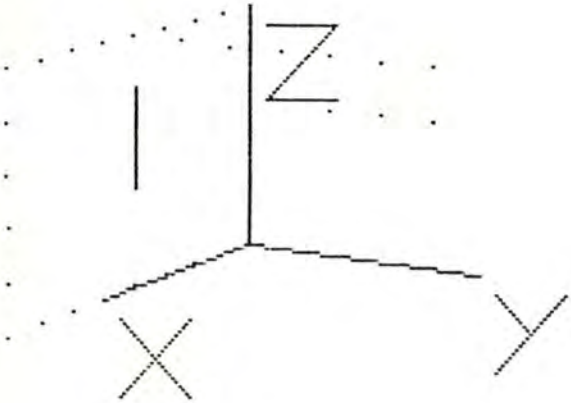
Il peut servir à déterminer les points caractéristiques des figures compliquées représentant, par exemple, des intersections de surfaces ou des pénétrations de volumes.

x:40 y:60 z:50 pas:10



Nous créons une arête dont le premier point est l'intersection entre un plan défini par trois points et une droite passant par deux points,

RRETE(P:I(Pl:(P:C,P:C,P:C),D:(P:C,P:
:40 y:40 z:40 pas:10



et le deuxième point est la projection orthogonale sur le plan courant d'un point donné.

),P:Pr(Pl:L,P:C)):

3.6 FONCTIONS

3.6.1 INTRODUCTION

Nous venons de voir qu'un type d'objet est défini par un ensemble structuré d'arêtes propres, et par un ensemble d'occurrences d'autres objets, une occurrence étant un type d'objet dont la définition apparait à un endroit précis de celle d'un autre.

Nous allons maintenant définir l'ensemble des fonctions qui nous ont semblé être indispensables pour que l'outil à réaliser soit opérationnel.

Cet ensemble contient des fonctions destinées à la conception des types d'objet et d'autres destinées à la gestion des types d'objet créés. Les fonctions de création regroupent la création d'éléments, de groupes d'éléments et d'occurrences de types d'objet, la manipulation des objets créés et le contrôle de l'espace.

Pour simplifier, convenons d'appeller un type d'objet un OBJET, et, puisqu'on ne peut se trouver que dans un seul objet à la fois, toutes les occurrences d'objets qu'il contient, des SOUS-OBJETS.

3.6.2 CONCEPTION DES OBJETS

3.6.2.1 CREATION D'ELEMENTS

3.6.2.1.1 POINT

L'élément le plus simple est le point. Il est en effet possible que l'on ait besoin de matérialiser la position d'un certain point pour des raisons de construction ou esthétiques. Pour créer un point, il suffit de donner sa position. Nous avons vu dans le chapitre PRECISION toutes les façons dont on voudrait pouvoir déterminer la position d'un point.

3.6.2.1.2 ARETE

Etant donné l'objectif choisi, tracer des arêtes dans l'espace va être la principale activité. Pour créer une arête, il suffit de donner deux points.

Pour la même raison, il n'y a actuellement pas de groupes prédéfinis (cercle, arc, rectangle, polygone, etc...)

3.6.2.1.3 TRIM, CUT

Afin d'éviter qu'une modification n'oblige chaque fois à exécuter une suppression suivie d'une création, les deux modifications les plus courantes font souvent l'objet de fonctions spécifiques appelées TRIM et CUT. TRIM permet d'allonger (ou de raccourcir) une arête donnée jusqu'à un point ou un plan donné. CUT crée à partir d'une arête donnée deux nouvelles arêtes en fonction d'un point de la première ou d'un plan donné.

Pour désigner une arête existante, il n'y a pas d'autre solution que d'aller la pointer avec le curseur. En réponse à quoi l'élément pointé clignotera ou changera d'intensité, afin que l'on puisse vérifier que l'on a bien pointé l'élément désiré.

Il y va de même avec un groupe d'éléments: étant donné que ces éléments sont indissociables, il suffit de pointer une seule arête de n'importe quel niveau pour désigner tout le groupe.

3.6.2.2 CREATION DE GROUPE

Pour grouper des éléments entre eux (arêtes ou groupes existants), il suffit de les désigner l'un après l'autre. Pour dégrouper un groupe existant, il suffit de le désigner, et tous les éléments

implique l'existance très fugitive de groupe, et pas mal de pas de clers lors de la manipulation. Vaudrait mieux avoir f° grouper et une autre dégrouper

composant le sommet de sa hiérarchie redeviendront autonomes.

3.6.2.3 CREATION DE SOUS-OBJETS

Au-delà de cette hiérarchie parmi les éléments, nous avons vu qu'un objet pouvait contenir d'autres objets. On peut donc à tout moment incorporer un sous-objet à l'objet dans lequel il se trouve.

Il suffit de signaler le nom de l'objet appelé à devenir sous-objet. Si ce nom n'existe pas, la fonction échouera, de même que s'il désigne l'objet hôte lui-même.

Il faut ensuite situer le point qui coïncidera avec l'origine de l'espace de référence de l'objet à incorporer. Il faut également introduire trois angles qui détermineront l'orientation du sous-objet par rapport aux axes de l'objet hôte, et un facteur d'échelle.

Pour désigner un sous-objet de l'objet courant, il suffit de le nommer. Il est en effet inutile d'aller pointer ses arêtes puisqu'elles sont inaccessibles depuis l'objet hôte. Pour la facilité, une zone de menus est réservée à la liste des sous-objets contenus dans l'objet courant.

mais ce serait facile pr désigner le sous objet.

3.6.2.4 MANIPULATIONS

Une série de manipulations sont possibles sur les éléments (arêtes, groupes) et les sous-objets de l'objet courant. Il suffit pour cela de les désigner comme nous l'avons vu précédemment.

Ces manipulations sont DEPLACER, REPETER et SUPPRIMER.

3.6.2.4.1 DEPLACER

Il y a trois façons de déplacer:
- TRANSLATER à un certain endroit,
- faire PIVOTER d'un certain angle autour d'un certain axe et par rapport à un certain point,
- CHANGER L'ECHELLE dans les trois directions et par rapport à un point fixe.

Ces trois façons sont combinables.

3.6.2.4.2 REPETER

Répéter un certain nombre de fois équivaut à déplacer sans supprimer, le déplacement se faisant chaque fois par rapport au résultat du déplacement précédent.

Une répétition particulière est le MIROITEMENT, c'est-à-dire une symétrie orthogonale par rapport à un certain plan.

3.6.2.4.3 DEPLACER UN SOUS-OBJET

Cela revient à aller modifier interactivement la valeur des paramètres spécifiés lors de son incorporation. Rappelons que le facteur d'échelle d'un sous-objet est global, ceci pour éviter une distorsion des distances dans les différentes directions, dans le cas où l'on rentre dans un sous-objet.

3.6.2.4.4 REPETER UN SOUS-OBJET

Ceci implique l'incorporation d'une nouvelle occurrence de l'objet correspondant. Lorsqu'un objet contient plusieurs occurrences d'un même objet, celles-ci sont numérotées d'après leur ordre d'incorporation et c'est le dernier numéro qui figure dans la liste des sous-objets.

3.6.2.4.5 SUPPRIMER

Supprimer un élément ou un sous-objet désigné ne requiert aucun paramètre supplémentaire.

3.6.2.4.6 UNDO

Enfin, toutes ces opérations étant assez abstraites à commander, il arrive fréquemment que le résultat obtenu ne corresponde pas à celui désiré. Pour éviter toute complication, il existe

couramment une commande qui annule les effets de la dernière manipulation effectuée.

3.6.2.5 CONTROLE

A chaque objet est associé un espace de référence qui est repéré par trois axes perpendiculaires par rapport auxquels sont exprimées les coordonnées des points appartenant à l'objet.

Cet espace est virtuel et l'utilisateur ne le "voit" que par l'intermédiaire de l'écran sur lequel il est projeté suivant une perspective vraie. Tout se passe exactement comme si une caméra imaginaire filmait l'objet dans l'espace virtuel et en renvoyait l'image sur l'écran de l'ordinateur.

3.6.2.5.1 TRAME

Nous avons déjà parlé de l'utilité de la trame. La trame est une zone de l'espace repérée par des points situés à espaces réguliers. On peut déterminer l'espacement entre deux points, délimiter dans l'espace cette zone parallélépipédique au moyen de deux points, la rendre visible ou invisible, l'activer ou la desactiver.

Rendre visibles tous les points de la trame serait malhabile. On va se contenter des points situés

dans un, deux ou trois plans particuliers: les plans debout, frontal et horizontal, situés à un certain endroit de la trame.

3.6.2.5.2 CURSEUR

Afin de situer dans cet espace les éléments qu'il crée ou qu'il manipule, l'utilisateur dispose d'un curseur qui se déplace dans les trois dimensions. Ce curseur, une petite croix à six branches dans l'espace, va être déplacé grâce à une série de commandes, ou à l'aide d'une souris ou d'un digitaliseur. Les commandes permettent de le déplacer de haut en bas, de gauche à droite, et d'avant en arrière, d'un nombre quelconque de positions. A tout instant, trois compteurs affichent la position exacte du curseur.

La souris, elle, permet le déplacement rapide du curseur, mais seulement dans un plan. On peut imaginer que deux touches situées sur la souris-même permettent d'approcher ou d'éloigner le curseur perpendiculairement au plan de la souris. Ce plan peut être frontal (X=cst), horizontal (Z=cst), ou debout (Y=cst, cette constante pouvant prendre la valeur désirée).

Pour repérer plus clairement le curseur, on fait appel à une trame. Le curseur y est relié par un ou plusieurs segments de droite perpendiculaires aux plans visibles.

3.6.2.5.3 CAMERA

Il va être possible de déplacer la caméra virtuelle, soit en désignant le point visé et l'endroit de la caméra (ce qui ne sera pas toujours possible avec le curseur), soit grâce à une série de commandes contrôlant ses différents mouvements possibles.

On a également à sa disposition un zoom qui permet d'agrandir la vision de l'objet sans s'en rapprocher, donc, sans le déformer, zoom que l'on règle en spécifiant un angle d'ouverture.

3.6.2.5.4 ELAGUER

Afin de ne pas surcharger l'espace de travail de l'objet courant, il est permis de rendre invisible une sous-branche de la hiérarchie "contient", en désignant le sous-objet correspondant.

3.6.3 GESTION DES OBJETS

3.6.3.0.1 LISTES

Il est pratique de pouvoir consulter la liste des objets créés afin de choisir celui dans lequel on va rentrer ou celui qu'on va supprimer ou

incorporer dans un autre.

3.6.3.0.2 ENTRER DANS UN OBJET

Cette fonction a pour but de donner accès à l'espace de référence d'un objet, et d'afficher tout ce qui s'y trouve.

Elle n'aboutit que si le nom donné par l'utilisateur figure bien dans la liste des objets existants déjà ou dans la liste des sous-objets de l'objet dans lequel il travaille.

Si l'utilisateur ne mentionne pas de nom, cela signifie qu'il désire travailler sur un nouvel objet et il disposera alors d'un espace vierge. Il devra alors choisir la précision de l'objet.

Pour terminer, on pourrait désirer que certains objets soient protégés et qu'il faille donner un mot de passe pour pouvoir y accéder

3.6.3.0.3 SAUVER

A tout moment, on peut sauver l'objet courant, c'est-à-dire le recopier sur un support externe. Si aucun nom n'est spécifié, l'ancienne version de l'objet est remplacée par l'actuelle. Si un nom est spécifié, et que ce nom ne désigne aucun objet existant, un nouvel objet portant ce nom est créé, ayant comme définition celle de l'objet courant.

3.6.3.0.4 SORTIR

Lorsque l'on sort d'un objet et que des modifications ont été effectuées depuis le dernier sauvetage, le système demande si l'on veut le sauver afin d'éviter de perdre par inadvertance le résultat de cette mise à jour.

3.6.3.0.5 SUPPRIMER

Il est pratique de pouvoir se débarrasser des objets dont on n'a plus besoin. Mais il faut faire attention au fait que si un objet est supprimé, il disparait de tous les objets dans lesquels il était incorporé.

3.6.4 RESUME

Nous insistons sur le fait que cette liste de fonctions est arbitraire et ne constitue qu'un noyau de base destiné à rendre utilisable les applications developpées dans ce travail.

CONCEPTION D'UN OBJET

CREATION D'ELEMENTS
POINT
ARETE
TRIM-CUT

CREATION DE GROUPE

CREATION DE SOUS-OBJETS
DEPLACER
REPETER
SUPPRIMER
UNDO

CONTROLE DE L'ESPACE
TRAME
CURSEUR
CAMERA
ELAGUER

GESTION DES OBJETS

LISTES
ENTRER
SAUVER
SORTIR
SUPPRIMER

3.7 CONCLUSION

Au cours de cette deuxième partie, nous avons défini les trois objectifs particuliers de cette étude, à savoir:

- une perception exclusivement tridimensionnelle des objets créés,
- l'utilisation d'une structure hiérarchique de types et d'occurrences d'objets correspondant à une approche par décomposition en sous-objets,
- la manipulation d'un outil qui permet de déterminer précisément des points caractéristiques de l'espace.

Nous avons ensuite spécifié un outil capable de réaliser ces objectifs compte tenu des contraintes liées aux objectifs principaux.

Il nous reste à réaliser et à expérimenter cet outil avant de pouvoir tirer des conclusions générales.

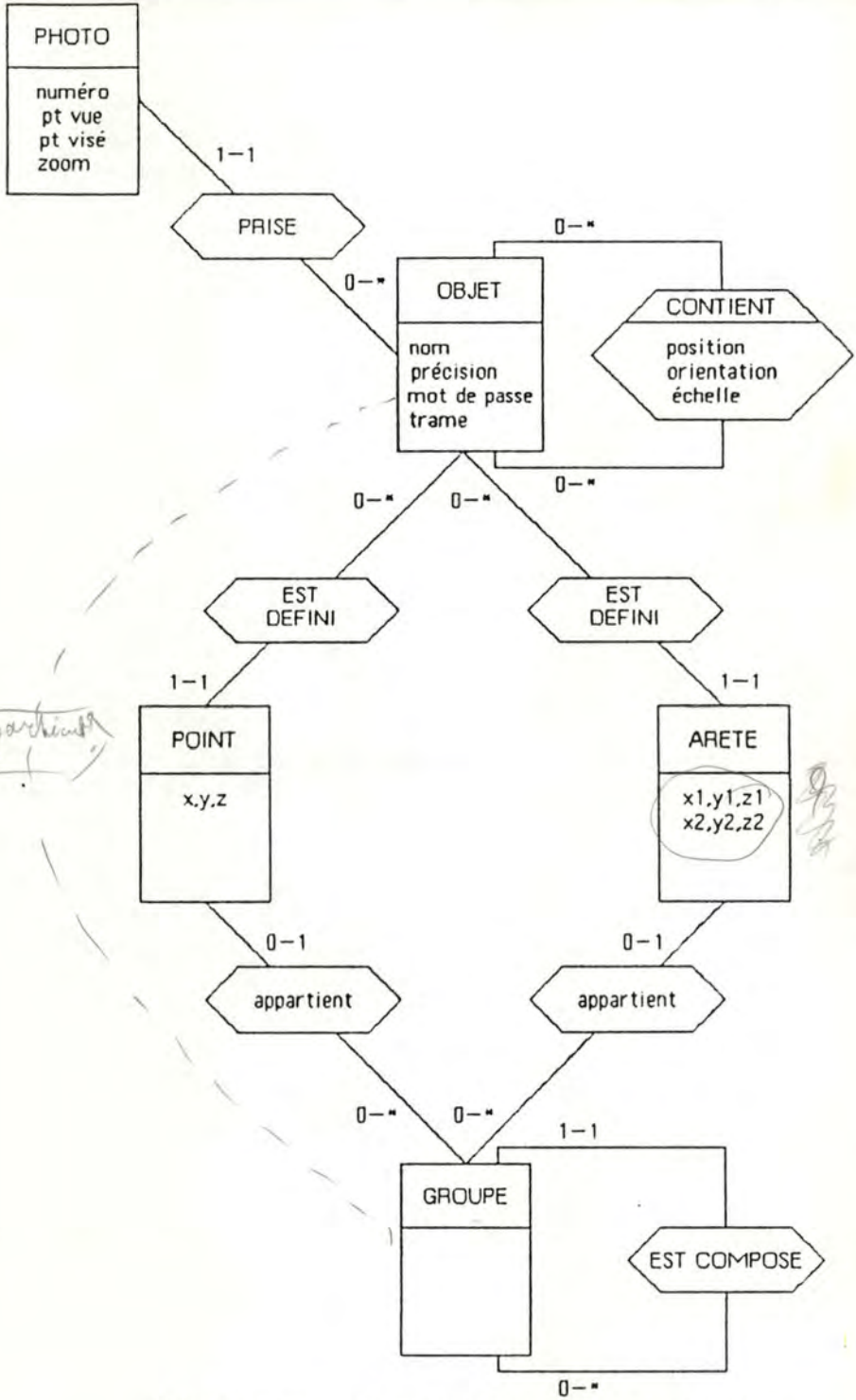
Nous allons dans la partie qui suit développer l'analyse fonctionnelle correspondant aux besoins définis dans le chapitre FONCTIONS et en proposer une architecture logique.

4 TROISIEME PARTIE

4.1 ANALYSE FONCTIONNELLE

4.1.1 SCHEMA ENTITE-ASSOCIATION

Les informations traitées par CIOS sont spécifiées suivant le modèle Entité-Association développé à l'Institut d'informatique de Namur, sous l'égide du professeur F. BODARD.



Shéma Entité—Association des informations traitées dans CIOS

4.1.2 SPECIFICATIONS(FONCTIONS)

Les fonctions correspondent à celles définies dans la deuxième partie et sont spécifiées selon le modèle dit "de la boite noire". Chaque fonction est décrite par cinq paragraphes.

Le paragraphe précédé de "OBJECTIF" décrit brièvement la raison d'être de la fonction.

Le paragraphe précédé de "INPUT" décrit en termes de valeurs d'attributs les informations dont la fonction a besoin.

Celui précédé de "OUTPUT" décrit celles qui sont générées par la fonction.

Le paragraphe précédé de "CONSULTATION" décrit en termes d'entités et d'associations les informations internes que la fonction consulte.

Celui précédé de "MODIFICATION" décrit celles que la fonction modifie.

pré/post conditions ?

4.1.2.1 LISTE DES OBJETS

OBJECTIF : Communiquer à l'utilisateur la liste des objets qu'il a déjà créé. Cette liste peut lui servir à choisir l'objet qu'il va supprimer, renommer, imprimer, ou dans lequel il va rentrer.

Afin d'éviter qu'il ne supprime sans le savoir un objet dont il veut conserver certaines occurrences, il peut connaitre la liste des objets qui en contiennent au moins une occurrence. Il peut enfin connaitre la liste des occurrences d'objets contenues dans chaque objet.

INPUT : le type de liste désiré.

CONSULTATION : T.E. OBJET et T.A. CONTIENT.

MODIFICATION :

OUTPUT : La liste telle qu'elle est définie dans l'objectif.

4.1.2.2 SUPPRESSION D'UN OBJET

OBJECTIF : ELimimer les objets dont on n'a plus besoin. Etant donné qu'un sous-objet n'est qu'une occurrence d'un objet, si l'on supprime un objet, toutes ses occurrences disparaissent.

INPUT : Le nom de l'objet.

CONSULTATION : T.E. OBJET.

MODIFICATION : Si cet objet existe, suppression de l'occurrence du T.E. OBJET correspondant, et de toutes celles du T.A. CONTIENT qui l'impliquent dans le sens "est contenu dans" ; suppression de toutes les occurrences des T.E. POINT et ARETE

associées à cet objet par un T.A. EST DEFINI ; suppression de toutes les occurrences du T.E. GROUPE qui sont associées à ces points ou a ces arêtes par un T.A. APPARTIENT.

OUTPUT :

4.1.2.3 RENOMMER UN OBJET

OBJECTIF : Modifier le nom qui sert à identifier un objet. L'ancien nom pourra, alors seulement, etre réutilisé pour désigner un nouvel objet, car deux objets ne peuvent pas porter le même nom.

INPUT : L'ancien et le nouveau nom désiré.

CONSULTATION : T.E. OBJET.

MODIFICATION : Si l'ancien nom existe et si le nouveau n'existe pas, mise à jour de la valeur de l'item NOM de l'occurrence du T.E. OBJET portant ce nom.

OUTPUT : Message d'erreur si la modification est impossible.

4.1.2.4 IMPRIMER UN OBJET

OBJECTIF : Obtenir un document représentant une vue de l'objet au moyen d'une imprimante, d'un

"plotter" ou d'une table traçante. La façon la plus simple de procéder consiste à imprimer la vue que l'on a à ce moment-là de l'objet dans lequel on se trouve. Etant donné qu'une telle impression dure un temps relativement long, l'alternative consiste à avoir pris une photo de l'objet, c-à-d d'avoir enregistré la vue en question, et de l'imprimer à un autre moment que pendant la création. Cela permet de se constituer un catalogue de photos numérotées, associé à chaque objet.

INPUT : Le nom de l'objet et la liste des photos à imprimer. Par défaut, la vue courante de l'objet courant. Il y a trois façons de spécifier cette liste: soit en en énumérant les numéros, soit, par défaut, la liste complète des photos associées à cet objet, soit en les passant toutes en revue et en sélectionnant celles désirées. Cette troisième façon est aussi l'occasion de supprimer de son catalogue les photos obsolètes.

CONSULTATION : Tout les T.E. et les T.A.

MODIFICATION : Si il y a lieu, suppression des occurrences du T.E. PHOTO dont les numéros ont été selectionnés.

OUTPUT : Si le nom de l'objet existe et si la liste est non vide, impression des vues correspondant aux photos dont le numéro existe.

4.1.2.5 RENTRER DANS UN OBJET

OBJECTIF : Fonction préliminaire indispensable pour pouvoir agir sur les éléments d'un objet (points, arêtes, groupes, sous-objets), elle sert aussi à créer un nouvel objet.

INPUT : Le nom de l'objet. Si ce nom n'existe pas et qu'on ne se trouve pas dejà dans un objet, une précision est demandée. Le mot de passe éventuel.

CONSULTATION : T.E. OBJET et T.A. CONTIENT.

MODIFICATION : Si le nom de l'objet n'existe pas, qu'on ne se trouve pas dejà dans un objet, et que la précision est strictement positive, création d'une nouvelle occurrence du T.E. OBJET portant le nom et la précision introduite.

OUTPUT : Si le mot de passe est correct ou si il n'y a pas de mot de passe, et si on n'est pas dejà dans un objet ou on y est et le nom introduit figure dans la liste des sous-objets de ces objets, alors il y a affichage de la vue standard de cet objet avec en fond son éventuel objet hôte.

4.1.2.6 COPIER UN OBJET

OBJECTIF : Permettre de créer automatiquement un nouvel objet étant la réplique exacte d'un autre. Cela peut servir par exemple à mémoriser les

différentes étapes de l'évolution d'un objet.

INPUT : Le nom de l'objet à recopier, par défaut l'objet courant dans l'état où il se trouve, et le nom du nouvel objet.

CONSULTATION : T.E. OBJET.

MODIFICATION : Si le nom du nouvel objet n'existe pas et que l'objet à recopier existe, création d'une nouvelle occurrence du T.E. OBJET ainsi que de toutes les entités et associations qui sont associées à l'objet à recopier, toutes ces occurrences ayant les mêmes caractéristiques que leurs homologues, à l'exeption du nouvel objet qui porte le nouveau nom.

OUTPUT :

4.1.2.7 SORTIR D'UN OBJET

OBJECTIF : Cette fonction est le pendant de celle "rentrer dans un objet" et permet de se libérer de la contrainte que tous les éléments reférences appartiennent a l'objet courant. C'est seulement à ce moment-là que les modifications apportées vont être mémorisées définitivement, si l'utilisateur le désire.

INPUT : Ordre de mémorisation.

CONSULTATION :

MODIFICATION : Si l'ordre de mémorisation est affirmatif, mise à jour des T.E. et des T.A. associés à l'objet que l'on quitte, en fonction des modifications effectuées depuis l'entrée dans cet objet.

OUTPUT : Si l'on se trouvait déjà dans un sous-objet, affichage de l'objet hôte.

REMARQUE : Toutes les fonctions qui suivent ne peuvent précisement être activées que dans le cadre d'un objet courant, c-à-d si l'on est préalablement rentré dans un objet.

4.1.2.8 MOT DE PASSE

OBJECTIF : Permettre de modifier la valeur du mot de passe de l'objet dans le quel on se trouve.

INPUT : La valeur du nouveau mot de passe.

CONSULTATION :

MODIFICATION : Mise-à-jour de la valeur de l'attribut "mot de passe" du T.E. OBJET correspondant à l'objet courant, même si cette valeur est vide.

OUTPUT :

4.1.2.9 DEPLACER LE CURSEUR

OBJECTIF : Déplacer la position du curseur.

INPUT : Le déplacement et le pas du déplacement (par défaut, le pas du déplacement précédant).

CONSULTATION :

MODIFICATION :

OUTPUT : Si le pas est positif et la nouvelle position appartient à l'espace, affichage à l'écran de la nouvelle position du curseur, après avoir effacé l'ancienne.

4.1.2.10 DEPLACER LA CAMERA

OBJECTIF : Cette fonction permet de modifier les caractéristiques de la caméra, à savoir sa position dans l'espace, son orientation, son inclinaison et la valeur associée à son zoom.

INPUT : Les éventuelles nouvelles valeurs, les autres restant inchangées. Initialement, tous ces paramètres reçoivent une valeur par défaut.

CONSULTATION :

MODIFICATION :

OUTPUT : Si la nouvelle position appartient à l'espace et si la valeur du zoom est positive, réaffichage de l'objet suivant son nouveau point de vue.

4.1.2.11 MODIFIER LA TRAME

OBJECTIF : Permettre de redéfinir les paramètres de la trame associée à l'objet, ces paramètres ayant une valeur initiale par défaut et représentant l'écart entre deux points de la trame, les limites fond-inférieur- gauche et avant-supérieur-droite de la trame dans l'espace, l'endroit et la visibilité ou l'invisibilité des traces horizontale, verticale et frontale de cette trame.

INPUT : Les éventuelles nouvelles valeurs, les autre restant inchangées.

CONSULTATION :

MODIFICATION : Si l'écart est supérieur à la précision, si les limites appartiennent à l'espace et si les endroits se situent entre ces limites, mise à jour éventuelle de ces paramètres.

OUTPUT : Modification de la vue à l'écran en fonction de ces mises à jour.

4.1.2.12 ELAGUER LA VUE

OBJECTIF : Rendre momentanément invisible un sous-objet de l'objet courant, ceci pour en clarifier la vue et accélèrer l'affichage.

INPUT : Le nom du sous-objet à rendre invisible.

CONSULTATION : T.E. OBJET et T.A. CONTIENT.

MODIFICATION :

OUTPUT : Si le sous-objet en question existe bien dans l'objet courant, réaffichage de la même vue sans les éléments de ce sous-objet, et ceci jusqu'à nouvel ordre.

4.1.2.13 REAFFICHER L'OBJET

OBJETIF : Pendante de la précédante, cette fonction permet de revoir l'objet avec la totalité de ses sous-objets.

INPUT :

CONSULTATION : T.E. OBJET, POINT, ARETE, GROUPE, et T.A. CONTIENT.

MODIFICATION :

OUTPUT : Affichage suivant la vue courante de la

totalité de l'objet.

4.1.2.14 NETTOYER L'OBJET

OBJECTIF : Effacer les éventuelles lignes de construction.

INPUT :

CONSULTATION : T.E. OBJET, GROUPE, POINT et ARETE, et T.A. CONTIENT.

MODIFICATION :

OUTPUT : Réaffichage de la vue courante élaguée, sans les lignes de construction.

4.1.2.15 PRENDRE UNE PHOTO

OBJECTIF : Mémoriser une vue particulière de l'objet dans le but de l'imprimer par la suite. Ces photos sont numerotées automatiquement d'après leur ordre de prise.

INPUT :

CONSULTATION :

MODIFICATION : Création d'une occurrence du T.E. PHOTO ayant comme valeurs celles de la caméra

courante et le numéro d'ordre de prise.

OUTPUT :

4.1.2.16 COPIE D'OBJET

OBJECTIF : Rajouter aux arêtes de l'objet courant une copie des arêtes d'un autre objet. Ces arêtes devienent un groupe de l'objet hôte et pourront donc être déplacées par la suite. Elles conservent en outre leur structure hiérarchique de groupe. L'objet hôte hérite également des sous-objets de l'objet copié. Cela ressemble un peu à la fonction "copier un objet" dans l'autre sens, exepté qu'ici l'objet hôte existe déjà et contient déjà des arêtes et des sous-objets.

INPUT : Le nom de l'objet a copier, un endroit, une échelle et une orientation.

CONSULTATION : T.E. OBJET.

MODIFICATION : Si le nom de l'objet à recopier existe et n'est pas celui de l'objet hôte, création d'un double de toutes les occurrences de CONTIENT, POINT, ARETE et GROUPE associées à l'objet à recopier et association de ces occurrences à l'objet hôte.

OUTPUT : Réaffichage de la vue courante avec ses nouvelles arêtes. Il est possible qu'un problème de précision se pose. En effet, si l'objet à

recopier a été conçu avec une plus grande précision que celle de l'objet hôte, il risque d'arriver une perte d'information due au fait que le système aura dû arrondir certaines valeurs.

4.1.2.17 CREER UN POINT

OBJECTIF : Ajouter dans l'espace de l'objet courant la matérialisation d'un point particulier de cet espace.

INPUT : L'introduction d'un endroit précis de l'espace se fait à l'aide de la fonction POINT.

CONSULTATION : T.E. POINT.

MODIFICATION : Si ce point appartient bien à l'espace de réfférence de l'objet courant, et si il n'existe pas déjà, création d'une nouvelle occurrence du T.E. POINT ayant ses coordonnées et associé à cet objet.

OUTPUT : Affichage de ce point symbolisé par une petite croix a six branches.

4.1.2.18 CREER UNE ARETE

OBJECTIF : Rajouter une arête dans l'espace de l'objet courant. Le point et l'arête sont actuellement les deux seuls éléments que l'on

puisse créer. On peut imaginer que d'autres fonctions viennent s'ajouter, permettant de créer des quadrilatères orthogonnaux, des cercles, des ellipses, des parallélépipèdes rectangles, des polyèdres, des ellipsoïdes, conoïdes, hyperboloïdes, etc,...

INPUT : Deux points à l'aide de POINT.

CONSULTATION : T.E. ARETE.

MODIFICATION : Si ces deux points appartiennent bien à l'espace de réfférence de l'objet courrant et si l'arête n'existe pas déjà, création d'une nouvelle occurrence du T.E. ARETE associée à cet objet.

OUTPUT : Affichage de cette arête.

4.1.2.19 PROLONGER UNE ARETE

OBJECTIF : Cette fonction couramment utilisée dans les logiciels grapiques commercialisés permet de rallonger ou de raccourcir une arête jusqu'à un endroit précis. Il s'agit donc d'une modification particulière qui équivaut à une suppression suivit d'une création.

INPUT : L'arête est désignée grâce à une des fonctions surnommées ED (pour Element Detection), l'endroit en question est introduit à l'aide de POINT.

CONSULTATION : T.E. ARETE.

MODIFICATION : Si l'arête existe et si ce point appartient à l'espace et se situe sur la droite comprenant cette arête, mise-à-jour de l'occurrence du T.E. ARETE correspondant à l'arête désignée.

OUTPUT : Effacement de l'ancienne arête et affichage de la nouvelle.

4.1.2.20 COUPER UNE ARETE

OBJECTIF : Cette fonction similaire à la précédante permet de créer deux arêtes à partir d'une seule, à condition de désigner un point situé entre ses deux extrémités. Contrairement aux apparences, elle est égallement très usitée.

INPUT : L'arête est désignée à l'aide d'une ED, et le point à l'aide de POINT ou d'un plan.

CONSULTATION : T.E. ARETE.

MODIFICATION : Si cette arête existe, mise-à-jour de l'occurrence de l'ancienne arête et création d'une nouvelle occurrence du T.E. ARETE.

OUTPUT : Conceptuellement, il n'y a aucun changement visible à l'écran. Mais si cela pose des problèmes à l'utilisation, il suffit de rendre visible ce point caractéristique jusqu'au

réaffichage suivant.

4.1.2.21 REGROUPER DES ELEMENTS

OBJECTIF : Relier virtuellement entre-eux des éléments (points, arêtes ou groupes existants) pour en faire un groupe. L'avantage d'un groupe est qu'il peut être manipulé globalement.

INPUT : Chacun des éléments désignés à l'aide d'une ED.

CONSULTATION : T.E. POINT, ARETE et GROUPE.

MODIFICATION : Si au moins deux de ces éléments existent, création d'une occurrence du T.E. GROUPE ainsi que des T.A. EST_COMPOSE et APPARTIENT l'associant aux éléments qui le forment.

OUTPUT :

4.1.2.22 DEGROUPER DES ELEMENTS

OBJECTIF : Rendre indépendants et par conséquent accessibles les éléments d'un groupe existant.

INPUT : Le groupe à dissoudre à l'aide d'une ED.

CONSULTATION : T.E. GROUPE.

MODIFICATION : Si ce groupe existe, suppression de l'occurrence du T.E. GROUPE correspondant au groupe à dissoudre, ainsi que celle des T.A. EST_COMPOSE et APPARTIENT auquels il est associé.

OUTPUT :

4.1.2.23 AJOUTER UN SOUS-OBJET

OBJECTIF : Faire figurer les éléments d'un objet existant en permanence dans l'objet courant, sans qu'ils ne deviennent accessibles : on ne pourra manipuler ce sous-objet que globalement.

INPUT : Le nom de l'objet appelé à devenir sous-objet, un endroit de l'espace grâce à POINT, trois angles déterminant l'orientation et valant zéro par défaut, et un facteur d'échelle valant un par défaut.

CONSULTATION : T.E. OBJET.

MODIFICATION : Si le nom de l'objet appelé à devenir sous-objet existe et n'est pas celui de l'objet hôte, si cet endroit appartient à l'espace et si le facteur est stictement positif, création d'une nouvelle occurrence du T.E. CONTIENT associant les deux objets et prenant les valeurs d'item introduites.

OUTPUT : Affichage dans la position choisie et d'après la vue courante du nouveau sous-objet.

Ajout dans la liste des sous-objet de l'objet courant affichée dans la partie menu, du nom de l'objet introduit, ou, s'il possédait déjà une occurrence dans l'objet corrant, incrémentation de l'indice qui figure à coté de son nom.

4.1.2.24 DEPLACER UN ELEMENT

OBJECTIF : Il s'agit de la manipulation la plus simple que l'on peut opérer sur un point, une arête, un groupe ou même un sous-objet. Elle comprend une translation, une rotation dans trois plans et un changement d'échelle dans trois directions.

INPUT : L'élément à déplacer désigné par une ED ou par son nom s'il s'agit d'un sous-objet. Deux points introduits grâce à POINT désignant respectivement un point de l'élément avant et ce même point après le déplacement (dans le cas d'un sous-objet, un seul point suffit et désigne l'endroit de l'origine de l'espace de réfférence du sous-objet après le déplacement). Les trois angles de rotation dans les trois plans directeurs et trois points par rapport auquels doivent s'effectuer ces rotations respectives (par défaut l'origine dans le cas de sous-objets). Les trois facteurs de changement d'échelle dans les trois directions et les trois points qui resteront fixes dans ces trois directions (dans le cas d'un sous-objet : un seul facteur et pas de point).

CONSULTATION : T.E. ARETE, POINT et GROUPE, ou T.E. OBJET et T.A. CONTIENT.

MODIFICATION : Si l'élément désigné existe, si les points introduits appartiennent à l'espace et si le facteur d'échelle est strictement positif, mise-à-jour des valeurs d'item COORDONNEES des occurrences des T.E. POINT et ARETE correspondant à l'élément déplacé. Dans le cas d'un sous-objet : mise-à-jour des valeurs des items POSITION, ORIENTATION et ECHELLE de l'occurrence du T.A. CONTIENT associé à ce sous-objet.

OUTPUT : Effacement de l'élément avant le déplacement et affichage de celui-ci après celui-là.

4.1.2.25 REPETER UN OBJET

OBJECTIF : Créer une ou des copies d'un élément sans devoir le reconstruire complètement. Répéter revient à déplacer un certain nombre de fois sans supprimer l'élément déplacé, chaque déplacement se faisant par rapport au résultat du déplacement précédant.

INPUT : Les mêmes que pour le déplacement plus le nombre de répétitions.

CONSULTATION : T.E. ARETE, POINT et GROUPE, ou T.E. OBJET et T.A. CONTIENT.

MODIFICATION : Si l'élément désigné existe, si les points introduits appartiennent à l'espace et si le facteur et le nombre sont strictement positifs, création d'autant d'occurrence des T.E. POINT, ARETE et GROUPE que nécéssaire. Dans le cas d'un sous-objet, création d'une nouvelle occurrence du T.A. CONTIENT.

OUTPUT : Affichage des nouveaux éléments.

4.1.2.26 SUPPRIMER UN ELEMENT

OBJECTIF : Eliminer de l'espace de référence de l'objet courant un point, une arête, un groupe ou un sous-objet.

INPUT : Le nom du sous-objet, sinon l'élément désigné au moyen d'une ED.

CONSULTATION : T.E. ARETE, POINT et GROUPE, ou T.E. OBJET et T.A. CONTIENT.

MODIFICATION : Si l'élément désigné existe, suppression de l'occurrence du T.A. CONTIENT correspondant au sous-objet, ou suppression de toutes les occurrences des T.E. POINT, ARETE et GROUPE correspondant à cet élément.

OUTPUT : Effacement de l'élément supprimé.

4.1.2.27 ANNULATION

OBJECTIF : Annuler tous les effets de la dernière fonction activée.

INPUT :

CONSULTATION :

MODIFICATION : Dépend de la fonction annulée.

OUTPUT : Dépend de la fonction annulée.

4.2 ARCHITECTURE LOGIQUE

4.2.1 DECOMPOSITION EN MODULES

Les différents modules de l'architecture logique forment une hiérarchie "utilise" à cinq niveaux.

Le niveau le plus bas renferme les outils de base qui permettent de stocker les informations et de les manipuler, ainsi-que de dialoguer avec l'utilisateur au moyen de l'écran, du clavier, de la souris, de l'imprimante, du plotter, etc,... Le contenu de ces module est très dépendant du hardware et du langage de programmation, et constitue donc la partie de l'application dans laquelle est caché ce qu'il y a de moins "portable".

Le deuxième niveau est un outil intermédiaire particulier à l'application: il interprète les messages en provenance du clavier et de la souris, et gère en même temps le curseur.

Le niveau trois contient des outils évolués qui servent d'intermédiaires entre les modules principaux et l'utilisateur.

Le niveau quatre se compose d'une série de modules intelligents qui dialoguent avec l'utilisateur par l'intermédiaire des outils des niveaux inférieurs.

A chacun d'eux est confié la responsabilité d'une fonction de l'analyse fonctionnelle.

Au dernier niveau se trouve un coordinateur qui est chargé d'activer l'un ou l'autre module intelligent, en fonction du résultat de l'activation précédante.

4.2.1.1 MODULES DE NIVEAU 1

Il y a deux modules de base. L'un contient les outils de base internes, l'autre, les outils de base externes. Le premier gère les informations stockées, le second gére l'affichage à l'écran de graphiques ou de textes, la lecture au clavier ou à la souris, et l'impression.

4.2.1.2 MODULE DATA

Ce module contient d'une part la définition des informations qui seront stockées en permanence, et d'autre part la définition de celles qui ne sont nécessaires que lors de l'exécution.

Il n'est visible que par un "interface" constitué d'un ensemble de primitives qui permettent de créer, consulter, mettre-à-jour et supprimer des informations.

4.2.1.3 MODULE ECRAN

De même, ce module n'est visible que par une liste de primitives qui permettent de manipuler les supports d'information particuliers que sont l'écran, le clavier, etc.

4.2.1.4 MODULE DE NIVEAU 2

4.2.1.5 MODULE IN

La principale fonction de ce module est de recevoir et de transformer en un code un message en provenance de l'utilisateur, et de le renvoyer au module supérieur qui lui a fait appel. C'est ce module supérieur qui interprétera ce code comme étant un caractére, un chiffre ou une commande.

Sa deuxième fonction se justifie par le fait que l'utilisateur peut, à tout moment ou il a la parole, vouloir faire se déplacer le curseur, information qui n'intéresse en rien le module supérieur. Dans ce cas, le module IN exécute la requête de l'utilisateur, mais reste actif tant qu'il ne s'est pas acquité de sa principale fonction.

4.2.1.6 MODULES DE NIVEAU 3.

Les quatre modules du niveau 3 sont des outils évolués de saisie. En effet, outre les commandes ou les caractères fournis par le module IN (niveau 2), les modules de niveau 4 se nourrissent de quatre denrées particuliaires: des chaines de caractères, des nombres entiers, des coordonnées de points et des éléments existants (points, arêtes, ou groupes).

4.2.1.7 MODULE STRING

Les chaines de caractères permettent à l'utilisateur de désigner des objets existants ou à créer en les nommant.

4.2.1.8 MODULE ENTIER

Les nombres entiers servent à introduire différentes choses comme la valeur du zoom ou les coordonnées d'un point.

4.2.1.9 MODULE POINT

Nous avons vu dans l'étude des besoins qu'il y avait bien d'autres façons d'obtenir les coordonnées d'un point, mais il est intéressant de constater que l'outil évolué chargé de saisir un

point utilise parfois l'outil chargé se saisir un entier. Il y a donc une petite hiérarchie "utilise" au sein des outils évolués du niveau 3.

4.2.1.10 MODULE ELEMENT

Désigner un élément pour par exemple le déplacer ou le supprimer, ne peut se faire en le nommant à l'aide de l'outil de saisie de chaine de caractères puisque ces éléments ne portent pas de noms. La seule possibilité est d'aller les pointer dans l'espace-même à l'aide du pointeur. Vu l'imprécision de cette technique, il est bon que l'utilisateur percoive se détacher l'élément choisi et puisse confirmer ou infirmer son choix. C'est tout cela dont a la charge le module de saisie d'élément.

4.2.1.11 MODULES DE NIVEAU 4

Chaque module de niveau 4 reprend une fonction de l'analyse fonctionnelle. Il y en a donc 26 ! La seule différence se trouve dans le huitième module correspondant à la huitième fonction: "déplacer le curseur". Nous avons vu que le déplacement du curseur était géré par le module de niveau 2: IN. Il ne reste au module huit du niveau 4 que la charge de modifier le pas de ce déplacement en fonction du désir de l'utilisateur.

Tous ces modules n'ont aucun arguments: ils

s'occupent eux-même de se procurer les données dont ils ont besoin, à l'aide des outils de saisie, et, en fonction de leur validité, de mener à bien la tâche qui leur est assignée, ou simplement de s'auto-avorter. Le résultat est si tout va bien, une mise-à-jour des informations mémorisées, et dans tous les cas, la commande suivante de l'utilisateur.

4.2.2 SPECIFICATION DES MODULES

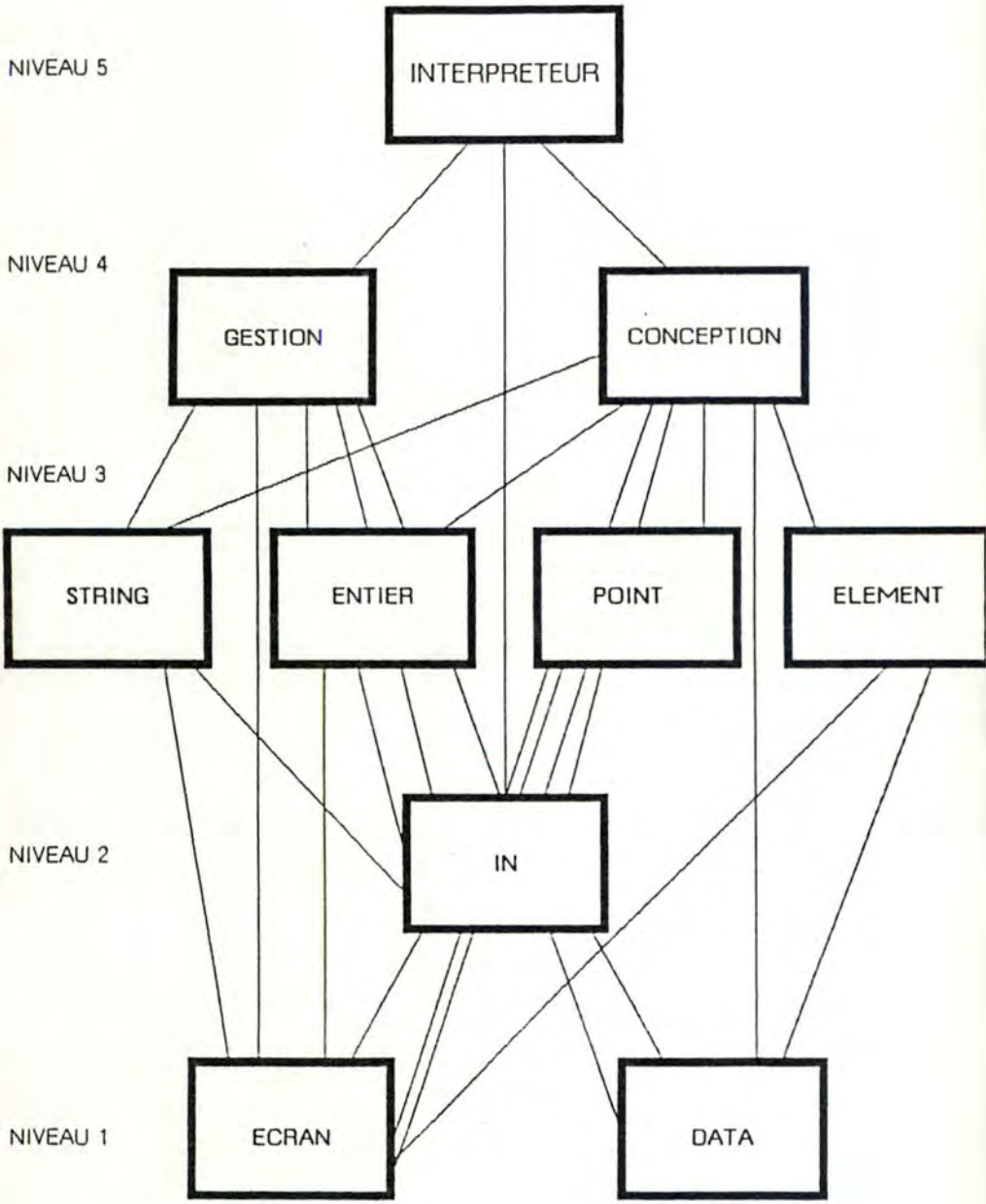
Ces modules ont été spécifié d'après le modèle de la "boîte noire":

1- arguments.
2- résultats.
3- pré-conditions.
4- post-conditions.

4.2.2.1 MODULES DE NIVEAU 1

4.2.2.2 MODULE DATA

(Liste de primitives)



Hiérarchie "utilise" des modules de l'architecture logique de CIOS

LISTOBJ.

1-
2-
3- une liste de noms.
4- cette liste contient les noms de tous les objets existants.

LISTSSOBJ.

1- un nom d'objet.
2- l'objet désiné par ce nom existe.
3- une liste de noms.
4- cette liste contient les noms de tous les objet qui sont sous-objets au premier niveau de la hiérarchie "contient" de l'objet désiné en argument.

LISTOBJETSS.

1- un nom d'objet.
2- l'objet désigné par ce nom existe.
3- une liste de noms.
4- cette liste contient les noms de tous les objet qui contiennent au premier niveau de leur hiérarchie l'objet désigné en argument.

CREER.

1- un nom d'objet.
2- ce nom ne désigne aucun objet existant.

3- mise-à-jour de la base de données.
4- ajout d'un élément portant le nom argument dans la liste des objets et d'une liste vide de sous-objets et d'arêtes associées a ce nouvel objet.

CHARGER.

1- un nom d'objet.
2- ce nom désigne un objet existant.
3- mise-à-jour de la mémoire centrale.
4- affectation des valeurs de l'objet désigné en argument ainsi que de ses sous-objets et arêtes associées, respectivement à l'objet courant et aux listes des arêtes courantes et des sous-objets courants.

SAUVER.

1- un nom d'objet.
2- ce nom désigne un objet existant.
3- m-à-j de la b.d.
4- affectation des valeurs contenues dans l'objet courant, les arêtes courantes et les sous-objets courants à l'endroit de la liste des objets correspondant au nom argument.

DETRUIRE.

1- un nom d'objet.
2- ce nom désigne un objet existant.
3- m-à-j de la b.d.

4- suppression de la liste des objets de l'élément portant le nom argument, ainsi que de ses deux listes associées.

EXIST.

1- un nom d'objet.
2-
3- une valeur bouléenne.
4- cette valeur vraie si le nom argument correspond à un objet existant, fausse sinon.

INLISTSSOBJET.

1- un nom d'objet.
2-
3- une valeur bouléenne.
4- cette valeur est vraie si le nom argument désigne un sous-objet courant.

AJOUTPILE.

(pourrait se faire automatiquement lors du chargement, mais on a choisi de dissocier l'objet courant du sommet de la pile afin d'éviter de passer chaque fois par un pointeur.)

1-
2-
3- m-à-j de la m.c.
4- ajout au sommet de la pile un élément ayant

comme valeurs celles de l'objet courant.

RETIRERPILE.

1-
2-
3- m-à-j de la m.c.
4- suppression du sommet de la pile.

PILEVIDE.

1-
2-
3- une valeur bouléenne.
4- cette valeur est vraie si la pile d'objets courants en m.c. est vide, fausse sinon.

MODNOMOBJET.

1- deux noms d'objets.
2- le premier déisgne un objet existant et le deuxième ne correspond à aucun objet existant.
3- m-à-j de la b.d.
4- le nom de l'objet désigné par le premier prend la valeur du deuxième. ATTENTION : cette m-à-j doit également se faire dans la liste des sous-objets de tous les objets contenant cet objet.

MODPASSEOBJET.

1- un nom d'objet et un mot quelconque.
2- ce nom désigne un objet existant.
3- m-à-j de la b.d.
4- le mot de passe de l'objet désigné par le nom argument prend la valeur du mot argument.

Remarque : on entre seulement à partir d'ici dans la partie consacrée à la conception proprement dite.

AJOUTARETE.

1- six entiers.
2- ces entiers sont compris dans les limites et compatibles avec la précision de l'objet courant.
3- modification de la m.c.
4- ajout d'un élément dans la liste des arêtes courantes contenant les valeurs passées en argument.

ARETEPOINTEUR.

1- six entiers.
2- ces entiers sont compris dans les limites et compatibles avec la précision de l'objet courant.
3- une valeur de pointeur.
4- cette valeur pointe vers l'élément de la liste des arêtes courantes dont les valeurs se rapprochent le plus de celles passées en argument.

SUPPRIMARETE.

1- une valeur de pointeur.
2-
3- modifiction de la m.c.
4- suppression de l'élément de la liste des arêtes courantes pointé par la valeur argument.

AJOUTSSOBJ.

1- un nom d'objet et sept entiers.
2- le nom désigne un objet existant, les trois premiers entiers représentent une position possible dans l'objet courant, les trois suivants, une orientation possible et le dernier, un facteur possible d'échelle.
3- modification de la m.c.
4- ajout dans la liste des sous-objets courants d'un élément contenant les valeurs entrées en argument, une visibilité vraie et un indice à jour (c'est-à-dire un si ce sous-objet est la première occurence de l'objet correspondant à être introduite dans l'objet courant, son numéro d'ordre sinon).

SUPPSSOBJET.

1- un nom d'objet et un entier.
2- ce nom correspond à un sous-objet courant.
3- modification de la m.c.
4- suppression de l'élément de la liste des sous-objets courants dont le nom et l'indice

correspondent aux arguments, rien sinon.

MODVISISSOBJ.
MODPOSISSOBJ.
MODORISSOBJ.
MODECHSSOBJ.

1- un nom d'objet, un entier et soit une ou trois valeurs.
2- ce nom correspond à un sous-objet courant, la valeur seule est bouléenne ou correspond à un facteur d'échelle possible, les trois valeurs sont soit une posision possible, soit une orientation possible.
3- modification de la m.c.
4- mise-à-jour de l'élément de la liste des sous-objets courants correspondant au nom et à l'indice argument, d'après la ou les valeurs données en argument.

4.2.2.3 MODULE ECRAN

MVTCURSEUR.

1-
2-
3-4- affichage a l'écran du curseur d'après l'endroit donné par la position du curseur en m.c., rien si cette position donne un point situé

hors de l'écran.

DESSINETOUT. 1-
2-
3-4- affichage de toutes les arêtes et de tous les points de l'objet courant, de tous les sous-objets courants et de tous leurs leurs sous-objets, à l'exeption de ce qui sort de l'écran.

DESSINESSOBJ.

1- une valeur de pointeur.
2- cette valeur pointe vers un sous-objet courant.
3-4- affichage de toutes les arêtes du sous-objet courant pointé et de celles de tous ses sous-objets.

DESSINEARETE.

1- une valeur de pointeur.
2- cette valeur pointe vers une arête courante.
3-4- affichage de l'arête ou du point pointé, sauf s'il sort de l'écran.

EFFACEARETE.

1- une valeur de pointeur.
2- cette valeur pointe vers une arête courante.
3-4- effacage de l'écran de l'arête ou du point pointé.

EFFACESSOBJ.

1- une valeur de pointeur.
2- cette valeur pointe vers un sous-objet courant.
3-4- affichage de l'écran de toutes les arêtes du sous-objet courant pointé et de tous ses sous-objet.

EFFACETOUT.

1-
2-
3-4- vide l'écran de ses arêtes et redessine la trame éventuelle.

RESSORTIR.

1- une valeur de pointeur.
2- cette valeur pointe vers une arête courante ou vers un sous-objet courant.
3-4- affichage d'une manière particulière de l'élément pointé.

4.2.2.4 MODULE DE NIVEAU 2 : IN

1- une touche du clavier ou de la souris.
2-
3- une valeur de code et la mise-à-jour de mémoire centrale.

4- cette valeur est un entier qui correspond à une touche du clavier suivant un code défini. La mise-à-jour concerne la position courante du curseur dans le cas ou l'argument correspond à une des commandes du curseur, auquel cas cet argument n'est pas traduit en code.

4.2.2.5 MODULES DE NIVEAU 3

4.2.2.6 MODULE STRING

1- un nombre.
2- ce nombre est un entier compris entre 1 et N et représente le maximum de caractères désirés.
3- une chaine de caractères,
   un nombre,
   un code.
4- la chaine de caractère ne contient que des caractères alphanimériques dont le nombre valides est dans le nombre résultat et varie de zéro au nombre de l'argument, le code a la valeur de la touche "return" si le string est valide, la commande suivante sinon.

4.2.2.7 MODULE ENTIER

1- un nombre.
2- ce nombre est un entier compris entre 1 et N et

représente le maximum de chiffres désirés.
3- deux nombres et un code.
4- le deuxième nombre, compris entre zéro et le nombre argument plus un (à cause du signe), donne le nombre de chiffre de l'entier contenu dans le premier, le code contient la valeur de "return" si l'entier est valide, la commande suivante sinon.

4.2.2.8 MODULE ELEMENT

1-
2-
3- une valeur de pointeur et un code.
4- cette valeur pointe vers un élément de l'objet courant, le code vaut "return" si l'utilisateur a confirmé son choix, la commande suivante sinon.

4.2.2.9 MODULE POINT

1-
2-
3- trois entiers et un code.
4- les trois entiers représenttent les coordonnées d'un point possible de l'objet courant, le code vaut "return" si ces coordonnées sont valides, la commande suivante sinon.

4.2.2.10 MODULES DE NIVEAU 4

Ces modules, comme nous l'avons vu au début de ce chapitre, ne possèdent pas d'arguments. Ils se chargent eux-même d'interroger l'utilisateur ou la base de données en fonction de leurs besoins, et prennent les décisions qu'il convient pour accomplir leur fonction, ou s'il le faut, avorter le traitement. Ce sont donc des modules qui n'exécutent eux-même aucun traitement

Le résultat de ces module est soit une mise-à-jour de la base de données et -ou- de la mémoire centrale, soit un affichage à l'écran ou une impression, soit les deux.

Ce sont en fait des modules qui n'exécutent eux-même aucun traitement, ils se contentent d'activer des modules inférieurs et d'éffectuer des tests en suivant un scénario précis.

Dans les spécifications qui vont suivre, le "1-" renseigne le type de résultat et le "2-" le scénario.

Dans tous les modules qui utilisent les modules de niveau 3, il y a une constante au point de vue scénario qui ne sera pas répétée : lorsqu'on lui demande un point, un entier, un nom ou de désigner un élément, il est normal qu'il arrive que l'utilisateur ne puisse ou ne veuille pas honorer cette demande. Dans ce cas, il est évident que le module concerné ne peut qu'avorter.

Il est également possible que l'utilisateur désire garder l'ancienne valeur si on lui demande un nombre ou un nom. Dans ce cas, il tappe par convention "return", ce qui a pour effet de renvoyer un nom ou un nombre valide mais vide. Au module appelant de l'interpréter.

D'autre part, il a été décidé que tous les modules de ce niveau renvoient le code de la commande suivante. Ceci va notamment permettre aux effets momentanés de ces modules (affichage de messages, etc,...) de persister tant que ceux-ci gardent la main, c'est-à-dire jusqu'au choix de la commande suivante, moment auquel ils profitent pour "nettoyer" leur écran.

Pour dialoguer avec l'utilisateur, tous ces modules vont lui envoyer différents messages qui seront affichés à des endroits bien définis selon le type de message...

MODULE LISTE

1- Affichage.
2- Il y a le choix parmis trois listes différentes : celle de tous les objets existants, celle des sous-objets d'un objet précis ou celle des objets qui contiennent un objet donné. Ces listes ne sont affichées à l'écran que le temps de l'activation du module.

MODULE DESTRUCTION.

1- Modification de la b.d. et de la m.c.
2- Si l'objet à détruire existe et que l'utilisateur confirme, il sera supprimé de la liste des objets et de celle des sous-objets des tous les objets qui le contiennent, y compris des sous-objets courants si il y figure.

MODULE RENOMMER.

1- Modification de la b.d. et de la m.c.
2- Si l'objet à renommer existe et si le nouveau nom n'est pas déjà utilisé, la mise-à-jour peut se faire dans la liste des objets et dans celle des sous-objets de tous les objets qui le contiennent, y compris dans la pile des objets courants et dans les sous-objets courants.

MODULE IMPRIMER.

MODULE RENTRER.

1- Modification de la m.c. et affichage.
2- Si on n'est pas en mode conception, l'utilisateur peut rentrer soit dans un objet existant si il en connait le mot de passe éventuel, soit dans l'objet sans nom à condition de donner la précision. Si on est en mode conception, il peut rentrer dans un des sous-objets courants à condition toujours d'en connaître le mot de passe. Cette opération a pour effet d'assigner les valeurs de l'objet

correspondant (l'objet vide ne contient rien, sinon une précision) dans l'objet courant, et de rajouter cet objet courant au sommet de la pile des objets courants. Les valeur de la position du curseur, de la carméra et du point courant sont initialisées aux valeurs par défaut. La valeur de la visibilité de tous les sous-objets courants est initialisée à "vrai. Le nouvel objet courant est ensuite affiché avec éventuellement l'objet dont il est l'hôte.

MODULE DUPLIQUER.

1- Modification de la b.d.
2- L'effet est la création d'un nouvel objet dans la liste des objets, portant un nouveau nom et contenant les valeurs d'un objet existant. Cet objet existant est soit l'objet courant si l'on se trouve en mode conception, soit un objet désigné par l'utilisateur.

MODULE RECOPIER.

1- Modification de la b.d.
2- L'effet est de recopier les valeurs contenues dans l'objet courant dans l'objet correspondant, à condition que l'on se trouve en mode conception et que l'objet courant ne soit pas l'objet vide.

MODULE SORTIR.

1- Modification de la m.c. et affichage.
2- Sortir signifie simplement supprimer le sommet de la pile des objets courants. Si cette pile est alors vide, on passe en mode management, sinon, l'objet courant prend les valeurs de l'objet se trouvant au sommet de la pile, et est affiché, la position du curseur et du point courant ayant été réinitialisé aux valeurs par défaut. La visibilité de tous les objets courants prend la valeur "vrai".

MODULE MOTPASSE.

1- Modification de la m.c.
2- Mise-à-jour de la valeur du mot de passe de l'objet courant d'après la valeur introduite par l'utilisateur.

MODULE PASCURSEUR.

1- Modification de la m.c.
2- Mise-à-jour de la valeur du pas du déplacement du curseur d'après la valeur introduite par l'utilisateur.

MODULE CAMERA.

1- Modification de la m.c. et affichage.
2- Mise-à-jour des valeurs de la position, de l'orientation et du zoom de la caméra en fonctions des valeurs introduites par l'utilisateur, en

tenant compte du fait que cette position doit faire partie de l'espace, que l'orientation est un cap de 0 à 360 degrés et une inclinaison de 90 à -90 degrés, et que le zoom doit être strictement positif. Affichage de la nouvelle vue.

MODULE TRAME.

1- Modification de la m.c. et affichage.
2- Mise-à-jour des valeurs de la trame de l'objet courant en fonction des valeurs introduites par l'utilisateur et en respectant le fait que l'écart entre deux points de la trame doit être supérieur à la précision, que les limites de la trame doivent être comprises dans l'espace de l'objet et que leurs endroits doivent être compris dans ces limites. Affichagge du résultat de ces modifications.

MODULE ELAGUER.

1- Modification de la m.c. et affichage.
2- Effacement de l'écran du sous-objet courant désigné par l'utilisateur, et mise à "faux" de sa valeur de visibilité.

MODULE REAFFICHER.

1- Modification de la m.c. et affichage.
2- Affichage de tous les sous-objets courants dont la valeur de visibilité valait "faux" et mise à

"vrai" de ces valeurs.

MODULE COPIER.

1- Modification de la m.c. et affichage.
2- Création d'autant d'arêtes courantes qu'il faut pour accueillir toutes celles de l'objet désigné par l'utilisateur et toutes celles de ses sous-objets, à l'endroit, l'orientation et l'échelle désirée. Affichage de toutes ces nouvelles arêtes.

MODULE POINT.

1- Modification de la m.c. et affichage.
2- A moins que ce point n'existe déjà, création d'une arête courante accuillant la position introduite par l'utilisateur, et affichage de ce point à l'écran.

MODULE ARETE.

1- Modification de la m.c. et affichage.
2- A moins que cette arête n'existe déjà, création d'une arête courante pour accueillir les deux positions introduites par l'utilisateur, et affichage de cette nouvelle arête.

MODULE SUPPRIMELEMENT.

1- Modification de la m.c. et affichage.
2- Suppression de l'arête courante contenant l'élément désigné par l'utilisateur (point ou arête), si celui-ci confirme. Effacement de cet élément de l'écran.

MODULE AJOUTSSOBJ.

1- Modification de la m.c. et affichage.
2- Création d'un sous-objet courant ayant comme valeurs le nom introduit par l'utilisateur et désinant un objet existant autre que l'objet courant, la position, l'orientation et l'échelle introduite par l'utilisateur et respectant les mêmes spécifications que pous le module "copier",la visibilité à "vrai" et l'indice adéquat. Affichage de toutes les arêtes de ce nouvel objet courant.

MODULE SUPPRIMESSOBJ.

1- Modification de la m.c. et affichage.
2- Suppression du sous-objet courant portant le nom et l'indice introduits par l'utilisateur, et effacement de toutes ses arêtes.

MODULE DEPLACERSSOBJ.

1- Modification de la m.c. et affichage.
2- Même effet que la composition de la suppression d'un sous-objet et de l'ajout de ce même sous-

objet.

MODULE REPETERSSOBJ.

1- Modification de la m.c. et affichage.
2- Même effet que la succession d'un certain nombre, introduit par l'utilisateur, d'ajouts d'un sous-objet existant, également désiné par l'utilisateur, exepté le fait que la position, l'orientation et l'échelle ne sont introduites qu'une seule fois et recalculées chaque fois par rapport leurs valeurs précédantes.

4.3 IMPLEMENTATION

Pour des raisons évidentes de délais, nous n'avons pu implémenter qu'une partie des modules du niveau 4. En voici la liste accompagnée de celle des fonctions correspondantes de l'analyse fonctionnelle.

LISTE	Liste des objets
DESTRUCTION	Suppression d'un objet
RENTRER	Rentrer dans un objet
SORTIR	Sortir d'un objet
MOTPASSE	Mot de passe
PASCURSEUR	Déplacer le curseur
CAMERA	Déplacer la caméra
ELAGUER	Elaguer la vue
REAFFICHER	Réafficher l'objet
POINT	Créer un point
ARETE	Créer une arête
AJOUTSSOBJ	Ajouter un sous-objet
SUPPRIMELEMENT	Supprimer un élément
SUPPRIMESSOBJ	Supprimer un élément

4.4 PROGRAMMATION

CIOS est programmé en TURBO Pascal sur OLIVETTI M24.

5 CONCLUSION

Conclusion

L'homme a depuis toujours développé différents moyens d'expression afin de pouvoir dialoguer avec ses semblables.

Pour pouvoir exprimer ses impressions, ses états d'esprit ou ses objections, il a inventé le langage.

Puis il s'est rendu compte du fait que le langage l'aidait à réfléchir : le fait de pouvoir extérioriser ce qu'il pensait lui permettait de penser de nouvelles choses. Et il se mit à se parler à lui-même... Plus il avait envie de dire des choses, plus il enrichissait ce langage, et plus il avait des choses à dire.

Le langage serait donc un outil qui sert à concrétiser une idée dans le but de l'élaborer puis de la communiquer à ses congénères.

Il apparut rapidement que l'utilisation de symboles était une chose pratique pour exprimer les objets. De même, ces symboles permirent

d'imaginer d'autres objets et par la même occasion se sont considérablement enrichis.

Aujourd'hui, nous l'avons vu, l'homme dispose d'images très proches de la réalité pour représenter les objets qu'il imagine.

Jusqu'à présent il a principalement exprimé ces objets en deux dimensions, faute de moyens rapides. Le fait de s'exprimer en deux dimensions l'a amené à raisonner et à imaginer en deux dimensions. S'il pouvait disposer d'outils maniables permettant l'expression en trois dimensions des objets, peut-être réapprendrait-il à raisonner et à imaginer dans l'espace...

Mais un tel outil est-il envisageable de manière réaliste ? Nous espérons avoir apporté un petit élément de réponse grâce à ce travail.

Le moment est venu de dresser le bilan des objetctifs poursuivis par C.I.O.S., en se posant les questions suivantes:

- la perception de l'espace tridimensionnel est-elle bonne?
- la structure de sous-objets est-elle utile?
- l'outil de précision est-il performant?

En ce qui concerne la perception spatiale, il nous faut rester prudents. Il semble que l'appréhension de l'espace soit bien simulée et que la caméra se "conduise" facilement. Néanmoins, la vision risque de s'embrouiller rapidement à mesure que

l'objet créé devient complexe. Qu'adviendra t-il lorsque les objet seront des volumes? S'il faut attendre à chaque mouvement que le système efface les arêtes cachées et remplisse les faces, on perd l'intérêt de l'interactivité. On peut imaginer que la vision "fils de fer" ne serve qu'à la recherche de l'angle de vue. Une autre approche consisterait à ne composer qu'au moyen de surfaces qui génèreraient ultérieurement des volumes.

En ce qui concerne la structure de sous-objets, il semble que l'idée était bonne puisque le logiciel ARC+ l'avait adoptée. Nous pensons que cette approche est riche en possibilités et que toutes les audaces sont permises, y compris l'intégration de mini-systèmes experts dans la définition des objets.

Quant à l'outil de précision, il est performant mais ne représente actuellement qu'un outil de "D.A.O." (dessin assisté par ordinateur) en raison du fait qu'il ne traite que des problèmes de positions de points. En réalité il devra résoudre les problèmes concernant des arêtes (intersections de faces) et des surfaces (intersections de volumes). Le modèle simpliste des trois notions de base dans l'espace (point, droite et plan) que nous avons adopté supportera t-il la complexité des situations faisant intervenir des surfaces et des volumes?

Nous n'avons jamais prétendu apporter de solutions aux problèmes liés au graphisme 3D. Nous sommes

conscient du fait que les problèmes et leurs solutions sont à un niveau cent fois plus élevé que celui de ce mémoire. Nous voulions simplement, avant de peut-être l'aborder dans la vie professionnelle, avoir effectué une réflexion personnelle sur le sujet. Nous savions qu'une démarche isolée n'aboutirait à rien. Nous l'avons faite malgré tout,... pour "voir".

6 TABLE DES MATIERES

Chapitre Page

1 INTRODUCTION 2

2 PREMIERE PARTIE 6

2.1 INTRODUCTION 7

2.2 QUELS SONT LES GENS CONCERNES ? 8

2.2.1 TECHNIQUES TRADITIONNELLES 8
2.2.2 DIFFERENTS BESOINS 10

2.3 EVOLUTION DE LA SITUATION 12

2.4 QUELQUES LOGICIELS EXISTANTS... 15

2.4.1 MAC-DRAW 16

2.4.1.1 SUPPORT 17
2.4.1.2 ELEMENT DE BASE 17
2.4.1.3 CREATION 18
2.4.1.4 MANIPULATIONS 18
2.4.1.5 STRUCTURE DES OBJETS 18

2.4.2 PC-DRAFT 20

2.4.2.1 SUPPORT 20
2.4.2.2 ELEMENTS DE BASE 21
2.4.2.3 CREATION 21
2.4.2.4 MANIPULATIONS 23
2.4.2.5 STRUCTURE DES OBJETS 24

Chapitre Page

2.4.3 STAR 30

2.4.3.1 SUPPORT 30
2.4.3.2 ELEMENT DE BASE 31
2.4.3.3 CREATION 31
2.4.3.4 MANIPULATIONS 33
2.4.3.5 STRUCTURE DES OBJETS 34
2.4.3.6 ASSOCIATION DE COMPOSANTS 35

2.4.4 ARC+ 38

2.4.4.1 SUPPORT 39
2.4.4.2 ELEMENTS DE BASE 39
2.4.4.3 CREATION 40
2.4.4.4 MANIPULATIONS 45
2.4.4.5 STRUCTURE DES OBJETS 46

2.5 CONCLUSION 48

3 DEUXIEME PARTIE 50

3.1 INTRODUCTION 51

3.1.1 OBJECTIFS 52

3.2 ELEMENT DE BASE 54

3.3 TROIS DIMENSIONS 57

3.3.1 INTRODUCTION 57
3.3.2 AXONOMETRIE OU PERSPECTIVE? 59

Chapitre Page

3.3.3 PROBLEMES DE LA PERSPECTIVE 61

3.3.3.1 REPERE 61
3.3.3.2 CURSEUR 62
3.3.3.3 TRAME 63
3.3.3.4 CAMERA 64
3.3.3.5 ELAGUER 66

3.3.4 CONCLUSION 67

3.4 DEFINITION DE LA STRUCTURE 69

3.4.1 INTRODUCTION 69
3.4.2 HIERARCHIE EST COMPOSE DE . 70
3.4.3 INCORPORATION D'OBJETS 73
3.4.4 HIERARCHIE CONTIENT 74

3.4.4.1 INTRODUCTION 74
3.4.4.2 EXEMPLE DE LA CHAMBRE ... 75
3.4.4.3 TYPES ET OCCURRENCE 77
3.4.4.4 COMPARAISON 78
3.4.4.5 A APPROFONDIR... 79

3.4.5 CONCLUSION 81

3.5 PRECISION 82

3.5.1 AVEC UNE PRECISION DE 82
3.5.2 AVEC PRECISION 84

3.5.2.1 INTRODUCTION 84
3.5.2.2 MECANISME DE L'OUTIL 86
3.5.2.3 DEFINITION DE L'OUTIL ... 89
3.5.2.4 RESUME 94

Chapitre Page

3.5.2.5 CONCLUSION 95

3.6 FONCTIONS 96

3.6.1 INTRODUCTION 96
3.6.2 CONCEPTION DES OBJETS 97

3.6.2.1 CREATION D'ELEMENTS 97
3.6.2.2 CREATION DE GROUPE 98
3.6.2.3 CREATION DE SOUS-OBJETS . 99
3.6.2.4 MANIPULATIONS 100
3.6.2.5 CONTROLE 102

3.6.3 GESTION DES OBJETS 104
3.6.4 RESUME 106

3.7 CONCLUSION 108

4 TROISIEME PARTIE 109

4.1 ANALYSE FONCTIONNELLE 110

4.1.1 SCHEMA ENTITE-ASSOCIATION . 110
4.1.2 SPECIFICATIONS(FONCTIONS) . 111

4.1.2.1 LISTE DES OBJETS 111
4.1.2.2 SUPPRESSION D'UN OBJET .. 112
4.1.2.3 RENOMMER UN OBJET 113
4.1.2.4 IMPRIMER UN OBJET 113
4.1.2.5 RENTRER DANS UN OBJET ... 115
4.1.2.6 COPIER UN OBJET 115
4.1.2.7 SORTIR D'UN OBJET 116
4.1.2.8 MOT DE PASSE 117

Chapitre Page

4.1.2.9 DEPLACER LE CURSEUR 118
4.1.2.10 DEPLACER LA CAMERA 118
4.1.2.11 MODIFIER LA TRAME 119
4.1.2.12 ELAGUER LA VUE 120
4.1.2.13 REAFFICHER L'OBJET 120
4.1.2.14 NETTOYER L'OBJET 121
4.1.2.15 PRENDRE UNE PHOTO 121
4.1.2.16 COPIE D'OBJET 122
4.1.2.17 CREER UN POINT 123
4.1.2.18 CREER UNE ARETE 123
4.1.2.19 PROLONGER UNE ARETE 124
4.1.2.20 COUPER UNE ARETE 125
4.1.2.21 REGROUPER DES ELEMENTS .. 126
4.1.2.22 DEGROUPER DES ELEMENTS .. 126
4.1.2.23 AJOUTER UN SOUS-OBJET ... 127
4.1.2.24 DEPLACER UN ELEMENT 128
4.1.2.25 REPETER UN OBJET 129
4.1.2.26 SUPPRIMER UN ELEMENT 130
4.1.2.27 ANNULATION 131

4.2 ARCHITECTURE LOGIQUE 132

4.2.1 DECOMPOSITION EN MODULES .. 132

4.2.1.1 MODULES DE NIVEAU 1 133
4.2.1.2 MODULE DATA 133
4.2.1.3 MODULE ECRAN 134
4.2.1.4 MODULE DE NIVEAU 2 134
4.2.1.5 MODULE IN 134
4.2.1.6 MODULES DE NIVEAU 3. 135
4.2.1.7 MODULE STRING 135
4.2.1.8 MODULE ENTIER 135
4.2.1.9 MODULE POINT 135
4.2.1.10 MODULE ELEMENT 136

Chapitre Page

4.2.1.11 MODULES DE NIVEAU 4 136

4.2.2 SPECIFICATION DES MODULES . 137

4.2.2.1 MODULES DE NIVEAU 1 137
4.2.2.2 MODULE DATA 137
4.2.2.3 MODULE ECRAN 144
4.2.2.4 MODULE DE NIVEAU 2 : IN . 146
4.2.2.5 MODULES DE NIVEAU 3 147
4.2.2.6 MODULE STRING 147
4.2.2.7 MODULE ENTIER 147
4.2.2.8 MODULE ELEMENT 148
4.2.2.9 MODULE POINT 148
4.2.2.10 MODULES DE NIVEAU 4 149

4.3 IMPLEMENTATION 158

4.4 PROGRAMMATION 159

5 CONCLUSION 160

6 TABLE DES MATIERES 165

LISTE DE QUELQUES LOGICIELS EXISTAMTS

STAR. Star info S.A., chaussée de Tongres, 75, 4420 Rocourt. HP9000.

ARCHITECTE PERSONNEL. Computervision Coop., Annaconsult, place de l'université, 25, 1348 Louvain La Neuve. IBM XT-AT, HP VECTRA.

PC BAT. Batisoft,(même adresse).(mêmes machines) OLIVETTI M24.

ARCADE. Cact. (même adresse). HP9000.

SIMAB. Iez-Cad. système. ToemaaHragel ,1,9000 Gent> VAX,PRIME.

PC-DRAFT. Rhv.Georg-Glock-Strasse, 3, 4000 Düsseldorf 30. IBM XT.

ARC+. Aca. Dizengoff Center, 50,64332 Tel-Aviv. IBM AT.

BIBLIOGRAPHIE

BOSSARD Etienne, Etude et comparaison du système graphique, mise en oeuvre sous unix, mémoire, 1977-78

MATAIGNE et PEROZZO, Réalisation d'un progiciel graphique, (outil: DI 3000), mémoire,1981-82

VAN WONTERGHEM, Conception et développement d'un logiciel graphique interactif destiné à la résolution de problèmes scientifiques, (Architecture du système graphique, exécution d'un programme graphique), mémoire,1983

LUCAS M., PEYRIN J.P. et SCHOLL P.C.,Algorithme et représentation de données, Ed. MASSON, 1983-84

VULDY J.L.,Graphisme 3D sur votre micro-ordinateur, Ed. EYROLLES, Paris 1984

NEWMAN W.M. and SPROUL R.F.,Principles Of Interactive Computer Graphics, Ed. MAC GRAW-HILL BOOK COMPANY

KELLOGG S. BOOTH, Tutorial: Computer Graphics, IEEE Catalogue numéro EHO 147-9

HINDIN H.J., Special Report On Graphics Technology, COMPUTER DESIGN, May 1984

EFIMOV N.,Eléments de Géométrie Analytique, Ed. MIR. Moscou 1969.

DELANDE Georges, Géométrie Analytique, FUNDP Namur