TRANSFORMATION PIPES:
A model to manipulate data, using graphical interaction, based on "transformation pipes, between the graphical devices and a data base.

Marius FREGIER
Architect, Computer scientist

LABORATOIRE GAMSAU (Groupe d'etude pour l'application des methodes scientifiques a l'architecture et a l'urbanisme

Ministere de l'urbanisme et du logement. Unite pedagogique d'architecture de Luminy, 70 Route Leon Lachamp,
13288 Marseille CEDEX 9

ABSTRACT:

Conditions of the manipulation of data, using a high level of graphical interaction within a data base in computer_aided architectural design, are examined .

A so called "TRANSFORMATION PIPES" model (data stucture and processes) is described, its goal being to carry out the communications from the graphical devices to the data base and vice versa.

This model is built upon the virtual device concept of CORE and GKS standards, and is constructed on the principle of a permanent minimal representation of the project item morphology which is associated with their abstract representation (description in terms of class, components, relations and properties)

KEE WORDS

Abstract data stucture._ Architecture._ Building industry. CAD._ Computer graphics._ Data manipulation._ Graphics standards,_ High level graphical interaction._ High level input primitives._ Morphological description

CONDITION OF THE MANIPULATION OF DATA USING HIGH LEVEL GRAPHICAL INPUT PRIMITIVES

An architectural computer aided design system must be able to give help in two decisive tasks of the architectural process:

_The manipulation of graphical information with the aim of design

_The production of extra information which is partly obtained by using the graphical information.

That means there are in the 3D reality of an object, several possible interpretations of ascene showned in the sketch of a certain plane representation.

One can object [3] that this 3D reality could be infered from the three view projection of the object. That is correct, but does not take consideration to three major facts of the real conditions of the use of drafting with the aim of design in the architectural practice.

The main part of the architectural design process is done in the "horizontal view plane". This introduce an unbalance of information to the detriment of the other views which are generally used exclusively to reduce certain problems of diachrony in the vertical dimension (EX: the passage of the head in the "stair well" [1].

The first stages of the architectural design are carried out by using topological relations (proximity, separation, etc...). These kind of relation is capital during the whole first stage of the design process which is eminently "infralogical" [1].

The coherence rules between views, proper to industrial drafting, cannot be applied to architectural drafting, because of the symbolical nature of most of the object representations

In other words we can not speak of a bi_univocal correspondence between views, at the most, we can simply say that there is a correspondence in the localisation of the different symbolical representations of a single object in different views.

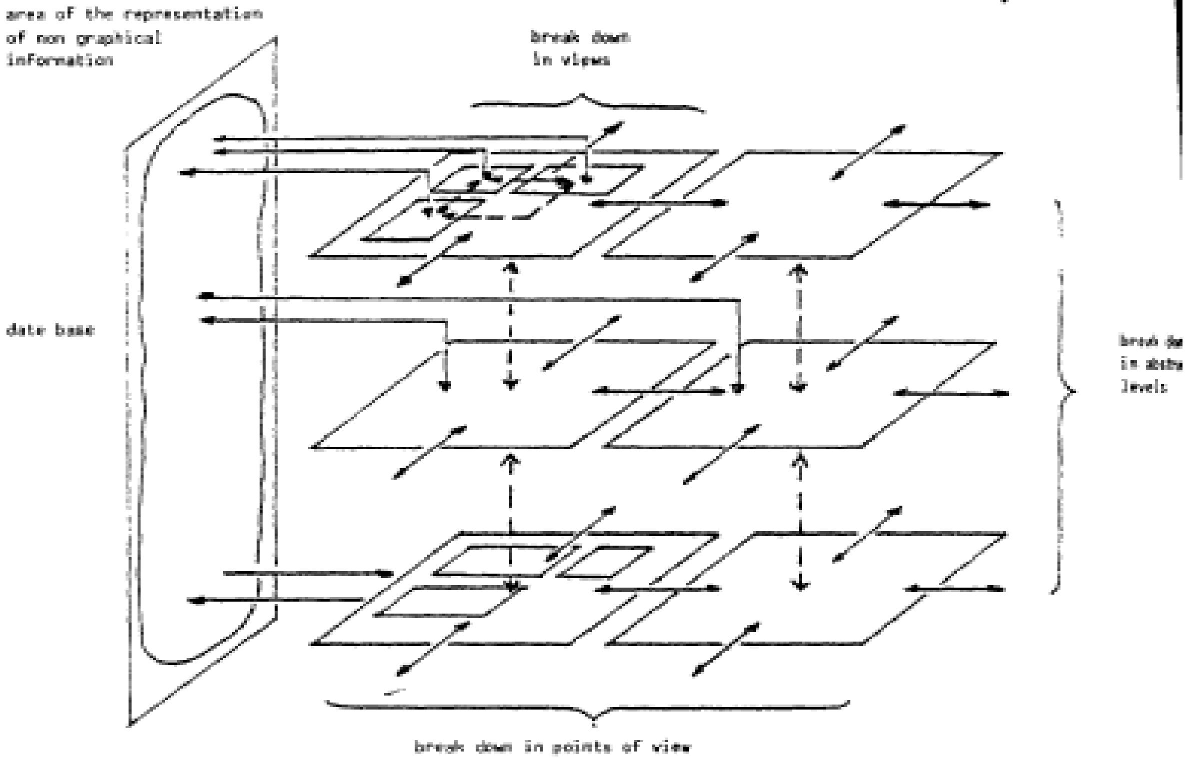
3 / Because of the complexity of the object that we are going to design, when we represent the object we need to break it down into different parts, each corresponding to different points of view.

4 / The architectural design process, like most design processes, depends on the breaking down of the project in levels of abstraction.

Close consideration of the four points descibe above is necessary to obtain the "traceability" as shown in figure 1

[3] CIMA, Programme Modul pour la saisie d'objets 3D par les trois vues de projection

FIG 1 : Traceability on the break down in terms of abstract levels, points of view and views



To be carried out, the four points listed above need a commun fundamental function. This function will manage the coherence between the data describing the information contained in the different representations used to manipulate the abstact representation

This method consist in the extension of the notion of "traceability", as it has be reported in [4] :
the "correspondence between levels of abstraction" (point number 4), is extended to the correspondence between different views (point number 2), between different points of views (point number 3), and also between graphical information and non_graphical information (point number 1) .

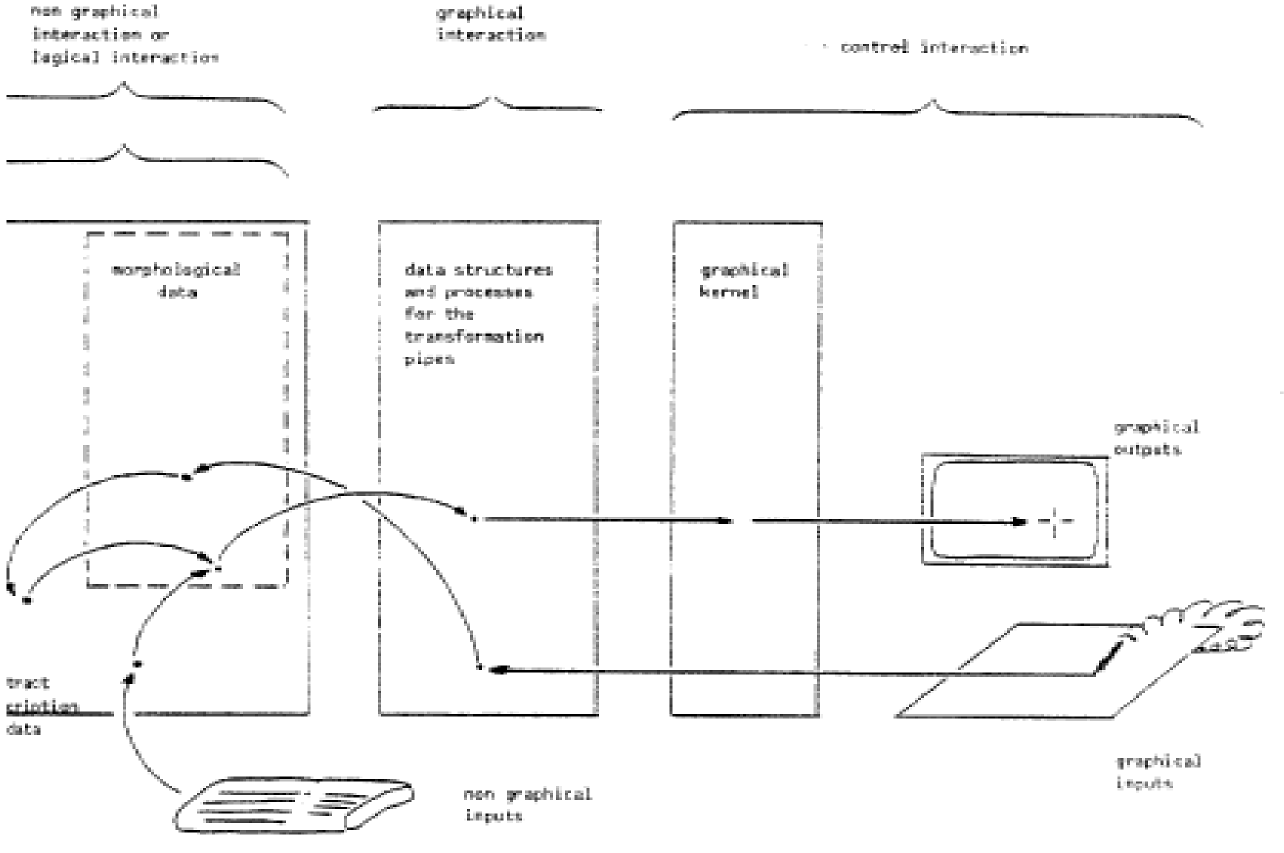


FIG.2 The three levels of interaction

GRAPHICAL REPRESENTATION AND NON_GRAPHICAL REPRESENTATION

In the proposed model graphical information no longer has the status of principal and permanent information, but it is the data necessary to the graphical representation (see figure 2).

These data are considered as a particular stucture of the information contained in the data base and are deductible from this base, and more specifically from that part of the data base which concerns the information describing the morphologie of the project items .

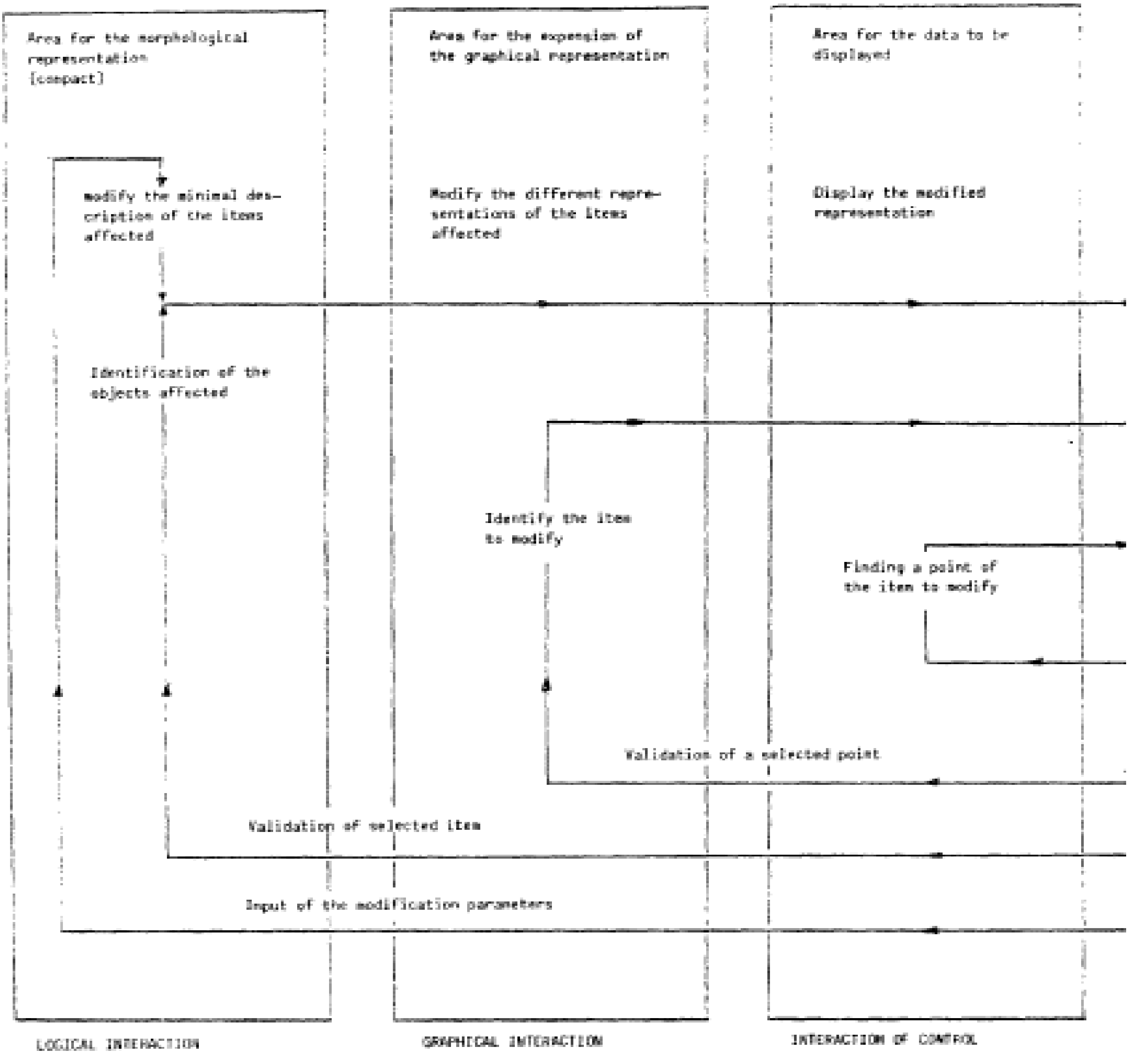


FIG 3. Organigramme of graphical query levels
Example : Modification of an item of the D.B.

The actions, (input, output and manipulation) carried out on the data base by means of using graphical interactive communication, are executed by a specific processor (the graphical interactive communication processor) which acts on the data base in the same way as any other processor (see figure 2 and 3).

THE MODEL

The aim of the model (the data structures and their processes) is to carry out the functions of interactive communication between a data base and a standard graphical kernel like CORE or GKS as shown in figure 2.

In this model we distinguish three levels of interaction:

-The control interaction (the closest to the hardware) consists in checking the echo function within the graphical kernel.

-The graphical interaction operating on the area of graphical representation is used to check the results of the search for graphical variables manipulated during the input or modification of the abstract data contained in the data base.

-The non_graphical interaction or logical interaction operates on the area of abstract description. It consists in checking, at the level of minimal representation contained in the data base, modifications produced by a transformation process, whether this be done with the help of a graphical input or not.

The checking is done by observing, as an output on a graphical device, the graphical representation which is obtained from the state of minimal representation, after its modification by any means graphical or not.

Below we shall discuss how they function.

THE ROLE OF THE DATA BASE

We will not expose the features concerning the management of an architectural data base, this being the subject of a specific research [5] parallel to ther present study within the gamsau laboratory.

However, we will recal the role of a data base within CADCAM systems [4,6], which we can sumarise as follows:

_To ensure that the treatment applied to the data and the data are independent of each other.

_To have a tool which insures that the description, the modification and the interrogation of a data file, which describes the objects processed in design or their associated knowledge, are coherent.

THE SPECIFIC ROLE OF THE GRAPHICAL KERNEL

The graphical kernel is a software tool (data structure and associated programs) of which the function is to assure the coherence of format between the different input output graphical devices on the one hand, and between the different application programs [6,9], on the other hand.

The transformation of data, from the normalised format of the graphical kernel, to the format known by the graphical devices, is carried out by so called "device drivers".

The aim of such a structure is to ensure for a large number of application programs their versatility in matter of graphical devices, without needing to rewrite for each new application program, a new interface for each graphical device (the figure 5, due to W.M. Newman and A. Van Dam [7], shows very well these features).

Thus, to use of a new graphical device, we need to write only one interface for all the present and future application programs.

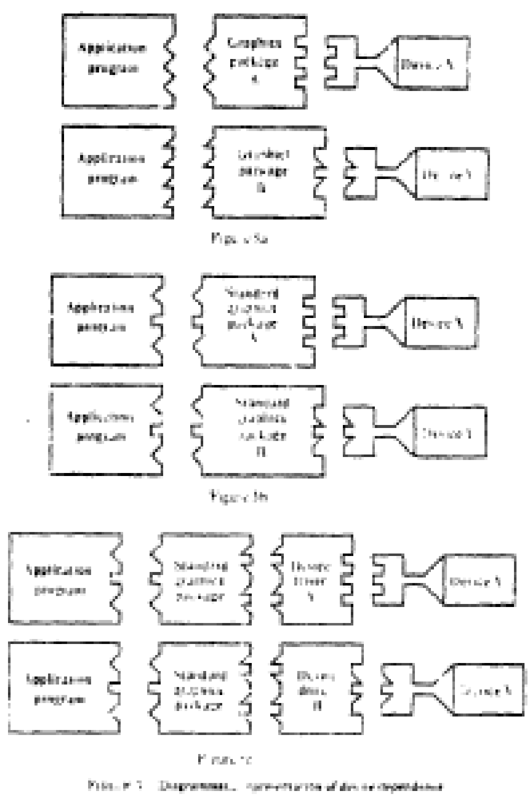
The user sees this structure as a virtual screen in which is application program can write and read, in the normalised format of the graphical kernel, the information wich is to be handled or to displayed, on the different graphical devices "activated" during the program execution.

Between the two well known propositions of graphical standards (CORE for the ACM SIGGRAPH Graphical Standard Commitee "GSPC" [9], and GKS "Graphical Kernel System" for the working group ISO/TC97/WH2 [10]), the GAMSAU recently chose GKS for the following reasons:

_GKS being a 2D system, is smaller than CORE; it is also better suited to the matter of "transformation pipes" of which some features bear on the "2D/3D one to many" problem.

_The GKS graphical standard has reached the status of ISO standard (ISO DIS 7942) [11,12]

W. M. Newman and A. van Dam



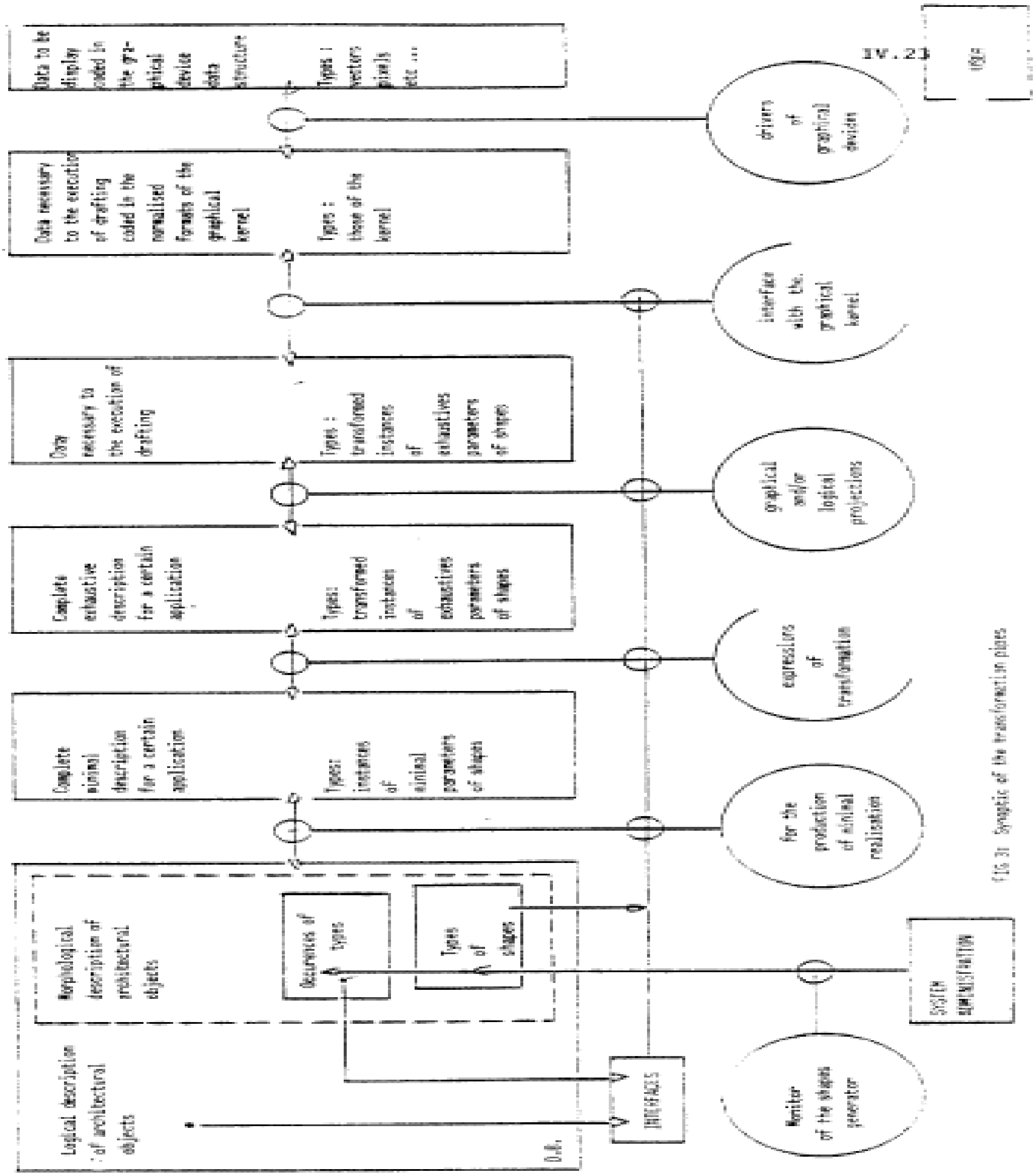


FIG 3: Synoptic of the transformation plans

_The C4 version of GKS in C [13] for the PDP 11 and VAX 11 UNIX environment, is being transported on SMX [14], a 68000 UNIX multiuser monoprocessor development environment package for the SM90 [15], which is a heterogeneous multimicroprocessor machine from the CNET.

BREAKING DOWN OF THE MODEL (figure 4)

The data structure and the process necessary to carry out interactive graphical communication are divided into three areas, each corresponding to the three levels of interaction defined above.

1/ The area containing the data for visual representation.

It consist of the standard graphical kernel and its graphical device "drivers".

2/ The area containing the data for graphical representation (of the morphologie of the project items).

It contains the different information intermediary states which allow the transformation of information from the state of minimum and permanent representation (the morphological description contained in the data base) to the state of the information communicated to the standard graphical kernel in a recognised format.

3/ The area containing the data for abstact description.

It contains the permanent minimum information of the data base, from which any manipulation can be executed, whether it be of graphical nature or not.

CONCLUSION

The survey, of some features of the problem of the manipulation of data using high_level graphical primitives within a CADCAM system, leads us to introduce the result of some recent research which seems to give answers to these questions.

We have exposed our proposals to elaborate a CADCAM system using these results:

The principle of a minimal and permanent represenation (the morphological description of object) linked to the non graphical information within the abstact description contained in the data base.

The principle of the "transformation pipes" betwen this representation and the 2D graphical information to be displayed on the graphical devices.

At the present stage of research a lot of points are insufficiently developped; netherthelesss we can already point this new problem:

Because of the large number of tasks which will be necessary to manipulate in real time and by graphical means, the large number of data of an architectural project, a study on the hierarchy between these tasks is needed; but we also look towards a multi_microprocessor solution to carry out this multi_task stucture.

REFERENCES

[1] J.C. Lebahar Le role du dessin dans le processus de conception. GAMSAU, Marseille 1978

[2] David SH. Rosenthal. Models of interactive graphics software. BSI OPS 13/WGS/30. 22 fevrier 1979

[4] J. Foisseau et al. Proposition pour pour un formalisme conceptuel et pour une architecture de systeme general CAO. CERT_DERI et CETE Aix. INRIA 79

[5] J. Autran and M. Florenzano. Etude exploratoire en vue de la conception et de la realisation d'un SGBD en CAO en Architecture. GAMSAU. Mai 78.

[6] JP. Crestin and J. foisseau. Base de donnees en CAO. CERT_DERI. Garin 81.

[7] W. M'Newman and A. Van Dam. Recents efforts toward graphics standardisation. in Co. Survey Vol.10 No 4, 78

[9] Computing Survey Vol.10. No 4. Special Issue. Graph Standards. Dec 75.

[10] Draft Intenational Standard ISO/DIS GKS Functional description. ISO TC97/SC5 WG2 No 117.

[11] G. Enderle. GKS_Implementation Over_view and Inquiry. in Comp. Graphics Forum Vol 2, N 1. pp.15_22. March 83

[13] D.S.H. R osenthal. GKS in C. in Proc Eurographics'82 pp. 371_382, ed. E.A Warman. North-Holland 1982.

[14] Equipe SM 90 de l'INRIA. L'architecture SM 90 Developpements Materiels et Logiciels. INRIA. Mai 1983.

[15] U. Finger and G. Medigue. Multiprocessor machines for enhanced availability: the SM 90. in L'echo des recherches English issue. pp.3-12. 82