

MODELATGE GEOMETRIC DE SOLIDS POLIEDRICS SOBRE MICROCOMPUTADOR

Giralt, J. (*), Torres, E. (*), Brunet, P. (*)

Dep. Mètodes Informàtics ETSEIB-UPC.

Josep Giralt i Adroher finalitzà els estudis d'Enginyeria Industrial l'any 1986 i des de 1985 treballa al Departament de Mètodes Informàtics en feines de modeltat geomètric i altres aspectes del CAD, que han donat lloc a articles en revistes Internacionals.

El present article descriu una adaptació per a microcomputadors del sistema de modelat DMI, discutint les prestacions del model i de les principals operacions associades. Es presentaran les característiques fonamentals del model de fronteres emprat i una descripció de les possibilitats de les diferents comandes així com els algorismes associats i la interacció amb l'usuari.

1 INTRODUCCIO

Els sistemes de CAD s'utilitzen en àrees tan diverses com el disseny mecànic, l'enginyeria civil, l'arquitectura, la decoració, el disseny electrònic, el disseny tèxtil, etc. Vàries d'aquestes àrees es caracteritzen per tractar objectes tridimensionals. Els sistemes CAD que consideren aquests tipus d'objectes poden oferir les següents prestacions: definició interactiva i senzilla de la forma tridimensional dels objectes, generació de vistes, seccions, visions realistes i plànols, càlcul de propietats geomètriques, realització de processos de càlcul i simulació complexos i connexió a sistemes de CAM. Aquests darrers sistemes generen, automàticament, una cinta de control numèric per a la construcció de prototipus dels objectes.

(*) Departament de Mètodes Informàtics, ETSEIB, Universitat Politècnica de Catalunya. El treball descrit ha estat desenvolupat per un equip format, a part dels autors, per J. Adrián, J. Carrasco, I. Navazo, J. Pascual i A. Trill del Departament de Mètodes Informàtics, i per A. Llisterri del Centre de Càlcul de la UPC en la part corresponent a l'editor 2D.

Per a poder oferir les prestacions anteriors, el computador emmagatzema una representació tridimensional completa dels sòlids, anomenada model. El mòdul d'un sistema de CAD que facilita la creació, emmagatzematge i modificació dels models rep el nom de Sistema de Modelat Geomètric.

Les prestacions que pot oferir un Sistema de Modelat Geomètric depenen del tipus de model escollit per a emmagatzemar els sòlids. Actualment, els dos models no ambigus (un model representa a un únic sòlid real) més estesos són el model de fronteres i la Geometria Constructiva de Sòlids (CSG).

Al model de fronteres, els sòlids queden determinats pels punts de la seva frontera, que són, en definitiva, els que separen el seu interior de l'exterior. La frontera es representa per un conjunt disjunt de cares que poden ser planes o corbes. Cada cara és acotada per un paràmetre anular d'arestes que s'intersecten en vèrtex. Si la cara té forats, queda acotada, a més a més, per un o més anells interns d'arestes.

El model de fronteres és poc concís, donat que necessita emmagatzemar una elevada quantitat d'informació. El seu domini (quantitat de sòlids que permet representar) està determinat pel tipus de superfícies admeses, que poden ser planes, còniques, cilíndriques, cúbiques o d'altres.

La creació d'objectes realitzant operacions booleanes entre sòlids és una operació costosa pel que fa al temps de càlcul ja que cal intersectar totes les cares d'un sòlid amb totes les de l'altre, classificar totes les cares resultants respecte als sòlids i, segons l'operació a realitzar, quedar-se amb unes cares o altres. La visualització del model i el càlcul de propietats geomètriques són senzills de realitzar, cal només conèixer la geometria de l'objecte que, en aquest cas, s'emmagatzema en el model.

La Geometria Constructiva de Sòlids es basa en la combinació de sòlids via operadors regularitzats. La representació consisteix en un arbre binari ordenat. La informació que contenen els nodes d'aquest arbre depèn de la implementació efectuada: per exemple, els nodes no terminals poden representar operadors booleanes, com unió, intersecció o diferència, i els nodes terminals, instanciacions a sòlids de determinades famílies, paràmetres definidors de semiespais o arguments de transformacions geomètriques. Així doncs, es tracta d'un sistema concís.

El domini del sistema depèn dels semiespais que limiten als seus sòlids primitius i de les operacions booleanes i transformacions geomètriques implementades. La creació del model és simple donat que l'usuari elabora l'arbre d'operacions

mitjant un conjunt de sentències.

La realització d'operacions booleanes és senzilla, ja que consisteix en combinar dos arbres per a produir l'arbre final. En canvi, degut a la informació tan poc elaborada que emmagatzema, la visualització del model i la realització de càlculs geomètrics són complexes.

El present article descriu una adaptació per a microcomputadors del sistema de modelat DMI (Bru'86), (NAB'86), discutint les prestacions del model i de les principals operacions associades. El següent apartat presenta les característiques fonamentals del model de fronteres emprat, mentre que l'apartat 3 descriu les possibilitats de les diferents comandes així com els algorismes associats. Finalment, l'apartat 4 exposa els trets fonamentals de la interacció amb l'usuari.

2 EL MODEL DE REPRESENTACIO

A continuació es justifica el model de representació emprat en el sistema que es presenta, tant des del punt de vista de la seva estructura interna com des del d'estudiar el mínim nombre d'operacions entre sòlids necessàries.

2.1 Estructura Interna Del Model

Tant el model de fronteres com la Geometria Constructiva de Sòlids són esquemes de representació vàlids per la modelització d'objectes volumètrics. Encara que el domini -o quantitat d'objectes reals modelables- és molt elevat en els dos cassos, ja s'ha mencionat que els dos sistemes tenen prestacions diferents en les diferents operacions d'edició i accés al model.

Al cas d'un sistema encarat a microcomputador, s'ha cregut més convenient usar el model de fronteres, per tal d'aconseguir una major velocitat en els processos de sortida que permeti un grau acceptable d'interactivitat. Entre l'opció de permetre únicament cares planes i la de permetre cares més complexes (per exemple quàdriques), s'ha preferit la primera, per les següents raons,

-Els algorismes i el tractament de cassos singulars en processos com els de secció i visualització realista són de complexitat molt més reduïda.

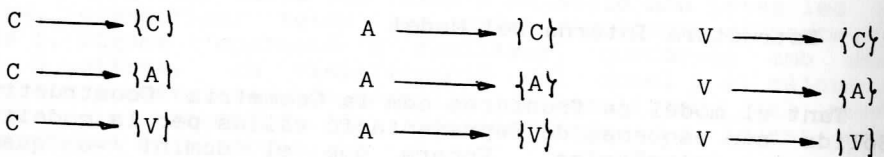
-El temps de càlcul en les principals operacions és menor, la qual cosa fa que aquestes puguin ser més interactives.

-Com es veurà al següent apartat, l'error en la representació de cilindres i altres superfícies quàdriques és controlable per l'usuari, i el sistema el corregeix en els processos de visualització.

Concretant-nos ja a la representació d'objectes delimitats per cares planes, el model de fronteres emmagatzema informació geomètrica dels elements (habitualment cares i vèrtex) de la superfície, i les relacions de connectivitat entre cares, arestes i vèrtex. La informació geomètrica més relevant per a les cares es l'equació del pla orientat, P_k , que conté cada una de les cares C_k . Pel que fa als vèrtex, s'emmagatzema les seves coordenades espacials. Cada un dels plans orientats P_k defineix un semiespai $\Pi_k \in \mathbb{R}^3$, tal que $P_k = \text{frontera}(\Pi_k)$. Amb aquesta notació, és sempre possible definir el sòlid S com una expressió booleana realitzada entre els semiespais definits per les seves n cares planes $c_1 \dots c_n$,

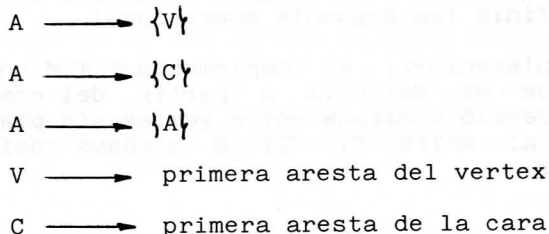
$$S \triangleq \text{expr} - \text{bool}(\pi_1, \pi_2, \dots, \pi_n) \quad |1|$$

Les relacions de connectivitat representades en el model han de permetre respondre qualsevol de les 9 interrogacions bàsiques (BEH'79),



(donada una cara, ha de ser possible determinar el conjunt de cares colindants, el conjunt d'arestes que la limiten i el conjunt dels seus vèrtex; donada una aresta, s'ha de poder conèixer les cares que uneix, el conjunt d'arestes que conflueixen als seus vèrtex extrems, i aquests mateixos; i, a partir d'un vèrtex, ha de ser possible determinar el conjunt de cares i arestes confluents, així com el conjunt de vèrtex contigus).

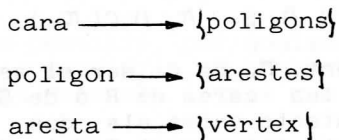
La representació "winged edge" (Nav'86), que està força extesa, emmagatzema les relacions,



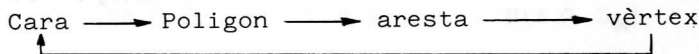
i és immediat veure que permet respondre qualsevol de les interrogacions. No obstant, i degut a que tots els processos de sortida requereixen les relacions (apuntadors) directes,

cara → poligon (frontera o forat) → aresta → punt ó vèrtex

Al sistema que es presenta es guarden, a part de les característiques geomètriques de cares i vèrtex, les relacions,



Per tal de poder respondre interrogacions en sentit invers al de les relacions emmagatzemades (per exemple, $V \rightarrow \{C\}$, ó $V \rightarrow \{A\}$), cal disposar d'una o més relacions inverses. Evidentment, és suficient guardar la relació $V \rightarrow \{C\}$, que produeix el tancament cíclic del graf de relacions:



No obstant, donat que les interrogacions inverses són poc freqüents i que l'algorisme de càlcul de la relació $V \rightarrow \{C\}$ és lineal, aquesta relació només es calcula explícitament en els processos de càlcul que la necessiten.

2.2 Operacions Bàsiques Sobre El Model

Suposem que, sobre el conjunt de sòlids delimitats per cares planes, definim les següents operacions:

- Complementari: el complementari T d'un sòlid S és el que es defineix a partir del complementari de l'expressió booleana entre semiespais plans que donava lloc al sòlid S: Si S quedava definit per $|1|$, llavors

$$T=C(S) \triangleq \neg \text{expr-bool}(\pi_1, \pi_2, \dots, \pi_n)$$

o, d'una altra forma,

$$T=C(S) = \{P/P \notin S\} \cup \{\text{frontera}(S)\} \quad |2|$$

- Enganxat: és una operació de reunió només definida en parelles de sòlids R, S tals que

$$R \cap S = \pi_k \cap C(\pi_k) \quad |3|$$

on, lògicament, π_k ha de ser el semiespai definit per alguna de les cares de R ó de S; aquesta cara és la única de contacte entre els dos sòlids. En aquest cas, per tant, l'enganxat T dels R i S és simplement,

$$T= E(R,S) = R \cup S$$

- Secció: Donat un sòlid S i un pla P_s , la secció de S per P_s no és més que el sòlid intersecció entre S i el semiespai π_s definit per P_s ,

$$T= \text{Seccio}(S, P_s) = S \cap \pi_s \quad |4|$$

A continuació es demostra que si un sistema de modelatge de sòlids delimitats per cares planes té definides aquestes tres operacions, pot realitzar qualsevol operació booleana entre sòlids. Per fer-ho, es començarà estudiant el cas de l'operació d'intersecció i després s'extindrà a les altres.

Proposició 1: Un sistema de modelatge de sòlids delimitats per cares planes [1] que pugui realitzar les operacions de complementari, enganxat i secció, permet calcular la intersecció entre qualsevol parella de sòlids representables.

a) Demostració en el cas de que un dels sòlids que s'operen sigui convex.

En aquest cas, si es vol calcular $T=R \cap S$ i S , per exemple, és convex, l'operació booleana que el defineix, [1] conté únicament operacions d'intersecció:

$$S = \pi_1 \cap \pi_2 \dots \cap \pi_n \quad |5|$$

llavors

$$T = R \cap S = R \cap \pi_1 \cap \pi_2 \dots \cap \pi_n$$

i la propietat associativa de la intersecció, implica que, T es pot obtenir per aplicació reiterada dels operadors de secció corresponents a $p_1 \dots p_n$, al cos R .

b) Demostració en el cas general, on ni R ni S són convexes.

Atès que sempre és possible, per qualsevol cos S , trobar un conjunt de cossos $S_1 \dots S_m$ convexes, disjunts excepte en parts de la frontera i tals que

$$S = S_1 \cup S_2 \cup \dots \cup S_m \quad |6|$$

(veure, per exemple Rog'85), es pot escriure,

$$T = R \cap S = R \cap (S_1 \cup \dots \cup S_m) \quad |7|$$

per la propietat distributiva,

$$T = (R \cap S_1) \cup (R \cap S_2) \cup \dots \cup (R \cap S_m) \quad |8|$$

ara bé, les interseccions $R \cap S_k$ es poden fer, amb el mètode indicat al cas a), en ser S_k convexes; i les unions entre $R \cap S_i$ i $R \cap S_j$ són, o bé enganxats o bé unions de cossos totalment disjunts, que no requereixen cap operació (a part d'incorporar les dues representacions, conjuntament, a la base de dades).

Proposició 2: En les mateixes hipòtesis de la proposició 1, es poden realitzar les operacions de reunió i diferència entre qualsevol parella de sòlids representables.

La demostració és immediata, ja que qualsevol d'aquestes operacions es pot posar en funció de les d'intersecció i complementació, les quals es poden realitzar per la proposició 1 i per hipòtesi:

$$A \cup B = C(C(A) \cap C(B))$$

$$A - B = A \cap C(B)$$

|9|

Corol·lari: Un sistema de modelatge de sòlids delimitats per cares planes i que admeti les operacions de complementari, enganxat i secció té el mateix domini (permet de generar els mateixos objectes) que un sistema CSG basat en primitives formades per semiespais plans.

Cal observar que, per a obtenir operacions booleans regularitzades (Req'80), únicament cal que l'operació de secció sigui també regularitzada:

$$\text{Secció}^*(S, P_s) = \text{Recubriment}(\text{Interior}(\text{Secció}(S, P_s)))$$

en canvi, en les operacions d'enganxat i complementari no cal cap condició especial (no poden donar lloc a objectes que calgui regularitzar).

A la pràctica, les possibilitats del sistema de modelatge que acabem de descriure són inclús més elevades: una gran quantitat d'objectes reals es poden generar realitzant un nombre reduït d'operacions de secció i enganxat escollides adequadament.

D'altra banda, si s'incorpora al sistema un mòdul generador de primitives suficientment potent, moltes de les transformacions que caldria efectuar mitjançant operacions booleans a un Sistema CSG, queden absorbides per aquest procés inicial. Si es disposa, per exemple, de funcions que permetin la generació del model de fronteres a partir d'escombrats de translació i rotació (Bru'85), es generen directament els objectes primitius habituals de la Geometria Constructiva de Sòlids (paral·lepíped, cilindre, con, esfera, torus), així com un bon nombre de sòlids més complexos (entre d'altres, els que apareixen als sistemes de 2 1/2 D).

A més a més, si s'admet la possibilitat d'escombrats múltiples -escombrats de polígons pertanyents a cares d'objectes ja generats-, s'obtenen automàticament sòlids del tipus.

$$E(S_1, S_2)$$

$$S_1 - S_2$$

en casos simples en que

$$\text{Frontera}(S_1) \cap \text{Frontera}(S_2) \subset \pi_{\kappa} \cap C(\pi_{\kappa}) \quad |10|$$

on π_{κ} és el semiespai definit per alguna de les cares de S_1 , sòlid inicial sobre el qual es realitza l'escombrat corresponent a S_2 .

El sistema que es presenta a l'apartat següent incorpora totes les característiques justificades en aquest apartat:

- Model de fronteres
- Aproximació per cares planes
- Operacions de enganxat, secció plana i complementari
- Sistema d'entrada d'objectes primitius, per escombrat múltiple de translació, rotació o conificació.

A més a més, l'escombrat múltiple de diferència no està limitat únicament a la condició expressada per |10|, sino que possibilita que S_2 talli plans P_i de S_1 paral·lels a P_{κ} ; d'aquesta forma es poden generar, ja en el procés inicial d'escombrat, sòlids amb perforacions passants.

L'adaptació del sistema DMI, inicialment desenvolupat sobre un mini-computador VAX, s'ha realitzat sobre un microcomputador PC compatible amb 640 Kb de memòria central i treballant sota sistema operatiu MS-DOS. Per a la sortida gràfica s'utilitza una placa gràfica MATROX PG640 amb 256 colors i paleta de 4096.

El sistema permet la generació de cossos en 3 dimensions basada en la tecnica de l'escombrat múltiple amb l'ajut d'un sistema d'edició de poligons en 2 dimensions. Els cossos, emmagatzemats segons model de fronteres, poden esser sotmesos a operacions i a transformacions geomètriques (translacions, rotacions,...), així com esser visualitzats segons diferents sistemes de representació que permeten, generar la sortida gràfica per diferents dispositius gràfics.

A la figura 1 es representa l'esquema dels diferents blocs que formen el sistema de disseny.

3.1 EDICIO DE COSSOS 3D

L'edició de cossos 3D es basa en la tècnica de l'escombrat d'un poligon 2D a l'espai. L'escombrat del poligon 2D es pot realitzar segons 3 formes diferents:

1. ESCOMBRAT PARAL.LEL: el pla d'escombrat del poligon es manté paral.lel al pla inicial del poligon.
2. ESCOMBRAT CONIC: El poligon varia homotèticament al llarg de l'escombrat.
3. ESCOMBRAT ROTACIONAL: el pla d'escombrat gira respecte d'un eix de rotació.

L'escombrat paral.lel dona com a resultat cossos prismàtics. El cònic genera cossos piramidals, i el rotacional cossos de revolució i toroidals. A la figura 2 apareixen uns exemples simples d'aquests escombrats.

Un cop generat un cos a partir d'un escombrat inicial, es pot modificar la seva forma afegint protuberancies i forats (passants o no) a partir de nous escombrats de poligons 2D definits sobre les cares del cos inicial. A la figura 3 s'han definit escombrats sobre les cares dels cossos de la figura 2.

Per a l'edició dels poligons 2D es disposa d'un editor 2D que

permet utilitzar dades dels cossos 3D ja generats. El sistema permet també realitzar l'escombrat de polígons 2D generats des de programes externs al propi sistema.

Malgrat el model utilitzat només permet emmagatzemar cares planes, a tota cara d'un cos que approximi una superfície no plana se l'hi assigna un flag que ho indica (en direm flag de cara cilíndrica). Això fa possible, a la fase de visualització, una representació molt més realista de l'objecte dissenyat.

3.2 CLASSIFICACIO I GESTIO DELS COSSOS.

El dissenyador pot treballar amb més d'un cos dins el seu disseny. Donada la limitació de memòria del microcomputador i per a que això no suposi un inconvenient important per a l'usuari, el sistema permet tenir cossos a memòria central i cossos a disc. Així, el volum d'informació de que es pot disposar en un disseny s'amplia a la capacitat del disc.

Els cossos que intervenen en un disseny es distribueixen en 3 conjunts:

1. COSSOS SELECCIONATS: residents en memòria, són aquells sobre els que es realitza tota operació del sistema (llevat de les de gestió, que es poden fer sobre qualsevol cos).
2. COSSOS ACTIVATS: residents en memòria, s'utilitzen com a referència de treball a les operacions que es fan sobre els seleccionats.
3. RESTA DE COSSOS: resideixen a disc i en tot moment es poden activar o seleccionar.

A la figura 4 es mostra el diagrama d'aquesta classificació dels cossos dins del disseny.

Per a la gestió dels cossos es disposa d'una sèrie de comandes com ara la selecció, activació, deselecció, desactivació, eliminació, duplicació, canvi de nom, etc.

Per a facilitar l'agrupació dels cossos dins del disseny, tot cos té assignat un nivell i un atribut mitjancament els quals es permet gestionar més d'un cos de forma conjunta. Així, es pot seleccionar un cos en concret ó seleccionar tots els del nivell "n", o seleccionar tots aquells que tenen com a atribut el "ser de fusta".

3.3 VISUALITZACIO.

La visualització dels objectes definits en un disseny es pot realitzar segons quatre sistemes :

1. WIRE-FRAME o MODEL DE FILFERROS: es dibuixen totes les arestes del cos.
2. ELIMINACIO DE LINIES OCULTES: només es dibuixen els trams d'aresta visibles. A més, les arestes de contacte entre cares amb flag de cara cilíndrica no es dibuixen.
3. SIMULACIO D'ILUMINACIO (FACETAT): dibuixa cada cara amb una intensitat de color corresponent a l'il.luminació que l'hi pertoca.
4. SIMULACIO D'ILUMINACIO AMB SUAVITZAT: afegeix al sistema anterior el suavitzat de les cares amb flag de cara cilíndrica, aconseguint millor sensació de realisme.

A la figura 5 apareix la visualització, en aquests quatre sistemes, d'un mateix disseny.

El sistema d'eliminació de línies ocultes és del tipus aresta-cara (Bru'85)

Com a sistema amb il.luminació per facetat s'ha implementat un sistema basat en la partició binaria de l'espai (BSP, Fuchs'83) que genera un arbre de prioritats amb totes les cares de l'escena. L'avantatge d'aquest sistema està en que la generació de l'arbre és totalment independent dels angles de visualització i de la posició del focus de llum. D'aquesta manera, un cop generat l'arbre, davant qualsevol modificació d'un d'aquests paràmetres el temps de generació de la nova imatge es redueix al temps de redibuixat.

El quart sistema de visualització segueix un algorisme Z-buffer amb suavitzat de Gouraud de les cares amb flag de cara cilíndrica.

La visualització del disseny es pot realitzar en planta, perfil, alçat o perspectiva o amb tots quatre a l'hora.

Hi ha una sèrie de paràmetres de visualització que es poden modificar: posició de l'observador i focus de llum, zoom, pan, etc. El primer permet, si l'observador s'acosta prou a un objecte, veure'l amb perspectiva cònica i fins i tot penetrar al seu interior.

Tot cos té un color que es pot escollir entre 15 de possibles (modificables per l'usuari). Així mateix, es pot assignar color d'una cara independentment del cos al que pertany.

3.4 TRANSFORMACIONS GEOMETRIQUES

Les transformacions geomètriques que es poden realitzar sobre els objectes inclosos en un disseny són la translació, la rotació i l'escalat.

La translació es pot realitzar segons un vector qualsevol a l'espai. La rotació es realitza al voltant d'un eix també qualsevol a l'espai. Per últim, l'escalat es pot realitzar de forma global o bé independentment per als tres eixos coordenats.

3.5 OPERACIONS SOBRE EL MODEL

Tot seguit es descriuran les operacions que el sistema permet fer sobre el model. Es a dir, aquelles operacions que modifiquen el contingut del model de fronteres sobre el que es treballa.

Aquestes operacions són :

1. SECCIO: Donat un pla qualsevol a l'espai, es secciona el cos, donant com a resultat dos nous objectes.
2. ENGANXAT: Donats dos cossos amb una cara en contacte, es realitza la seva unió a través d'aquesta cara.
3. COMPLEMENTARI: Donat un cos, s'envolta d'un bloc sòlid, quedant el volum definit pel cos com a concavitat del bloc. Es realitza, per tant, el motllo de l'objecte.

Així doncs, i com a conseqüència del que s'ha exposat a l'apartat 1, aquest sistema de modelatge, basat en sòlids delimitats per cares planes, té el mateix domini que pot tenir un sistema CSG basat en primitives formades per semiespais plans.

A la figura 6 apareixen uns exemples d'operacions sobre cossos. A la part superior de la figura hi ha els tres cossos originals, i a la part inferior el resultat de realitzar un enganxat, una secció i un complementari (motllo),

respectivament.

4 INTERFACE AMB L'USUARI.

Gran part de l'èxit d'un sistema, no radica només en la seva potència alhora de realitzar operacions, sino també en la facilitat de que diposa l'usuari per tal de realitzar les mateixes. Cal per tant tenir cura en el disseny del interface amb l'usuari per tal de fer un sistema utilitzable.

L'interface amb l'usuari es pot dividir d'una manera natural en 4 components (New'79). La primera i la que sosté totes les demes és el "Model d'usuari". El model d'usuari es el model conceptual de manegament del sistema, sense el qual es podria controlar al mateix i per tant no podriem realitzar cap operació.

Una vegada definit el model de usuari, el sistema necessita comandes per tal de manipular-lo. Hem de donar el que s'anomena "Llenguatge de comandes" que forma el segon component del interface d'usuari.

El tercer component del interface d'usuari és la "realimentació", que és la part del sistema amb el qual s'ajuda a l'usuari. La realimentació pot venir de moltes maneres : missatges d'explicació, indicació d'objectes seleccionats, ...

El quart component és "l'Àrea d'informació", que és necessari per tal de mostrar l'estat de la informació que està manipulant.

4.1 EL MODEL D'USUARI - EL LLENGUATGE DE COMANDES.

El model de sistema proposat esta basat en una estructura que funciona normalment com a cua FIFO. Quan es demana una comanda les accions a realitzar són introduïdes dins la cua i es van executant succesivament sota el control d'un programa principal rector de tot el sistema. El conjunt d'accions que realitzaran les comandes són carregades a la inicialització del sistema i són definibles pel propi usuari.

La estructura de la comana és la següent :

aabbcc xxxNom Accions d'execució

aabbc : és un número identificador de comanda que permet donar una estructura jeràrquica de tres nivells a les mateixes.

aa : és l'identificador de grup.
bb : és l'identificador de comanda.
cc : és l'identificador de paràmetre de comanda.

xxx : Són els qualificadors de comanda i que permeten tenir un control de la mateixa. Són :

| : Comanda de defecte.
\$: Comanda executable en qualsevol moment.
? : Comanda que no permet executar una altra mentre ella s'esta executant.
% : Comanda no directament accessible per l'usuari.
¿ : Comanda no executable : títol de grup.

Només 3 d'aquest 5 paràmetres tenen sentit ahora.

Nom : Es el nom de la comanda.

Accions d'execució : Són les accions que realment s'executaran durant la comanda "Nom". Les accions principals són :

- Escriptura de textos, per tal d'indicar a l'usuari el que es va a fer.

- Utilitats: que serveixen per subministrar les dades al sistema. Les utilitats bàsiques considerades seràn :

- Real.
- Enter.
- String.
- Punt a pantalla.
- Selecció d'un punt 3D.
- Selecció d'una aresta.
- Redibuixat.

Aquestes utilitats deixen la informació en uns stack, informació que haurà de ser recollida per les rutines.

- Crides a rutines que són les que faràn realment les operacions sol·licitades a partir de les dades que han deixat al stack les utilitats.

- Sentències de Control de flux : Es donen també dos tipus de sentències de control de flux dintre una comanda, que permeten realitzar estructures alternatives i repetitives.

- Crides a altres comandes : Permet per tant enllacar comandes i donar més potència al sistema.

Uns exemples de comandes serien :

```
010100 $ZFINESTRA "Entra la finestra " |XYP| |XYP| (ZFIN)
030300 SECCIO      [PLA3D] (SECCIO) |REDIBUIX|
070400 PLA3D
070401 |PLA3P     [PUNT3D] [PUNT3D] [PUNT3D] (PLA1)
070100 PUNT3D
070101 |VERTEX   |selpunt| (PUNT1)
070102  SEGTPC   "1 punt" |selpunt| "2 punt" |selpunt|
          "Dist en %" |real| (PUNT2)
```

La comanda "ZFINESTRA" fa un zoom a partir d'una finestra entrada com dos punts a pantalla. Aquest punts seràn depositats al stack de punts per la utilitat |XYP| d'entrar punts a pantalla i seràn recollits per la rutina (ZFIN) que farà el zoom.

La comanda "SECCIO" fa la secció per un pla determinat per tres punts. Ara bé podem veure que existeixen dos alternatives per la selecció d'un punt: la opció "VERTEX" (opció de defecte al portar el qualificador de comanda "|") que deixa al stack de punts un punt seleccionat com a vertex d'un cos mitjançant la utilitat |SELPUNT| mentre que la opció "SEGTPC" que deixa al mateix stack un punt seleccionat com un percentatge entre dos vertex del cos.

Al executar una comanda, el sistema carrega les accions en la cua i les va executant una a una. Quan troba que l'acció a executar és una crida a una altra comanda (en el cas de la SECCIO [PLA3D]), inserta enlloc d'aquesta les seves accions i segueix l'execució. Si la comanda és directament executable o no té cap opció (com és el cas de "PLA3D") no cal fer res més ara bé si la comanda a executar té varies alternatives possibles (com és el "PUNT3D") es pren la de defecte.

El sistema farà lloc a la cua introduint les accions d'aquesta nova comanda dins de la mateixa, bé davant de l'acció que anava a executar, bé substituint alguna de les accions ja

existents per la nova, en el cas que la nova comanda fos una alternativa d'una de les existents en la cua. Així doncs mentre s'esta executant la comanda "SECCIO" i el sistema ens demana qualsevol dels punts 3D mitjançant la comanda de defecte "VERTEX", l'usuari pot reemplaçar aquesta per la comanda "SEGTPC". De la mateixa manera també pot demanar la comanda "ZFINESTRA" que encara que no és germana de cap de les comandes existents a la cua porta el qualificador "\$" que ens permetra fer el zoom i continuar amb la secció tal com estavem abans.

Així doncs, és possible sempre que es desitji executar qualsevol comanda en qualsevol moment, sempre sota el control de l'usuari ja que la totalitat de la comanda i per tant també els seus qualificadors són definibles per l'usuari a través d'un arxiu editable. A més a més es dona la possibilitat a l'usuari, ja no tant sols de canviar les comandes ja definides, sinó també crear-ne de noves a partir de les ja existents.

4.2 LA REALIMENTACIO I L'AREA D'INFORMACIO.

L'usuari ha de estar en qualsevol moment guiat pel sistema i tenir la possibilitat de demanar en qualsevol moment una ajuda o explicació de les accions que està fent. Per tal de donar una major claretat al sistema i unes màximes possibilitats de decisió s'ha adoptat la distribució de la pantalla en 6 finestres clarament diferenciades (SPLI'84):

- Finestra gràfica : En ella es realitza tota la interacció gràfica entre usuari i sistema ja sia d'entrada com de sortida. La entrada en aquesta àrea es realitza sempre a través del cursor gràfic però en qualsevol moment es disposa de la possibilitat de accedir a una entrada mitjançant el teclat o bé accedir a les àrees anomenades de Menús o d'Història (mitjançant una tecla previament definida) per tal d'accedir a una nova comanda.

- Finestra de Menús : En ella es disposa dels menús del sistema que serveixen tant d'ajuda com de mètode de selecció de comandes. L'accès a aquesta àrea no està restringit pel tipus de comanda que esta executant, sinó que l'usuari pot accedir-hi en qualsevol moment. Una vegada dintre pot moure's lliurement per tota l'estructura de comandes del sistema fins escollir la comanda desitjada i poder-la executar.

En la figura 7 veiem la disposició del menú de la comanda ZTPC, entrada mitjançant el teclat a l'àrea d'entrada. La comanda ZTPC podria tenir la següent estructura jeràrquica :

```
010000 VISUALITZACIO
010600     ZOOM
010601         ZTOTS
010602         ZTPC
010603         ESCALA
010604         TOTDIS
010605         ZFINESTRA
```

Com podem veure en ella apareixen el grup de comandes a que pertany ZTPC que és visualització la comanda que té el mateix identificador de comanda que és ZOOM, a més a més de tots els paràmetres de comanda de ZOOM, com són ZTOTS,ZTPC,ESCALA,TOTDIS i ZFINESTRA.

- Finestra d'Història : En ella es té reflectida tota la història de la comanda o comandes que s'estan executant. Anomenarem història d'una comanda a tota la sèrie d'altres comandes que han anat apareixent durant l'execució de les accions pertinents. Així doncs la història de la comanda "SECCIO" anteriorment executada seria la que veiem a la figura 8.

Ara bé, l'àrea d'històrics no té com a única finalitat el guiar a l'usuari, sinó que permet al mateix el canvi de les opcions que s'estan executant per qualsevol opció de les disponibles. Com havíem vist l'usuari disposa durant l'execució d'una comanda amb diverses opcions, de canviar l'opció de defecte per qualsevol de les altres existents, ja sia donant la mateixa mitjançant el teclat o bé accedint a l'àrea d'històrics, aon pot escollir entre qualsevol de les opcions disponibles en aquell moment que aniran apareixent a mesura que ho sol·liciti. Aquest procés de canvi dins l'àrea d'històrics el podem veure a la figura 9 en que canviem la opció de defecte VERTEX per l'altra opció existent com és SEGTPC.

- Finestra d'Status : En ella es reflecteix l'status actual del sistema es a dir el tipus d'informació que està esperant (comanda, enter, real, string, punt a pantalla,...) a més a més altres informacions de caire general com la data, l'hora, el disk de defecte, les coordenades actuals del cursor, l'existència de journal o no, ...

En les figures 7 i 8 podem veure dos estats del sistema; en la primera el sistema espera una comanda, en el segon espera una selecció d'un vertex.

- Finestra d'entrada : En ella es realitza tota entrada mitjancant el teclat per part de l'usuari. La longitud de la comanda no vé limitada per l'espai físic de la finestra sinó que el sistema realitza un scroll automàtic de la mateixa per tal de donar la possibilitat d'enllacar comandes.

- Finestra de sortida : En ella apareixen totes les respostes i tots els missatges que el sistema dona a l'usuari. Aquests missatges poden ser tant missatges d'error com missatges orientatius o d'ajut. El missatges d'ajut poden ser demanats en qualsevol moment mitjancant la selecció de la opció desitjada a la finestra de menús o d'històrics i demanant el Help de la mateixa.

La disposició d'aquestes finestres no es fixa ni en la seva posició dins la pantalla ni en la grandària de les mateixes. Aquestes magnituds són definibles per l'usuari ja sia via arxiu de definicions inicials o via comanda del sistema. En la figura 10 veiem una alternativa de finestres a la presentada a les figures 7, 8 i 9. Aquesta possibilitat de modificar la grandària de les finestres introdueix la necessitat de la existència d'un scroll de qualsevol de les finestres, scroll que pot ser realitzat en qualsevol moment, en qualsevol de les direccions.

A més a més de la possibilitat de modificar la distribució i grandària de les finestres, existeix també la possibilitat d'eliminar alguna de les finestres, permetent per tant d'augmentar la mida de la finestra gràfica si l'usuari creu que el seu coneixement del sistema es prou important per prescindir d'alguna de les finestres anteriorment citades.

5 REFERENCIAS.

BRU'85. P. Brunet, "DMI: A solid modelling system based on boundary representation and octtree encoding" in spanish, Raport DMIO3/85, ETSEIB, UPC(1985).

NAB'86. I. Navazo, D. Ayala and P. Brunet, "A Geometric Modeller Based on the Exact Octtree Representation of Polyhedra" North Holland, Computer Graphics Forum 5 (1986) 91-104.

BEH'79 Baer, A., Eastman, C. and Henrion, M., "Geometric Modelling: A Survey". Computer-Aided Design, Vol.11 Number 5, 1979.

NAV'86 I. Navazo, "Geometric Modelling of Octtree encoded polyhedral objects", PhD Thesis, Dept de Metodes Informatics, Polithecnical University of Catatlonia, Spain(1985).

REQ'80 Requicha, A.G. "Representations for Rigid Solids", Computing Surveys of the ACM Vol 12 Number 4, December 1980.

FUCHS'83 Fuchs, H., Abram, G.D., and Grant, E.D., "Near Real-Time Shaded Display of Rigid Objects", ACM Computer Graphics, Vol 17, Number 3, July 1983.

NEW'79 Newman, W. and Sproul "Principles of interactive Computer Graphics", 2nd Edition, McGraw-Hill New-York, 1979.

SPLI'84. Monrad Spliid, A. and Sorgen, A. "User Interfase : Concepts and Specifications", Proc. of Eurographics'84, Elsevier Science Pubi (North Holland Publ. Comp.), 1984

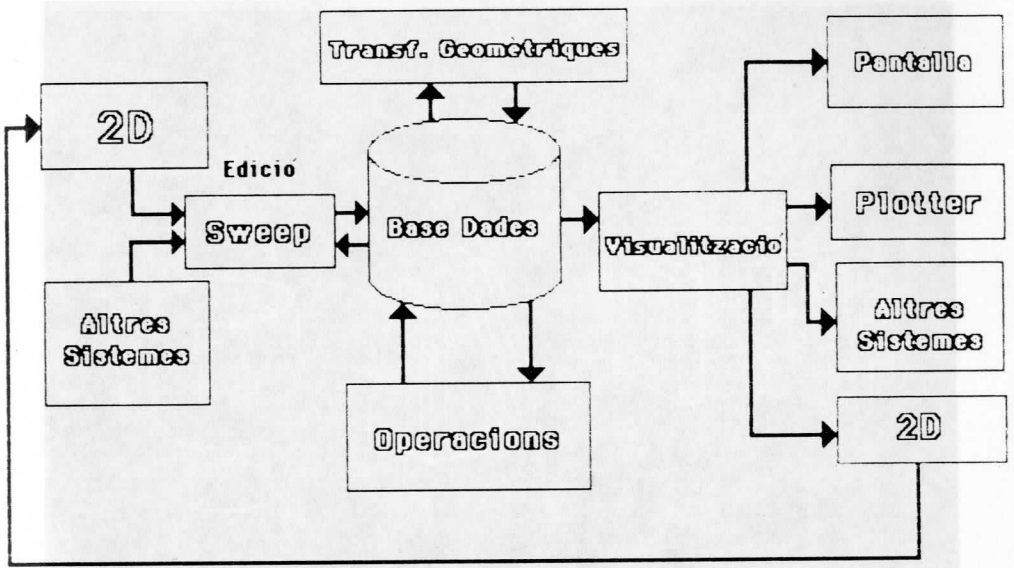


Figura 1. Esquema de blocs del Sistema.

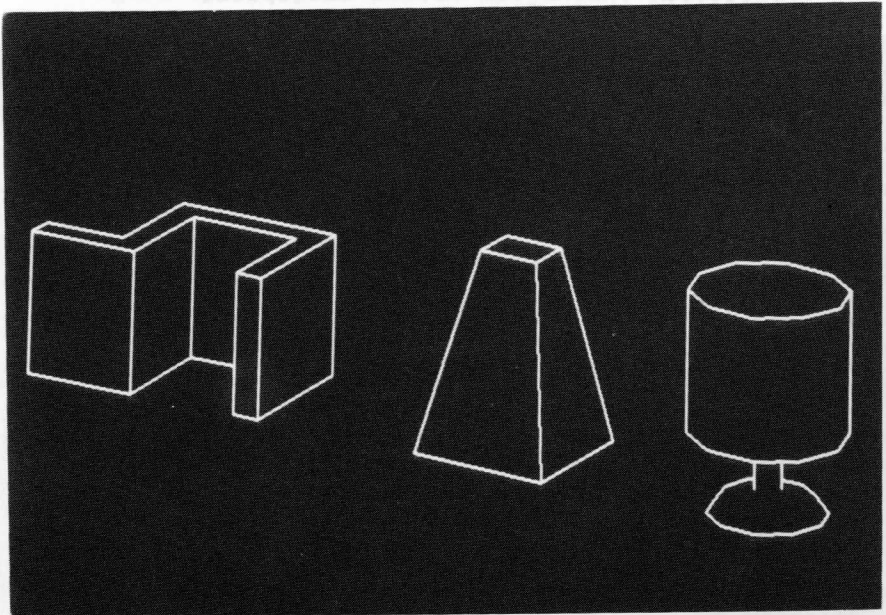


Figura 2. Escombrats simples.

Figura 5.2. Representació mitjançant línies ocultes

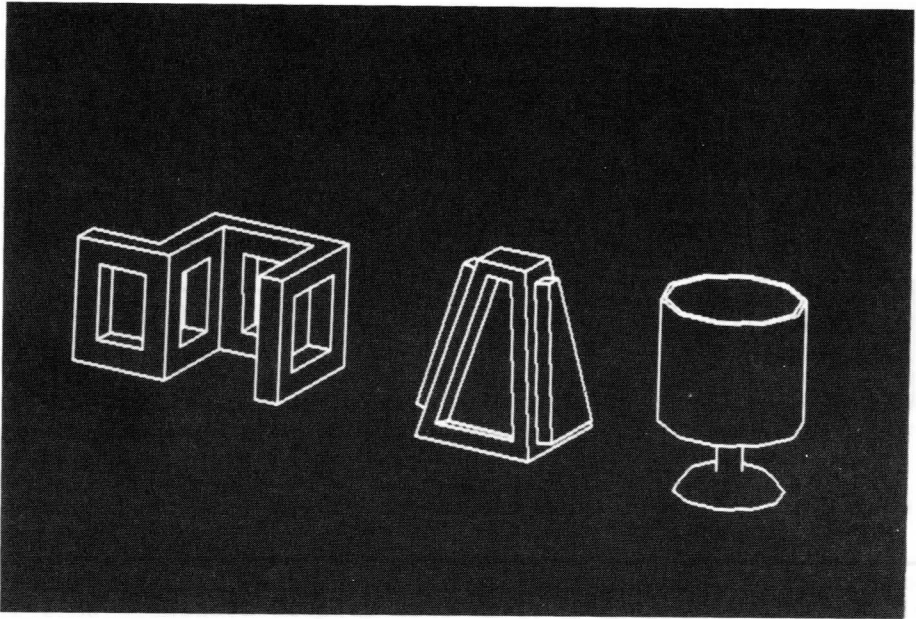


Figura 3. Escombrats multiples.

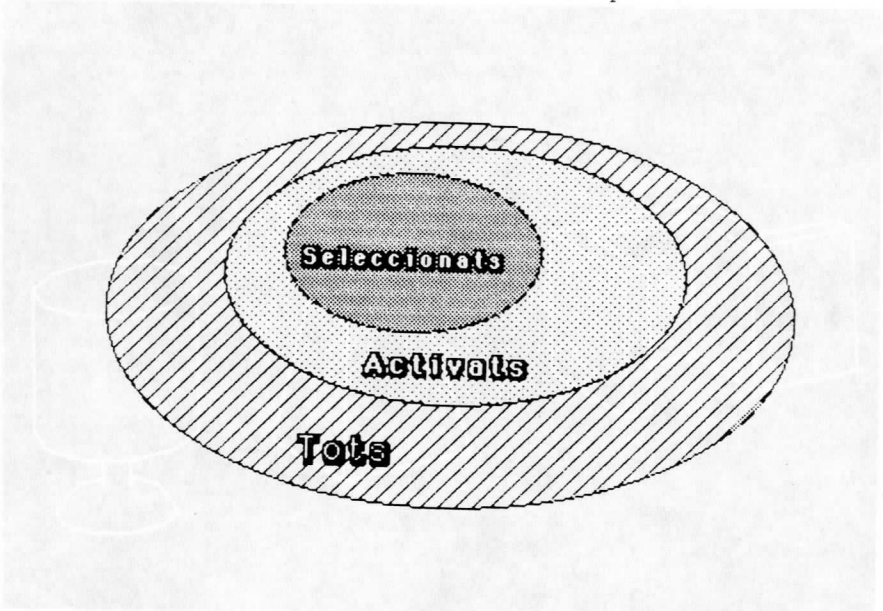


Figura 4. Classificacio del cossos.

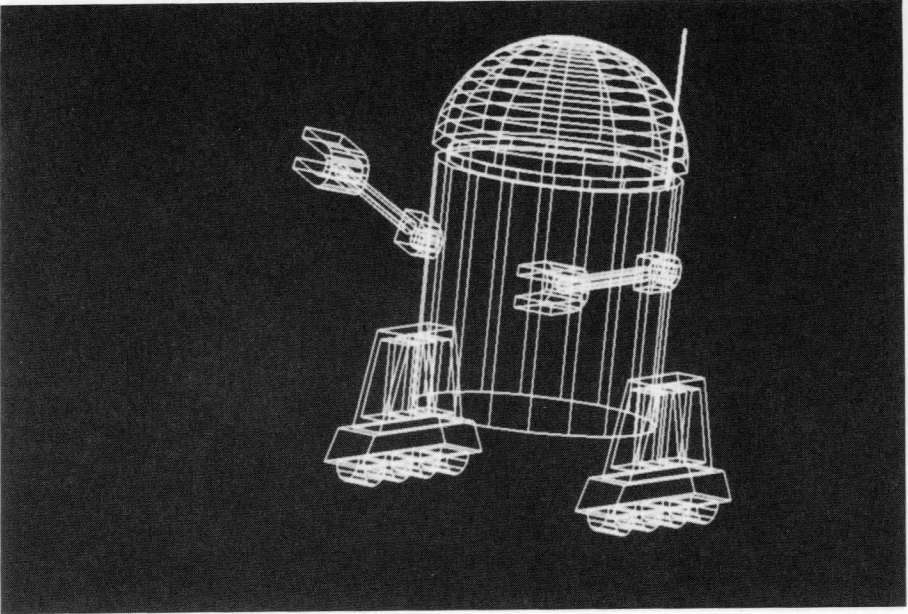


Figura 5.1. Representacio mitjancant Filferros.

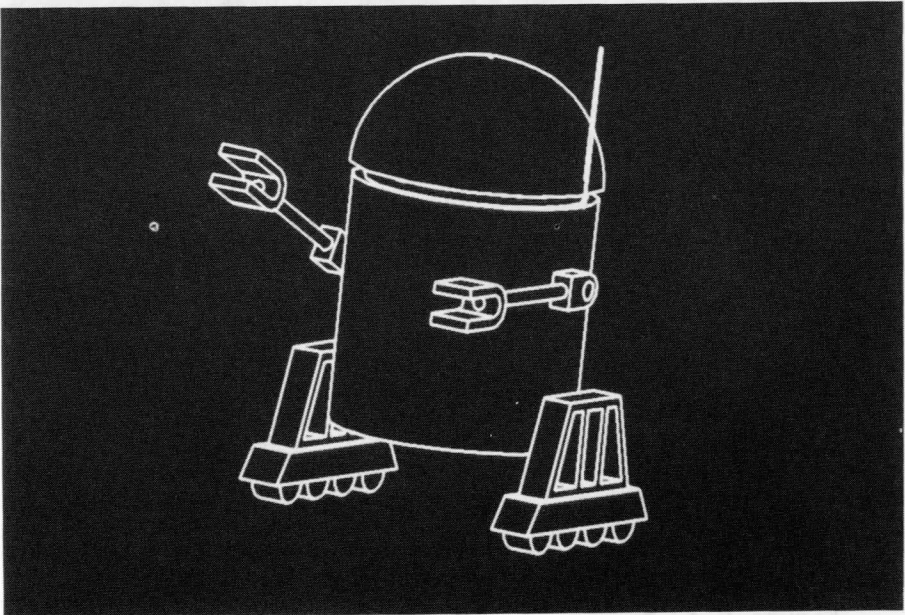


Figura 5.2. Representacio mitjancant Linies ocultes

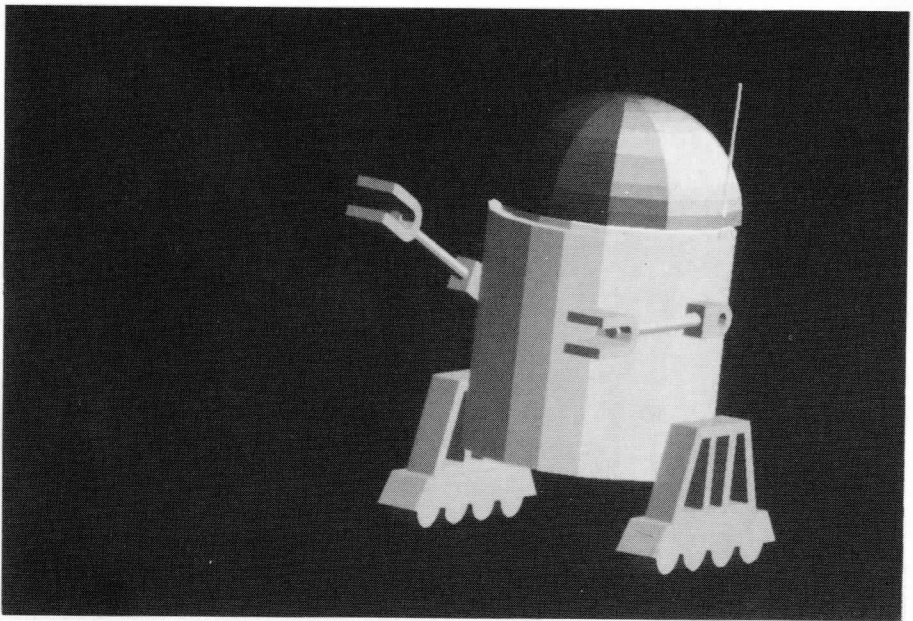


Figura 5.3. Representacio mitjancant Facetat.

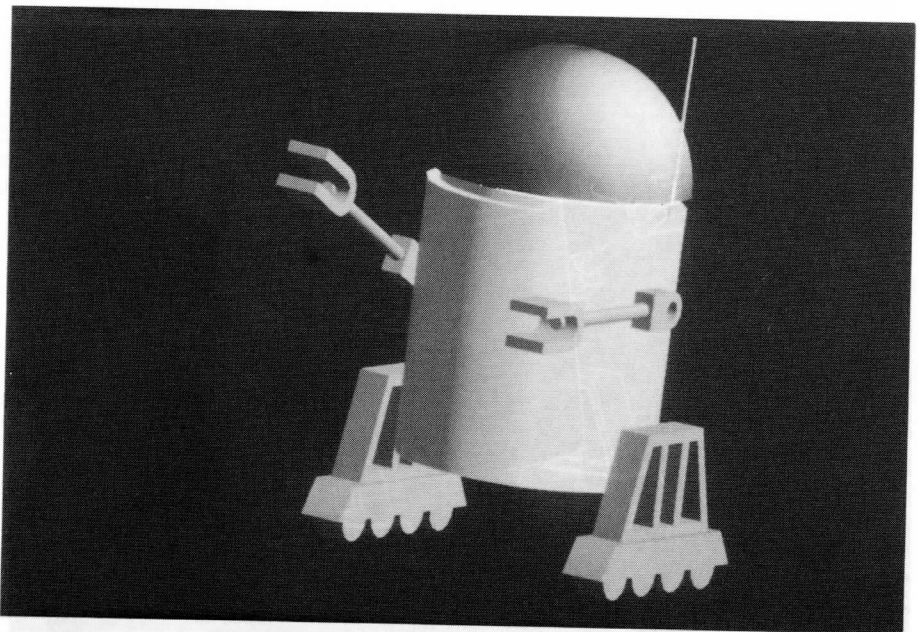


Figura 5.4. Representacio mitjancant Suavitzat.

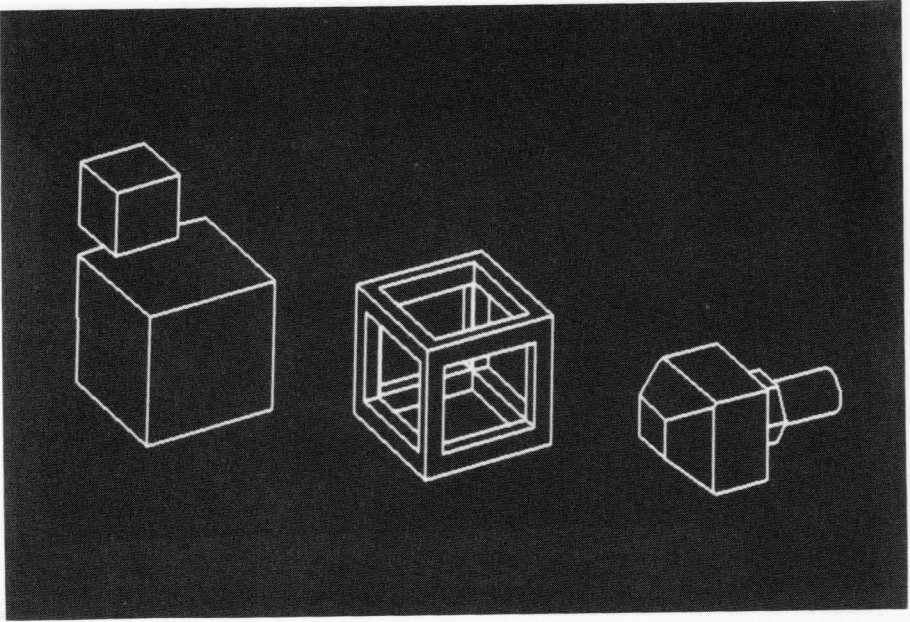


Figura 6.1. Tres cossos generats per escombrat multiple.

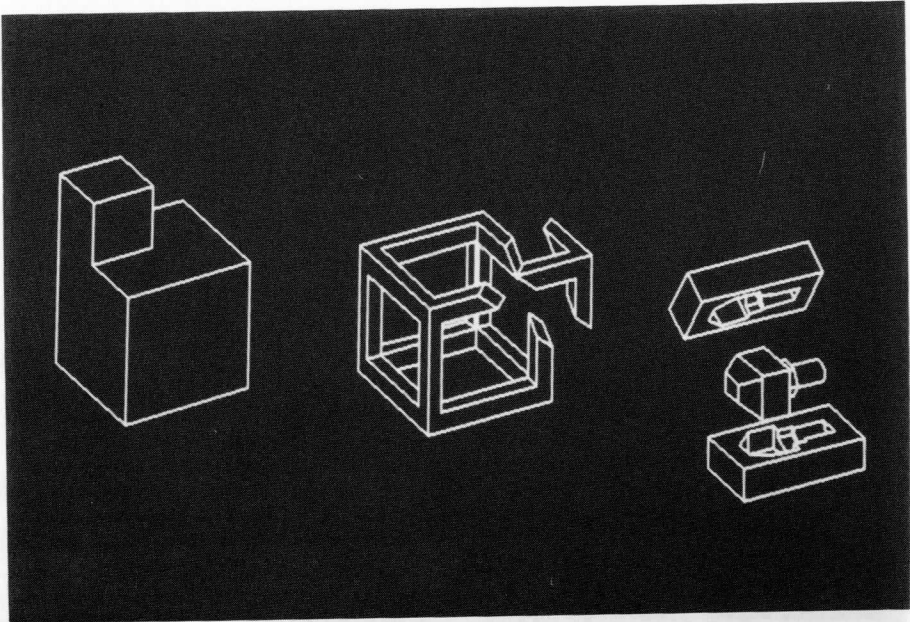


Figura 6.2. Enganxat, Seccio i Motllo dels cossos de 6.1.

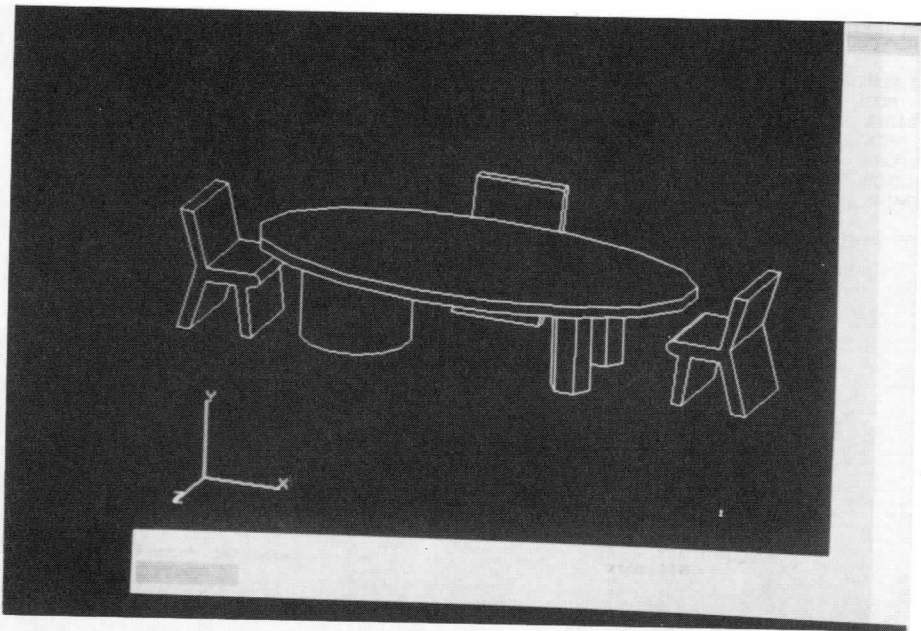


Figura 7. Disposicio de menus a la finestra de Menus.

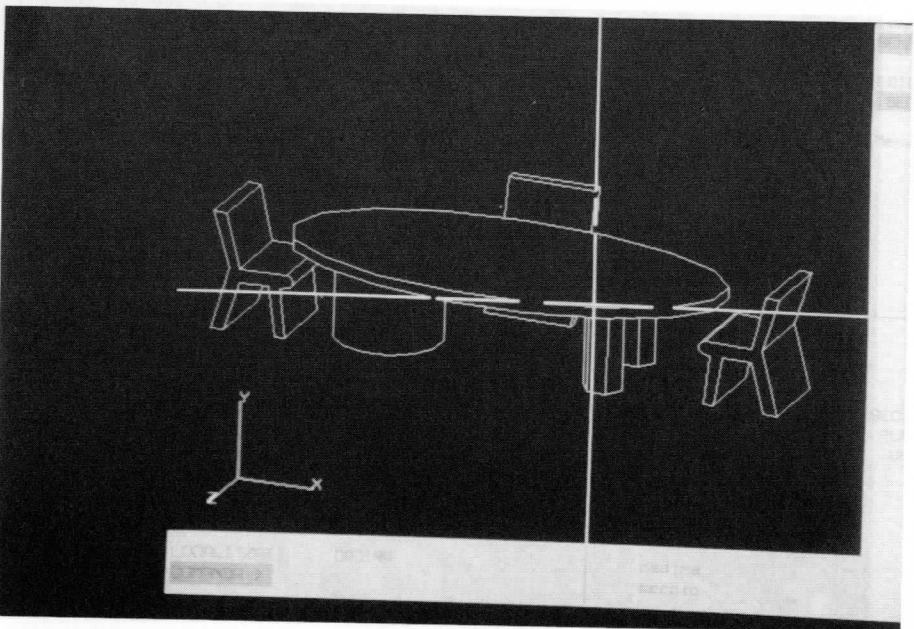


Figura 8. Disposicio de menus de la finestra d'Historics.

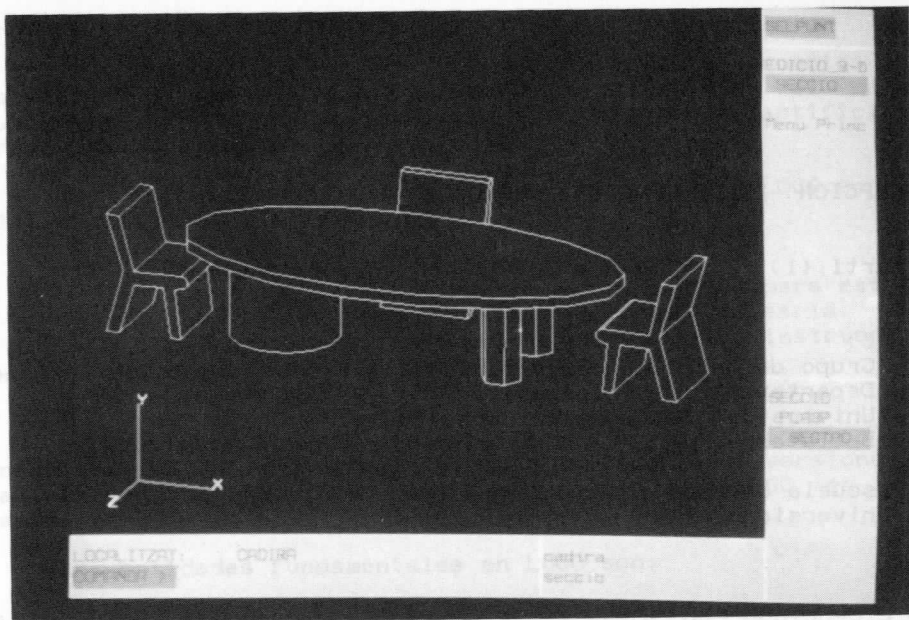


Figura 9. Canvi d'opció a la finestra d'Historics.

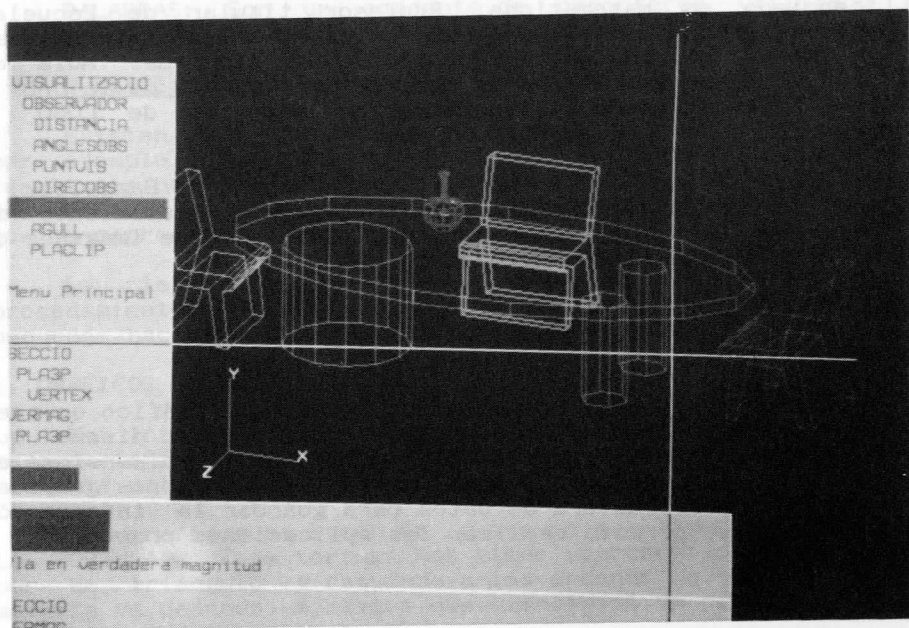


Figura 10. Altra disposicio de les finestres.