

CONGRES : un système informatique pour la tenue des réunions à travers un réseau d'ordinateurs

Yvon Gardan

▶ **To cite this version:**

Yvon Gardan. CONGRES : un système informatique pour la tenue des réunions à travers un réseau d'ordinateurs. Modélisation et simulation. Institut National Polytechnique de Grenoble - INPG, 1977. Français. NNT : . tel-00287661

HAL Id: tel-00287661

https://theses.hal.science/tel-00287661

Submitted on 12 Jun 2008

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THESE

présentée à

l'Institut National Polytechnique de Grenoble

pour obtenir le grade de

DOCTEUR-INGENIEUR
Génie Informatique

par

Yvon GARDAN



CONGRES

UN SYSTEME INFORMATIQUE POUR LA TENUE DE REUNIONS A TRAVERS UN RESEAU D'ORDINATEURS



Thèse soutenue le 20 décembre 1977 devant la Commission d'Examen :

Président : L. BOLLIET

Examinateurs : C. BOKSENBAUM
M. LUCAS
G. VEILLON

INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE

Monsieur Philippe TRAYNARD : Président

Monsieur Pierre-Jean LAURENT : Vice Président

PROFESSEURS TITULAIRES

MM.	BENOIT Jean	Radioélectricité
	BESSON Jean	Electrochimie
	BLOCH Daniel	Physique du solide
	BONNETAIN Lucien	Chimie minérale
	BONNIER Etienne	Electrochimie et électrométallurgie
	BOUDOURIS Georges	Radioélectricité
	BRISSONNEAU Pierre	Physique du solide
	BUYLE-BODIN Maurice	Electronique
	COUMES André	Radioélectricité
	DURAND Francis	Métallurgie
	FELICI Noël	Electrostatique
	FOULARD Claude	Automatique
	LESPINARD Georges	Mécanique
	MOREAU René	Mécanique
	PARIAUD Jean-Charles	Chimie-Physique
	PAUTHENET René	Physique du solide
	PERRET René	Servomécanismes
	POLOUJADOFF Michel	Electrotechnique
	SILBER Robert	Mécanique des fluides

PROFESSEUR ASSOCIE

M.	ROUXEL Roland	Automatique

PROFESSEURS SANS CHAIRE

MM.	BLIMAN Samuel	Electronique
	BOUVARD Maurice	Génie mécanique
	COHEN Joseph	Electrotechnique
	LACOUME Jean-Louis	Géophysique
	LANCIA Roland	Electronique
	ROBERT François	Analyse Numérique
	VEILLON Gérard	Informatique fondamentale et appliquée
	ZADWORNY François	Electronique

MAITRES DE CONFERENCES

MM.	ANCEAU François	Mathématiques appliquées
	CHARTIER Germain	Electronique
	GUYOT Pierre	Chimie minérale
	IVANES Marcel	Electrotechnique
	JOUBERT Jean-Claude	Physique du solide
	MORET Roger	Electrotechnique nucléaire
	PIERRARD Jean-Marie	Mécanique
	SABONNADIERE Jean-Claude	Informatique fondamentale et appliquée
Mme.	SAUCIER Gabrièle	Informatique fondamentale et appliquée

MAITRE DE CONFERENCES ASSOCIE

M.	LANDAU Ioan	Automatique

CHERCHEURS DU C.N.R.S. (Directeur et Maîtres de Recherche)

MM.	FRUCHART Robert	Directeur de Recherche
	ANSARA Ibrahim	Maître de Recherche
	CARRE René	Maître de Recherche
	DRIOLE Jean	Maître de Recherche
	MATHIEU Jean-Claude	Maître de Recherche
	MUNIER Jacques	Maître de Recherche

Je tiens à remercier,

Monsieur le Professeur BOLLIET pour la confiance qu'il m'a témoignée et pour l'honneur qu'il me fait de présider le jury,

Monsieur le Professeur VEILLON, en particulier pour ses remarques critiques du manuscrit de cette thèse,

Monsieur BOKSENBAUM, Maître de Conférences à Montpellier, pour avoir guidé mes premiers pas dans la recherche et pour l'attention critique qu'il a manifestée pour ce travail dont le sujet lui tenait à coeur,

Monsieur LUCAS pour ses critiques constructives et pour son soutien actif qui m'ont été très utiles ; c'est un réel plaisir pour moi de pouvoir continuer à travailler dans son équipe.

Je voudrais également remercier tous les chercheurs avec qui j'ai pu travailler et tous ceux qui d'une manière ou d'une autre, ont favorisé l'aboutissement de ce travail.

Je ne saurais oublier Madame CHALAND qui a eu la redoutable tâche de frapper ce texte, ce dont elle s'est acquittée avec sa compétence habituelle, ainsi que le Service Tirage de l'ENSIMAG qui en a assuré l'impression.

Enfin, comme il serait trop long de dire tout ce que l'achèvement de cette thèse doit à Michèle, je lui dédie ce travail, modeste récompense pour tous ses sacrifices.

S O M M A I R E

INTRODUCTION p. 1

CHAPITRE I - LES COMMUNICATIONS ET L'ORDINATEUR

I.1. La théorie de la communication p. 4
I.2. Les communications p. 8
I.2.1. Les communications dans un groupe p. 8
I.2.2. Les techniques de communication p. 10
I.3. Les systèmes de réunion assistés par ordinateur p. 14
I.3.1. Généralités p. 14
I.3.2. Les systèmes actuels p. 15
I.3.2.1. Les réseaux d'ordinateurs p. 15
I.3.2.2. Les systèmes de réunions assistées par ordinateur p. 18
I.4. Buts du projet p. 19
I.4.1. Concepts généraux p. 19
I.4.2. Concepts techniques p. 21
I.4.3. Comparaison entre deux types de réunions p. 21

CHAPITRE II - GÉNÉRALITÉS

II.1. Introduction p. 24
II.2. Remarques générales p. 26
II.2.1. Contenu des interventions p. 26
II.2.2. Influence de l'asynchronisme p. 27
II.2.3. Facilités pour retrouver l'information p. 27
II.2.4. Rôle du leader dans la résolution des problèmes p. 28

II.3. Système orienté utilisateurs p. 31
II.3.1. Dispositifs de dialogue p. 31
II.3.2. Transmission de dessins p. 32
II.3.3. Réunion minimale p. 33
II.3.4. Modes de transmission p. 34
II.3.5. Retours dans la conversation p. 34
II.3.6. Réunions synchrones (réunions "normales") p. 35
II.3.7. Autres problèmes p. 35
II.4. Aspects généraux de réalisation p. 37
II.4.1. Méthodes de réalisation p. 37
II.4.1.1. Logiciel "réparti" p. 37
II.4.1.2. Logiciel centralisé p. 38
II.4.2. Organisation des fichiers p. 40
II.4.2.1. Organisation du fichier p. 41
II.4.2.2. Modalités d'accès p. 41
II.4.2.3. Protection p. 43
II.4.2.4. Durée de vie p. 45
II.4.3. Description de deux systèmes p. 46
II.4.3.1. Système basé sur la méthode de Delphes p. 46
II.4.3.2. Un système de réunions p. 47
II.4.3.3. Conclusion p. 49
II.4.4. Aspects généraux de réalisation p. 50
II.5. Problèmes techniques spécifiques au système CONGRES p. 52
II.5.1. Niveau graphique p. 52
II.5.1.1. Programmation graphique p. 52
II.5.1.2. Structure de données p. 53
II.5.1.3. Codage p. 55
II.5.1.4. Indépendance du matériel par rapport au logiciel p. 61
II.5.2. Niveau général p. 65
II.5.2.1. Automatisation p. 65
II.5.2.2. Transmissions p. 66
II.5.2.3. Réunions synchrones automatisées p. 66

CHAPITRE III - POINT DE VUE UTILISATEUR

III.1. Généralités p. 68
III.1.1. LAGOR (Langage Avec Graphique Orienté Réunions) p. 68
III.1.2. Influence du type de console p. 69
III.1.3. Initialisation (généralités) p. 71
III.2. Construction et modification de dessins p. 72
III.2.1. Constructions p. 75
III.2.1.1. Environnement FIGURE p. 75
III.2.1.2. Environnement TEXTE p. 78
III.2.1.3. Environnement LIEN p. 80
III.2.2. Modifications p. 82
III.2.2.1. Niveau global : le dessin p. 82
III.2.2.2. Modifications d'objets p. 86
III.2.3. Mémorisation p. 95
III.2.4. Appel p. 95
III.2.5. Commandes spéciales p. 96
III.3. Tenue d'une réunion p. 98
III.3.1. Début d'une session p. 98
III.3.2. Retours dans la conversation p. 101
III.3.3. Préparation d'un commentaire p. 103
III.3.4. Modes de transmission p. 103
III.3.5. Réceptions p. 104
III.3.6. Cas particulier : réunion normale p. 105
III.4. Conclusions p. 108

CHAPITRE IV - CONCEPTION ET RÉALISATION DE CONGRES

IV.1. Choix généraux p. 110
IV.1.1. Indépendance du logiciel par rapport aux consoles p. 110
IV.1.2. Les fichiers p. 112
IV.1.2.1. Le fichier de conversation p. 113
IV.1.2.2. Le fichier personnel p. 115
IV.1.2.3. Autres fichiers p. 115

IV.1.3. Partage du logiciel p. 117
IV.1.3.1. Module d'interprétation p. 117
IV.1.3.2. Module de gestion de la réunion p. 124
IV.2. Les outils p. 126
IV.2.1. CP 67 et accès par CYCLADES p. 127
IV.2.2. Les consoles p. 130
IV.2.2.1. Les consoles graphiques p. 130
IV.2.2.2. Les consoles alphanumériques p. 133
IV.2.3. Programmes de service p. 133
IV.3. Conception et réalisation de CONGRES p. 135
IV.3.1. La console virtuelle p. 142
IV.3.1.1. L'écran virtuel p. 143
IV.3.1.2. Structure de données p. 144
IV.3.2. Interpréteur de commandes INTER p. 151
IV.3.2.1. Analyseur lexicographique p. 151
IV.3.2.2. Analyseur syntaxique p. 152
IV.3.2.3. Interprétation des commandes p. 154
IV.3.2.4. Interprétation des informations venant du SUPLOC p. 162
IV.3.3. Résolution des problèmes graphiques p. 164
IV.3.3.1. Exemples de problèmes graphiques p. 164
IV.3.3.2. Cas de quelques consoles p. 173
IV.3.4. Tenue d'une réunion p. 176
IV.3.4.1. Les fichiers et la sécurité p. 176
IV.3.4.2. Le SUPLOC p. 182
IV.3.4.3. Le SUPGEN p. 185
IV.3.4.4. Cas particulier d'une réunion normale p. 195
IV.3.4.5. Déroulement d'une réunion p. 199
IV.3.5. Conclusion p. 202

CHAPITRE V - CONCLUSION

V.1. Bilan et avenir p. 205
V.1.1. Propositions techniques p. 206
V.1.2. Propositions générales p. 206
V.2. Evaluation du coût p. 208
V.3. Conclusion p. 214

RÉFÉRENCES p. 216

INTRODUCTION

Les développements récents des réseaux d'ordinateurs, tant aux Etats Unis (ARPA, TYMNET) qu'en Europe (CYCLADES, EIN) permettent des applications informatiques d'un type nouveau. On peut considérer en particulier, que l'un des buts essentiels d'un réseau est de permettre à des entités distantes d'agir les unes sur les autres. A la base de toutes ces actions, se trouve une communication entre entités distantes (programmes ou, par leur intermédiaire, êtres humains). Or, on peut constater que jusqu'à présent, les communications se limitaient à un accès à des services distants pour tout utilisateur du réseau. En particulier, dans un réseau hétérogène général, tel que CYCLADES, comprenant des matériels très différents (IBM, CII, CDC, PHILIPS, ..), la connexion d'un ordinateur signifie, d'une part, mettre les services dont on dispose localement à la disposition d'utilisateurs distants, d'autre part, offrir aux utilisateurs locaux l'accès aux services réseau.

Les communications entre utilisateurs à travers un réseau sont actuellement à un stade primaire, consistant en des transmissions de fichiers. Considérant donc que peu de recherches ont été entreprises pour permettre à des utilisateurs de communiquer à travers un réseau, nous avons étudié un logiciel permettant des réunions entre personnes éloignées géographiquement, ayant à leur disposition des consoles connectées à un réseau (par exemple, CYCLADES).

A partir de cette idée de conception de réunion, relativement vague au départ, on peut émettre quelques idées de base :

. l'introduction de l'ordinateur dans la boucle de communication devrait permettre des réunions d'un type nouveau. Il s'agira en particulier d'utiliser au mieux les possibilités inhérentes à un ordinateur, telles que puissance de calcul, facilités de mémorisation et de recherche d'information ;

. ce service devra être accessible à des non informaticiens et éviter ainsi l'inconvénient de voir un réseau tel que CYCLADES rester un outil pour initiés ;

. le logiciel mis en place sera suffisamment complet pour permettre une utilisation en vraie grandeur pouvant servir de base à une étude de l'impact de ce moyen de communication sur les utilisateurs ;

. enfin, et c'est un des aspects originaux de ce projet, les participants auront la possibilité de se transmettre des dessins.

On voit donc que sur le plan de la recherche, les intérêts du projet étaient :

. d'étudier, en partant d'une idée simple (réunion à travers un réseau), un type tout à fait nouveau de communication ;

. de réaliser effectivement ce système et d'en tester l'exploitation, non seulement pour vérifier que "ça marche", mais également pour étudier l'impact que ce type d'outil pourrait avoir sur les utilisateurs.

L'étude de ce projet comporte deux aspects qui, pour n'être pas totalement indépendants, n'en sont pas moins distincts :

. un aspect graphique qui consiste à mettre à la disposition des utilisateurs des outils de construction et de modification de dessins. Etant donnée la grande diversité des consoles (graphiques, alphanumériques, ..), on peut d'ores et déjà noter que des problèmes d'indépendance du logiciel par rapport au matériel (consoles) se poseront avec une grande acuité ;

. un aspect réunion qui comprend un ensemble de mécanismes assurant la transmission de dessins ou de commentaires entre les participants et gérant la réunion, telle qu'elle sera définie dans le chapitre III.

On présentera dans le premier chapitre un certain nombre d'aspects généraux des communications et le rôle que joue, ou que pourrait jouer, un ordinateur dans l'environnement actuel des moyens de communication.

Le second chapitre montrera les buts essentiels que l'on veut atteindre et abordera les principaux problèmes techniques qui se posent au niveau de la conception d'un tel système. Ces problèmes se présentent d'une part pour satisfaire au mieux les utilisateurs, d'autre part sur un plan plus technique, pour assurer par exemple l'indépendance à un certain niveau du logiciel par rapport au matériel.

Le chapitre suivant décrira le système du point de vue de l'utilisateur. Il sera illustré par un certain nombre de figures, créées au moyen du système CONGRES lui-même.

Le quatrième chapitre présente les aspects techniques du système CONGRES. Il comporte trois parties qui sont :

. la description des choix généraux de conception, indépendants du choix du matériel et qui font suite à ce qui a été présenté au chapitre II,

. la présentation des outils, tant matériels que logiciels, qui ont été choisis, en suivant toujours les idées générales exprimées dans le chapitre II,

. l'exposé technique du système CONGRES, tant du niveau graphique que du niveau général.

Enfin, le dernier chapitre exposera les conclusions que l'on peut tirer de cette étude et abordera l'aspect coût d'utilisation.

-CHAPITRE I

LES COMMUNICATIONS ET L'ORDINATEUR

I.1. LA THÉORIE DE LA COMMUNICATION

Ce chapitre va servir d'approche au problème posé. On commencera par placer l'ordinateur par rapport aux moyens de communication existants (les possibilités futures seront abordées dans le chapitre V). Ensuite, on soulignera quelques généralités sur les systèmes de réunions assistées par ordinateur et on en fera un bref historique. Enfin, on donnera les grandes lignes du projet CONGRES.

Avant d'aborder les communications sur un plan technique, il n'est pas inutile de dire quelques mots de la théorie de la communication. Les types de communication sont nombreux, leur but commun étant la transmission d'un message.

Le schéma général d'une communication est le suivant [SH] :

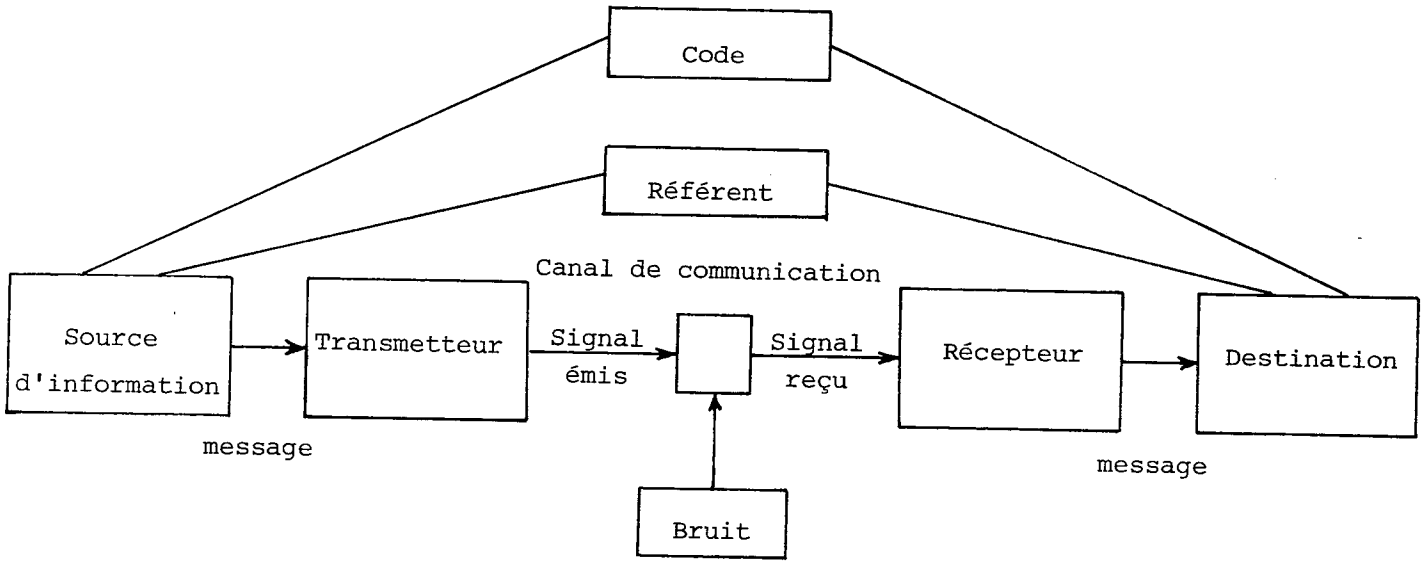


Figure 1

Explicitons les éléments impliqués dans ce schéma :

· une source d'information produit un message ou une suite de messages. Ces messages peuvent être de types différents, par exemple une fonction du temps f(t) pour un message transmis par téléphone, une séquence de caractères pour un message transmis par télétype, ou des fonctions plus complexes pour la télévision, de la forme f(x,y,t) pour un système en noir et blanc, ou plus complexe encore pour la couleur ;

· un transmetteur a pour rôle de transformer le message pour qu'il puisse être transmis à travers le canal de communication ; pour le téléphone, cette opération consiste à transformer une pression (due au son) en un courant électrique ;

· le canal de communication est la voie d'acheminement des messages ; on peut l'assimiler aux moyens techniques auxquels le transmetteur a recours pour acheminer un signal vers le récepteur ; ce peut être un cable coaxial, une bande de fréquence, etc .. ;

· le récepteur a un rôle inverse à celui du transmetteur, puisqu'il consiste à transformer un signal en un message ;

· le destinataire est celui à qui est destiné le message (personne, machine, ..) ;

· le référent est constitué par le contexte réel ou imaginaire auquel fait référence un message ;

. le code est un ensemble de signes et de règles de combinaison de ces signes (grammaire). La source d'information se sert de ce code pour constituer un message correct et le destinataire reconstitue le sens du message à partir de ce code (à condition qu'ils aient le même code ou au moins une partie commune : un anglais et un français ignorant la langue qui n'est pas la leur, n'ont aucune chance de se faire comprendre au téléphone, en revanche s'ils sont en présence l'un de l'autre, ils peuvent se faire des signes) ;

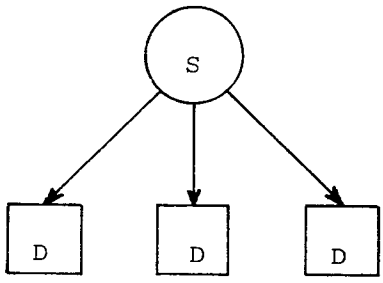
. le bruit désigne tout ce qui, d'une manière ou d'une autre, affecte la transmission du message ; ce peut être non seulement une perturbation d'ordre sonore comme la signification courante du mot bruit semble l'indiquer, mais encore une faute de frappe dans un message tapé à la machine à écrire, etc ..

Les principaux types de communication sont la diffusion et l'échange (cf. figure 2) :

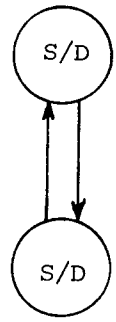
. la diffusion est une communication unilatérale s'établissant donc d'une source d'information vers un destinataire, sans qu'il y ait réciprocité ;ainsi, un professeur durant un cours magistral, un poste de télévision, diffusent des messages ;

. l'échange est une communication bilatérale, c'est-à-dire dans laquelle source d'information et destinataire font alterner leur rôle. C'est le cas d'une conversation courante.

Figure 2



diffusion



Echange

S = source d'information

D = destinataire

S/D = source d'information et destinataire

On peut remarquer que certains moyens de communication de type diffusion établissent un échange (faible, mais réel) avec les destinataires, en utilisant en général un canal de communication différent (courrier des lecteurs pour la presse écrite ou la télévision, questions posées par téléphone dans une émission de radio ou de télévision, etc ..).

I.2. LES COMMUNICATIONS

I.2.1. Les communications dans un groupe

Un groupe est un ensemble de personnes ayant une oeuvre commune à accomplir. Il existe, à l'intérieur de ce groupe, des communications entre individus. Les processus de la vie d'un groupe ne peuvent pas être étudiés sans référence aux processus de communication qui l'accompagnent. L'un des moyens de faciliter ces communications et faire progresser la résolution des problèmes posés au groupe, est de provoquer une réunion [DE], [MA].

Une réunion est un moment de la vie d'un groupe, dont l'objectif est de permettre des communications entre les membres du groupe, afin de régler un certain nombre de problèmes.

Toutes les réunions ne sont pas de même nature, du fait même que tous les problèmes ne nécessitent pas une solution de même type. On peut distinguer par exemple :

. les réunions d'information : les responsables de la réunion choisissent un thème et un ou plusieurs spécialistes venant exposer leurs idées sur ce thème ;

. la dynamique de groupe : le groupe a une liberté totale dans tous les domaines, programme, animation, la seule contrainte étant de respecter l'unité de lieu "ici" et l'unité de temps "maintenant" ;

. brain storming : chacun exprime librement tout ce qui lui passe par la tête, en partant du principe que cela est profitable au groupe, soit directement, soit en suscitant chez un autre participant une idée intéressante

. la réunion décision : le but de la réunion est de prendre une décision sur un sujet donné. Ces réunions sont en général précédées d'une convocation comportant une introduction au sujet à traiter ; cette introduction sera reprise au début de la réunion avant de lancer la discussion proprement dite.

L'analyse de ces réunions est faite dans de nombreux ouvrages. Cependant, il faut remarquer que l'étude des processus de communication ne doit pas se limiter aux seuls processus socio-psychologiques, mais accorder de l'importance à l'infrastructure matérielle permettant la communication (transmetteur, canal de communication, récepteur).

On appelle réseau de communication l'ensemble des moyens matériels permettant à A de communiquer avec B, à B de communiquer avec C, etc ..

Si l'on considère les réunions où le groupe est dans une même salle à un même moment, quelle que soit la nature proprement dite de la réunion, elle respecte de façon générale les principes suivants :

. unité de temps : la réunion se tient à partir d'un moment donné pendant une certaine durée ;

. unité de lieu : tous les participants se trouvent dans la même salle ;

. tous les participants peuvent communiquer entre eux ;

. chaque participant voit tous les autres protagonistes de la réunion et sait donc toujours qui est la source d'une communication ;

. un animateur peut être chargé d'animer la réunion, c'est-à-dire de la guider, de la relancer, etc ..

. autres caractéristiques : deux personnes ne peuvent parler en même temps ; chacun peut disposer de moyens pour prendre des notes ; un tableau aide éventuellement l'explication des problèmes posés ; etc ..

Cette description sommaire concerne les réunions telles qu'on les pratique habituellement. Elles peuvent être accompagnées de moyens audio-visuels.

I.2.2. Les techniques de communication [DA], [SR]

Après avoir utilisé les moyens les plus immédiats, tels que les signes et la parole, l'être humain a cherché à développer des techniques de communication : on peut placer parmi celles qui ont eu une grande influence : l'écriture et les moyens qu'on a trouvés pour la diffuser (courrier, journaux). Ensuite, sont venus les moyens techniques de l'ère moderne, parmi lesquelles le téléphone et la télévision :

. le téléphone permet des communications bilatérales entre individus (échange) ;

. la télévision permet à un nombre très élevé de personnes d'assister aux événements importants (diffusion).

Considérant que l'un des meilleurs moyens de résoudre les problèmes qui peuvent se poser à un groupe de personnes est de mettre sur pied une rencontre, que ce soit sous forme de conférence ou sous forme de réunion de travail plus restreinte, le développement de nouveaux moyens de communication est un objectif fondamental. On peut considérer ces développements sous la forme de nouveaux types de communication, comme peuvent l'être les réseaux d'ordinateurs, ou sous la forme de compléments aux moyens actuels.

De notre point de vue, on peut se demander ce que peut apporter l'utilisation d'un ordinateur, par exemple dans le cadre bien connu de la télévision. On peut par exemple imaginer le système suivant :

Les téléspectateurs ont à leur disposition un moyen d'entrer des données dans un ordinateur, par exemple par l'intermédiaire de boîtes à cinq boutons permettant de voter sur une échelle de 1 à 5 (très mauvais, mauvais, moyen, bon, très bon). L'ordinateur se charge alors de traiter ces informations et de les afficher sous forme de diagrammes ou de tableaux à l'intention des producteurs d'émissions et éventuellement des téléspectateurs eux-mêmes. Le programme peut permettre des votes anonymes (il suffit qu'il ne sorte par le nom des votants), tout en assurant que tous les membres de l'échantillon ont voté une et une seule fois.

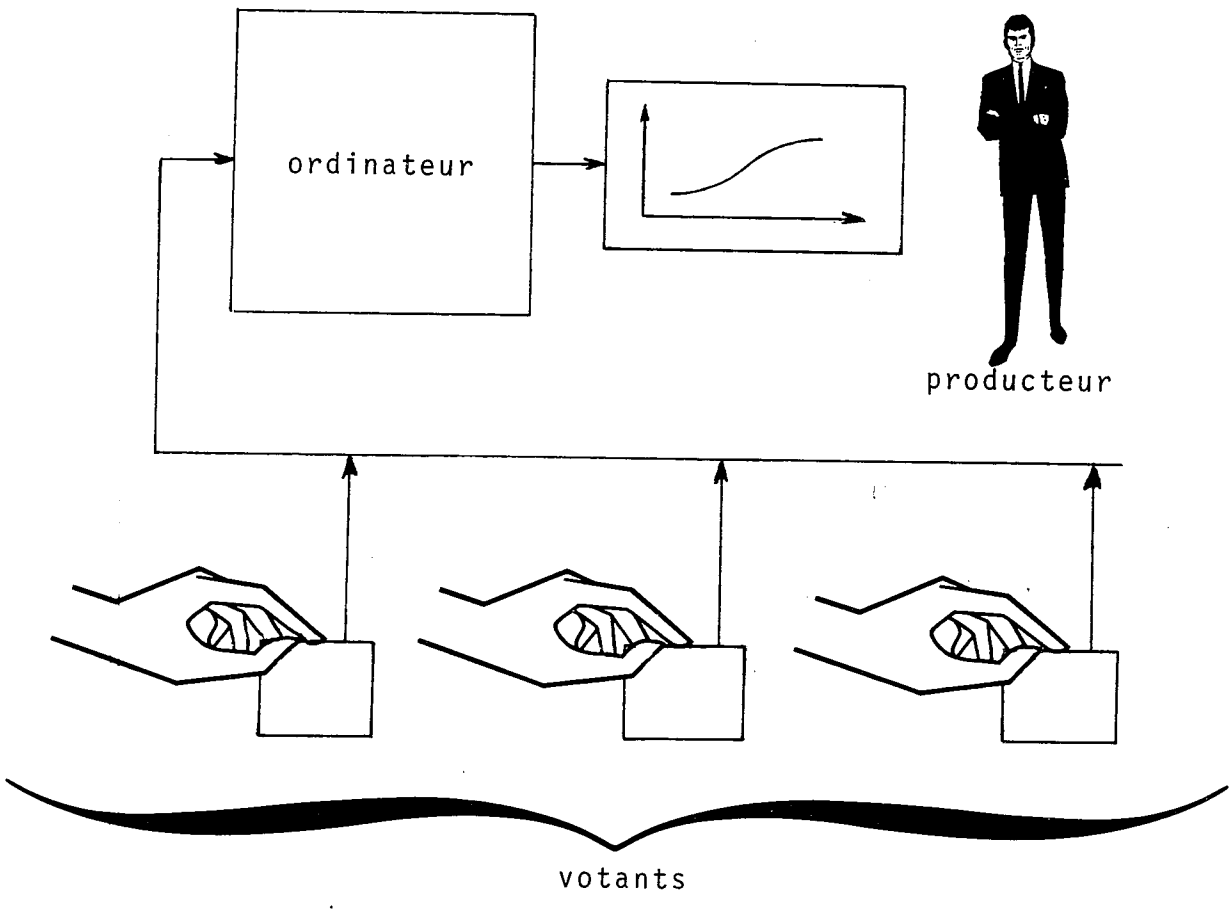


Figure 3

On peut également dire quelques mots de la "méthode de Delphes" qui a été une des premières à trouver des applications assistées par ordinateur. [TU3]

Cela consiste à réunir un certain nombre d'experts sur un sujet donné. La première phase de la méthode consiste en un échange de vues sur le problème et la mise sur pied d'un certain nombre de propositions sur lesquelles les participants sont invités à donner leur avis par l'intermédiaire d'un vote. Une fois tous les votes recueillis par l'ordinateur (ou un pourcentage important), celui-ci les traite puis les fournit aux votants sous forme condensée (tableaux ou diagrammes). Un nouveau vote sur la même proposition, après une éventuelle discussion, peut avoir lieu. Le même processus peut ainsi être répété plusieurs fois. Les deux caractéristiques de cette méthode sont :

. respect de l'anonymat,

. "feedback" important (rétro-action).

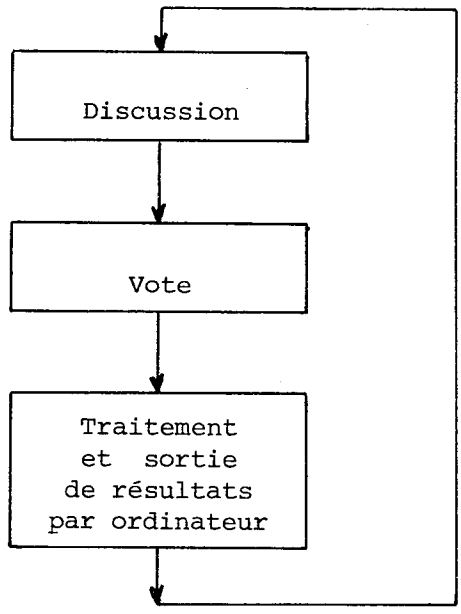


Figure 4

Ces deux exemples ne sont qu'un reflet de ce que peut apporter l'ordinateur dans le domaine des communications entre individus. Dans cette optique, on peut arbitrairement mettre en relief trois générations de moyens de communication électroniques :

. radio : transmet des messages parlés (en général du centre vers la périphérie),

. télévision : transmet des messages audio-visuels (du centre vers la périphérie),

. les systèmes de communication assistés par ordinateur ayant un certain nombre de caractéristiques nouvelles, en particulier :

- aucune perte d'information : lorsqu'on écoute la radio ou regarde la télévision, on ne peut pas revenir en arrière, alors que les systèmes assistés par ordinateur permettent de retrouver rapidement, à n'importe quel moment, une information diffusée auparavant ;

- simultanéité non obligatoire : plusieurs participants, même s'ils sont ensembles dans le système, peuvent très bien ne pas être dans la même partie du programme ;

- les styles de participation pourront être très différents suivant la personnalité des différents participants.

Dans l'environnement technique actuel, il faut ajouter que cette technique n'en étant qu'à ses balbutiements, les moyens d'entrée et de sortie sont presque exclusivement des claviers alphanumériques et du papier, à l'extrême limite des écrans graphiques. On conçoit donc qu'actuellement ces systèmes sont moins riches que la télévision par exemple, ne serait-ce qu'au point de vue dynamique des images. On peut cependant considérer que dans un avenir proche, les entrées se feront tout simplement par la voix et les sorties sur des écrans très perfectionnés (écrans de télévision).

I.3. LES SYSTÈMES DE RÉUNION ASSISTÉS PAR ORDINATEUR

I.3.1. Généralités

Dans de tels systèmes, l'ordinateur est chargé de permettre à des participants éloignés géographiquement, de communiquer entre eux.

Les caractéristiques générales de ces systèmes sont les suivantes (elles peuvent varier suivant les désirs du concepteur) :

. les réunions peuvent être synchrones ou asynchrones : dans une réunion synchrone, tous les participants ont décidé d'une heure précise et sont présents en même temps. Pendant cette période, les communications entre individus peuvent être asynchrones (c'est-à-dire que émetteur et récepteur n'ont pas à se synchroni$er (cf. réunions normales)). Dans une réunion asynchrone, chacun participe lorsqu'il le désire, c'est-à-dire qu'à un instant donné, zéro, une ou plusieurs personnes peuvent être présentes. On parlera dans toute la suite de réunions asynchrones (ou étendues), pouvant être exceptionnellement synchrones.

. N'importe qui peut entrer des messages à n'importe quel moment. En particulier, contrairement aux faces à faces où chacun doit attendre son tour, plusieurs personnes peuvent taper un message au même moment (asynchronisme des transmissions).

. Tout ce qui est transmis peut être conservé dans un fichier spécial. Ainsi, lorsqu'un participant revient dans la réunion après une absence, il a la possibilité de revoir tout ce qui lui a été transmis depuis sa dernière participation.

. Plusieurs types de transmission pourront être autorisés. Par exemple, des transmissions publiques ou privées, anonymes ou signées.

. Il est possible d'implémenter des systèmes de vote illustrant, par exemple, la méthode de Delphes.

D'une manière générale, on essaie d'utiliser les possibilités inhérentes à un ordinateur. On verra plus précisément dans le chapitre III la façon dont se déroule une réunion dans le système CONGRES.

I.3.2. Les systèmes actuels

I.3.2.1. Les réseaux d'ordinateurs

L'apparition des systèmes de communication assistés par ordinateur, date du début des années 1970. A cela une raison bien simple : la nécessité de disposer de réseaux d'ordinateurs. Or, les premiers réseaux ayant dépassé le stade expérimental pour atteindre un niveau d'utilisation en vraie grandeur et qui soient utilisables pour ce type d'application, sont apparus vers la fin des années 1960. Parmi ceux-ci, ont peut citer ARPA, le plus connu, qui relie des universités américaines et TYMNET1 qui est un réseau commercial.

Les réseaux peuvent se distinguer par leur structure, leurs fonctions et leur composition.

. Structure :

- réseaux centralisés :

il s'agit de réseaux de terminaux avec un ou plusieurs gros ordinateurs au centre d'une étoile. Les premiers réseaux d'ordinateurs étaient de cette forme :

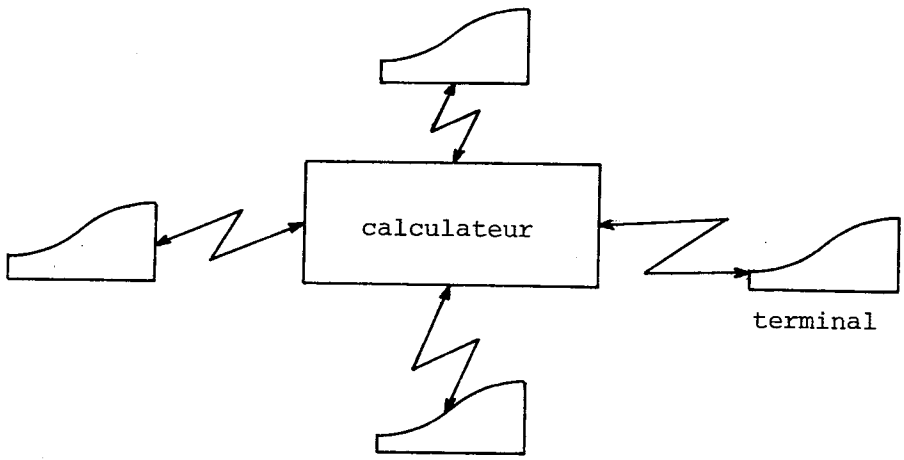


Figure 5

- réseaux décentralisés :

on dispose alors de plusieurs ressources (ordinateurs) accessibles à travers un réseau de commutation qui, quelle que soit la nature des ressources, est un interface homogène :

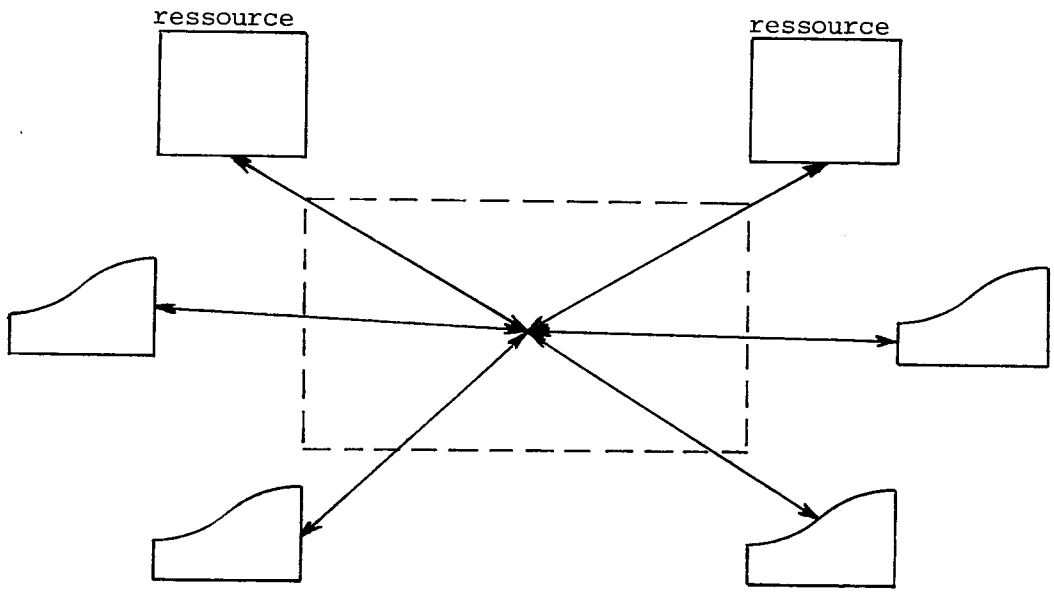


Figure 6

. Fonction :

- réseaux spécialisés : il s'agit de réseaux créés en fonction d'une application bien particulière ; ils font en général appel à un type unique de matériel ;

- réseaux généraux : au contraire des réseaux spécialisés, les réseaux généraux d'ordinateurs se caractérisent par la diversité des applications qu'ils permettent.

. Composition :

- réseaux homogènes : tous les ordinateurs faisant partie du réseau sont du même type (même marque) ;

- réseaux hétérogènes : les matériels composant le réseau sont de types (marques) différents.

Parmi les réseaux hétérogènes généraux d'ordinateurs, on peut citer ARPA et, parce qu'il nous touche de plus près, CYCLADES.

Le réseau hétérogène général d'ordinateurs CYCLADES est l'aboutissement de recherches entreprises en 1972. Il est entré dans une phase d'utilisation réelle en 1975 et un certain nombre de logiciels tendent à en développer l'efficacité et à rendre son emploi plus commode. Les principaux ordinateurs participants viennent de la CII-HB (IRIS 80, IRIS 50, MITRA 15, 10070), d'IBM (360/67), de CDC (6600), de PHILIPS et SIEMENS. Le réseau de commutation CIGALE est constitué de mini-ordinateurs CII-HB MITRA 15.

I.3.2.2. Les systèmes de réunions assistées par ordinateur

Parmi les premières recherches effectuées dans ce domaine, on peut citer les travaux de M. TUROFF et de T.W. HALL [TU], [HA]. Les premiers systèmes implémentés étaient essentiellement des applications de la méthode de Delphes. Par la suite, un certain nombre de personnes, informaticiens ou sociologues, se sont intéressées à ce moyen de communication.

Le seul système de réunion assistée par ordinateur qui, à notre connaissance, a été implémenté, est FORUM, réalisé par le "Group Communication through Computers" aux Etats Unis [GR], [LI]. Cette recherche a été faite avec des moyens assez importants et les tests ont été réalisés tant d'un point de vue technique que d'un point de vue sociologique. En 1972, les membres de cette équipe étudiaient un système permettant de mettre en rapport des experts éloignés géographiquement afin de résoudre un problème donné. Dans une deuxième phase, un système de communication plus général était implémenté sur le réseau ARPA. Après un certain nombre d'expériences réalisées sur ce prototype, le système PLANET était implémenté sur le réseau commercial TYMNET1 et mis à la disposition de la NASA [VA1], [VA2].

Cette technique n'en est donc encore qu'à ses balbutiements et, comme tout nouveau moyen de communication, son impact sur les utilisateurs est difficile à prévoir. C'est pourquoi, il semble intéressant de réaliser des prototypes expérimentaux aux possibilités diverses, afin de voir d'abord si ce type de communication est un moyen d'avenir et, dans ce cas, d'en tirer des conclusions qui puissent servir de base aux systèmes futurs.

I.4. BUTS DU PROJET

Il s'agit de concevoir et réaliser un logiciel permettant des réunions entre personnes éloignées géographiquement, ayant à leur disposition des consoles graphiques ou alphanumériques connectées à un réseau.

A l'origine, on se proposait de concevoir des réunions calquées sur des réunions de type face à face, le système tenant le rôle du président de séance (donne et retire la parole, etc ..). On verra plus loin que ce type de réunion n'est qu'un sous-ensemble de ce qui est réalisé, tout simplement parce qu'il n'utilise pas au maximum les possibilités inhérentes à un ordinateur, en particulier la conservation et la recherche faciles et rapides de l'information.

Sur un plan général, les principales caractéristiques d'une réunion sont détaillées dans les paragraphes suivants.

I.4.1. Concepts généraux

. Asynchronisme : on dit qu'une réunion est asynchrone lorsque à un moment donné, zéro, un, ou plusieurs participants sont présents. Cela signifie que chacun participe quand il le veut (cf. § II.2.1.).

. Transmissions simultanées : à un instant donné, plusieurs personnes peuvent taper et transmettre de l'information. Cette notion est tout à fait contraire aux normes habituelles que l'on rencontre dans les face à face et même dans les conversations par téléphone où deux personnes ne peuvent parler en même temps sous peine de rendre la discussion inaudible.

. Transmission de dessin : on veut donner la possibilité aux participants de se transmettre des dessins. Il faudra bien sûr définir, d'une part les types de dessins transmissibles, d'autre part les méthodes de construction de ces dessins. On pourra élaborer un logiciel de construction et de modification de dessins assez complexe, sachant qu'il existera un sous-ensemble facile à utiliser par des non-informaticiens (en particulier pour la transmission de textes).

. Revue d'interventions : on désigne par intervention ce qui est transmis par un participant (par exemple un dessin). Etant donné qu'une réunion est en général asynchrone, il faudra donner à un participant la possibilité de consulter les interventions qui lui ont été transmises durant son absence (au minimum), ou toute intervention transmise auparavant.

. Fichier personnel : on désigne sous le terme de fichier personnel, une base de données permettant à l'utilisateur de conserver des dessins ou des messages qu'il a créés, ou des interventions qu'il a prises dans la conversation et de les réafficher à sa guise par la suite. Cela devrait permettre, par exemple, à quelqu'un qui a des dessins complexes à envoyer, de les préparer et de les transmettre ensuite dans l'ordre et en temps voulus.

I.4.2. Concepts techniques

. Indépendance par rapport au matériel : le logiciel devra être disponible à partir de consoles aux possibilités très différentes allant des consoles alphanumériques aux consoles graphiques.

. Statistiques : on s'attachera à mettre en place, ou au moins à rendre facile la mise en place d'un certain nombre de statistiques permettant une étude de l'impact d'un tel moyen de communication sur les utilisateurs et ainsi un recensement des points à améliorer et des possibilités à ajouter ou à supprimer.

I.4.3. Comparaison entre deux types de réunions

Si l'on compare ce type de réunion à des réunions telles qu'on les a décrites au début de ce chapitre, on constate que :

. l'unité de lieu n'est plus respectée : chacun est dans son coin, devant sa propre console ;

. l'unité de temps devient une notion floue : une réunion peut durer un temps assez long (plusieurs jours, plusieurs mois) et chacun ne vient participer que lorsqu'il le désire, le système se chargeant de l'informer ;

. plusieurs personnes peuvent taper de l'information en même temps ;

. les moyens matériels de communication sont bien particuliers : en général un clavier, un écran et, de façon un peu moins visible, un ordinateur.

On fera dans le chapitre II quelques remarques générales sur l'influence des caractéristiques d'un système de réunion assistée par ordinateur tel qu'on veut en construire une version. On peut cependant, en regardant la figure 7, avoir une idée assez juste des différences essentielles entre une réunion assistée par ordinateur et une réunion où tout le monde se trouve dans la même salle.

Remarque :

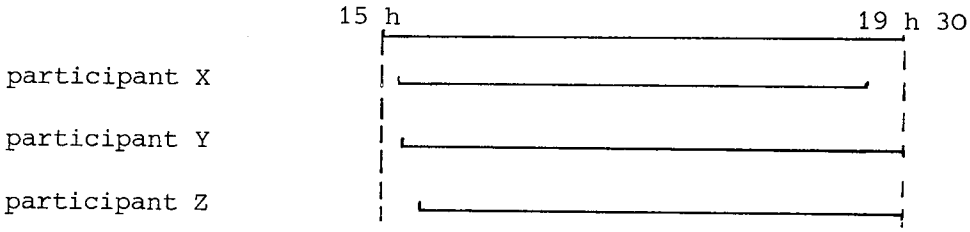
On implémentera également un système de réunion calqué de façon assez précise sur les réunions où le groupe se trouve dans une même salle, en chargeant une partie du logiciel de jouer le rôle de l'animateur. Seule alors l'unité de lieu ne sera plus respectée, les autres caractéristiques telles que unicité de parole, présence de tous les participants, etc .., étant reconduites d'une certaine façon. On désignera ce type de réunion par le terme *réunion normale*.

Figure 7

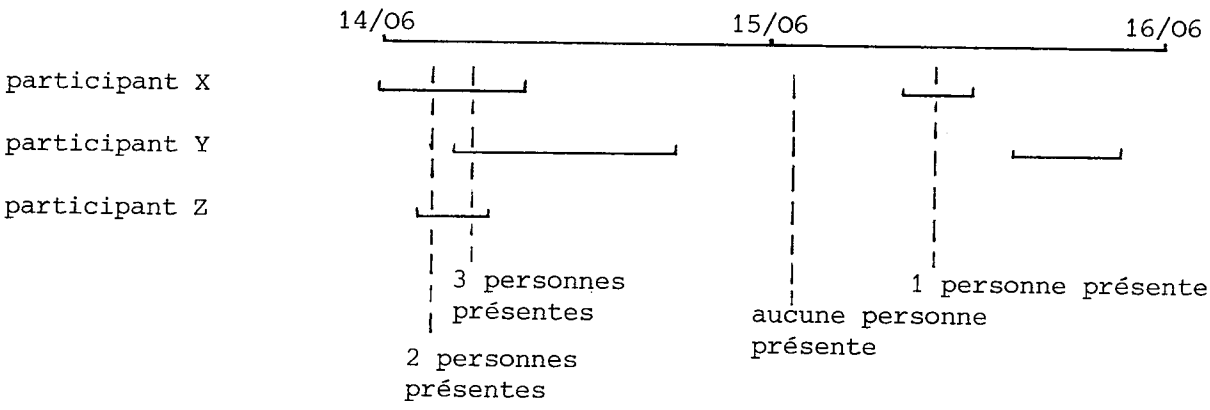
	REUNION A	REUNION B (assistée par ordinateur)
Participants	X de Lyon Y de Grenoble Z de Paris :	X de Lyon Y de Grenoble Z de Paris :
Lieu	Grenoble	chacun chez soi (Lyon, Grenoble, Paris ..)
Date de début	le 11/06/77 à 15 h	le 11/06/77 à 15 h
Durée probable	2 heures	3 mois
Présence	obligatoire	au gré du participant

Les schémas de présence peuvent être décrits par :

Réunion A :



Réunion B (exemple sur 2 jours)



CHAPITRE II

GÉNÉRALITÉS

II.1. INTRODUCTION

Ce chapitre présente ce que l'on entend par réunion assistée par ordinateur et on y expose un certain nombre de choix à faire au point de vue de la conception de ce type de système.

On commence par exprimer les différences essentielles entre une réunion telle qu'on la comprend en général (type face à face) et une réunion assistée par ordinateur. On montre les avantages et les inconvénients que l'on peut trouver à l'une par rapport à l'autre. Cette comparaison est importante pour la bonne compréhension de la suite de cet exposé, car elle permet de ne plus faire de confusion sur le terme *réunion* que l'on continuera à employer.

On introduira ensuite l'aspect utilisateur de ce système, en montrant particulièrement tous les outils que l'on désire mettre à sa disposition. Ceci sera fait sans tenir compte des moyens matériels fournis à l'utilisateur, afin de ne pas trop baser notre étude sur des cas particuliers.

On abordera enfin, de façon un peu plus approfondie, les problèmes de conception posés par l'étude d'un tel système, aussi bien pour l'aspect écriture d'un logiciel que pour l'aspect utilisation de matériels. On donnera dans une première partie, les idées générales sur les aspects généraux de la conception, ce qui, à partir de quelques lois simples, nous amènera à juger de façon plus précise les matériels et les logiciels que l'on pourra utiliser (cf. chapitre IV).

On décrira ensuite un certain nombre de problèmes techniques spécifiques au système que l'on veut implémenter, tant d'un point de vue graphique (programmation, structure de données, codage, ..) que d'un point de vue logiciel indépendant du matériel et gestion de la réunion. Cette dernière partie peut être considérée comme une introduction approfondie aux aspects conception et réalisation exprimés dans le chapitre IV.

II.2. Remarques générales

Nous nous proposons, dans ce paragraphe, de revenir un peu sur la comparaison des moyens usuels de communication avec les communications assistées par ordinateur, essentiellement d'un point de vue que l'on pourrait qualifier d'impact sur le public. Un certain nombre de points sont remarquables à ce sujet (ce n'est évidemment qu'une approche de ces problèmes qui pourraient à eux seuls faire l'objet d'une étude très intéressante).

II.2.1. Contenu des interventions

Dans tous les systèmes assistés par ordinateurs actuels, ce qui est transmis l'est sous forme de message écrit. Nous pensons que cette limitation peut être repoussée par l'utilisation de consoles graphiques.

Cependant, du fait de l'isolement des participants les uns par rapport aux autres, les échanges perdent évidemment de la richesse, comparés aux face à face (où les gestes, les intonations, les mimiques, ..., apportent de l'information), ou au téléphone (où le ton et les silences apportent de l'information). Dans l'état actuel des choses (les personnes ne se voyant pas), cela a tout de même l'avantage de permettre l'anonymat (le ou les destinataires ne connaissent pas l'identité de l'émetteur).

II.2.2. Influence de l'asynchronisme

Cette influence agit sur deux points :

. chacun peut transmettre à tout instant et n'est donc pas obligé d'attendre qu'un autre ait fini de transmettre ;

. chacun participe lorsqu'il le désire. Ceci a l'avantage de laisser à chacun le soin de s'organiser, de lire tranquillement ce qui a été transmis, etc .. En revanche, personne n'est obligé de répondre à une question (au moins pas rapidement), ce qui peut être frustrant pour celui qui a posé cette question et surtout, une certaine confusion peut être produite par un mélange temporel entre les interventions.

II.2.3. Facilités pour retrouver l'information

Chacun peut revoir à tout instant toute information transmise auparavant (qui lui était destinée), ce qui n'est pas le cas dans les face à face habituels où l'on doit prendre des notes et où l'on ne conserve pas en général le libellé exact de l'intervention, ce qui peut provoquer des confusions. A remarquer, toutefois, que cet inconvénient dans le face à face, peut être en partie écarté par un enregistrement sur magnétophone, par exemple.

II.2.4. Rôle du leader dans la résolution des problèmes [FL], [HI]

La quantité et le type d'information transmise par un participant dans une réunion en face à face pour résoudre un problème, dépend fortement du fait qu'il soit considéré comme un leader.

Les plus anciennes études auxquelles on peut se référer à ce sujet datent des années 1950, en particulier celles de French et Bavelas. Ils ont montré que les phénomènes de la vie de groupe sont directement déterminés par les diverses propriétés du réseau de communication. Plus tard, on a montré que la quantité d'information transmise par un participant était beaucoup plus liée au fait qu'il soit choisi comme leader qu'à l'intérêt réel du contenu de ce qu'il transmet. De même, on a constaté que la facilité d'un individu à répondre rapidement ou à interrompre la conversation, est un facteur très important et de plus une caractéristique individuelle stable (L.V.R. Latency of Verbal Response).

Ce qui est important est donc qu'un individu très rapide dans la conversation, même n'étant pas le plus intelligent, peut tendre à dominer la discussion et la prise de décision dans un petit groupe.

On peut penser que les réunions assistées par ordinateur supprimeront en grande partie l'influence du LVR, d'une part parce que plusieurs personnes peuvent transmettre de l'information simultanément, d'autre part parce que chacun est isolé et peut répondre quand il le veut à une question. Celui qui répond après un certain temps ne sera pas forcément écrasé par celui qui répond immédiatement. Dans ce cas, l'intérêt et l'exactitude de l'information seront fortement corrélés avec le rôle de leader dans la discussion et la prise de décision au sein de petits groupes. Les sociologues pensent d'ailleurs que la mise en évidence d'un leader sera beaucoup moins facile que dans les face à face. Enfin, les interventions anonymes permettent à chacun de s'exprimer sans s'exposer personnellement.

Afin d'illustrer ces différences, nous donnons sur un même graphique, les courbes donnant le pourcentage de messages transmis par les participants en fonction de leur rang (de celui qui transmet le plus vers celui qui transmet le moins), dans une conférence assistée par téléphone et une conférence assistée par ordinateur. [VA2]

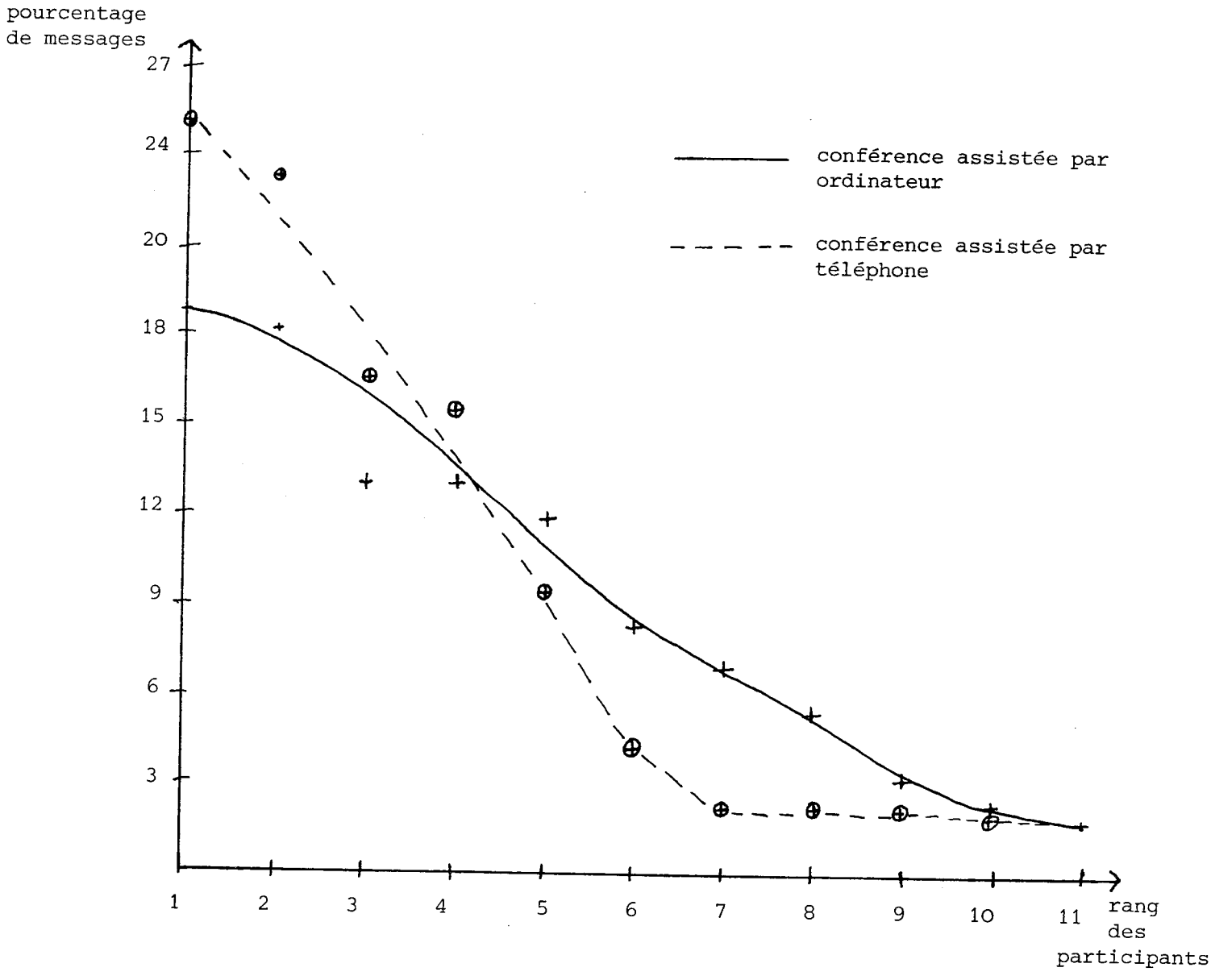


Figure 1

	Réunion	Conférence	Téléphone	Téléphone + télévision	Réunions assistées par ordinateur
Interaction	forte	faible	bonne	bonne	forte
Taille des interventions	petite à grande	grande	petite à grande	petite à grande	petite à moyenne
Compte rendu	possible	existe	possible (bande magnétique)	possible (bande magnétique)	Oui
Services ordinateur	non	non	non	non	oui
Contenu émotionnel	haut	haut	moyen	moyen	faible
Anonymat	non	non	non	non	oui
Sensibilité aux pannes	non	non	faible	faible	oui
Nécessité de taper à la machine	non	non	non	non	oui
Taille du groupe	petit à moyen	petit à grand	petit	petit	petit à moyen
Principaux coûts	. voyages . perte de temps	. voyages . hotels .. . perte de temps	. communications	. communications	. communications . utilisation de l'ordinateur
Autres caractéristiques	. information de et pour chacun . Effets psychologiques importants (leader ..)	. information d'un seul vers beaucoup	. Information de et pour chacun . Effets psychologiques importants (leader ..)		. Information de et pour chacun . Effets psychologique dans le sens rapport avec un leader minimisé . Isolement du participant

Comparaisons entre divers moyens de communication

II.3. SYSTÈME ORIENTÉ UTILISATEURS

Il s'agit de concevoir un système qui soit utilisable par des personnes venant d'horizons divers. Il a fallu, au départ, faire un certain nombre de choix, dont certains ont été abordés au chapitre précédent. On verra que ces choix ont été discutés à l'utilisation, ce qui était un des buts du projet.

II.3.1. Dispositifs de dialogue

Les dispositifs de dialogue devront permettre à l'utilisateur de construire des dessins et participer à la réunion. Comme on le verra plus loin, les dispositifs de dialogue associés aux matériels graphiques sont nombreux. Pour les consoles alphanumériques, seul en général un clavier alphanumérique est disponible. Celui-ci offre l'avantage par rapport aux matériels plus sophistiqués, d'être bien connu des utilisateurs non-informaticiens.

Bien que nous soyons amenés à faire un choix (chapitre III), celui-ci ne sera fait qu'au moment de l'implémentation proprement dite du système. Pendant toute la description de la conception du système, on raisonnera en termes d'outils logiques (on dit, par exemple, qu'il existe un outil logique de désignation d'un point sur l'écran : le dispositif physique associé peut être un photostyle, un manche à balai, ou une simulation par logiciel).

II.3.2. Transmission de dessins

Jusqu'à présent, les seuls systèmes existants ne permettaient de transmettre que des lignes de texte. On a cherché à ajouter la transmission de dessins afin d'enrichir la conversation. Les choix au niveau des dessins devaient être faits d'une part sur leur forme, d'autre part sur les méthodes de construction.

Etant donné que les dessins apparaissant dans les réunions sont en général schématiques, il nous a semblé qu'on pouvait les définir comme des figures (ou des textes) reliées ou non par des lignes droites ou brisées.

Un dessin est donc composé de :

. figures : figures géométriques prédéclarées (CARRE, CERCLE, LOSANGE, ..) ou construites segment par segment par l'utilisateur. Un nom peut être associé à une figure.

. textes : chaînes de caractères alphanumériques.

. liens : lignes droites ou brisées, avec ou sans flèche, joignant deux figures, deux textes, ou une figure et un texte (les textes sont des figures particulières).

Dans toute la suite, on désignera par OBJET les éléments de base d'un dessin, c'est-à-dire une FIGURE, un TEXTE ou un LIEN.

Méthode de construction

On définira un langage de commande pour construire et modifier un dessin. On utilisera un dispositif de désignation d'un point de l'écran pour placer les figures et les textes. Les liens seront créés automatiquement en donnant, par exemple, les noms des figures (ou textes) à relier.

Le participant disposera également d'un "bloc note" (FICHIER PERSONNEL) où il pourra conserver puis réafficher des dessins. Ces mémorisations et appels se feront en donnant un nom au dessin. Cela semble intéressant lorsque les dessins sont complexes, auquel cas on peut les construire par morceaux ou lorsqu'il s'agit d'une conférence, auquel cas le conférencier peut préparer une suite de dessins qu'il pourra ensuite transmettre dans l'ordre désiré.

Tout ce qui concerne les méthodes de construction sera expliqué dans le chapitre III.

II.3.3. Réunion minimale

Une réunion minimale est une réunion au cours de laquelle les participants ne se transmettent que des lignes de caractères (pas de dessins). Si on la mentionne ici, c'est parce que l'on verra que la construction et la modification de dessins peuvent être assez complexes, au moins pour un utilisateur non informaticien qui serait de plus novice dans l'utilisation du système.

En revanche, s'il ne transmet que des lignes de caractères, l'apprentissage du système est extrêmement simple (le participant peut d'ailleurs *recevoir* les dessins puisque cela ne demande aucune intervention de sa part).

II.3.4. Modes de transmission

Les participants se transmettent des INTERVENTIONS. Une intervention peut être :

- un DESSIN (cf. § II.3.1.),
- un COMMENTAIRE : une ou plusieurs lignes de caractères.

Etant donné que la présence d'un ordinateur dans la boucle de communication le permet, on a prévu des transmissions publiques ou privées, anonymes ou signées. Les interventions publiques sont destinées à tous les participants ; les transmissions privées ne sont destinées qu'à un participant. Si une intervention est signée, chaque récepteur connaît le nom de l'émetteur ; si elle est anonyme, ce nom n'est dévoilé à personne.

Un participant pourra transmettre des interventions à des participants présents ou non, à n'importe quel moment et indépendamment de ce que peuvent faire les autres participants présents.

II.3.5. Retours dans la conversation

D'une manière générale, les réunions étant asynchrones, il faut donner aux participants la possibilité de revenir sur des interventions précedemment transmises, ne serait-ce que celles qui lui ont été envoyées durant son absence. On mettra donc à la disposition des participants un petit langage très simple lui permettant d'afficher n'importe quelle intervention.

II.3.6. Réunions synchrones (réunions "normales")

On a voulu (et c'était d'ailleurs l'idée initiale) réaliser un système de réunion où une partie du logiciel tienne lieu de président de séance. Dans ce type de réunion, tous les participants sont présents et une seule personne peut transmettre à un instant donné. Le système s'assure donc qu'une seule personne a la parole à chaque instant, gère les demandes de parole, donne et retire le droit de transmettre, etc ..

On remarque que l'on n'utilise pas toutes les facilités offertes par l'ordinateur (conservation de l'information → asynchronisme), mais ce type de réunion devrait permettre des études de groupe dont tous les éléments sont sur un pied d'égalité avec des possibilités telles que anonymat, interventions publiques ou privées.

II.3.7. Autres problèmes

On aborde dans ce paragraphe quelques idées qui peuvent sembler intéressantes du point de vue utilisateur, ou qui permettent de mieux comprendre les particularités des systèmes de réunions assistées par ordinateur.

. Temporisation :

lors d'une réunion asynchrone, il peut y avoir un certain nombre de personnes présentes à un instant donné. Supposons qu'en quelques secondes un participant A reçoive plusieurs interventions venant d'autres participants présents. Si A dispose d'un terminal à papier, les interventions s'afficheront les unes à la suite des autres et il pourra les lire. En revanche, s'il dispose d'un terminal à écran, il risque de ne pas avoir le temps de lire certaines interventions.

Il semble donc que l'on doive imposer qu'une intervention reçue reste obligatoirement affichée un certain temps. On rejette cette solution, car d'une part ce temps doit être variable suivant la complexité de l'information (et de la facilité de compréhension du participant) et d'autre part parce que cela risque de retarder fortement la réception d'autres interventions. (La seule solution à peu près valable serait de laisser au participant le soin de préciser à quel moment il est disponible pour d'autres interventions).

Afin de minimiser ce problème, on essaiera de laisser affichées à un moment donné le plus grand nombre possible d'interventions parmi les plus récentes. On rappelle également que le participant peut faire un retour dans la conversation pour revoir une intervention.

. Réunions simultanées :

plusieurs réunions pourront exister à un instant donné. Un participant inscrit à plusieurs réunions simultanées devra évidemment reconnaître à quelle réunion appartient telle ou telle information. Il semble que la solution la plus simple (et qui engendre le moins de confusion possible pour un utilisateur) soit la suivante :

- il peut être inscrit à plusieurs réunions,
- à un instant donné, il ne peut être considéré comme présent que dans une seule réunion (donc toute l'information qu'il peut accéder dépend de cette réunion). Pour participer activement à une autre réunion, il est obligé de quitter celle à laquelle il est présent.

. Date et heure d'émission :

comme on l'a vu au début de ce chapitre (II.1.2.), une certaine confusion sur l'ordre d'émission des interventions peut intervenir. C'est pourquoi, lors de tout affichage d'intervention, ou de tout retour dans la conversation, on précise la date et l'heure auxquelles cette intervention a été transmise.

II.4. ASPECTS GÉNÉRAUX DE RÉALISATION

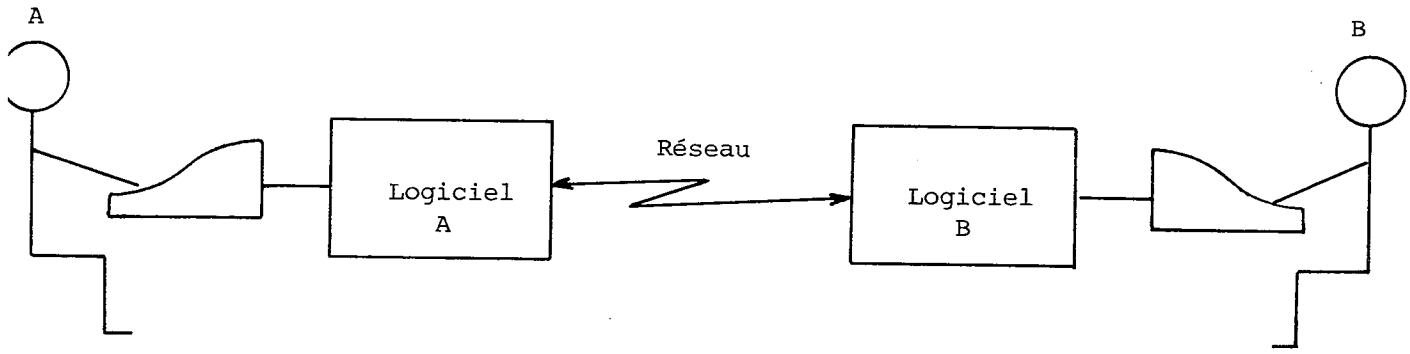
On va tout d'abord expliquer les différentes méthodes que l'on pourrait utiliser pour réaliser un tel système, puis on dégagera des lois qui nous serviront pour le choix du matériel et du logiciel.

II.4.1. Méthodes de réalisation

Ce que l'on entend par méthode de réalisation est une considération d'ordre général sur la façon de placer le logiciel dans le réseau.

II.4.1.1. Logiciel "réparti"

Etant donné que les utilisateurs sont sur des sites locaux, on peut considérer que, pour permettre à A et B (personnes physiques) de communiquer entre eux, la première idée est de fournir à chacun, sur son site, un logiciel de communication. Cela donne le schéma suivant :



. En premier lieu, il est probable que les logiciels A et B seront identiques, ce qui veut dire que l'on duplique de l'information (si ce sont les mêmes calculs, sinon on réécrit pour B dans un autre langage et sous une autre forme ce que l'on a déjà écrit pour A).

. Il faudra de toutes façons centraliser une partie du logiciel, ceci afin de savoir quelles sont les personnes présentes (sinon on est obligé de transmettre des messages vers chacun afin de savoir s'il est présent), d'éviter qu'une partie du logiciel A ne soit toujours active (ce qui serait obligatoire, car même si A est absent, il peut recevoir de l'information), etc ..

. De plus, on sait que dans tout réseau efficace, on peut accéder, à partir d'un terminal situé sur le site B, au calculateur situé sur le site A, le réseau se chargeant des problèmes de communication.

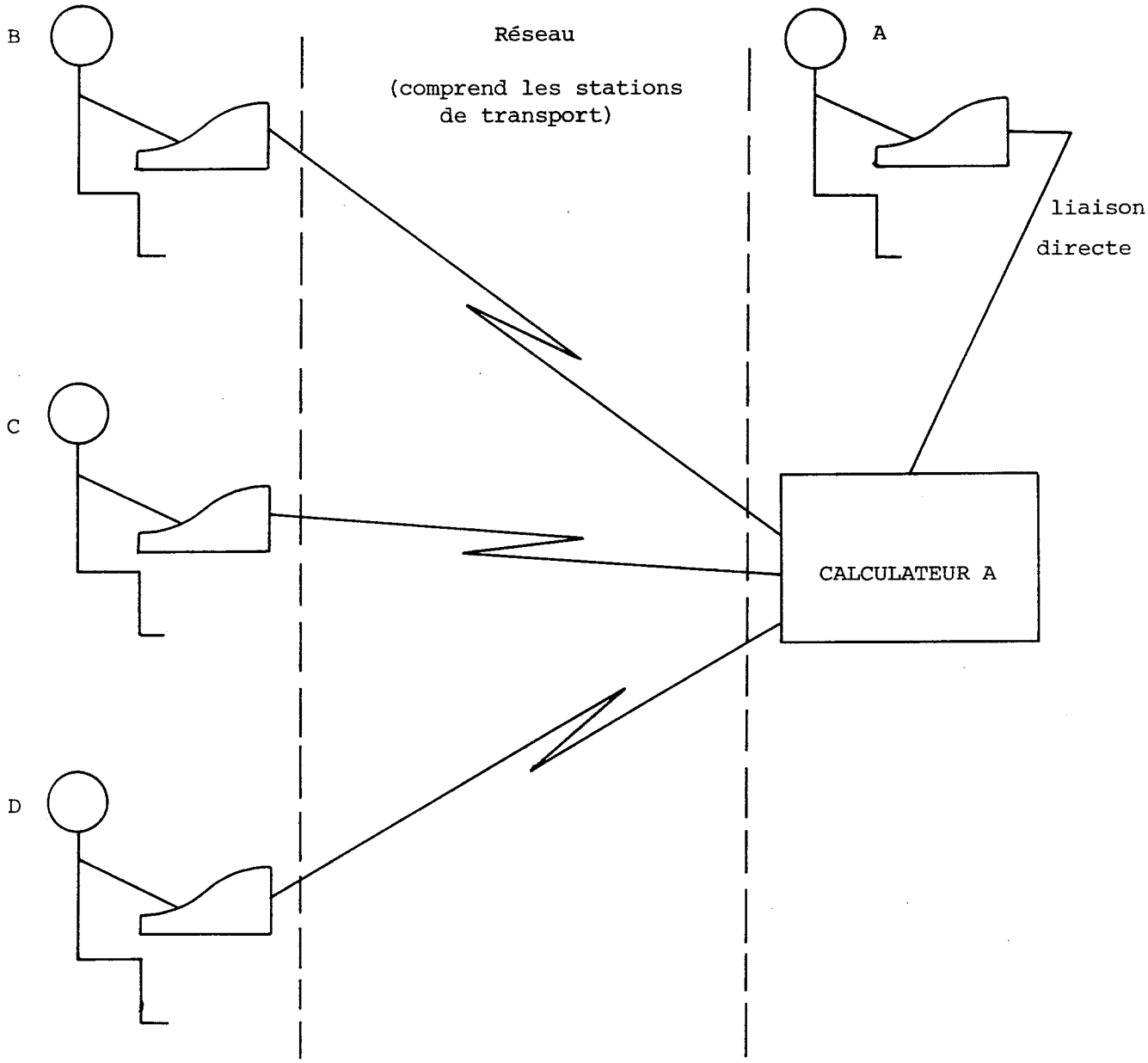
Pour toutes ces raisons, on rejette cette solution pour adopter celle qui suit.

II.4.1.2. Logiciel centralisé

On verra dans le paragraphe II.4.4. quelles sont les caractéristiques que l'on demandera au calculateur A choisi.

On peut faire une remarque sur le type de réseau utilisé : il est évident qu'un réseau général d'ordinateurs mettant à la disposition d'utilisateurs distants les facilités offertes sur un site donné, conviendra très bien à ce type de système, mais on peut remarquer qu'un réseau orienté terminaux (tel que TYMNET1) conviendrait tout aussi bien.

Logiciel centralisé



Remarque :

Les problèmes posés par les systèmes de réunion par ordinateurs concernent pour une bonne partie la gestion des données. On verra dans le chapitre IV que ces données seront centralisées. Pour certaines réalisations de tels systèmes, on pourrait envisager une répartition des données. Nous pensons que cette solution était mauvaise dans notre cas, d'une part parce que le logiciel sera centralisé, d'autre part parce que les bases de données réparties constituent un sujet mal connu. On pourra se reporter pour plus d'information à [CH2], [LB], [SE].

II.4.2. Organisation des fichiers

Si l'on se place à un niveau logique, c'est-à-dire en se libérant des caractéristiques de la structure technologique du support, on peut caractériser un fichier par :

- une organisation logique définissant un ensemble d'objets élémentaires,
- la ou les fonctions d'accès aux composants du fichier,
- les opérations possibles : lecture, écriture,
- une durée de vie.

Un système de communications assistées par ordinateur est essentiellement basé sur l'existence de fichiers lui permettant de conserver et de retrouver à tout moment de l'information. Si l'on prend comme exemple le fichier de conversation, qui contient à un instant donné tout ce qui a été transmis, on voit que quatre problèmes se posent :

- organisation,
- modalités d'accès,
- protection,
- durée de vie.

Ces problèmes se poseront également de façon plus ou moins simplifiée pour les autres fichiers.

II.4.2.1. Organisation du fichier

Cette organisation dépendra du rôle du fichier. Le fichier contenant tout ce qui a été transmis pourra être organisé par interventions, un fichier contenant des dessins étant organisé par dessins. On pourra lire et écrire dans tous les fichiers, mais, par exemple, on ne pourra écrire dans le fichier de conversation qu'en fin de fichier (zone libre).

II.4.2.2. Modalités d'accès

Considérons l'ensemble U = {Ui} des utilisateurs pouvant accéder à ce fichier. Soit Uk et Up deux utilisateurs voulant écrire une intervention dans le fichier de conversation. Il faut que Up attende que Uk ait terminé l'écriture avant de pouvoir écrire lui-même, afin de conserver chaque intervention séquentiellement.

Ce fichier devra donc être considéré comme une ressource critique. Ce problème pouvant se poser pour d'autres ressources, nous donnons brièvement les principales solutions. Pour plus de renseignements à ce sujet, on pourra se reporter aux nombreux ouvrages traitant de la question, en particulier [CR].

. Attente active :

une variable v accessible aux processus prend les valeurs 1 ou 0 suivant que la ressource critique est occupée ou non. Un processus Uk consulte v pour pouvoir entrer dans la section critique et remet v à 0 en sortant. Un processus bloqué sur v boucle sur l'instruction de test. Cette solution n'est donc acceptable que si le phénomène d'exclusion mutuelle est rare et que la boucle de test dure peu de temps.

. Verrous :

un verrou est une variable v à laquelle on associe une file d'attente A(v). v est initialisé à 0, ce qui indique qu'aucun processus ne se trouve dans la section critique. Si un processus ne peut entrer dans la section critique (v = 1), il est mis dans la file A(v). Si un processus sort de la section critique, un des processus de la file d'attente est activé si la file A(v) n'est pas vide, sinon on met v à 0.

. Les sémaphores :

un sémaphore s est constitué de :

- une variable entière e(s), positive, négative ou nulle, initialisée à une valeur positive ou nulle,
- une file d'attente A(s) initialement vide.

Les deux primitives P et V suivantes sont les seules pouvant agir sur un sémaphore :

P(s) e(s) := e(s) - 1
si e(s) < 0 on met le processus dans la file A(s)

V(s) e(s) := e(s) + 1
si e(s) $\leq$ 0 on sort un processus de la file A(s) et on l'active.

L'exclusion mutuelle se résout alors comme suit : soit s un sémaphore avec e(s) initialisé à 1. Chaque processus s'exécute suivant le schéma :

P(s) ;
section critique
V(s)

ce qui assure qu'un processus au plus peut se trouver dans sa section critique et que s'il n'y en a pas, aucun autre ne peut être bloqué derrière le sémaphore d'exclusion mutuelle.

Remarque :

Les solutions précédentes ne garantissent pas qu'un processus entre dans sa section critique au bout d'un temps fini, suivant les priorités définies dans la file d'attente (problème résolu si la file d'attente est gérée en FIFO par exemple).

II.4.2.3. Protection

Le problème de la protection au sujet des fichiers se pose à deux niveaux :

. protection globale sur un fichier : par exemple un utilisateur Ui ne pourra ni lire ni écrire sur un fichier appartenant en propre à un utilisateur Uk ;

. protection au niveau objet : une intervention dans le fichier de conversation qui est destinée uniquement à l'utilisateur Ui ne pourra être lue par un autre participant.

Se posera donc un problème de protection que l'on peut exprimer de façon générale par :

- un ensemble d'objets R = {ri} qui sont les éléments à protéger,
- un ensemble d'utilisateurs U = {uj} qui sont les entités capables d'accéder aux objets ri.

On appelle DROITS ("capabilities") d'un utilisateur Ui sur un objet Rj l'ensemble des modes d'accès à Rj pour Ui. On définit une matrice des droits dont chaque élément Mij indique les droits de Ui sur Rj (ε = aucun chemin de Ui vers Rj ; λ = interdiction d'accéder de Ui à Rj)

Exemple :

	fichier 1	fichier 2	fichier 3
U1	lire	écrire lire	λ
U2	lire écrire	ε	lire écrire

Les exemples les plus connus de l'implantation de la matrice des droits sont les suivants :

. Liste d'accès :

c'est une liste des utilisateurs ayant des droits associés à chaque objet Ri. Chaque élément est de la forme (u, M(r,u)). Cette liste devant être consultée à chaque tentative d'accès à l'objet Ri ; elle n'est valable que si la liste est courte. Diverses méthodes permettent de la raccourcir (groupement des utilisateurs ayant les mêmes droits, ..).

. La liste des droits :

contrairement à la liste d'accès qui est associée à un objet Ri, la liste des droits est définie pour chaque utilisateur Ui. Elle indique, pour chaque objet auquel il peut accéder, les droits d'accès correspondants. Elles peuvent être raccourcies par des méthodes similaires à celles utilisées pour la liste des droits et lorsqu'elles sont trop longues, restructurées en sous-listes (sous-liste correspondant à une classe d'objets, ..). L'avantage de la liste des droits est qu'elle n'est employée que lorsque l'utilisateur associé est actif, d'où l'introduction de mécanismes d'accès plus rapides (mémoire associative, ..).

. Clés et verrous :

chaque utilisateur (regroupés si possible) dispose d'une clé (un code). Chaque objet (regroupés si possible) possède un verrou. Il y a une clé et un verrou par mode d'accès possible. L'utilisateur Ui ne pourra accéder à l'objet Rj que si sa clé permet d'ouvrir le verrou associé à l'objet Rj.

II.4.2.4. Durée de vie

La durée de vie d'un fichier dépend de son rôle dans le système. On peut, par exemple, considérer en première approximation que le fichier de conversation a une durée de vie au moins égale à la durée de la réunion correspondante. Ces problèmes de durée de vie seront explicités fichier par fichier dans le chapitre IV.

II.4.3. Description de deux systèmes

On donne ici les grands choix qui ont été faits pour deux systèmes :

. un système basé sur la méthode de Delphes,

. un système de réunions.

II.4.3.1. Système basé sur la méthode de Delphes

Il s'agit d'un système implémenté au début des années 1970 par Thomas W. Hall [HA]. Ses caractéristiques d'utilisation sont les suivantes :

. chacun peut accéder au contenu d'un fichier comprenant essentiellement les propositions présentées par les participants ;

. chacun peut ajouter des propositions ou exprimer des opinions ;

. chacun peut voter par oui ou par non sur une proposition.

Les choix qu'il a faits pour l'implémentation du système sont les suivants :

. le programme est écrit en XBASIC (extension de BASIC) sur un UNIVAC 1108 sous l'operating system standard EXEC8, dont les particularités qui nous intéressent sont d'être un système conversationnel à partage de temps accessible à distance ;

. les données de chaque conférence sont stockées dans deux fichiers dont les noms et l'accès sont contrôlés directement par le système (de façon invisible pour l'utilisateur) :

- le fichier des votes contient toutes les informations relatives aux votes des participants pour chaque proposition,

- le fichier de conférence contient toutes les propositions.

La gestion de ces fichiers est la suivante :

- afin de donner une limite supérieure à la place occupée par ces fichiers, on ne conserve que les 50 plus récents enregistrements. Les enregistrements sont donc placés circulairement dans le fichier, le 51ème prenant la place du premier, etc ..

- les conflits d'accès à une zone commune sont résolus par un moyen fourni par EXEC8 : on peut demander l'accès exclusif à un bloc d'un fichier, le verrouiller et si quelqu'un d'autre tente d'y accéder, sa demande est rejetée jusqu'au déverrouillage. Ainsi, quand un utilisateur veut ajouter un enregistrement, il verrouille le premier bloc libre, ajoute son enregistrement, met à jour le pointeur de bloc libre, puis déverrouille cet accès. Il est évident que plus les verrouillages sont longs et nombreux, plus les temps de réponse aux autres participants sont longs. Afin de remédier à cet inconvénient, Hall propose :

= chaque fonction de verrouillage est distincte des autres : par exemple, quelqu'un qui ajoute un enregistrement ne verrouille que ceux qui veulent faire la même chose ;

= toute la préparation est faite avant le verrouillage, afin de minimiser ce temps de verrouillage ;

= aucun verrouillage n'est effectué pour une lecture.

II.4.3.2. Un système de réunions

Il s'agit du système FORUM implémenté vers les années 1975 à l'Institute for the Future (USA). Son fonctionnement externe est le suivant :

. discussion : des blocs de texte sont entrés et distribués aux participants destinataires,

. proposition et vote.

Les aspects essentiels de réalisation sont les suivants :

. FORUM a été implémenté sur un PDP 10, sous le système à partage de temps TENEX, accessible à distance par l'intermédiaire du réseau général d'ordinateurs ARPANET. Les programmes sont écrits entièrement en langage d'assemblage, ce que ses concepteurs jusfifient essentiellement par le fait que c'est un système orienté entrées/sorties.

. La structure de donnée est composée de trois fichiers dont la gestion, tout comme dans le système décrit précédemment, est interne à FORUM, donc invisible pour l'utilisateur :

- un fichier contenant toutes les interactions,
- un fichier contenant toute l'information ayant été transmise durant la conférence,
- un fichier de contrôle contenant toute l'information utile sur chacun des participants.

. Pour règler les conflits d'accès à une zone commune, essentiellement l'écriture dans la partie libre d'un nouvel enregistrement, la solution adoptée consiste à :

- le premier mot de chaque fichier est utilisé comme pointeur vers la première place libre dans ce fichier ;
- avant de pouvoir écrire dans un fichier, toute l'information est stockée dans un buffer (intervention complète) ;
- l'espace nécessaire pour stocker cette information est alors réservé en ajoutant la longueur de l'entrée à faire au mot pointant vers la zone libre (une instruction machine) ;
- l'information peut être copiée sans danger qu'un autre essaye d'écrire dans la même zone puisqu'elle est déclarée non libre.

II.4.3.3. Conclusion

La description générale des choix faits dans ces deux systèmes impose quelques remarques :

. il semble que le choix de l'assembleur comme langage unique, ne soit pas entièrement justifié. Un langage de haut niveau pouvant communiquer avec l'assembleur aurait les mêmes avantages et de plus permettrait un contrôle d'erreurs plus facile et des possibilités telles que traitement de chaînes, modularité, ..

. Les solutions apportées aux problèmes des conflits d'écriture sont différentes dans leur conception. Celle proposée par T.W. HALL est classique, alors que celle utilisée dans FORUM est plus originale, mais très liée au fait que l'on ne peut écrire que dans un bloc libre (aucune modification sur une zone déjà remplie).

. Les langages de commande sont très simples, mais aucune transmission de dessins n'est autorisée (uniquement des chaînes de caractères).

II.4.4. Aspects généraux de réalisation

Cet ensemble de lois générales représente une approche du problème au point de vue de l'implémentation, cherchant surtout les avantages qu'apportent certains types de systèmes. Elles devront être prises en compte pour le choix des outils tant matériels que logiciels qui sont présentés dans le chapitre IV.

Loi 1 : utilisation d'un système conversationnel à partage de temps

A un moment quelconque d'une réunion, il peut y avoir zéro, un ou plusieurs participants présents. La première évidence est donc de ne pas réserver une machine simplement pour dérouler les programmes d'un logiciel qui risquent d'être la plupart du temps en état d'attente d'interruptions. On implémentera le logiciel de réunion comme sous-système d'un système à partage de temps. Le seul désavantage de cette méthode consiste à observer des variations de temps de réponse dues à la charge variable du système à partage de temps. En revanche, cela comporte deux autres avantages fondamentaux :

. on peut utiliser toutes les possibilités offertes par ce système d'exploitation,

. les allocations de quanta de temps aux participants sont faites directement par le système à partage de temps et le système de réunion est donc déchargé de ce travail.

Loi 2 : le système à partage de temps doit être accessible par un réseau

Etant donné que les participants peuvent être éloignés géographiquement, il est nécessaire qu'ils puissent accéder à des consoles reliées via un réseau au système à partage de temps choisi avec un nombre de portes aussi grand que possible.

Loi 3 : On peut écrire le logiciel dans un langage de haut niveau pouvant communiquer avec l'assembleur

Le système est orienté entrées/sorties et gestion de fichiers. L'assembleur permet de réaliser dans tous les cas les fonctions nécessaires. En revanche, l'utilisation d'un langage de haut niveau se justifie du fait d'une plus grande facilité d'écriture, de recherche d'erreurs et d'utilisation de packages (§ IV.2.2). *Il suffit donc* de choisir ce langage de haut niveau tel qu'il puisse communiquer avec le tout puissant assembleur.

Remarque :

il est évident que l'on pourra écrire tout le système en assembleur ou dans un langage tel que PL1.

II.5. PROBLÈMES TECHNIQUES SPÉCIFIQUES AU SYSTÈME CONGRES

On ne parle ici que des problèmes essentiels découlant immédiatement de la définition des buts généraux (cf. chapitre I). Ces problèmes sont donc liés au système CONGRES, les choix plus généraux étant explicités dans le paragraphe II.4.4 . On a vu, dans l'introduction, que l'étude globale du projet peut se décomposer en deux niveaux : un niveau graphique et un niveau réunion. Les problèmes vont se poser de manière différente à ces deux niveaux.

II.5.1. Niveau graphique

Il faut élaborer un système graphique permettant la construction, la modification, la mémorisation et l'appel de dessins. On rencontre un problème de spécification d'un système et du langage de commande associé. Un problème d'un autre ordre se pose du fait de la diversité des consoles utilisées.

II.5.1.1. Programmation graphique

Le langage graphique utilisé par un terminal graphique est en général de type très élémentaire, souvent limité à l'affichage d'une suite de points. On définit, quand cela est nécessaire, un langage de niveau immédiatement supérieur comportant des instructions de type "tracé de vecteur" et "tracé de caractères" (des générateurs de caractères et de vecteurs existent souvent sur les terminaux graphiques).

C'est ce langage que nous considérerons comme le "langage graphique de base" (LAB) ; il devra comporter :

- tracé de vecteurs et de points,
- affichage de caractères,
- fonctions de communication (entrée de caractères au clavier au minimum).

Le LAB peut exister sur les terminaux graphiques, soit qu'il ait été fourni par le constructeur, soit qu'il ait été développé au sein d'équipes (cf. GRIGRI au chapitre IV), mais il sera à définir pour certains terminaux

Le langage externe manipulé par le participant sera donc un langage graphique de niveau supérieur au LAB.

II.5.1.2. Structure de données

Les problèmes de représentation de données se posent pour nous essentiellement au niveau des figures et de contraintes entre ces figures (liens). Il existe une structure de données liée au LAB qui diffèrera suivant les terminaux (exemple : tableau de caractères sur un terminal alphanumérique) et une structure de données à définir pour la partie graphique du système CONGRES.

Les structures d'image les plus répandues sont les suivantes :

. Suite d'instructions :

une image est formée de la suite d'instructions permettant de la construire ; cette structure, de type linéaire, impose de recréer la suite lors d'une modification. Cette structure sera rejetée du fait des nombreuses possibilités de modification de dessin que l'on offrira à l'utilisateur et de la difficulté à résoudre les problèmes de contraintes entre figures.

. Structure de liste :

chaque élément comporte l'adresse de l'élément suivant et souvent, pour permettre de remonter au niveau supérieur, l'adresse de l'élément précédent. Ce type de structure, bien que commode pour les suppressions ou les adjonctions d'éléments, a pour défaut de nécessiter une place assez grande et d'imposer des recherches assez longues.

. Structure en table :

ce type de structure, bien que manquant de souplesse, peut être intéressant, car il permet de gagner de la place et le devient encore plus lorsqu'on a la possibilité d'utiliser des sous-programmes. L'image sera formée par des appels de sous-programmes à l'aide d'un minimum de renseignements contenus dans les tables. On peut également résoudre, à l'aide de tables de corrélation, certains problèmes de contraintes : lorsqu'une certaine contrainte doit être appliquée à une image, on met l'adresse de cette image dans une table de corrélation associée à cette contrainte. Ceci permet de retrouver aisément toutes les images soumises à une contrainte donnée.

II.5.1.3. Codage

Les figures prédéclarées sont des figures géométriques. Le codage le plus simple est donc un codage géométrique, une figure étant définie par :

f(NO,X,Y,R)

NO représentant un type de figure, donc un appel à un sous-programme,
X,Y représentant les coordonnées du centre de la figure,
R, que l'on nommera rayon dans la suite, étant en fait la longueur du côté du plus petit carré contenant la figure ("taille" de la figure).



Le passage de ce codage à la représentation segment par segment pour l'affichage réel d'une figure, est donc très simple en général : par exemple, pour un rectangle vertical tel que représenté dans la figure ci-dessus, le codage consiste à remplir un tableau de cinq points constituant les extrêmités d'une suite de segments jointifs :

1er point : X(1) = X + R/2
Y(1) = Y + R

2ème point : X(2) = X - R/2
Y(2) = Y + R

3ème point: X(3) = X - R/2
Y(3) = Y - R

4ème point : X(4) = X + R/2
Y(4) = Y - R

5ème point: X(5) = X + R/2
Y(5) = Y + R

Le premier et le cinquième point sont identiques, puisque c'est un ensemble de segments fermés.

Le passage du codage d'un cercle à sa représentation géométrique réelle est plus complexe, mais bien connu.

Le codage des figures déclarées segment par segment ne pourra pas se faire à partir d'une fonction de (NO,X,Y,R) car ces figures n'ont pas forcément des propriétés géométriques simples. Il faudra conserver dans le codage les coordonnées de toutes les extrêmités des segments. Le codage est de la forme :

f(NO,X,Y,R,D)

NO indique qu'il s'agit d'une suite de segments,
X,Y sont les coordonnées de l'origine du premier segment,
R représente le nombre de segments,
D est un déplacement dans une liste de points.

Le passage de ce codage à la représentation géométrique segment par segment est donc extrêmement simple.

Le codage d'un texte se fera à partir d'une fonction de la forme :

f(NO,X,Y,R,D)

NO indique qu'il s'agit d'un texte,
X,Y sont les coordonnées du premier caractère,
R représente le nombre de caractères,
D est un déplacement dans une suite de caractères.

La représentation réelle fait donc appel à une procédure d'affichage de caractères.

Les liens représentent une contrainte entre deux figures, deux textes, ou un texte et une figure ; le codage d'un lien consiste donc à donner le déplacement des figures concernées et des renseignements relatifs au type de lien à tracer (ligne droite ou brisée). Le codage est de la forme :

L(NF1,NF2,X1,Y1)

X1,Y1 représentent l'existence d'une barre horizontale ou verticale par laquelle passe le lien (cf. III.2.1.3.) X1 → barre verticale, Y1 → barre horizontale (l'un des deux au moins est nul) ; on verra dans le chapitre IV la méthode utilisée pour créer les liens à partir de ces renseignements.

Les transformations qui nous intéressent en particulier sont les suivantes :

. translation (ΔX, ΔY)

- figure prédéclarée

  f(NO,X,Y,R) → f(NO,X + ΔX, Y + ΔY, R)

- figure déclarée par segments

  f(NO,X,Y,R,D) → f(NO, X + ΔX, Y + ΔY, R, D)

  pour I = D jusqu'à D + R faire

  LISTX(I) = LISTX(I) + ΔX

  LISTY(I) = LISTY(I) + ΔY

- texte

  f(NO,X,Y,R,D) → f(NO, X + ΔX, Y + ΔY, R, D)

Remarque :

on n'a rien à modifier dans la table des liens, puisque celle-ci ne comprend que les déplacements des figures et non leurs caractéristiques (il suffit d'effacer et de réafficher ceux qui étaient attachés à la figure ou au texte ayant subi une transformation).

. Augmentation de taille d'une figure prédéclarée

f(NO,X,Y,R) → f(NO,X,Y,R + ΔR)

Remarque :

on n'a rien à modifier dans la table des liens (effacer et réafficher les liens attachés à la figure ayant subi la transformation).

. Translation globale (d'un dessin) :

il suffit de répéter pour toutes les figures et textes l'opération de translation définie plus haut et d'appliquer aux liens :

si X1 ≠ 0 L(NF1,NF2,X1,Y1) → L(NF1,NF2,X1 + ΔX,Y1) sinon

si Y1 ≠ 0 L(NF1,NF2,X1,Y1) → L(NF1,NF2,X1,Y1 + ΔX) sinon rien

. Modification de taille globale (d'un dessin) :

on verra plus loin que, comme on choisit un repère ayant son origine au coin bas gauche de l'écran, le plus simple est d'opérer une homothétie sur tous les objets concernés (on déplace le dessin, mais on a toujours la possibilité de translater) :

- pour les figures prédéclarées

  f(NO,X,Y,R) → f(NO,X*VAR,Y*VAR,R*VAR)

- pour les figures déclarées par segments

  f(NO,X,Y,R,D) → f(NO,X*VAR,Y*VAR,R,D)

  pour I = D jusqu'à D + R faire

  LISTX(I) := LISTX(I)*VAR ;

  LISTY(I) := LISTY(I)*VAR ;

- pour les textes

  f(NO,X,Y,R,D) → f(NO,X*VAR,Y*VAR,R,D)

On n'a toujours rien à changer dans la table des liens pour les noms de figures ; en revanche, il faut modifier les renseignements relatifs au type de tracé :

si X1 ≠ 0 L(NF1,NF2,X1,Y1) → L(NF1,NF2,X1*VAR,Y1) sinon

si Y1 ≠ 0 L(NF1,NF2,X1,Y1) → L(NF1,NF2,X1,Y1*VAR) sinon rien

En prévision de l'utilisation du clavier alphanumérique, il a fallu donner un nom aux objets afin de pouvoir les désigner par la suite. De plus, on autorisera plusieurs modes de tracé, d'où les fonctions définissant les objets.

. Figure prédéclarée

f(ID,NO,X,Y,R,M)

ID = nom de la figure

NO = type de la figure indiquant sa forme (carré, ..),

X,Y = coordonnées du centre

R = taille de la figure

M = mode de tracé.

Remarque 1 :

lors de toute modification sur la figure, il faut appliquer la modification sur ID (sauf pour un changement de taille) et on pourra ajouter une modification de nom.

Remarque 2 :

l'identificateur pourra être placé au centre, à droite, à gauche, au-dessus ou au-dessous de la figure. Les coordonnées du premier caractère de ID seront alors calculables à partir de X, Y et R.

. Figure déclarée segment par segment

f(ID,NO,X,Y,R,D,M)

ID, X, Y, M = idem figure prédéclarée

R = nombre de segments constituant la figure

D = déplacement dans la suite de segments

NO = indique qu'il s'agit d'une figure créée segment par segment.

. Texte

t(NO,X,Y,R,D)

NO = indique qu'il s'agit d'un texte

X,Y = coordonnées du premier caractère du texte

R = nombre de caractères du texte

D = déplacement dans une suite de caractères

. Liens

L(N1,N2,XB,YB,M)

N1, N2 = adresses des figures liées

XB, YB = façon de tracer le lien (direct, brisé)

M = mode de tracé (pointillés, avec ou sans flèche, ..).

Remarque :

On peut homogénéiser les fonctions représentant les figures et les textes par une fonction de la forme :

F(id,NO,X,Y,R,D,M)

en prenant : pour une figure prédéclarée : D = 0

pour un texte : ID = premières lettres du texte.

On verra en détails dans le chapitre IV toutes ces transformations.

II.5.1.4. Indépendance du logiciel par rapport au matériel

Le système devra s'appliquer à des consoles de types aussi différents que possible, puisqu'on peut travailler sur des terminaux graphiques, alphanumériques papier et alphanumériques écran. Le problème de l'indépendance du logiciel par rapport au matériel (logiciel transportable) est important et complexe lorsqu'il est étudié dans sa généralité.

Dans le cadre de notre étude, il est évident qu'il fallait concevoir un logiciel indépendant, à un certain niveau, du matériel, afin de ne pas avoir à tout réécrire pour chaque console.

On peut tout d'abord remarquer que l'indépendance sera nécessaire au niveau de la console de visualisation (et non pas au niveau du calculateur qu'on peut choisir sur le réseau). Les différences entre consoles qui nous intéressent sont : le type d'écran, les dispositifs de communication et le mode de connexion à l'ordinateur.

. Type d'écran :

d'un point de vue externe, les différences essentielles portent sur les modes d'affichage (graphique, alphanumérique), la taille de l'écran et la quantité d'information affichable (nombre de lignes sur des terminaux alphanumériques à écran). On constate que les différences sont grandes d'une part entre les terminaux graphiques et alphanumériques, d'autre part, parmi ceux-ci, entre les terminaux à écran et à papier. On ne s'intéresse pas aux caractéristiques plus sophistiquées telles que luminosité, couleur,.. Au niveau de l'écran, on se posera le problème d'un partage spatial pour permettre à l'utilisateur de préparer des dessins sans être coupé de la conversation.

. Dispositifs de communication :

parmi les dispositifs de communication les plus répandus, on peut citer :

. le clavier de fonction : l'enfoncement d'une touche de fonction provoque une interruption et l'émission d'un code permettant à un programme de déterminer de quelle touche il s'agit ;

. les dispositifs de désignation d'un point sur l'écran, en particulier le photostyle (cellule photoélectrique) et les réticules (deux lignes se croisant sur l'écran) qui permettent également l'utilisation d'un langage de commande par l'intermédiaire d'un réseau ;

. les tablettes : elles permettent d'obtenir des dessins sur l'écran en les traçant sur cette tablette ;

. le clavier alphanumérique : il permet de communiquer avec le calculateur au moyen de messages alphanumériques. Son fonctionnement peut d'ailleurs être rapproché de celui du clavier de fonction, le code étant reconnu par le programme suivant le libellé de la suite alphanumérique transmise : les avantages du clavier alphanumérique sont les suivants :

- il existe sur tous les terminaux d'usage courant,
- il permet d'entrer facilement des paramètres, soit sous forme de caractères, soit sous forme de nombre,
- il permet d'entrer des coordonnées.

Le désavantage essentiel concerne l'identification et la désignation d'un point sur l'écran sans utiliser de coordonnées. Pour l'identification, le problème est résolu si chaque élément accessible possède un nom ; il suffira alors de taper le nom de l'élément pour le désigner.

. Mode de connexion :

la vitesse de transmission des informations entre le terminal et le calculateur influe fortement sur les performances de la console. Etant donné que notre problème se situe dans un environnement réseau, cette vitesse de transmission sera très variable. Par ailleurs, un certain nombre de terminaux, même connectés directement au calculateur, le sont par une ligne lente (110 bauds sous CP). Il faudra tenir compte de cette restriction lors de l'implémentation du système.

Parmi toutes les possibilités qui nous sont offertes, on ne retient qu'un sous-ensemble :

. pour la communication entre l'utilisateur et le terminal, on s'en tient au clavier alphanumérique,

. pour le type d'écran, on l'utilisera dans son ensemble et on considèrera que l'on peut y tracer des points, des vecteurs et des caractères (simulés ou réels).

Etant donnée la grande diversité des matériels utilisés, un certain nombre d'études ont déjà été entreprises pour réaliser des logiciels graphiques interactifs indépendants du contexte d'utilisation, c'est-à-dire indépendants :

. par rapport au matériel,

. par rapport à l'application,

. par rapport au langage hôte,

. par rapport au système d'exploitation.

On peut citer le logiciel GRIGRI, réalisé à Grenoble par l'équipe graphique [LA], [LE], que l'on utilisera d'ailleurs pour les terminaux graphiques 2250 IBM, TEKTRONIX 4010, TEKTRONIX 4015.

On constate que notre problème ne se pose pas de la même manière, d'une part parce que l'on cherche essentiellement à être indépendants du matériel, d'autre part parce que l'on veut que les services du logiciel CONGRES soient accessibles sur les consoles alphanumériques.

Nous allons donner aux logiciels du type GRIGRI le nom de LGB (Logiciel Graphique de Base), ce qui nous intéresse dans ces logiciels étant qu'ils nous permettent, à partir d'un programme, de :

- tracer des vecteurs,
- tracer des points,
- écrire une chaîne de caractères,
- communiquer.

Le fait qu'un logiciel tel que GRIGRI permet d'accéder avec les mêmes fonctions à des terminaux différents, nous intéresse bien sûr car cela simplifie notre tâche, mais on pourrait utiliser un autre logiciel, du moment qu'il nous permet d'accéder aux mêmes types de fonctions. Il faudra d'ailleurs que l'on définisse un LGB pour les consoles alphanumériques. La technique la plus simple sera de créer le LGB sous forme d'une bibliothèque de sous-programmes, méthode qui impose d'utiliser un langage de programmation autorisant la compilation séparée.

Tout le logiciel sera indépendant du terminal utilisé, sauf en ce qui concerne le LGB qui pourra être différent suivant le terminal utilisé et également en ce qui concerne les caractéristiques physiques de l'écran. La méthode utilisée sera décrite plus longuement dans le chapitre V, dans la partie consacrée à ce que l'on appellera une "console virtuelle".

On peut dire, en guise de conclusion, que la diversité des terminaux pose des problèmes :

- au niveau du logiciel, car il faudra le rendre indépendant dans une certaine mesure du terminal utilisé,
- au niveau de l'utilisateur qui subira des contraintes en passant d'un terminal à un autre.

II.5.2. Niveau général

Parmi les problèmes que l'on peut mettre en évidence à partir de la définition des buts du projet (cf. § I.3), on peut distinguer trois groupes :

II.5.2.1. Automatisation

Dans le cadre général d'une réunion asynchrone (et a fortiori dans le cadre d'une réunion synchrone), une partie du logiciel sera chargée de gérer la réunion sur les points suivants :

. départs et arrivées des participants : tenue à jour de la liste des présents, etc ..

. stockage des interventions : afin de permettre des réunions asynchrones, il faut pouvoir stocker et retrouver l'information. Cette information devra être protégée. Cette base de données devra être assez complète afin de retrouver facilement les interventions voulues lors des cas suivants :

- retours dans la conversation demandés par des participants,
- lors d'une arrivée d'un participant, savoir indiquer le nombre d'interventions qui lui étaient destinées durant son absence et les retrouver.

II.5.2.2. Transmissions

Les participants peuvent se transmettre des commentaires et des dessins. Les transmissions de dessins posent un certain nombre de problèmes, en particulier au niveau du codage et décodage de l'information qui transite et au niveau de l'affichage des interventions reçues. L'affichage d'interventions reçues nécessitera la mise en place d'une structure de données relativement simple puisque les interventions affichées ne sont pas modifiables directement (on verra qu'elles peuvent l'être par d'autres méthodes). Cependant, cela pose un certain nombre de problèmes, d'une part à cause des différences éventuelles entre la console émettrice et la (ou les) console réceptrice, d'autre part à cause du postulat qui veut que l'on conserve un maximum d'interventions affichées à un instant donné.

II.5.2.3. Réunions synchrones automatisées

Dans ce type de réunions, en plus des problèmes déjà vus de transmission, se posent les problèmes d'automatisation du déroulement d'une réunion. Une partie du logiciel doit s'assurer, par exemple, qu'une seule personne peut transmettre des interventions à un instant donné. Cela imposera de toute évidence des vérifications, éventuellement des time-out, etc ..

CHAPITRE III

POINT DE VUE UTILISATEUR

Ce chapitre décrit d'une façon générale ce que peut faire un participant dans une réunion. On pourra se reporter, pour une plus grande précision, en particulier sur la syntaxe des commandes, aux notices utilisateurs [GA3] et [GA4]. Lorsque cela n'est pas explicitement infirmé, tout ce qui est décrit est implémenté et testé.

Les dessins qui illustrent ce chapitre ont été réalisés sous CONGRES, au moyen d'une procédure permettant d'obtenir au BENSON une copie de l'écran du 2250 IBM [ME]. On pourra noter certaines différences dans les dessins, particulièrement sur les liens, différences qui viennent du fait que deux versions de création de liens que l'on explicite dans le chapitre IV, ont été utilisées (elles sont toutes deux opérationnelles).

On commencera par montrer comment utiliser le logiciel CONGRES, en particulier la forme des commandes et l'entrée des paramètres.

Pour décrire ensuite CONGRES d'un point de vue externe, on a choisi de le décomposer en deux parties :

. le travail personnel qui consiste à construire et modifier des dessins ; c'est une activité à laquelle un participant peut accéder hors du cadre d'une réunion ;

. la tenue d'une réunion, qui montre la façon dont se tient une réunion, en particulier les moyens de transmission d'interventions.

Si ces deux parties peuvent être considérées comme indépendantes, elles n'en constituent pas moins un tout : en effet, un participant peut transmettre des dessins ou des commentaires qu'il a préparés (la préparation d'un commentaire est très simple).

III.1. GÉNÉRALITÉS

III.1.1. LAGOR (Langage Avec Graphique Orienté Réunions)

LAGOR se veut un langage facile à apprendre et à utiliser afin de ne pas rebuter les utilisateurs potentiels non informaticiens. Il est d'ailleurs plus juste de dire qu'un sous-ensemble de ce langage sera très facile à apprendre et que ce sous-ensemble permettra de tenir des réunions. Une autre partie de ce langage, consacrée à des modifications ou des constructions complexes de dessins, nécessitera un peu d'habitude de la part de l'utilisateur.

Ce langage est utilisé en conversationnel : le participant tape une ligne (plusieurs commandes peuvent être données sur une même ligne, séparées par au moins un blanc) et le système décode cette ligne, commande par commande. Si une commande est valide, le logiciel l'interprète, sinon il affiche un message permettant à l'utilisateur de trouver son erreur, puis attend une nouvelle ligne (une nouvelle commande).

Les commandes comportent au maximum huit caractères qui peuvent être données sous forme abrégée (une, deux ou trois lettres).

Exemple :

GAUCHE s'écrit G, GA, GAU, GAUC, GAUCH, GAUCHE
CARRE s'écrit CA, CAR, CARR, CARRE.

Les paramètres, lorsqu'ils existent, sont donnés entre parenthèses et ne sont séparés par aucun blanc de la commande.

Exemple :

G(2).

Le décodage ligne par ligne se justifie du point de vue de l'utilisateur par les raisons suivantes :

. s'il ne connaît pas LAGOR, il tape une commande par ligne ; il reçoit alors une réponse du système ou l'affichage de l'effet de la commande si elle était valide ; il peut donc prendre une décision ;

. s'il connaît suffisamment LAGOR, il tape plusieurs commandes par ligne et gagne ainsi du temps. En cas d'erreur, il aura simplement tapé un certain nombre de commandes pour rien et le système lui précisera la commande sur laquelle l'erreur a été commise.

III.1.2. Influence du type de console

Le type de console utilisée a une grande influence sur les méthodes de travail. En effet, des consoles alphanumériques aux consoles graphiques, les facilités, tant du point de vue construction de dessins que du point de vue travail en conversationnel, varient énormément.

Les deux aspects les plus importants aux yeux des utilisateurs seront le temps de réponse et le type d'affichage.

. Temps de réponse :

sur un terminal (surtout graphique) connecté au 360/67 au moyen d'une ligne rapide, l'affichage quasi instantané aux yeux de l'utilisateur permet de traiter immédiatement, tant du point de vue décodage et interprétation que du point de vue affichage. Sur les terminaux graphiques dont la ligne de connexion est lente, cette méthode devient pénible pour l'utilisateur.

Ce problème est encore plus aigü dans le cas de terminaux tels que le 2741 IBM. Il ne faudra donc pas, dans ce cas, afficher immédiatement l'effet d'une commande, mais l'utilisateur disposera d'une commande spéciale (PRINT) lui permettant de demander explicitement l'affichage.

. Type d'affichage :

par type d'affichage, on entend les possibilités de tracé de vecteurs, etc .. et le type d'écran. On peut dans ce contexte déterminer trois types d'affichage :

- terminaux graphiques :

on peut afficher des vecteurs, des points et des caractères en tout point de l'écran. On n'aura donc aucun problème de limitation de tracé de figure. L'écran sera en général partagé en deux parties : la partie supérieure sera réservée à la conversation, la partie inférieure sera utilisée pour la préparation de dessins. Dans le cas où le participant travaille hors du cadre d'une réunion, il pourra bien sûr disposer de tout l'écran.

- Terminaux alphanumériques à écran :

il y aura bien évidemment des problèmes au niveau de dessins complexes, puisqu'ils seront affichés sous forme de tableaux de caractères. Pour le partage de l'écran, on procèdera comme suit : l'utilisateur dispose de tout l'écran pour préparer une intervention ; lorsqu'une intervention est reçue, on lui fournit autant de lignes que nécessaire. Si le participant a perdu l'affichage de ce qu'il préparait, il le redemande.

- Terminaux alphanumériques à papier :

le problème d'affichage de dessins se pose dans les mêmes termes que pour les terminaux alphanumériques à écran. En revanche, comme on dispose, en principe, d'un nombre de lignes illimité, l'espace d'affichage est quasi infini (limité en largeur). Lorsqu'une intervention sera reçue, elle s'affichera donc tout naturellement à la suite.

III.1.3. Initialisation (généralités)

Au début d'une session, l'utilisateur reçoit un message lui demadant son numéro de passe (qui lui est indiqué lors de la décision de la tenue d'une réunion), le nom de la machine virtuelle sur laquelle il travaille, et son nom.

Si ces renseignements sont inexacts (par exemple, s'il n'est pas recensé dans la réunion), il a droit à deux nouveaux essais.

Si ces renseignements sont exacts, il peut accéder, dans l'ordre qu'il préfère, à trois activités essentielles :

. travail personnel : c'est l'activité qui lui permet de construire et modifier des dessins ;

. retours dans la conversation : c'est l'activité qui lui permet de revenir sur des interventions précédemment transmises (depuis sa dernière participation, par exemple) ;

. discussion avec les autres participants.

La section suivante décrira le travail personnel. La dernière section décrira les deux dernières activités qui font vraiment partie de la réunion (on reviendra également plus précisément sur l'initialisation d'une session).

III.2. CONSTRUCTION ET MODIFICATION DE DESSINS

Cette section décrit les facilités fournies à un utilisateur pour construire et modifier des dessins. On rappelle qu'un dessin est un ensemble de figures, textes et liens :

. figure :

figure géométrique prédéclarée ou déclarée segment par segment par l'utilisateur ; un nom affiché ou non peut être associé à une figure ;

. lien :

ligne droite ou brisée joignant deux figures ou deux textes, ou une figure et un texte ;

. texte :

suite de caractères alphanumériques placée dans le dessin à l'endroit désiré.

Le moyen de communication mis à la disposition de l'utilisateur est le clavier alphanumérique.

On désigne par objet la plus petite entité manipulée, c'est-à-dire une figure, un texte ou un lien.

Les commandes disponibles sont de cinq types :

. construction (de dessins ou d'objets),

. modification (de dessins ou d'objets),

. mémorisation (de dessins),

. appel (de dessins),

. commandes spéciales.

La désignation d'un point de l'écran est réalisée au moyen d'une croix que l'on désigne dans la suite par le terme "curseur". Cette croix apparaît sur l'écran dès que l'on entre dans l'environnement DESSIN. Quatre commandes (gauche, droite, haut, bas), permettent de déplacer ce curseur. Elles acceptent un paramètre entier (pris par défaut égal à un) désignant le nombre de fois le pas de l'échelle courante dont doit se déplacer le curseur dans la direction désignée par la commande.

Exemple :

si l'échelle courante a un pas de 10 unités, G(2) déplace le curseur de vingt unités vers la gauche. Si le curseur sort de l'écran, la commande est annulée.

Les échelles sont variables au gré de l'utilisateur et permettent ainsi une précision plus ou moins grande dans les déplacements du curseur. L'échelle peut être différente en X (gauche, droite) et en Y (bas, haut). Elles peuvent être affichées par les commandes X? et Y? (affichage sous forme de barres horizontales et verticales) (cf. figure 1).

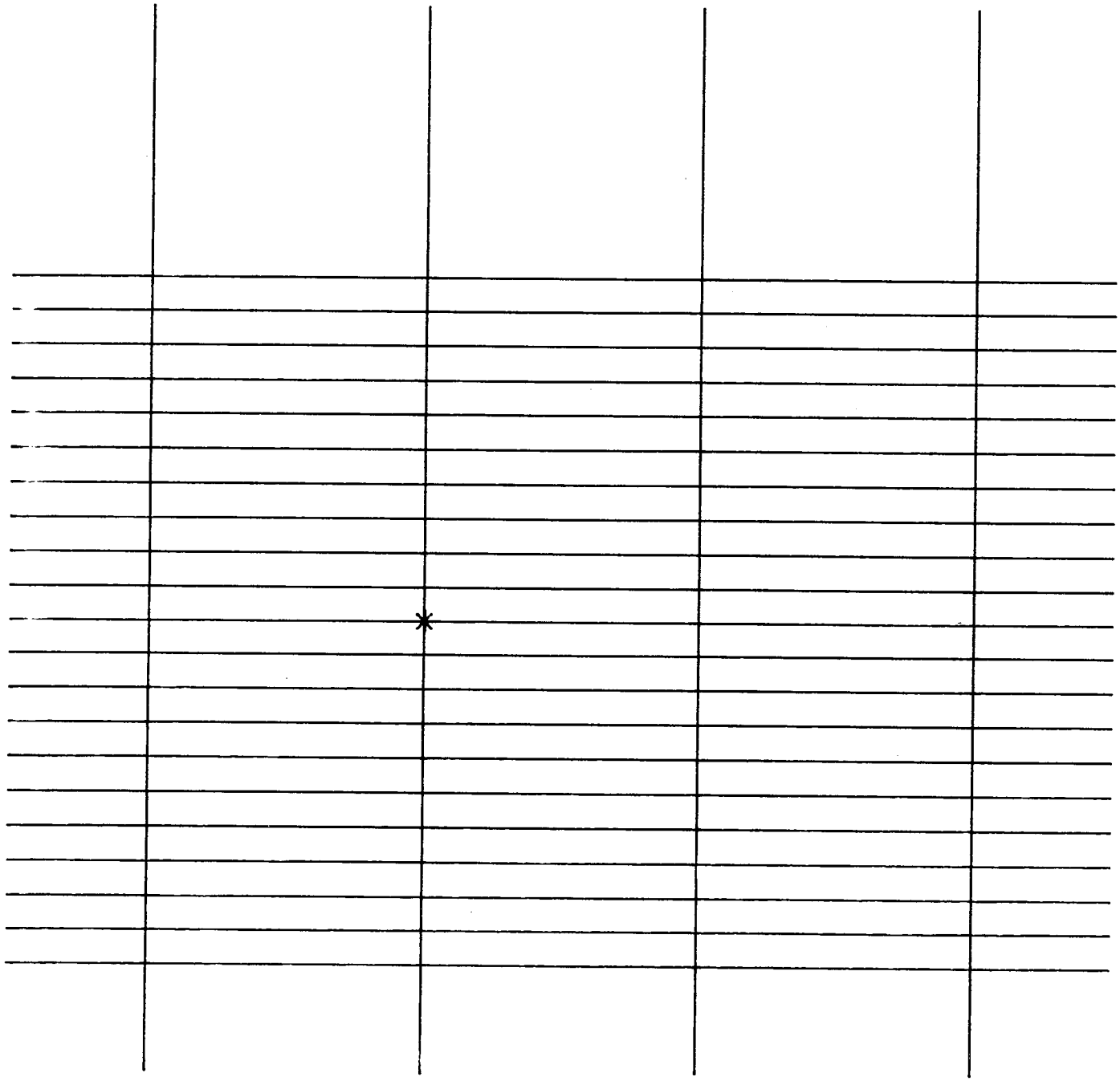
Remarque 1 :

mode de tracé : le tracé peut être réalisé à l'aide de traits pleins ou de traits pointillés. On verra dans les commandes spéciales, la méthode utilisée pour ce changement de mode.

Remarque 2 :

Lorsqu'on donne explicitement le nom d'une commande, il est écrit sous forme de caractères gras représentant la partie significative (abréviation maximum) suivis de caractères normaux.

Figure 1 - Echelles



FICHIER BENSON A JOUR

GARDAN LE 29/09/76 A 11:21:55 DESSIN NO 008 ECHELLE: 06/10

III.2.1. Constructions

Sous l'environnement DESSIN, on dispose de trois sous-environnements : FIGURE, TEXTE et LIEN, correspondant à la création de figures, textes et liens.

III.2.1.1. Environnement FIGURE

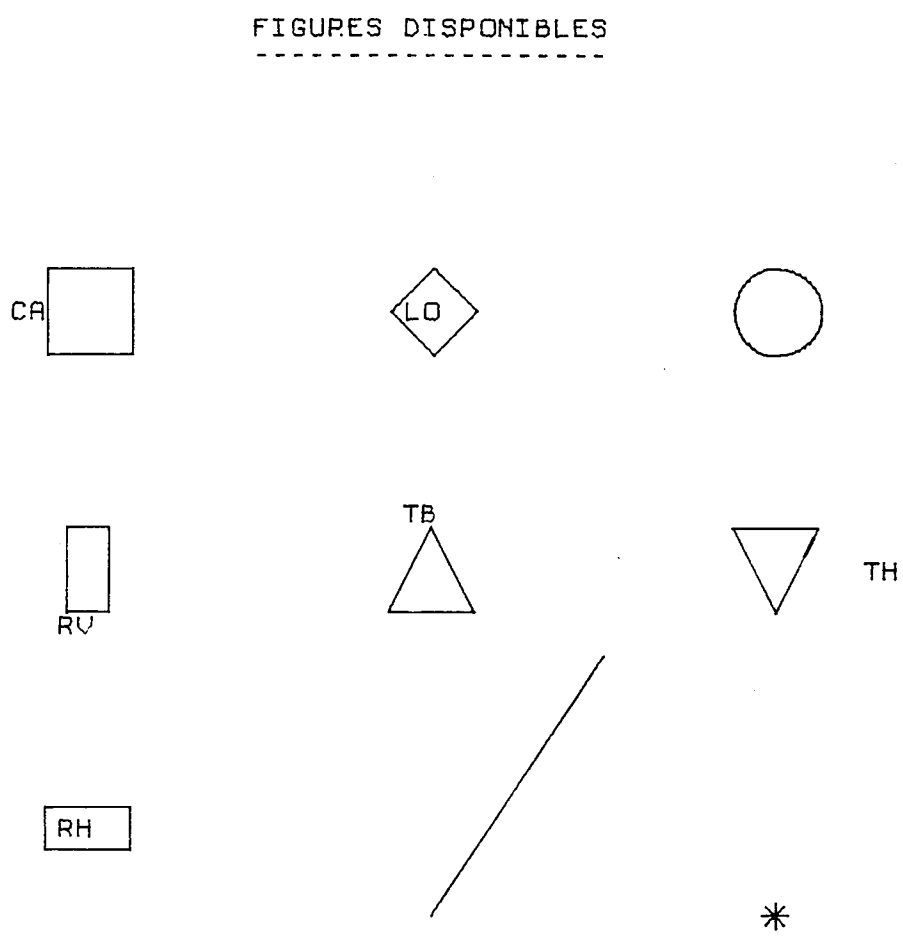
On y accède par la commande FIGURE. Deux paramètres permettent de lui donner un nom et de le placer par rapport à la figure (cf. figure 3). Ces noms sont nécessaires pour reconnaître les figures par la suite (pour créer des liens, modifier la figure, etc ..). Si l'utilisateur ne nomme pas explicitement une figure, le système lui attribue un nom par défaut (O1, O2,..). On autorise les doubles définitions d'identificateurs de figure, c'est-à-dire que plusieurs figures peuvent avoir le même nom. Lors d'une double définition, le système le signale à l'utilisateur qui peut aisément changer l'identificateur s'il le désire.

Les figures prédéclarées (CArré, CErcle, LOsange, TRHaut, TRBas, RHorizon, RVertica) sont créées automatiquement autour de la position courante du curseur à l'aide d'une seule commande (cf. figure 2). La taille de cette figure dépend d'un "rayon" courant qui peut être modifié au gré de l'utilisateur.

Les figures déclarées par l'utilisateur sont créées segment par segment. Un segment est créé de la façon suivante :

. l'utilisateur place le curseur à l'origine désirée du segment et signale qu'il commence la création d'une figure non prédéclarée ;

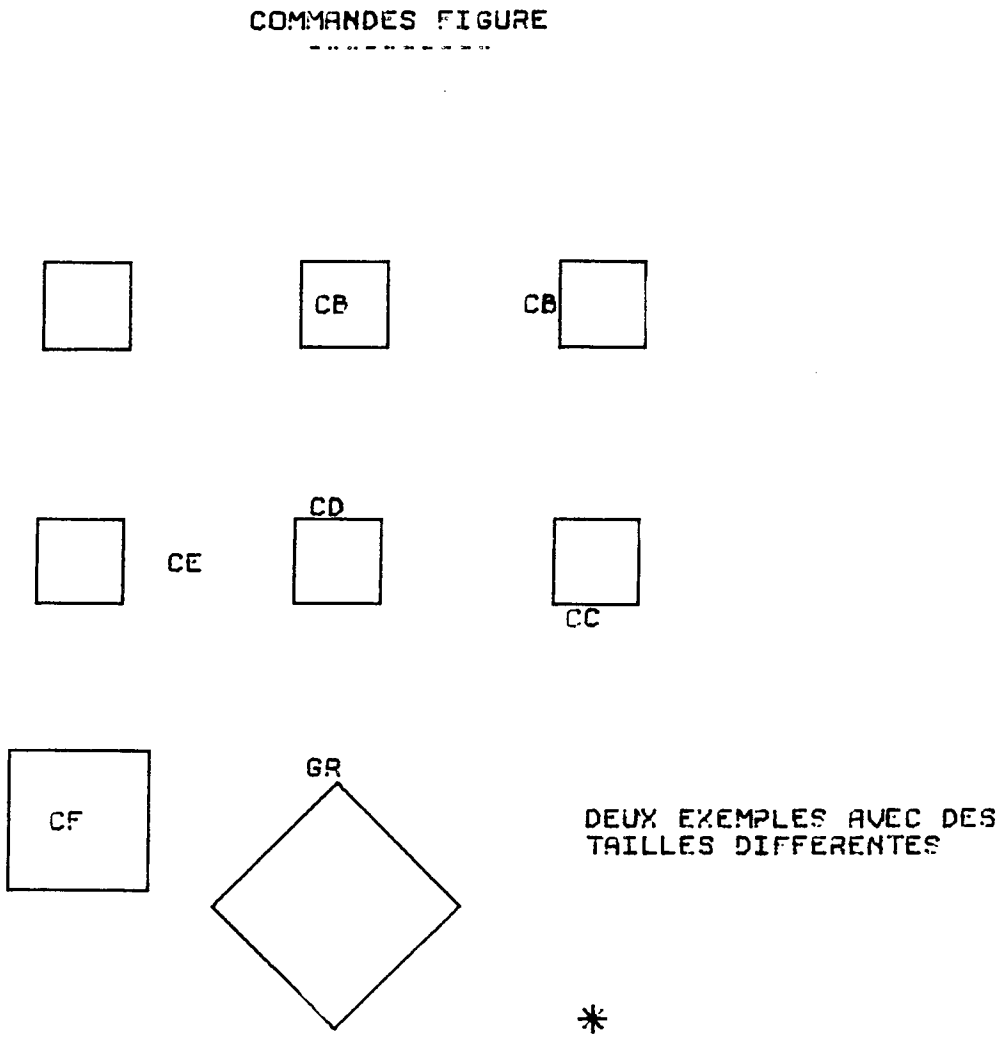
Figure 2 - Figures prédéclarées



FICHIER BENSON A JOUR

GARDAN LE 15/09/76 A 10:16:49 DESSIN NO 002 ECHELLE: 06/10

Figure 3



B(2) BENSON(6)
FICHIER BENSON A JOUR

GARDAN LE 16/09/76 A 15:44:47 DESSIN NO 003 ECHELLE: 06/10

. il bouge ensuite le curseur jusqu'à l'extrêmité désirée du premier segment et signale que c'est l'extrêmité du premier segment ;

. il peut ensuite créer des segments joints, en déplaçant le curseur pour avoir l'extrêmité du deuxième, puis du troisième, ..., jusqu'à ce qu'il ait terminé sa figure (non obligatoirement fermée).

La figure qui vient d'être créée peut être répétée un certain nombre de fois par une commande simple et on peut considérer dans la suite, au choix, l'ensemble (un sous-dessin) ou chaque figure le composant. Des liens peuvent être créés automatiquement entre ces figures, de la première (au point de vue ordre de création) vers la deuxième, de la deuxième vers la troisième, etc ... (cf. figure 5).

III.2.1.2. Environnement TEXTE

On y accède par la commande TEXTE, disposant d'un paramètre permettant de préciser si l'on veut placer le texte grâce au curseur ou si l'on veut qu'il soit placé automatiquement en haut et à gauche. On donne ensuite le texte que l'on désire afficher. Si l'on donne plusieurs lignes sans sortir de l'environnement TEXTE, elles sont placées les unes en-dessous des autres. Chaque ligne sera accessible indépendamment des autres.

```
*ECI EST UN EXEMPLE DE CHAINE CONSTRUITE AUTOMATIQUEMENT LIGNE
PAR LIGNE ....
```

Figure 4

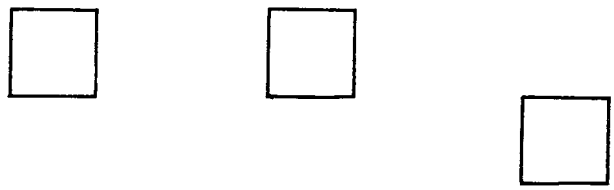
```
HIER BENSON A JOUR

      GARDAN   LE 04/10/76 A 10:29:26 DESSIN NO 006 ECHELLE: 06/10
```

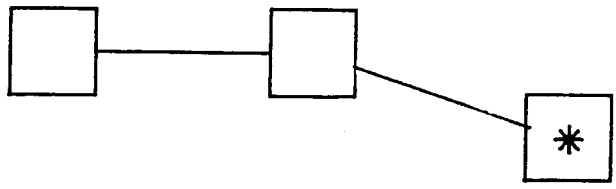
Figure 5

EXEMPLES DE FIGURES MULTIPLES

SANS LIENS INTERNES



AVEC LIENS INTERNES



F(A) CA D(3) M B(1) D(3) M LI BENSON(6)
FICHIER BENSON A JOUR

GARDAN LE 16/09/76 A 15:35:32 DESSIN NO 001 ECHELLE: 06/10

III.2.1.3. Environnement LIEN

Cet environnement permet de créer des liens entre deux figures, deux textes ou une figure et un texte.

S'il s'agit d'un lien entre la figure (ou le texte) courante et une autre figure (ou texte), il suffit de préciser son nom. En revanche, si l'on veut créer un lien entre deux figures quelconques, il faut préciser les noms des deux figures concernées.

Exemple :

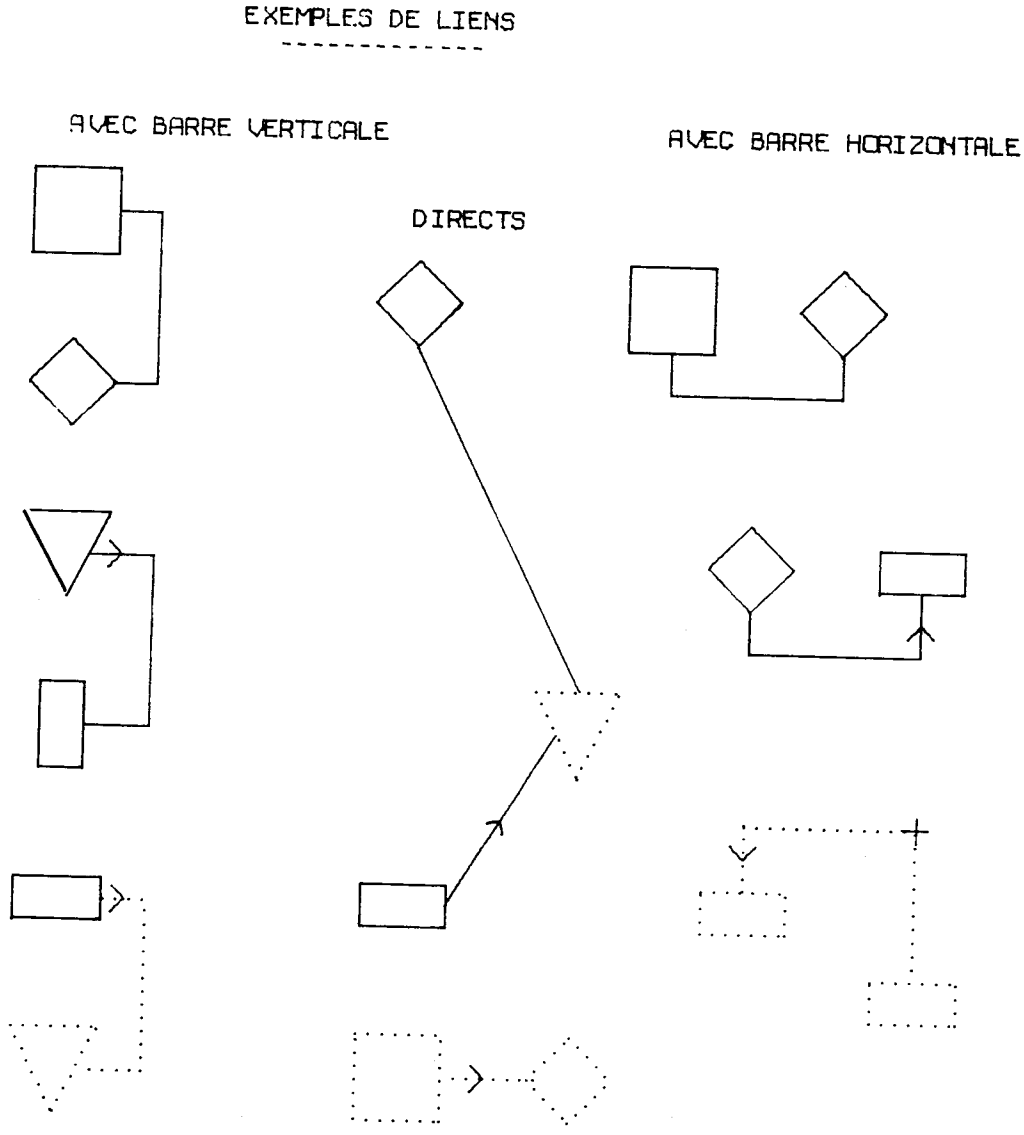
L(id) crée un lien entre la figure courante et la figure de nom id.
LE(id1,id2,id3,id4,..) crée un lien entre les figures (textes) de noms id1 et id2, un lien entre les figures (textes) de noms id3 et id4, etc ..

Les liens peuvent être sous forme de lignes droites (liens directs) ou sous forme de lignes brisées, de la manière suivante (cf. figure 6) :

L'option de départ est la création de liens directs. Cette option reste valable tant qu'aucune barre n'est créée. Une barre est un trait vertical ou horizontal qui peut être placé grâce au curseur. Cette barre sera considérée comme courante (tous les liens passent par cette barre) tant qu'elle n'est pas effacée, auquel cas les liens seront à nouveau directs, ou remplacée par une autre barre.

Les liens peuvent être également orientés au moyen d'une flèche, créée automatiquement si l'utilisateur l'a demandé. L'option de départ est la création de liens sans flèche (cf. figure 6).

Figure 6


EXEMPLES DE LIENS
AVEC BARRE VERTICALE
AVEC BARRE HORIZONTALE
DIRECTS
FICHIER BENSON A JOUR
GARDAN LE 15:49:29 A PAPI DESSIN NO 004 ECHELLE: 06/10

III.2.2. Modifications

Les modifications peuvent s'appliquer à deux niveaux :

. un niveau global : le dessin,

. un niveau objet : figure, texte ou lien.

III.2.2.1. Niveau global : le dessin

Les trois types de modifications sur un dessin implémentés sont les suivants :

. Modification de taille :

une commande permet de diminuer ou d'augmenter globalement la taille d'un dessin, à l'aide d'un paramètre entier N. La variation est de N/10. Etant donné que l'on peut difficilement choisir un milieu du dessin courant, cette modification est une homothétie de centre le coin bas gauche de l'écran et de rapport N/10 (cf. figure 7). Ceci n'est pas gênant pour l'utilisateur, puisqu'il peut ensuite appliquer une translation du dessin.

. Translation :

quatre commandes permettent de déplacer globalement le dessin dans les directions horizontales et verticales. Elles acceptent un paramètre entier N indiquant le nombre de fois le pas de l'échelle courante dont doit se déplacer le dessin dans la direction précisée par la commande (cf. figure 7).

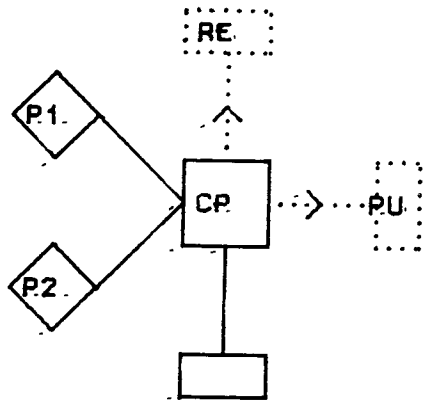
. Effacement de dessin :

la commande EDessin permet d'effacer le dessin courant.

Figure 7

VARIATION GLOBALE

CONFIGURATION



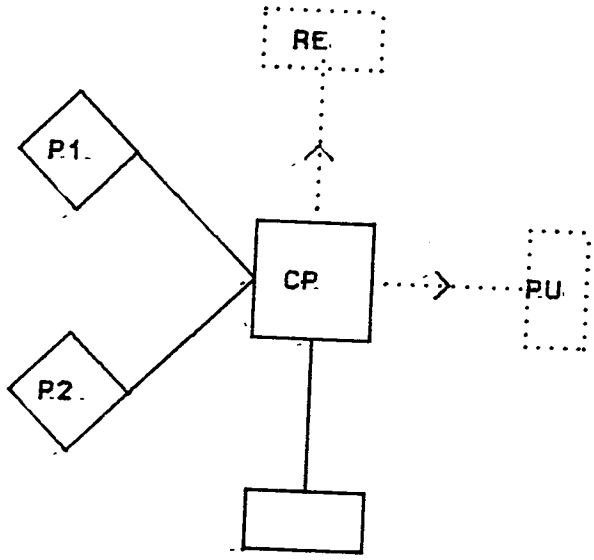
FICHIER BENSON A JOUR.

GARDAN LE 07/07/77 A 11:22:29. DESSIN NO 004 ECHELLE: 06/10.

Figure 7

APRES AUGMENTION

CONFIGURATION



TIER BENSON A JOUR.

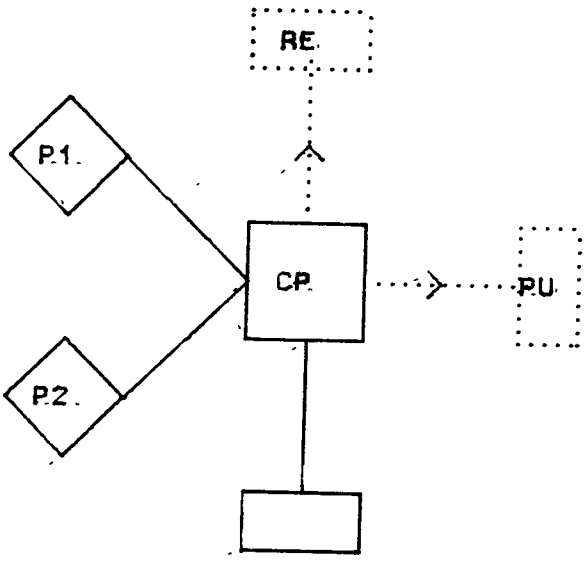
GARDAN LE 07/07/77 A 11:23:52 DESSIN NO 005 ECHELLE: 06/10.

Figure 7

APRES AUGMENTION

PUIS TRANSLATION VERS LA GAUCHE

CONFIGURATION



FICHIER BENSON A JOUR

GARDAN LE 07/07/77 A 11:24:51 DESSIN NO 006 ECHELLE: 06/10

III.2.2.2. Modifications d'objets

Ces modifications s'appliquent soit indifféremment à des figures, des textes et des liens, soit à un sous-ensemble de ces objets.

Toutes les modifications ont en paramètre le nom de l'objet considéré, sauf lorsqu'il s'agit d'une modification de l'objet courant.

Etant donné qu'un même identificateur peut désigner des figures ou des textes différents, il est nécessaire d'expliquer la façon dont on trouve le bon objet, connaissant son nom. La méthode est essentiellement conversationnelle et elle sera désignée dans toute la suite par : RECHERCHE OBJET ID. Son déroulement est le suivant : l'utilisateur veut faire subir une modification concernant l'objet de nom ID. Trois cas peuvent se présenter :

1) aucun objet de nom ID n'a été déclaré dans le dessin courant : le système retourne le message "objet inexistant" ;

2) un seul objet de nom ID a été déclaré dans le dessin courant. Le système estime que c'est l'objet recherché par l'utilisateur et passe donc à la phase suivante qui peut être soit l'application effective de la modification sur cet objet, soit la recherche d'un autre objet (cas d'une modification de liens) ;

3) plusieurs objets de nom ID ont été déclarés dans le dessin courant : le système repère le premier par une croix et demande à l'utilisateur si c'est l'objet qu'il recherche :

- si oui, le système peut passer à la phase suivante (application de l'effet de la commande, ou recherche d'un autre objet) ;
- sinon, le processus reprend en éliminant l'objet refusé par l'utilisateur, c'est-à-dire qu'à nouveau les cas 1, 2 et 3 peuvent se présenter.

a) Effacement d'objet

il est possible d'effacer n'importe quel objet en précisant son nom s'il s'agit d'une figure, les premiers caractères s'il s'agit d'un texte, ou les noms des figures liées s'il s'agit d'un lien. Ces commandes sont fortement interactives puisqu'elles utilisent la méthode RECHERCHE OBJET ID. Le déroulement de l'action d'effacement est donc le suivant :

. Effacement d'une figure :

si la figure recherchée est trouvée, elle est effacée, ainsi que tous les liens attachés à cette figure ;

. effacement d'un lien :

si les deux figures concernées sont trouvées, le ou les liens les joignant sont effacés ;

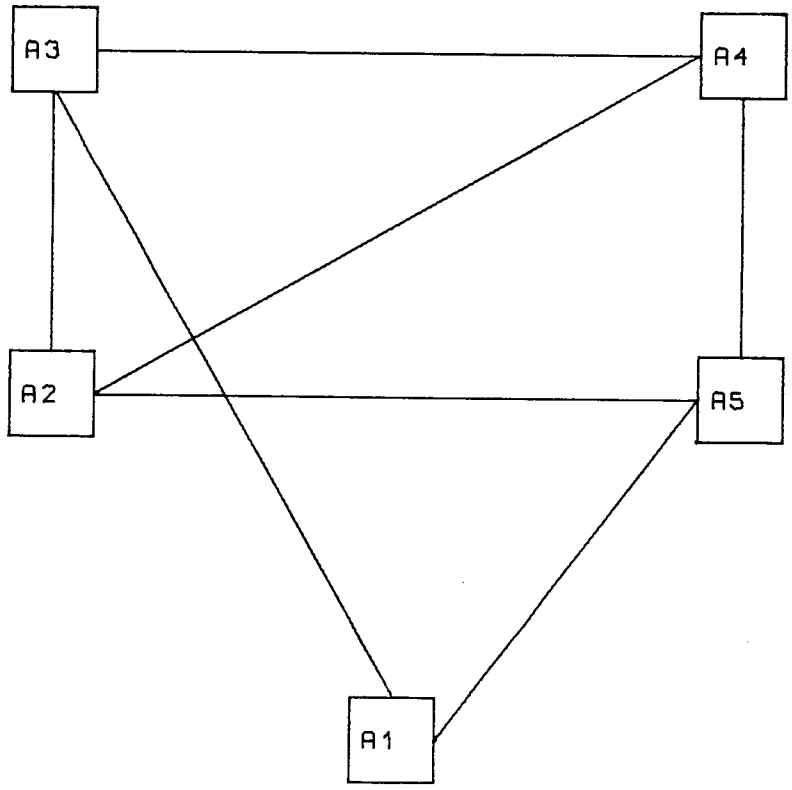
. effacement d'un texte :

si le texte recherché est trouvé, on efface toute la ligne concernée, ainsi que tous les liens qui y étaient attachés.

b) Modification de taille d'une figure

Si la figure recherchée est trouvée, son rayon est modifié (le centre reste au même endroit) et les liens se rattachant à cette figure sont donc modifiés en conséquence.

Figure 8 - Effacement de figure



#FFACEMENT DE A2

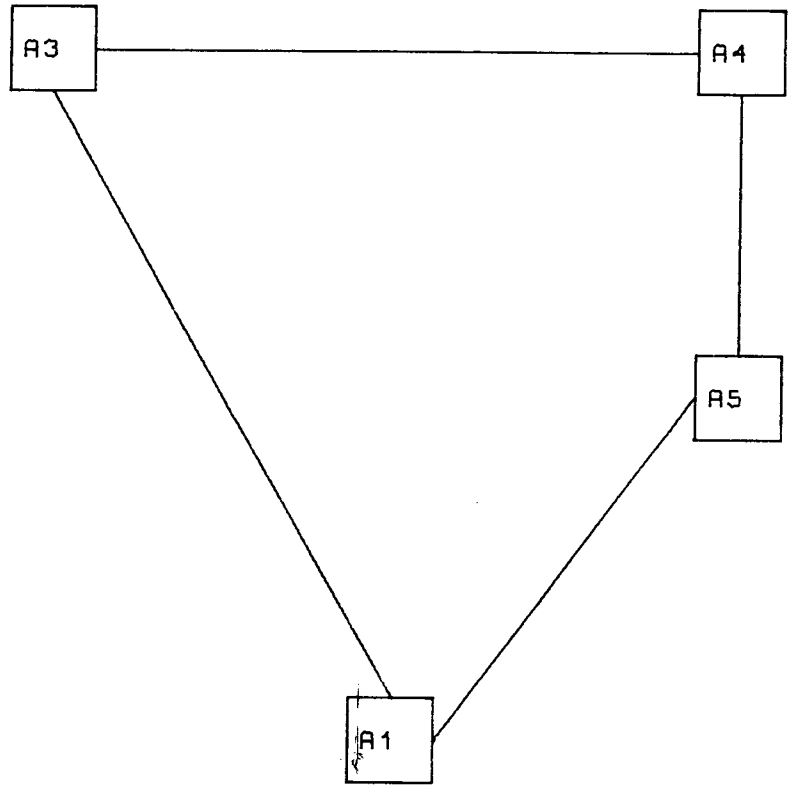
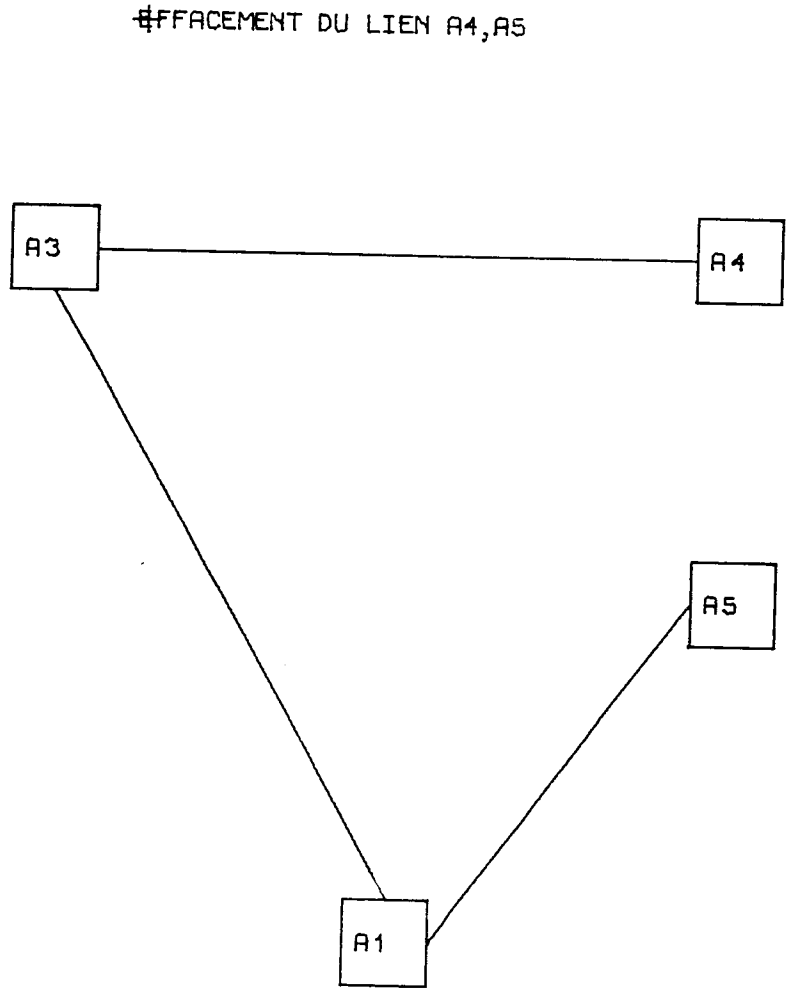


Figure 9 - Exemple d'effacement de lien



FICHIER-BENSEN A JOUR

GARDAN LE 16:02:26 DESSIN NO 007 ECHELLE: 06/10

Figure 10

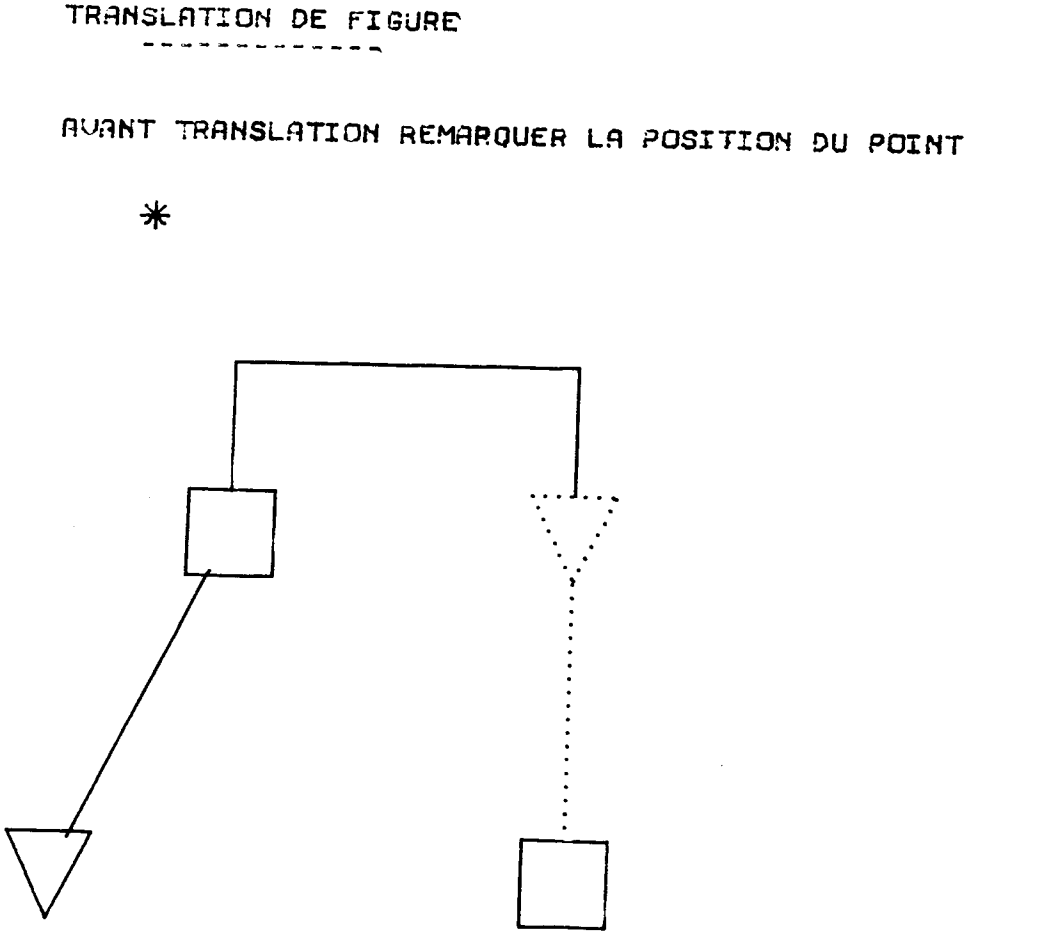
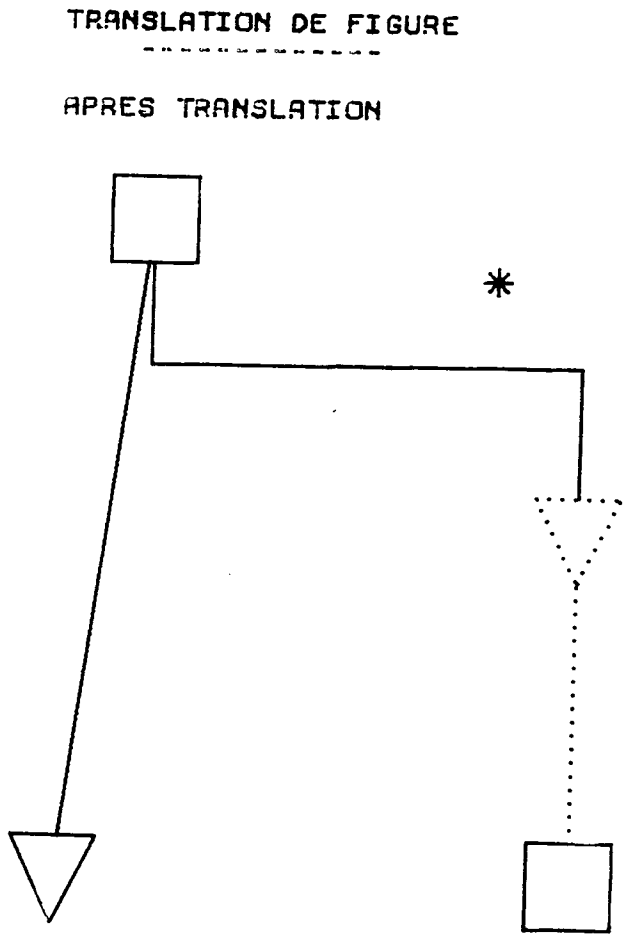

TRANSLATION DE FIGURE
AVANT TRANSLATION REMARQUER LA POSITION DU POINT
*
FICHIER BENSON A JOUR
GARDAN LE 16-09-76 A 15:55:54 DESSIN NO 004 ECHELLE: 06-10

Figure 10



FICHIER BENSON A JOUR

GARDAN LE 16/09/76 A 10:00:42 DESSIN NO 005 ECHELLE: 06/10

c) Translation d'une figure ou d'un texte

Une translation s'effectue en deux temps :

. l'utilisateur place tout d'abord le curseur à l'endroit où il veut translater la figure (ou le texte considéré) ;

. il précise ensuite le nom de la figure (ou du texte) à translater. On applique la procédure habituelle RECHERCHE OBJET ID. Si l'objet est trouvé, il est placé autour du curseur et tous les liens qui y étaient attachés sont modifiés en conséquence (cf. figures 10 et 11).

d) Changement de nom d'une figure

Etant donné que l'on autorise les doubles définitions de noms de figures, que celles-ci peuvent être involontaires et gênantes, ou tout simplement pour corriger des erreurs, il a semblé utile de mettre à la disposition de l'utilisateur une commande de changement de nom. La recherche du nom se fait toujours suivant la procédure RECHERCHE OBJET ID.



CHANGE (CA,CP)



La figure sera accessible, dans toute la suite, sous ce nouveau nom.

Figure 11

Exemple de translation de texte

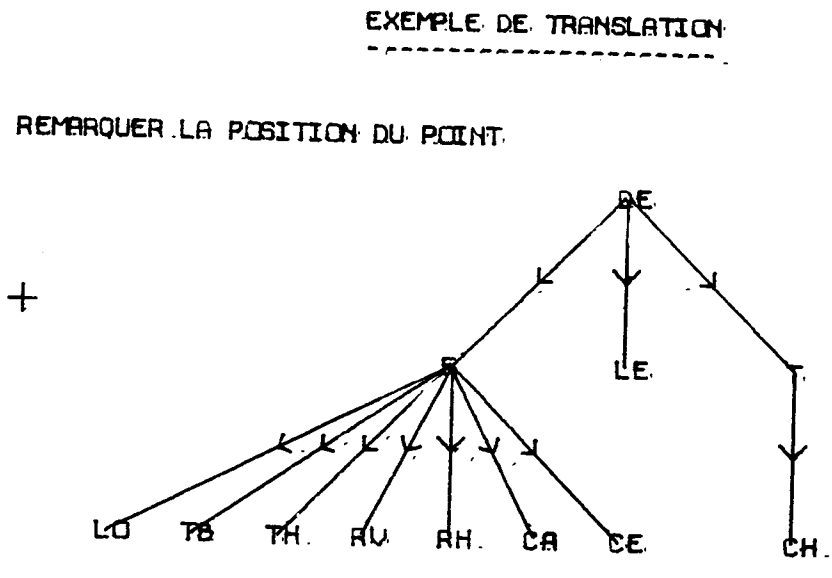
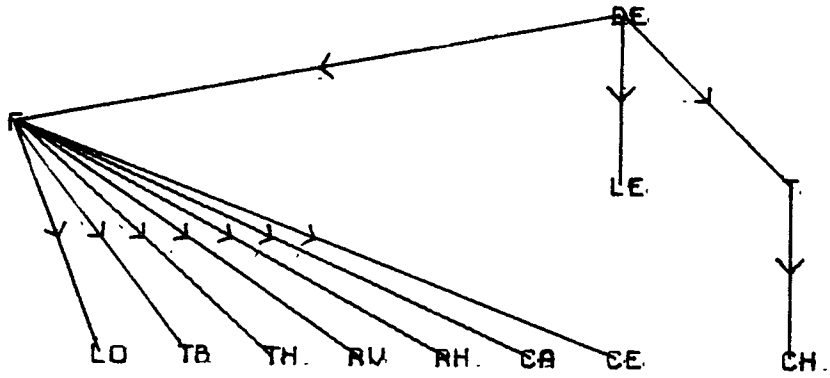


Figure 11

EXEMPLE DE TRANSLATION

APRES TRANSLATION



MODIFICATIONS

FICHIER BENSON A JOUR

GARDAN LE 07/07/77 A 11:20:35 DESSIN NO 003 ECHELLE: 06/10

III.2.3. Mémorisation

Cette facilité est offerte à l'utilisateur dans le but de préparer une réunion, conserver personnellement un certain nombre de dessins, ou même préparer un ensemble de sous-dessins.

La commande a la forme suivante : MEmorise(ident) ; ident est un identificateur d'au plus huit caractères. Tout ce qui se trouvait affiché dans la partie réservée au travail personnel est alors conservé sous le nom ident (les doubles définitions ne sont pas acceptées). Le dessin s'efface de l'écran, car on considère ce dessin comme terminé, mais dans le cas contraire, l'utilisateur peut le faire réafficher immédiatement.

III.2.4. Appel

La commande Appel(ident) affiche le dessin (s'il existe) mémorisé sous le nom ident. Ce dessin peut être placé comme l'on veut grâce au curseur considéré comme le centre du dessin appelé.

Plusieurs commandes Appel peuvent être données sans sortir de l'environnement dessin. Le dessin courant peut ainsi être formé de plusieurs appels de sous-dessins que l'on peut d'ailleurs modifier et auxquels on peut ajouter d'autres objets.

Cette commande, associée à la commande de mémorisation, permet donc à un utilisateur de se créer une bibliothèque de dessins. Par exemple, si l'on se sert souvent d'une figure particulière non prédéclarée, telle qu'un octogone, on crée cette figure segment par segment, on mémorise ce dessin (constitué d'une seule figure) sous un nom particulier (exemple : OCTO), puis on l'appelle et le place selon ses besoins dans des dessins plus complexes (on peut même le lier avec d'autres figures et textes, en se servant de son nom objet).

Remarque :

la commande TABliden permet d'afficher, au gré de l'utilisateur, les noms des dessins qu'il a mémorisés.

III.2.5. Commandes spéciales

Les commandes qui suivent constituent un ensemble d'aides à l'utilisateur et de commandes qui n'entrent pas expressément dans une des rubriques précédentes.

a) S.O.S.

C'est la commande de secours. Elle fournit un menu à l'utilisateur (listage de commandes) explicitant toutes celles qu'il peut utiliser à ce moment précis.

b) TABliden

Cette commande permet d'afficher les noms des dessins mémorisés (menu).

c) Flèche (rappel)

Commande utilisée pour créer des liens avec ou sans flèches.

d) POintil/PLein

Ces deux commandes permettent de changer le mode de tracé à chaque instant, de traits pointillés à traits pleins et inversement.

e) Retour

Annulle l'effet de la commande précédente.

f) PRint

Etant données les différences entre consoles, il a semblé utile de permettre à l'utilisateur de décider lui-même s'il veut ou non l'affichage de l'effet de ses commandes immédiatement après leur frappe. L'option de départ est que l'effet de chaque commande s'affiche immédiatement. Si l'utilisateur tape PRint(A), le dessin courant ne s'affichera que lorsqu'il le demandera explicitement par la commande PRint sans paramètre. S'il veut revenir à l'option de départ, il tape PRint(I).

III.3. TENUE D'UNE RÉUNION

Cette partie explique comment se tient une réunion pour un utilisateur. Pour avoir le droit de participer à une réunion, celle-ci doit avoir été décidée entre les participants. L'organisateur doit alors donner les noms des machines virtuelles utilisées, les noms des participants et les numéros de passe qui leur sont attribués. Une fois cette initialisation du système faite, la réunion se tient comme suit.

III.3.1. Début d'une session

Après s'être attaché une machine spéciale contenant les programmes et lancé la procédure EXEC correspondant au terminal utilisé (CONV 2250, CONV 2741, ...), un message s'affiche, demandant le numéro de passe, le nom de machine virtuelle (MV) et le nom personnel du participant. L'utilisateur fournit ces renseignements.

Exemple :

5 MACHIN DUPONT

. Si un des renseignements est inexact, CONGRES le lui signale et lui donne deux autres essais.

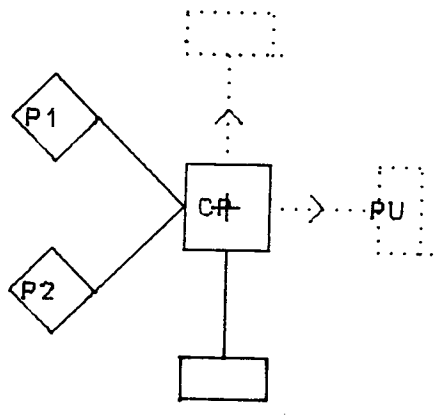
. Si les renseignements sont exacts, les renseignements suivants s'affichent :

- liste des participants, précisant s'ils sont présents ou non et la date et l'heure de leur dernière participation (cf. figure 12).

Figure 12

```
***LISTE DES PARTICIPANTS***
  P= PRESENT  A= ABSENT              DATE ET HEURE = DERNIERE PARTICIPATION
GARDAN     P    07*08*77   16:05:58
BENNET     A    06*27*77   16:24:26
VERAN      A          JA   MAIS
BALTER     A          JA   MAIS
```

CONFIGURATION



RECHIER BENSON A JOUR

GARDAN LE 16:07:37 A PAPI DESSIN NO 008 ECHELLE: 06/10

- un message indiquant le nombre d'interventions destinées à ce participant, transmises durant son absence.

Le participant peut alors accéder aux activités suivantes :

. retours dans la conversation,
. préparation d'un dessin (cf. § III.2),
. préparation d'un commentaire
. envoi d'interventions.

Indépendamment de ce qu'il fait, le participant reçoit des interventions et des messages.

Remarque :
si l'utilisateur ne désire pas participer à une réunion, mais uniquement préparer des dessins, il tape simplement comme numéro de passe le chiffre 0. Il dispose alors de toute la place sur l'écran virtuel et par projection, toute la place sur l'écran réel. Tout ce qui lui serait éventuellement transmis pendant ce temps ne serait donc pas affiché, mais seulement conservé. C'est donc aussi un moyen de travailler pour la réunion sans être dérangé.

Figure 12 bis

```
 start
EXECUTION BEGINS...
BONJOUR*NUMERO DE PASSE?NOM MACHINE?NOM?**IBME LIGNE
"1 Gardan Gardan"

  ***LISTE DES PARTICIPANTS***

  P= PRESENT  A= ABSENT

  DATE ET HEURE = DERNIERE PARTICIPATION
  GARDAN    P   08/24/77  10:28:12
  DENNET    A   08/27/77  10:24:20
  VERAN.    A   08/24/77  10:17:25
  WALTER    P   08/24/77  10:27:26
```

III.3.2. Retours dans la conversation

Le participant dispose de six commandes pour retrouver des interventions. Ces commandes ressemblent volontairement à celles utilisées sous EDIT dans CP, tant par leur nom que par leur effet. Les interventions sont numérotées par CONGRES. Un numéro d'intervention courant est initialisé à 1 au début d'une session.

. GOTO(n) :

affiche l'intervention de numéro n qui devient le numéro courant.

. NEXT :

affiche l'intervention de numéro immédiatement supérieur au numéro courant.

. UP :

affiche l'intervention de numéro immédiatement inférieur au numéro courant.

. TOP :

permet de retourner au début du fichier d'interventions (le numéro courant devient égal à 1).

. BOTTOM :

permet d'aller en fin de fichier d'interventions.

. LOCATE(A) :

affiche la première intervention trouvée transmise par A (nom d'un participant) en cherchant à partir de l'intervention de numéro courant (s'il en existe).

Remarque :

les interventions sont protégées en lecture, c'est-à-dire que le participant ne pourra lire que des interventions qui lui étaient destinées.

Une intervention affichée par un retour dans la conversation devient un dessin ou un commentaire (suivant son type) courant, c'est-à-dire, en particulier, que le participant peut ajouter ou retirer ou modifier des objets.

Cependant, l'intervention proprement dite n'est pas modifiée, en particulier s'il veut la réafficher plus tard, il la retrouvera dans l'état où elle avait été transmise.

L'intérêt de permettre à l'utilisateur de considérer l'intervention affichée par un retour dans la conversation est double :

. il peut mémoriser cette intervention en lui donnant un nom, ce qui lui permettra dans la suite de l'accéder très facilement et très rapidement ;

. il peut la compléter ou la modifier pour en faire une nouvelle intervention.

Lors d'un retour dans la conversation, les renseignements suivants s'affichent (cf. figure 13) :

. nom de l'émetteur (rien si elle était anonyme),

. date et heure d'émission,

. la mention PRIVE si c'était une intervention privée (rien si elle était publique).

Figure 13

I INTERVENTIONS EN RETARD.UTILISER GOTO,SP...
"GOTO(I)"

RETOUR SUR UNE INTERVENTION DE DALTER DU 08/24/77 A 10:24:44

IL FAUDRAIT REVOIR TOUT CE QUI A ETE

TRANSMIS DE PUIS LES VACANCES

III.3.3. Préparation d'un commentaire

Lorsque l'utilisateur veut transmettre une intervention composée uniquement de lignes de texte, il est inutile de le faire sous l'environnement DESSIN. Il suffit de donner plusieurs lignes à la suite, qui sont considérées comme intervention prête à transmettre tant qu'elles ne sont pas abandonnées par la commande d'effacement.

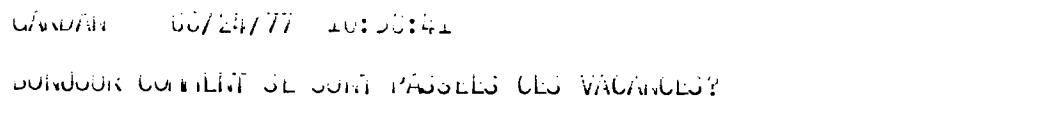
III.3.4. Modes de transmission

Le participant peut transmettre à tout moment un dessin ou un commentaire. Les transmissions peuvent être de quatre types par combinaison d'une destination et d'une signature :

. Destination :
  - privée : l'intervention est destinée à un participant unique (présent ou non),
  - publique : l'intervention est destinée à tous les participants (présents ou non).

. Signature :
  - signée : le nom de l'émetteur est connu du ou des destinataires,
  - anonyme : le nom de l'émetteur reste inconnu du ou des destinataires (qui n'ont aucun moyen de le connaître).

Figure 14



Toute intervention destinée à un participant présent s'affiche sur sa console, précédée d'une en-tête comprenant (cf. figure 14) :

. le nom de l'émetteur si l'intervention est signée, huit blancs si elle est anonyme,

. la date et l'heure d'émission de l'intervention,

. la mention PRIVE si l'intervention n'est destinée qu'à ce participant (rien si c'est une intervention publique).

Les quatre commandes disponibles sont les suivantes :

ENvoi : signée, publique

ENvoi(GARDAN) : signée, privée (uniquement destinée à GARDAN)

? : anonyme, publique

?(GARDAN) : anonyme privée (uniquement destinée à GARDAN).

III.3.5. Réceptions

Indépendamment de ce qu'il fait, le participant reçoit des interventions et des messages du système concernant la tenue de la réunion :

. les interventions lui sont transmises par des participants présents ;

. les messages proviennent du système et lui signalent les arrivées et les départs éventuels d'autres participants (on ne parle pas ici des différents messages d'erreur)

III.3.6. Cas particulier : réunion normale

On rappelle que le terme réunion normale désigne une réunion dans laquelle on respecte les caractéristiques du face à face, en particulier :

. unicité de parole : une seule personne peut transmettre de l'information à un instant donné ;

. tout le monde est présent : encore que l'on puisse facilement autoriser les arrivées en retard, puisque le participant pourra revenir sur ce qui a été transmis durant son absence ;

. un président de séance est chargé d'assurer le bon déroulement de la réunion (rôle tenu par une partie du logiciel).

Le participant peut être dans l'un des trois états suivants :

. PAROLE : il peut transmettre des interventions,

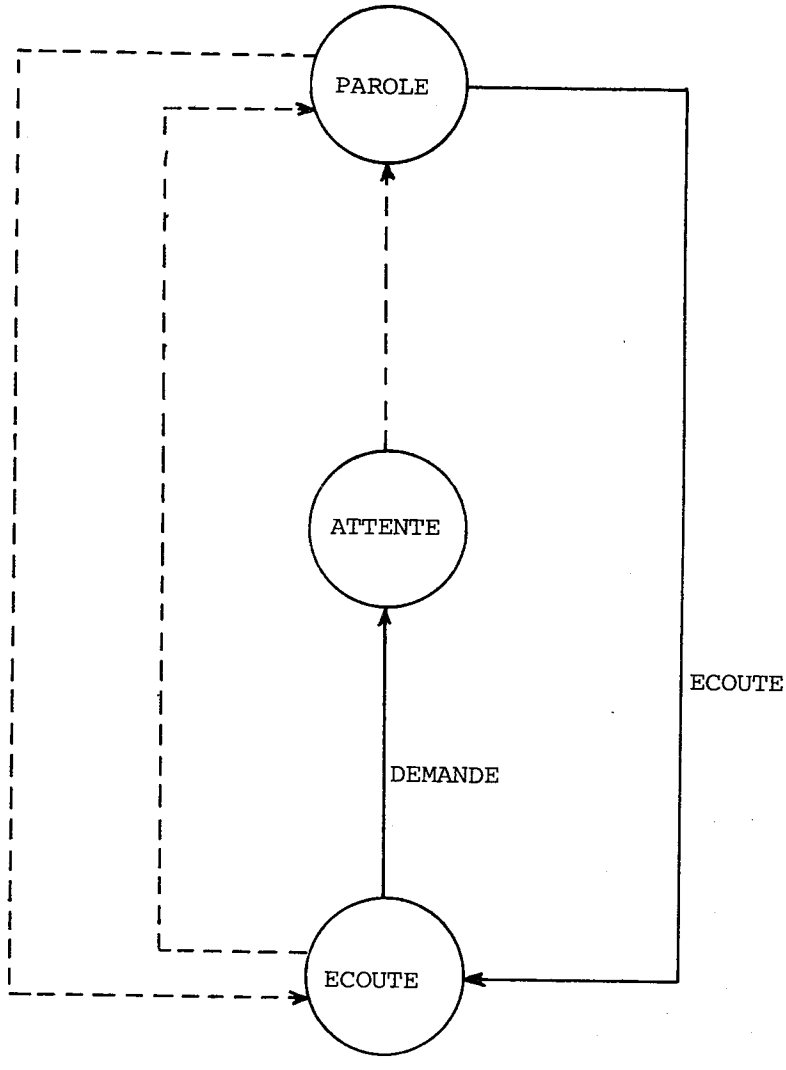
. ATTENTE : il a demandé l'autorisation de transmettre, mais elle ne lui a pas encore été accordée (il reçoit toujours ce qui est transmis par le participant en état PAROLE) ;

. ECOUTE : il est à l'écoute des autres participants et ne peut donc transmettre.

En dehors de tout ce qu'il peut faire, d'après ce que l'on a vu dans les paragraphes précédents (travail personnel, transmissions anonymes, ...), un participant dispose donc de deux commandes supplémentaires :

. DEMANDE : il demande l'autorisation de transmettre,
. ECOUTE : il signale qu'il a fini de transmettre.

Du point de vue de l'utilisateur, le schéma est donc le suivant :



——— commandes utilisateur

- - - - - commandes logiciel

L'utilisateur est tenu au courant à chaque instant de l'état dans lequel il se trouve, en particulier lors de changements d'état imposés par le SVPGEN (en pointillés sur le schéma).

. ATTENTE → PAROLE : l'utilisateur peut maintenant transmettre ;

. PAROLE → ECOUTE : au bout d'un certain temps, on décide de retirer le droit à la parole à celui qui l'a, si d'autres personnes sont en état attente (sinon, une personne pourrait garder le droit à la parole pendant toute la réunion) ;

. ECOUTE → PAROLE : lorsque personne n'a la parole et que personne n'est en attente, on propose à un participant de transmettre de l'information (s'il ne le désire pas, il tape la commande ECoute).

Remarque :

cette partie n'a pas été utilisée, mais toutes les primitives sont implémentées.

III.4. CONCLUSIONS

D'après le découpage de ce chapitre, on a vu que le système pouvait se décomposer d'un point de vue externe en deux sous-ensembles disjoints et supplémentaires de commandes, le premier concernant le travail personnel, le second la participation à la discussion.

. Le travail personnel (construction et modification de dessins) peut être considéré de façon indépendante d'une réunion. Un utilisateur (recensé ou non dans une réunion) peut en tapant comme numéro de passe le chiffre ZERO, accéder à toutes les activités incluses dans le travail personnel. Il peut le faire, par exemple, pour créer des dessins de type diagramme et les sortir sur BENSON ou les conserver en vue d'ajouts ou de modifications ultérieures. Il peut également préparer de cette façon des dessins qu'il aura à transmettre pendant une réunion.

. La participation à la discussion comprend toutes les commandes permettant en particulier de revenir sur ce qui a été transmis et d'intervenir dans la discussion. On y place également la commande de préparation de commentaire, ce qui permet de faire de ce sous-ensemble de commandes une partie suffisante pour participer à une réunion, en ne transmettant que des commentaires, mais en ayant l'occasion de recevoir des dessins.

Toute la partie travail personnel a été utilisée par diverses personnes. Cette utilisation a imposé quelques modifications et quelques ajouts. La partie concernant la tenue d'une réunion a été utilisée de la façon suivante :

. tests et utilisation entre deux terminaux de même type (TEKTRONIX 4010 et IBM 2250 ou deux terminaux alphanumériques) ;

. premiers tests de la version à n consoles suivant la méthode suivante : on n'utilise qu'une seule console ; on se connecte à CONGRES sur cette console, on effectue un certain nombre d'opérations, puis on se déconnecte de CONGRES sans qu'il ne le sache, ce qui permet de se connecter à nouveau à CONGRES sous un nouveau nom. Cette méthode a permis, en n'utilisant qu'une console, de tester les transmissions d'interventions et le fonctionnement de la gestion de la conversation ;

. tests et utilisation de la version générale.

Des petites réunions ont été testées sur CONGRES comme le montrent les exemples cités dans ce chapitre.

CHAPITRE IV

CONCEPTION ET RÉALISATION DE CONGRES

Ce chapitre est composé de deux parties :

. une discussion des grands choix que l'on a faits, en tenant compte des idées générales exprimées dans le chapitre II ;

. une description technique de la conception et de la réalisation du logiciel CONGRES. On mettra en particulier en évidence ce qui dépend du calculateur et du système à partage de temps choisis pour l'implémentation de CONGRES.

IV.1. CHOIX GÉNÉRAUX

IV.1.1. Indépendance du logiciel par rapport aux consoles

Afin d'assurer l'indépendance du logiciel par rapport aux terminaux réels utilisés, on définit une *console virtuelle* qui représente un matériel idéal. Elle est composée :

. d'un écran virtuel (système de coordonnées),
. de possibilités d'affichage et de communication,
. d'une structure de données.

Pour cette console virtuelle, l'écran sera un carré de 1023 points en abscisse et 1023 points en ordonnée. L'origine se trouve au coin bas gauche de l'écran, ce qui implique que toutes les coordonnées sont positives ou nulles. Cette console pourra afficher des segments, des points et des caractères.

Choix de la structure de données :

. Relative au dessin courant (liste d'affichage) :

La structure de données choisie pour la liste d'affichage est une structure de tables, pour les raisons suivantes :

- les figures et les textes peuvent être représentés par une fonction de la forme :

f(ID,NO,X,Y,R,D,M).

On peut donc constituer une table commune aux figures et aux textes faisant appel pour l'affichage graphique à un sous-programme fonction du numéro de forme NO.

- les liens sont représentés par une fonction de la forme :

L(N1,N2,X,Y,M).

On constitue une table de liens (table de corrélation) qui permet de retrouver très aisément toutes les figures et tous les textes soumis à la contrainte lien, deux à deux.

- Le rang des objets dans leurs tables respectives permet de leur associer un numéro que l'on peut définir de façon non ambigüe en prenant, par exemple :

figure ou texte de rang i → NV = 2i

lien de rang j → NV = 2j - 1

Inversement, connaissant un numéro d'objet, on pourra retrouver d'une part s'il s'agit d'une figure (ou d'un texte), d'autre part sa place dans la table qui le contient ; avec l'exemple ci-dessus, cela donne :

NV pair indique que c'est une figure ou un texte et que sa définition se retrouve au rang NV/2 de la table des figures et textes,

NV impair indique qu'il s'agit d'un lien et que sa définition se trouve au rang NV/2 + 1 dans la table des liens.

On associera donc à un objet un numéro virtuel par une relation bijective :

objet → numéro virtuel NV

(1) (1)

ensemble des objets courants ensemble des NV courants

- Enfin, cette structure permet, avec une assez grande facilité, les adjonctions, suppressions et modifications d'objets.

. Relative à la conversation :

On fait le choix de conserver le maximum d'interventions parmi les plus récentes. Cette information n'est pas accessible directement par le participant (ni modifications, ni adjonctions, ..).

On conserve donc une structure très simple comportant, pour toutes les informations affichées :

- leur type (dessin ou commentaire),
- un moyen de les retrouver dans le fichier de conversation.

IV.1.2. Les fichiers

La tenue d'une réunion est fortement basée sur l'utilisation de fichiers pour les actions suivantes :

. conserver et retrouver tout ce qui a été transmis,

. réinitialiser le système entre deux sessions,

. autoriser le participant à conserver pour lui, et lui seul, un certain nombre d'informations.

Avant de voir d'un peu plus près les caractéristiques générales de ces fichiers, on peut préciser qu'ils pourront atteindre des tailles respectables (surtout le fichier de conversation) et qu'il faudra donc les gérer en pages ou segments, c'est-à-dire qu'à chaque fois qu'on voudra accéder à une partie du fichier, on n'en chargera que ce qui est utile en mémoire centrale.

IV.1.2.1. Le fichier de conversation

Ses caractéristiques essentielles seront les suivantes :

. Unicité pour une réunion :

le fichier de conversation pouvant être grand, il est fondamental de n'en conserver qu'une copie par réunion. Cette unicité pose évidemment le problème de l'accès, particulièrement en écriture, de deux utilisateurs en même temps (accès exclusif).

. Accès exclusif :

la résolution de ce problème dépend en partie du système d'exploitation sous lequel on travaille. Cependant, on peut décider de principes généraux, en particulier :

- préparer au maximum dans un buffer tous les changements à réaliser pour n'avoir qu'à les faire écrire par la partie du programme qui constitue une section critique ;

- utiliser une variable s initialisée à 0 et une file d'attente associée A(s) ; lorsqu'un processus entre dans la section critique, s est mis à 1 et lorsqu'il en sort, s est remis à 0 ; tout processus arrivant lorsque s = 1 est mis dans la file d'attente ; il semble tout à fait normal de gérer cette file d'attente en FIFO (premier entré, premier sorti) ;

- aucune modification n'est autorisée, donc toute écriture se fait dans la première zone libre dont l'adresse est rangée dans un pointeur.

· Protection des enregistrements (interventions) :

on utilise une liste d'accès, c'est-à-dire qu'à chaque intervention I est associée une liste des utilisateurs pouvant l'accéder (forcément en lecture uniquement). Cette méthode est choisie parce que la liste d'accès pourra être facilement réduite pour exprimer trois types d'accès (alors qu'une liste des droits serait très longue puisqu'elle devrait indiquer toutes les interventions accessibles par usager) :

- accès autorisé à tous les usagers,
- accès autorisé à deux usagers,
- accès autorisé à un usager.

L'accès autorisé à tous les usagers représente le cas d'une intervention publique ; l'accès autorisé à deux usagers représente le cas d'une intervention privée (accessible en lecture au destinataire et à l'émetteur) ; l'accès autorisé à un seul usager représente le cas marginal d'un émetteur se transmettant une intervention privée.

· Accès au fichier :

afin de faciliter l'accès au fichier de conversation, on disposera d'une table qui pourra elle-même être recopiée dans un fichier, pour des raisons de sécurité, et pour réinitialiser rapidement le système à chaque début de session (cela évite de parcourir tout le fichier pour reconstituer la table). Elle contiendra tous les renseignements utiles pour retrouver une intervention. On pourra y implémenter les listes d'accès à chaque intervention.

IV.1.2.2. Le fichier personnel

Il s'agit d'un fichier contenant toutes les informations (dessins, interventions) que veut conserver un participant. Il est donc propre à un utilisateur. Les droits d'accès en sont simples : pour le fichier i personnel appartenant à Ui, on a : $k \neq i$, droit de Uk sur fichier i = AUCUN
droit de Ui sur fichier i = lecture, écriture, modification.

Chaque participant peut lire, écrire et modifier son fichier personnel. Il suffit donc que ce fichier appartienne à l'espace de travail de son propriétaire (qu'il n'existe aucun chemin d'un utilisateur Uk vers un fichier fi si $k \neq i$).

Ce fichier sera décomposé logiquement en dessins et sera accessible via une table de dessins.

IV.1.2.3. Autres fichiers

Il s'agit de fichiers de contrôle, tels que la liste des participants, l'historique de leur participation, etc .. Ils devront être protégés, mais ne seront en fait accessibles que par un processus système bien précis dont on verra le fonctionnement plus loin.

On peut résumer ces problèmes de protection par une matrice telle que :

	Fichier de conversation				Fichiers personnels			
	I1	I2	I3	enregistrement libre	f1	f2	f3	f4
U1	lire	i	lire	écriture	lire écrire	N	N	N
U2	lire	i	i	écriture	N	lire écrire	N	N
U3	lire	lire	lire	écriture	N	N	lire écrire	N
U4	lire	lire	i	écriture	N	N	N	lire écrire

où i veut dire interdiction d'accès

N veut dire aucun chemin d'accès.

Dans cet exemple, I1 est une intervention publique, I2 et I3 sont des interventions privées.

Remarque :

tous les fichiers seront gérés de façon invisible pour l'utilisateur, afin de ne pas l'encombrer d'aspects techniques.

IV.1.3. Partage du logiciel

Etant donnés les deux aspects essentiels du système que l'on veut réaliser :

. construction et modification de dessins,

. participation à la conversation,

on peut partager le logiciel en deux modules essentiels :

- un module chargé d'interpréter toutes les données fournies par l'utilisateur, quel que soit le moyen d'entrée utilisé, et de gérer l'espace propre à chaque utilisateur ;

- un module chargé de gérer la réunion, tant du point de vue intervention, que du point de vue départs et arrivées de participants.

IV.1.3.1. Module d'interprétation

Cette interprétation se fait relativement à la console virtuelle afin de ne pas avoir à redéfinir un interpréteur pour chaque type de console employée. Un interpréteur est composé :

. d'un module de communication avec l'utilisateur,

. d'un module d'interprétation proprement dit,

. d'un module graphique qui fait l'interface avec le LGB.

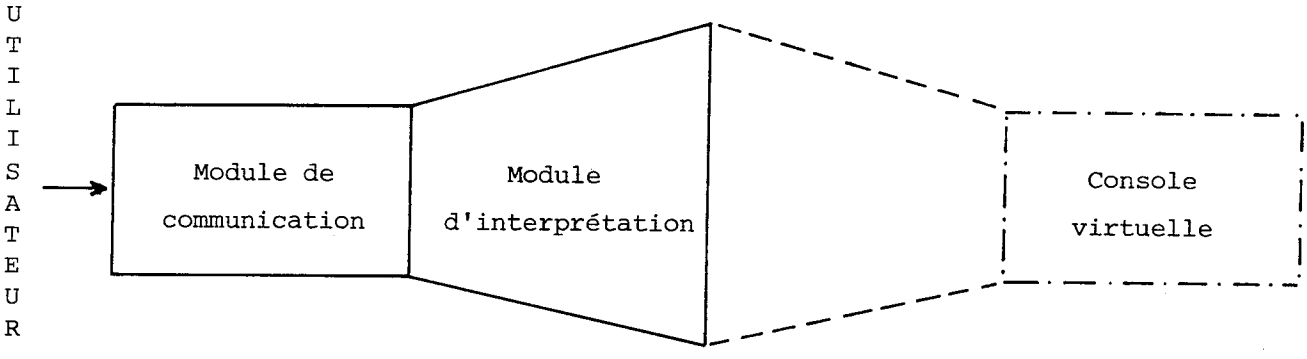


Figure 1

IV.1.3.1.1. Le module de communication

Il permet à l'utilisateur de communiquer avec le système.

Etant données les caractéristiques essentielles d'un dessin (cf. chapitre II) et sans préjuger des outils réels de dialogue (photostyle, claviers, ..), on peut définir en termes fonctionnels les actions relatives à la console virtuelle nécessaires :

. désignation d'un point de l'écran,
. identification d'un dessin ou d'un objet,
. introduction de valeurs alphanumériques,
. sélection d'une commande.

a) Désignation d'un point de l'écran

DESIGNATION(X,Y).

Etant donné que le système sera essentiellement conversationnel et qu'il s'agira de dessins à transmettre relativement simples, on considère que tout le dessin est entièrement affiché dans le système écran virtuel. On laisse donc de côté les problèmes de découpage d'un dessin.

b) Identification d'un dessin ou d'un objet

(objet contenu dans le dessin courant).

. IDENTDESSIN(N) ; N = numéro dans la table des dessins.

Tout dessin mémorisé est reconnu par un numéro dans la liste des dessins.

. IDENTOBJET(NV).

Tout objet du dessin courant est reconnu par un numéro virtuel puisque celui-ci est associé bijectivement à un objet. Quelle que soit la façon dont on le désignera, cela reviendra à déterminer son numéro virtuel.

c) Introduction de valeurs alphanumériques

Un certain nombre de zones seront réservées à l'introduction de valeurs alphanumériques par : RESERV(NV,ZONE,X,Y).
NV est le numéro associé à la zone (association bijective),
ZONE décrit la zone réservée (pointeur),
X, Y donnent la position du début de la zone sur la console virtuelle.

L'introduction de valeurs se fait ensuite par appel d'une fonction de la forme : LIRE(NV,CHAINE),
NV étant le numéro virtuel associé à une zone réservée, l'entrée effectuée est alors rangée dans cette zone.

d) Sélection de commandes

SELECTION(NOMBRE,LISTE).

La liste est formée d'une suite de nombres permettant d'atteindre une liste d'actions. Le nombre permet d'accéder par la liste à une action précise. L'entrée du nombre peut être réalisée par exemple par désignation sur un menu.

e) Message

ECRIRESB(NV,MESSAGE).

NV fait référence à une réservation faite auparavant (cf. § c). La zone indiquée dans cette réservation est remplie avec le message donné puis est éditée.

f) Affichage

AFFICHAGE(NV).

Un affichage est uniquement défini par le numéro NV de l'objet à afficher. Ce numéro permet de retrouver tous les éléments nécessaires à cet affichage.

g) Effacement

EFFACEMENT(NV).

De même que pour un affichage, le numéro NV de l'objet considéré suffit pour effacer cet objet.

IV.1.3.1.2. Module d'interprétation

Son rôle est d'interpréter relativement à la console virtuelle une série de données qui lui sont fournies, soit par le module de communication, soit par l'interface avec le logiciel chargé de la gestion de la réunion. Pour les données venant du module de communication, il faut reconnaître la donnée puis, si elle est valide, l'interpréter. Les types d'actions envisageables sont :

- Construction d'un dessin, comprenant :
  - constructions et modifications d'objets,
  - modification globale d'un dessin,
  - mémorisation et appel d'un dessin,
- Préparation d'un commentaire (mémorisation possible),
- Participation à la réunion, comprenant :
  - commandes particulières (début, fin, ..),
  - transmission d'interventions (dessins ou commentaires).

Les données venant de l'interface avec le logiciel de gestion de la réunion peuvent être de trois sortes :

. renseignements sur le déroulement de la conversation : l'interpréteur décode ces renseignements et informe le participant par un message (exemple : ARRIVEE DE X).

. intervention transmise par un participant présent : l'interpréteur décode l'information puis la passe au module graphique pour affichage.

. retour dans la conversation : il s'agit d'une intervention que le participant veut revoir ; l'interpréteur la décode et la fait afficher comme *dessin* ou *commentaire* courant afin que l'utilisateur puisse l'accéder.

On conçoit donc l'existence d'un petit moniteur qui permettra de décider quelles données interpréter. Les données fournies à l'interface avec le logiciel de gestion de la réunion sont de la forme suivante :

. transmission d'intervention :

il faut fournir au module de gestion de la réunion l'intervention codée dans un buffer. La fonction sera de la forme : INTERVENTION(Ui,BUFFER).

. Retour dans la conversation :

il s'agit de lire dans le fichier de conversation une intervention donnée que l'interpréteur récupèrera dans un buffer. La fonction sera de la forme : RETOUR(Ui,Ij,BUFFER).

. Autres commandes :

il s'agit de spécifier au module de gestion de la réunion un changement d'état par rapport à la réunion. Les deux principales commandes sont :

ARRIVEE(Ui,nom de passe),

DEPART(Ui).

Figure 2

Schéma d'interprétation de données
venant du module de communication :

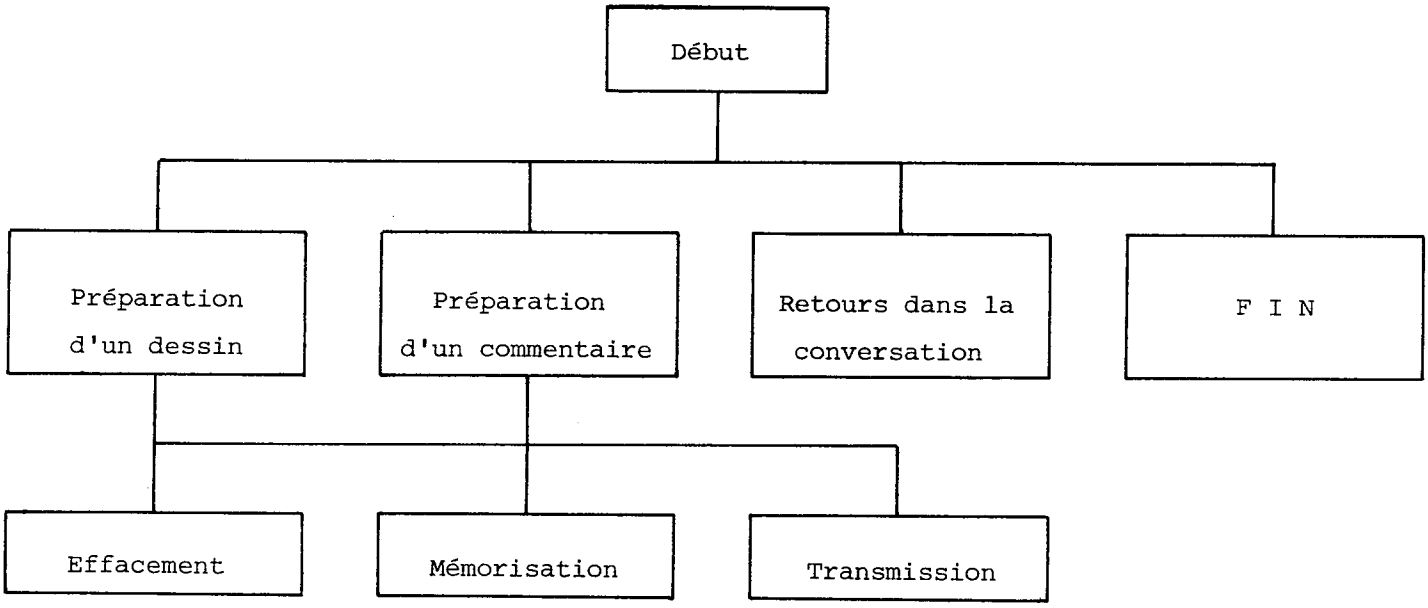
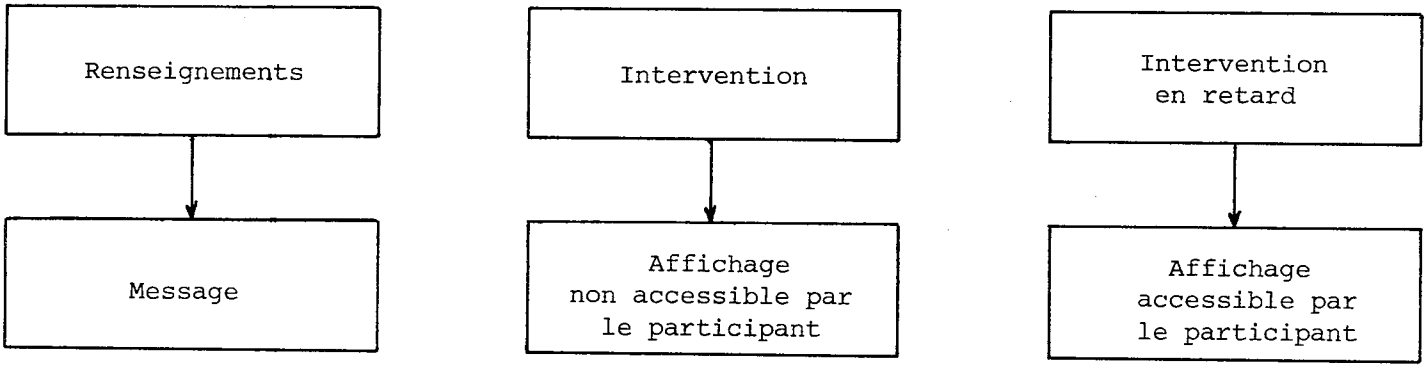


Schéma d'interprétation des données
venant de l'interface avec le logiciel de gestion de la réunion :



IV.1.3.1.3. Module graphique

On place ce module à l'intérieur du module d'interprétation, car il ne pourra être appelé que par lui. Il sera constitué d'un ensemble de procédures de mise en forme pour passer les données au LGB. On peut donc considérer qu'il est indépendant de la console utilisée, si le LGB répond bien aux conditions suivantes :

. construction de segments à partir de tableaux de points,

. construction de chaînes à partir d'une abscisse, d'une ordonnée et de la chaîne proprement dite.

. Affichage d'une figure :

il s'agit de reconstituer segment par segment une figure à partir d'une fonction : f(ID,NO,X,Y,R,D,M) et d'une zone de débordement SEG pour les figures déclarées segment par segment. On a vu dans le chapitre II qu'il était extrêmement facile de reconstituer une figure segment par segment à partir d'une telle fonction. On pourra donc écrire une procédure ayant en paramètre les variables de la fonction f et la zone de débordement. Cette procédure déterminera ensuite le traitement adéquat à partir du type de figure.

. Affichage d'un texte :

On reconstitue les données nécessaires au texte à partir d'une fonction : t(ID,NO,X,Y,R,D) et d'une zone de chaîne. On pourra placer ce traitement parmi les traitements de la procédure de mise en forme de figure, en ajoutant une chaîne en paramètre.

. Affichage de liens :

il s'agit de reconstituer les liens segment par segment, à partir d'une fonction l(N1,N2,XB,YB,M). On verra dans les exemples de problèmes graphiques rencontrés la façon de construire ces liens. Rappelons simplement la signification de XB et YB :

si (XB = O) et (YB = O) alors lien direct
sinon si (XB = O) alors lien avec barre horizontale
sinon lien avec barre verticale

. Affichage de figures système :

Il s'agit de figures particulières, pouvant représenter les échelles, le curseur, ...

IV.1.3.2. Module de gestion de la réunion

Il recevra des informations provenant du module d'interprétation. Il est chargé en particulier de gérer le fichier de conversation. C'est donc à son niveau que l'on devra résoudre les problèmes d'accès exclusif et de protection que l'on a vus dans le paragraphe consacré aux fichiers. Les types d'actions essentiels seront :

. demande de lecture de l'intervention Ik dans le fichier de conversation provenant de l'utilisateur Ui : vérification des droits d'accès de Ui sur Ik. En cas de droit à la lecture, passer cette intervention à Ui par l'intermédiaire d'un buffer. Une demande de lecture est fonctionnellement représentée par :

DEMLECTUR(Ui,Ik),

la fourniture de cette intervention étant :

INTLUE(Ui,BUFFER).

. Demande d'écriture d'une intervention en fin de fichier de conversation; cette intervention est fournie dans un buffer : vérification que personne n'est en train d'écrire, sinon on met le demandeur dans la file d'attente :

DEMECR(Ui,BUFFER)

On fournit l'intervention à tous les participants présents :

j présent : INTERV(Uj,Ui,BUFFER).

. Commandes de début ou de fin :

ARRIVEE(NUMERO,NOM)

DEPART(NUMERO,NOM)

Il s'agit alors de gérer, à partir du type de commande et des renseignements fournis, la liste des participants.

IV.2. LES OUTILS

On décrit dans cette partie le matériel que l'on a utilisé. On peut le décomposer en trois groupes :

. la machine et le système d'exploitation sous lequel sera implémenté CONGRES,

. les consoles,

. les programmes de service que l'on pourra utiliser.

Pour le choix de la machine et du système d'exploitation, on a essayé de respecter les lois que l'on a exposées dans le chapitre II.

Au sujet des consoles, on se contentera de décrire celles sur lesquelles CONGRES a été utilisé et on se limitera à une description externe, le lecteur intéressé pouvant se reporter aux nombreux ouvrages et fiches techniques les décrivant avec précision.

Le seul programme de service que l'on décrira d'un point de vue utilisateur est le logiciel graphique GRIGRI développé à Grenoble par l'Equipe Graphique de l'Université Scientifique et Médicale.

IV.2.1. CP 67 et accès par CYCLADES

● Le système CP 67 est en service au CICG sur un IBM 360/67 de 1024 K de mémoire centrale. CP *simule*, pour chaque utilisateur, un 360 qui est activé périodiquement. Il comprend en particulier :

. une mémoire continue où l'horloge et les double-mots pour échange de mots d'état lors d'une interruption sont simulés ;

. une portion de disque réel ;

. des unités virtuelles telles que lecteur, perforateur, simulés par spooling ;

. des unités réelles pouvant être attachées par l'opérateur.

Le transfert par spooling est le seul véritable moyen de communication entre machines virtuelles (si l'on excepte le partage de cylindres). On indique à CP que les sorties du perforateur virtuel doivent être dirigées vers le lecteur virtuel d'une autre machine virtuelle.

Un certain nombre de terminaux (2741 IBM, TEKTRONIX, TELETYPE, ..) sont connectés directement au 360/67 alors que le 2250 IBM, par exemple, doit être attaché à une machine virtuelle par l'opérateur.

● Accès par CYCLADES :

le but du réseau hétérogène général d'ordinateurs CYCLADES est l'expérimentation en vraie grandeur d'un réseau reliant des laboratoires de facultés. Se connecter au réseau pour un site, veut dire en particulier mettre les services dont il dispose localement à la disposition d'utilisateurs distants. Le 360/67 IBM du Centre de Calcul Interuniversitaire de Grenoble fonctionne alternativement avec les systèmes CP 67 (7 h - 12 h, 15 h - 20 h) et ASP/MVT. La connexion du 360 sous CP 67 à CYCLADES a été présentée par J.P. ANSART [AN].

La configuration d'une machine virtuelle est en général la suivante :

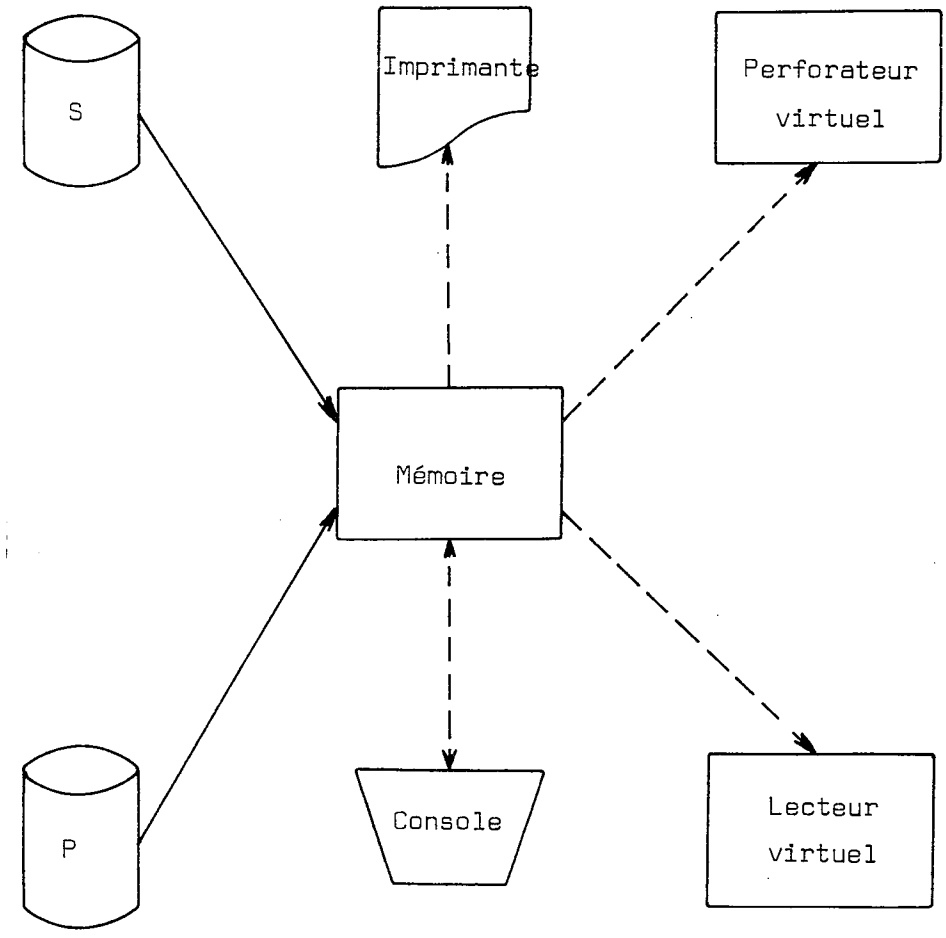


Figure 3

CYCLADES est composé de deux niveaux :

. le réseau de communication : constitué de CII MITRA 15 reliés entre eux par des lignes téléphoniques dont la fonction est de transmettre un datagramme d'une origine à une destination, avec un délai de transfert variable mais borné et un risque de perte non nul mais faible ;

. les ordinateurs connectés (CII 10070, CII IRIS 80, CII IRIS 50, IBM 360/67, CDC 6600 ...) : les utilisateurs de deux sites distants communiquent par le réseau de transport auquel ils accèdent par une station de transport située dans chaque site.

Dans le cas de la connexion du 360 sous CP 67, la station de transport est réalisée grâce à une machine virtuelle spécialisée. L'accès qui nous intéresse depuis un terminal conversationnel à CP 67 est le service TS CP 67.

. Justification du choix de CP 67 :

On a vu dans le chapitre II un certain nombre de caractéristiques que devrait avoir le système choisi. Sans revenir en détail sur chaque point, on remarque que CP 67 répond aux exigences suivantes :

. système à partage de temps,

. système accessible à distance via un réseau,

. possibilité de communication entre machines virtuelles,

. possibilité d'accéder en lecture à des programmes communs.

Remarque :

il est évident que CP n'est pas le seul système à partage de temps répondant aux lois évoquées dans le chapitre II. Des systèmes tels que SIRIS 8 peuvent convenir (exemple, à Grenoble : SIRIS 8 sur IRIS 80 accessible par CYCLADES), mais certains choix de réalisation peuvent être remis en cause.

IV.2.2. Les consoles

D'un point de vue externe, les consoles se répartissent en deux groupes :

. les consoles graphiques qui permettent l'affichage de points, vecteurs et caractères en tout point de l'écran ;

. les consoles alphanumériques qui ne permettent l'affichage de caractères qu'en certains points. L'affichage se fait sur papier ou sur écran.

Le but de cette partie n'est pas de décrire tous les types de consoles d'un point de vue technique, mais de voir les principaux aspects (essentiellement utilisation) en particulier des consoles disponibles à Grenoble.

IV.2.2.1. Les consoles graphiques

Nous décrivons sommairement trois consoles disponibles à Grenoble et sur lesquelles le système CONGRES a été utilisé : le 2250 IBM, le TEKTRONIX 4010 et le TEKTRONIX 4015.

a) Le 2250 IBM

Il se compose essentiellement d'un écran cathodique, d'une mémoire d'entretien et des dispositifs de communication (clavier alphanumérique, photostyle, clavier de touches de fonction, tablette sylvania).

. La mémoire d'entretien :

d'une taille de 8192 octets, elle renferme la liste de visualisation et décharge l'ordinateur de la fonction de régénération. Les communications entre l'unité centrale et cette mémoire d'entretien n'ont lieu que lors d'une modification d'image ou de lecture de données précédemment transmises à la mémoire d'entretien par l'intermédiaire des moyens de communication.

. L'écran cathodique :

la partie utile est un carré de 30 cm x 30 cm, avec 1024 x 1024 points adressables. Le temps d'affichage d'un point (et de points voisins) est de l'ordre de 16,8 µs. Un générateur de caractères permet de dessiner des petits caractères (52 lignes de 74 caractères ; temps d'affichage d'un caractère : 14 µs) et des grands caractères (35 lignes de 49 caractères ; temps d'affichage d'un caractère : 16 µs).

. Les moyens de communication :

parmi ceux cités précédemment, celui qui nous intéresse essentiellement (pour des raisons que l'on a vues au chapitre II) est le clavier alphanumérique. Un curseur peut être manipulé et matérialise l'endroit où sont entrées les données au clavier alphanumérique. Il existe en particulier des touches permettant de faire reculer ou avancer ce curseur.

b) Le TEKTRONIX 4010

Il se compose d'un écran cathodique, d'un clavier alphanumérique et d'un dispositif à molettes pour désigner un point sur l'écran.

L'écran a une dimension de 19,05 cm x 14,22 cm avec 1024 x 1024 points adressables. Le temps de tracé d'un vecteur est de 2,6 ms. L'écran peut contenir 35 lignes de 74 caractères en mode alphanumérique.

Le clavier alphanumérique : un curseur indique le point d'entrée de données au clavier alphanumérique qui comporte 63 caractères.

L'inconvénient essentiel du TEKTRONIX 4010 par rapport au 2250 IBM est l'absence de mémoire d'entretien, ce qui implique qu'à chaque fois que l'on efface une partie de l'image, il faut tout effacer et tout régénérer.

c) Le TEKTRONIX 4015

Il se compose d'un écran cathodique, d'un clavier alphanumérique et d'un dispositif à molettes pour désigner un point sur l'écran.

L'écran a une dimension de 38,1 cm x 27,9 cm, avec 1024 x 1024 points adressables et possède le trait continu, le trait pointillé, etc ..

En mode alphanumérique, quatre tailles de caractères peuvent être choisies (74 caractères par ligne et 35 lignes, ou 81 caractères par ligne et 38 lignes, ou 121 caractères par ligne et 58 lignes, ou 133 caractères par ligne et 64 lignes).

IV.2.2.2. Les consoles alphanumérigues

Elles peuvent se répartir d'un point de vue externe en deux groupes suivant que l'affichage se fait sur papier ou sur écran :

. Consoles papier (2741 IBM) : elles sont en général lentes et bruyantes. Le seul moyen de communication est le clavier alphanumérique.

. Consoles écran (IRISCOPE, SYNECRAN, ..) : elles ont l'avantage d'être silencieuses et, suivant la vitesse de la ligne de connexion, peuvent avoir un affichage rapide. Le seul moyen de communication est le clavier alphanumérique, un curseur indiquant l'endroit où sont entrées les données (ligne courante). Leur principal inconvénient est de n'avoir qu'un nombre limité de lignes (souvent 16 ou 24).

IV.2.3. Programmes de service

Comme on l'a vu au chapitre II, il suffit de choisir comme langage de programmation un langage de haut niveau pouvant communiquer avec l'assembleur. Le langage ALGOLW dispose sous CP 67 de la possibilité de communiquer avec l'assembleur ; il admet les procédures précompilées, ce qui permet d'utiliser le logiciel graphique GRIGRI [LE], [LA],[LL].

GRIGRI comporte un ensemble de fonctions permettant d'utiliser le 2250 IBM et le TEKTRONIX 4010 à partir de programmes écrits en ALGOLW ou en FORTRAN. Les procédures qui nous intéressent sont les suivantes :

. déclaration de figure,

. génération d'objets,

. effacement de figure,

. outils de communication,

avec les définitions suivantes :

Objet :

plus petite chose manipulable (point ou suite de points, vecteur ou suite de vecteurs, ligne de texte) ;

figure :

espace de travail associé à un système de coordonnées dans lequel sera défini un ensemble d'objets.

(Ces définitions ne doivent pas être confondues avec celles données dans le chapitre II et qui sont utilisées dans toute la suite).

Les principales caractéristiques du logiciel GRIGRI sont les suivantes :

. logiciel *graphique interactif*

. logiciel *indépendant* du contexte d'utilisation.

Remarque :

la version utilisée pour la réalisation de CONGRES est la première version de GRIGRI, moins complète que la seconde version qui n'est disponible que depuis le deuxième trimestre 1977.

IV.3. CONCEPTION ET RÉALISATION DE CONGRES

Cette troisième partie explique les aspects techniques de la conception et de l'implémentation du système CONGRES.

On a déjà vu au chapitre II et au début du présent chapitre, un certain nombre de choix fondamentaux dans la conception du système de réunion. On va maintenant entrer un peu plus dans les détails, en essayant de bien faire ressortir ce qui, du point de vue de la conception de CONGRES, est lié au système d'exploitation sous lequel on travaille. Le choix de CP impose immédiatement les réflexions suivantes :

. chaque participant dispose d'une machine virtuelle qui lui est propre et sur laquelle on peut dérouler un programme indépendamment de ce que font les autres participants ;

. les communications entre participants reviendront à des transferts entre leurs machines virtuelles respectives ;

. on peut conserver les programmes principaux sur une machine virtuelle accessible en lecture, chaque participant en recevant une copie lors de son entrée dans le système, copie détruite dès qu'il se désactive pour CONGRES.

Si l'on revient à la décomposition du logiciel en deux parties, exprimée au début de ce chapitre :

. un module interprétation,

. un module gestion de la réunion,

on constate que ce schéma peut être précisé.

Le module d'interprétation existe toujours et sera chargé sur chaque machine virtuelle activée par un participant. Cependant, ce module doit communiquer d'une part avec le module de gestion de la réunion, et d'autre part il peut avoir à communiquer avec les autres machines virtuelles participant à la réunion. Afin de résoudre ce problème de communications entre machines virtuelles, on définira un module associé au module d'interprétation qui sera chargé de gérer les communications relatives à la machine virtuelle sur laquelle il se trouve.

On peut donc décomposer le logiciel en quatre modules dont le fonctionnement sera explicité dans la suite :

. SUPGEN (superviseur général) :
chargé de gérer la réunion, il doit :
- gérer les départ et les arrivées des participants,
- gérer le fichier de conversation,
- jouer le rôle d'animateur dans une réunion "normale" ;

. INTER (interprétation) :
c'est le module chargé de :
- communiquer avec les participants,
- interpréter les commandes par rapport à la console virtuelle,
- interpréter les informations venant du SUPGEN via le SUPLOC,
- appeler les procédures graphiques nécessaires ;

. SUPLOC (superviseur local) :
son rôle est de gérer les entrées/sorties relatives à une machine virtuelle ; il peut :
- recevoir de l'information venant de l'interpréteur, qu'il devra transmettre soit à d'autres machines virtuelles participantes, soit au SUPGEN,
- gérer les interruptions d'entrées/sorties pouvant provenir de transmissions d'autres machines virtuelles du SUPGEN et fournir l'information nécessaire à INTER ;

. module graphique :
c'est le module contenant les procédures graphiques dont le rôle est de mettre en forme les données provenant de l'interprétation relative à la console virtuelle, afin de les passer sous une forme propre aux procédures du LGB. Cela consiste essentiellement à remplir des tableaux de réels représentant des points ou des segments (définis par deux points).

La conception et l'implémentation du SUPGEN doit se faire suivant les deux lois ci-après :

Loi 1 :
il est unique : dupliquer le SUPGEN sur chacune des machines virtuelles participantes serait d'une part une perte de place, d'autre part source d'erreurs car il faudrait à chaque instant assurer la compatibilité entre les différentes copies (accusés de réception, etc ..).

Loi 2 :
il doit être actif au moins à chaque fois qu'un participant est dans le système.

Les solutions envisagées ont été les suivantes :

1) Le SUPGEN est sur la machine virtuelle d'un participant : cette solution est à rejeter, car le participant donné n'étant pas toujours présent, la loi 2 ne sera pas toujours respectée.

2) Le SUPGEN est alloué dynamiquement à la machine virtuelle qui se connecte la première : cela revient à dire que si personne n'est présent, le SUPGEN ne sera pas actif. Dès qu'une machine virtuelle exécute la procédure d 'entrée dans CONGRES, le SUPGEN est mis sur cette machine. Tout le problème consiste alors à déterminer quel est le premier arrivé, ce qui implique bien qu'une partie du logiciel soit active à tout instant et il faudra gérer dynamiquement l'adresse du SUPGEN (nom de la machine virtuelle sur laquelle il se trouve). Enfin, lorsque le participant travaillant sur la machine virtuelle où le SUPGEN est actif s'en va, se pose à nouveau le problème de l'allocation du SUPGEN à une autre machine virtuelle.

3) Le SUPGEN est sur une machine virtuelle spéciale qui sera activée à chaque fois qu'il peut y avoir participation. Cette machine virtuelle sera donc active et le programme SUPGEN sera en attente d'interruptions venant des machines participantes. C'est la solution adoptée car elle répond bien aux lois 1 et 2. Cette partie du logiciel ne sera pas directement en rapport avec les consoles et elle sera donc indépendante de toutes manières des terminaux utilisés par les participants.

L'interpréteur de commande INTER sera un programme unique, conservé sur une machine virtuelle accessible en lecture. C'est la partie du logiciel qui devra être aussi indépendante que possible du matériel utilisé (consoles). Cette indépendance sera résolue par la définition d'une console virtuelle et d'un logiciel graphique de base (tracé de points, vecteurs, caractères, lecture de caractères).

Le SUPLOC sera également un programme unique conservé sur une machine virtuelle accessible en lecture. Son but essentiel est de résoudre les problèmes de communication entre machines virtuelles. Un petit moniteur de multiprogrammation chapeautera INTER et SUPLOC.

D'où le schéma suivant (le module graphique est compris dans INTER) :

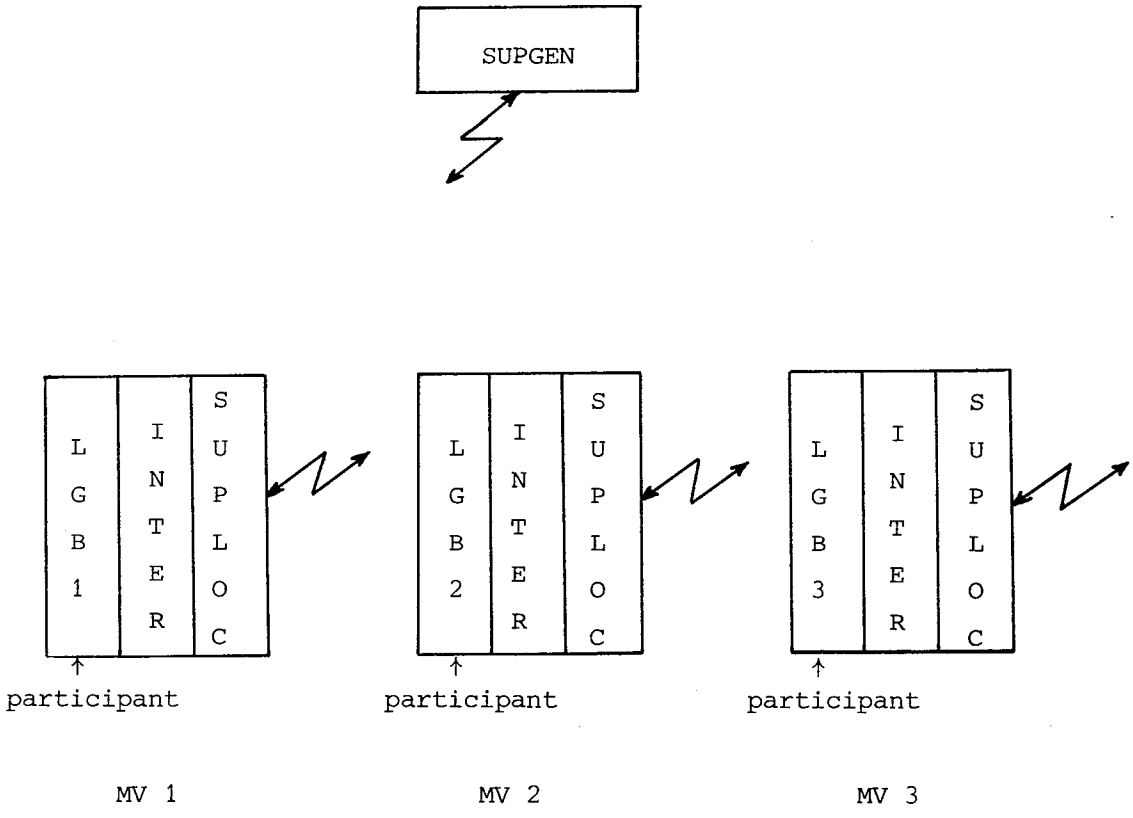


Figure 4

LGBi = logiciel graphique de base pour la console i (les logiciels de base pour des consoles différentes sont en général différents).

Il restait un choix à faire : si toutes les commandes sont forcément transmises entre un SUPLOC et le SUPGEN (demande de participation, etc ..), on pouvait se demander s'il fallait faire transiter les interventions par le SUPGEN (figure 5) ou charger les SUPLOC de les transmettre directement aux autres SUPLOC actifs. Aucune méthode ne semble vraiment comporter des avantages ou des inconvénients très supérieurs à l'autre. Si l'on a choisi la deuxième solution, c'est plus pour des raisons historiques que pour une supériorité de celle-ci sur celle-là. En effet, au tout début du projet, on voulait implémenter un système de réunion copié sur le face à face, c'est-à-dire que tout le monde est présent, d'où le rôle très faible du SUPGEN au point de vue interventions, puisqu'il n'avait même pas à les conserver pour les transmettre aux participants après une absence. (Ce choix peut d'ailleurs très facilement être inversé dans la réalisation de CONGRES).

Nous allons dans toute la suite de ce chapitre, expliciter les points suivants :

. définition d'une étape intermédiaire pour assurer à un certain niveau l'indépendance du logiciel par rapport au matériel, que l'on nommera console virtuelle ;

. présentation de l'interpréteur de commandes qui opèrera en fonction de la console virtuelle ;

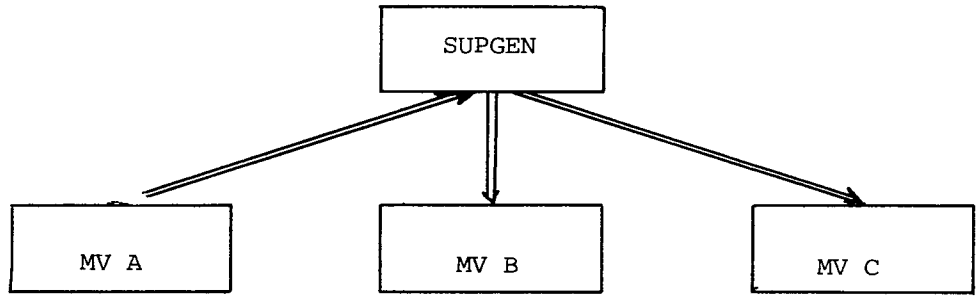
. résolution de quelques problèmes graphiques ;

. présentation des composants permettant la tenue d'une réunion (SUPLOC, SUPGEN) et spécification d'une réunion normale (synchrone) ;

. déroulement d'une réunion ;

. possibilités d'extension.

Transmission d'une intervention de A vers B et C transitant par le SUPGEN :



Transmission d'une intervention de A vers B et C directement :

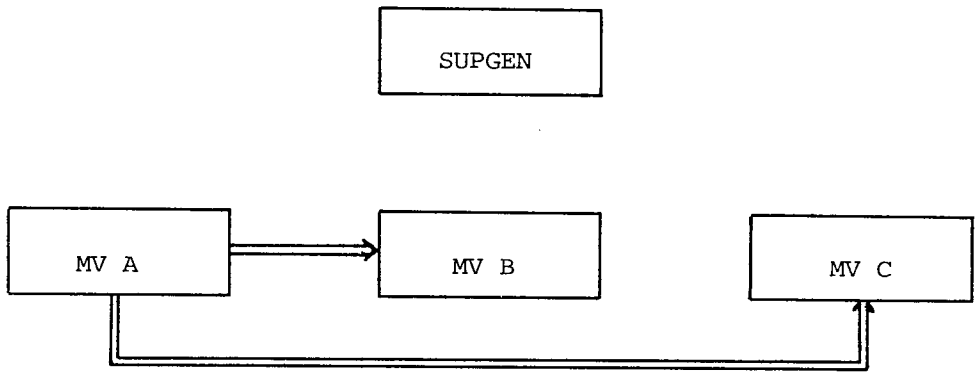


Figure 5 - Transmissions d'interventions

IV.3.1. La console virtuelle

Sous cette dénomination, on définit l'étape intermédiaire entre l'interpréteur de commandes et la console réelle. L'interpréteur sera donc défini relativement à cette console virtuelle, ce qui assurera son indépendance, à un certain niveau, par rapport au matériel. A cette console virtuelle sont associés :

- un écran virtuel : cet écran sera projeté sur les différents écrans réels,
- une structure de données.

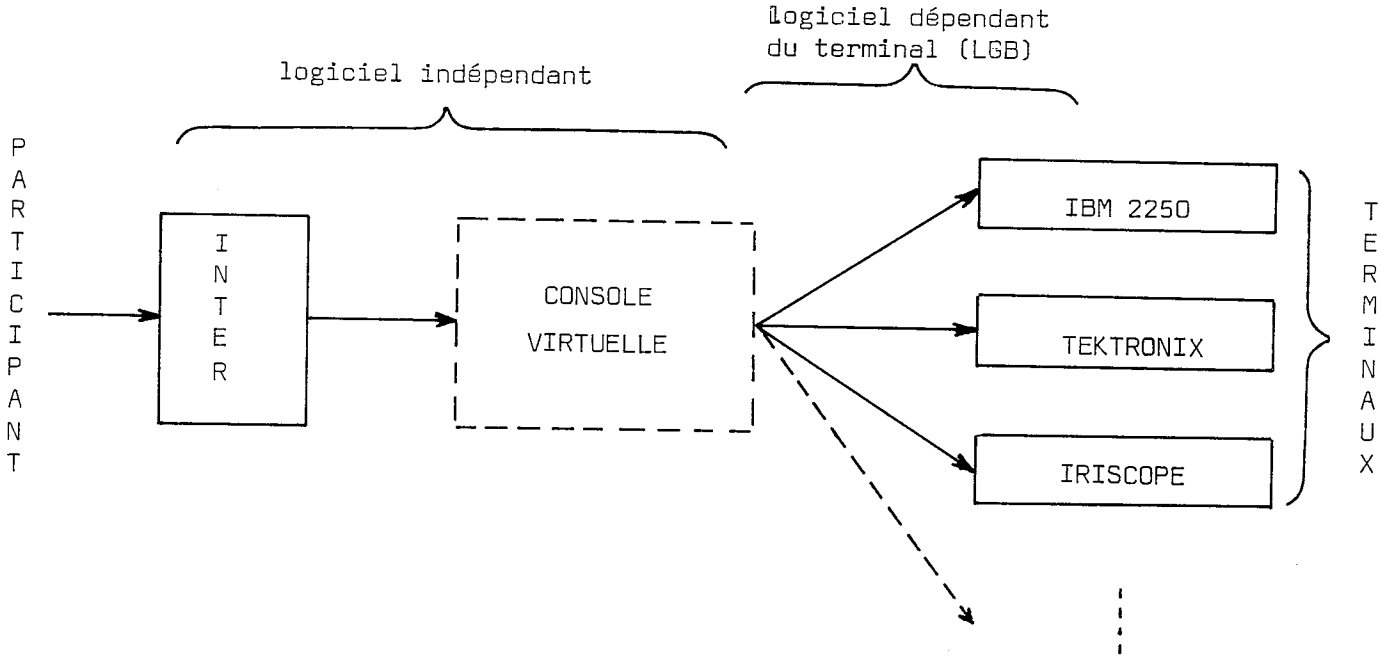


Figure 6

IV.3.1.1. L'écran virtuel

L'écran virtuel est un carré de 1024 x 1024, divisé en une partie supérieure (1024 x 512) réservée à la conversation et une partie inférieure (1024 x 512) réservée à la préparation des interventions (travail personnel). Pour la partie supérieure, l'unité logique est donc l'INTERVENTION, alors que pour la partie inférieure c'est l'OBJET.

Des numéros virtuels (NV) vont servir à reconnaître ce qui est affiché. Les numéros virtuels seront associés aux unités logiques, exception faite pour le demi-écran supérieur. On définit en effet huit unités particulières qui sont :

. NV = 1 : le demi-écran supérieur : il est interdit de modifier directement une intervention émise. Aucune unité logique du demi-écran ne peut donc être modifiée, ce qui permet de considérer la partie réservée à la conversation comme un tout du point de vue graphique ;

. NV = 2 : échelle verticale ;

. NV = 3 : échelle horizontale ;

. NV = 4 : barre horizontale ou verticale : un NV pour les deux barres suffit, car à un instant donné, au plus une barre peut être courante (donc affichée) ;

. NV = 5 : ligne utilisateur : c'est la ligne réservée à l'entrée des commandes ;

. NV = 6 : ligne système bas: reçoit les réponses de l'interpréteur d'un point de vue local ;

. NV = 7 : curseur ;

. NV = 8 : ligne système haut : reçoit les réponses d'un point de vue conversation (réponses de CONGRES).

Tous les objets créés ultérieurement par l'utilisateur recevront un numéro virtuel (à partir de 9). Afin de distinguer les figures (ou textes) des liens (cf. Structure de données), les numéros virtuels sont définis de la façon suivante :

soit le lien Li de rang i (i ème affiché)

soit la figure (ou le texte) Fj de rang j (j ème affiché),

on définit de manière arbitraire, en considérant qu'il y aura environ deux fois plus de figures et textes que de liens, le tableau suivant :

NV	9	10	11	12	13	14	15	16	17	18	19	20	...
	L1	F1	F2	L2	F3	F4	L3	F5	F6	L4	F7	F8	...

d'où les deux calculs :

. pour un lien :

$NV = 9 + 3\ (i - 1) = 3i + 6$

. pour une figure (ou un texte) avec les notations DIV (division entière) et REM (reste de la division entière) :

$NV = 3((j + 1)\ DIV\ 2) + 7 + (j + 1)\ REM\ 2$

$= (j + 1)\ DIV\ 2 + j + 8$

IV.3.1.2. Structure de données

Les structures de données associées aux demi-écrans supérieur et inférieur, sont différentes de par leur finalité même :

. pour le demi-écran supérieur, il suffit de pouvoir afficher une nouvelle intervention et aucune modification n'est possible ;

. pour le demi-écran inférieur, il faut pouvoir associer des figures (liens), ajouter ou modifier des objets, etc ..

La figure 9 explicite la structure de données générale et la figure 8 précise la structure associée à un dessin à un moment donné de sa construction (liste de visualisation).

IV.3.1.2.1. Demi-écran supérieur (figure 7)

Lors de la réception d'une intervention, la procédure est globalement la suivante :

1) lecture du nombre de lignes qu'elle occupe (on verra que ce paramètre est contenu dans un enregistrement associé à l'intervention) ;

2) y a-t-il suffisamment de place pour l'afficher à la suite des autres :
   . si oui, on l'affiche,
   . sinon, on élimine les interventions les plus anciennes, en nombre juste suffisant pour faire de la place à la nouvelle, on remonte les autres du nombre correspondant de lignes et on affiche l'intervention reçue.

La structure de données comprend donc :

. pour chaque intervention affichée, son type (dessin ou commentaire) et le nombre de lignes qu'elle occupe,

. le numéro (par rapport au fichier de conversation) de la première intervention affichée,

. le nombre total de lignes occupées.

Une table des interventions est utilisée pour accéder au fichier de conversation (cf.IV.3.4., le fichier de conversation).

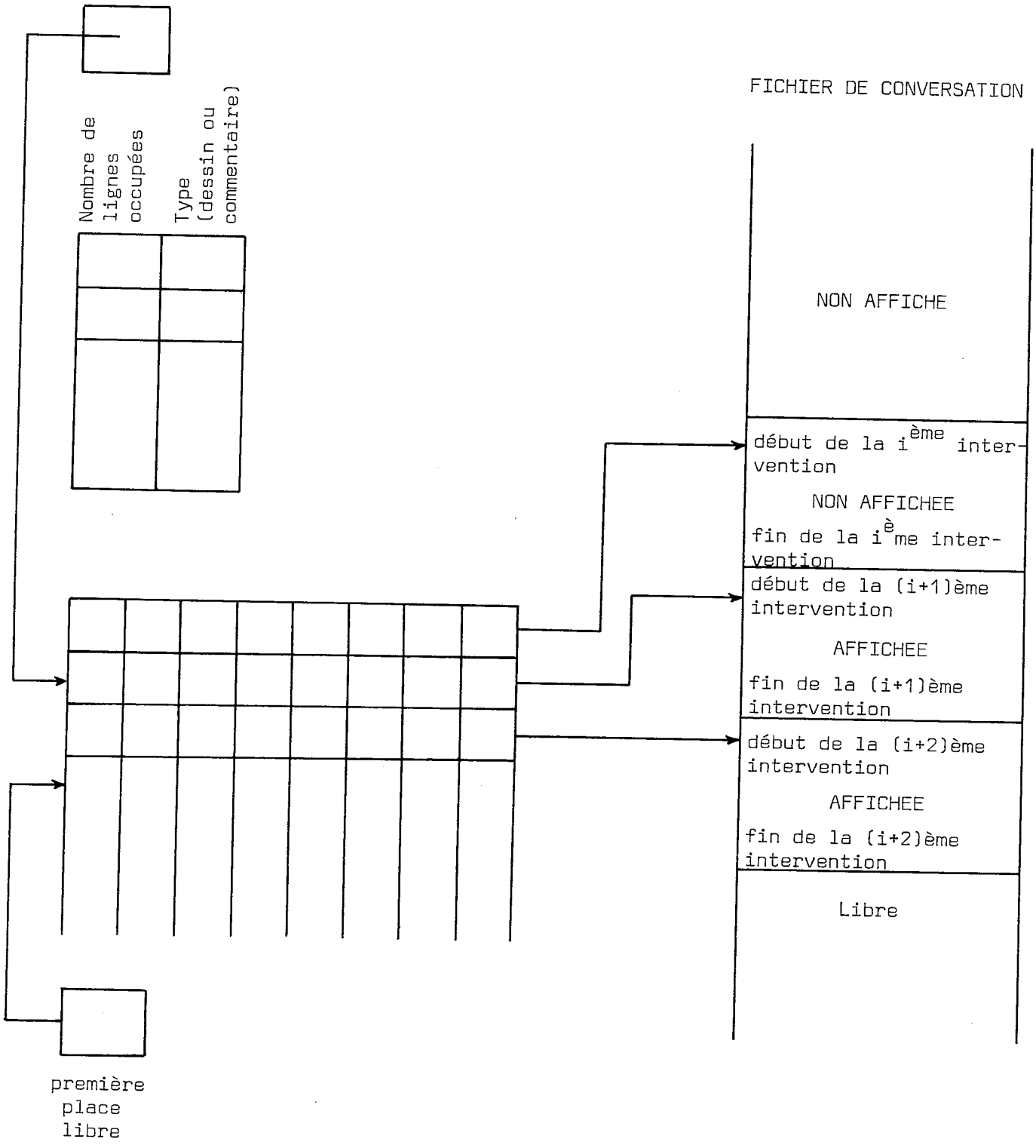

FICHIER DE CONVERSATION
Nombre de lignes occupées
Type (dessin ou commentaire)
NON AFFICHE
début de la i ème inter-vention
NON AFFICHEE
fin de la i ème inter-vention
début de la (i+1)ème intervention
AFFICHEE
fin de la (i+1)ème intervention
début de la (i+2)ème intervention
AFFICHEE
fin de la (i+2)ème intervention
Libre
première place libre

Figure 7

IV.3.1.2.2. Demi-écran inférieur (figure 8)

Cette liste de visualisation comprend tous les éléments que l'on a décrits au § II.5 .1.3. On a vu que les textes et les figures pouvaient être décrits par une même fonction :

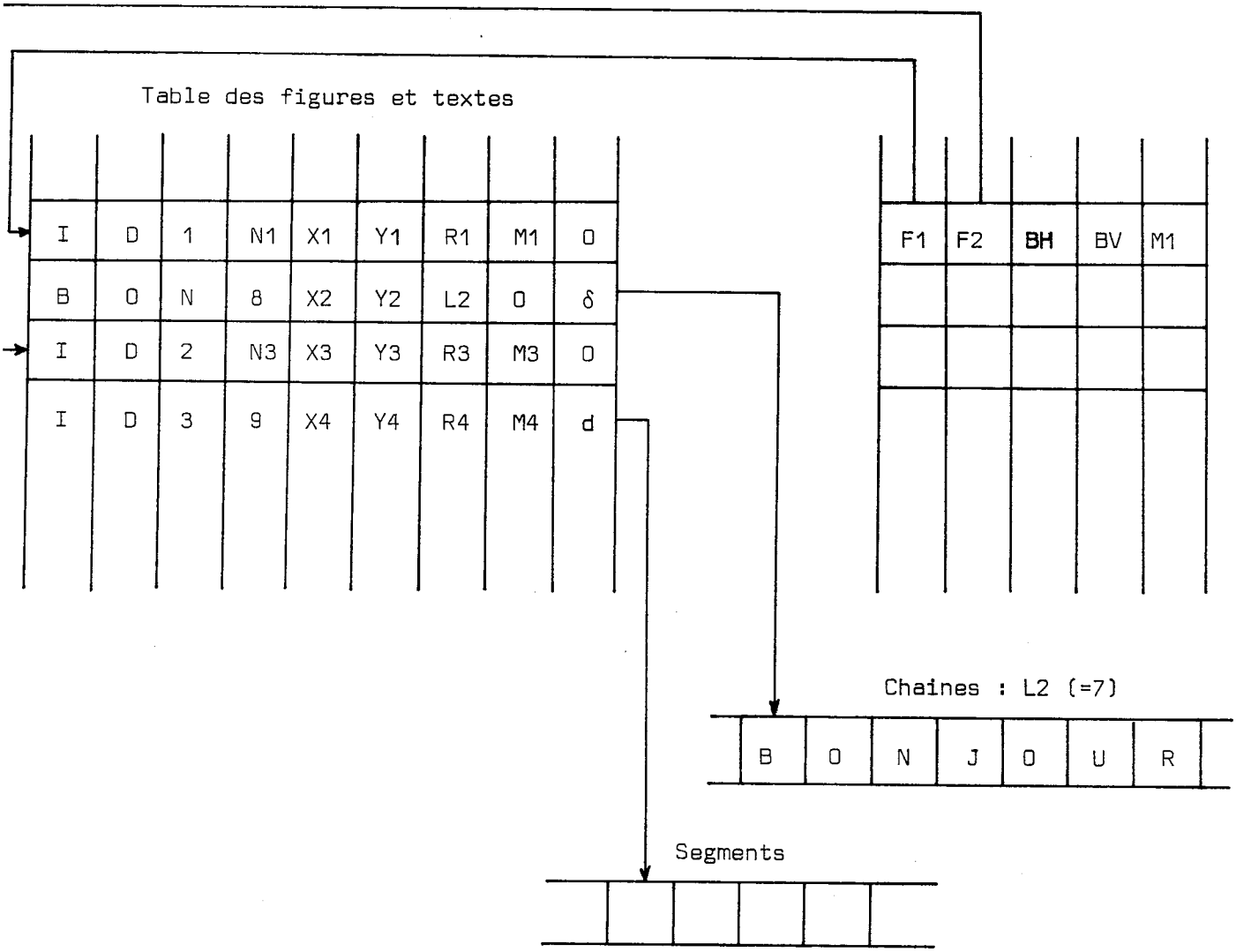
f(ID,NO,X,Y,R,D,M)

A partir de cette fonction, on constitue une table unique des figures et des textes. Elle est complétée par deux zones de débordement :

. une zone contenant toutes les chaînes définissant les textes. On accède pour un texte à sa chaîne de caractères avec le déplacement D dans la table des figures et on connaît le nombre de caractères correspondant par la longueur R ;

. une zone contenant tous les réels définissant les coordonnées des points extrêmités de segments pour une figure déclarée segment par segment. On y accède par le déplacement D et on connaît le nombre de paires de réels par le nombre de segments R.

On constitue une table de corrélations pour résoudre les liens, table qui comprend les déplacements des figures ou des textes concernés dans la table des figures et les renseignements indiquant le type de construction du lien.



ID_i	identificateur de figure (ou de texte)
N_i	numéro de figure
X_i, Y_i	coordonnées du centre pour une figure, du début pour un texte
R_i	rayon (L_i = longueur d'un texte) (ou nombre de segments)
F1, F2	numéros des figures liées
BH, BV	coordonnées de la barre si elle existe
M_i	mode (plein, pointillés, place du nom ...)

Figure 8

Structure de donnée du 1/2 écran inférieur

Liste de visualisation

Signification des modes

Figures :

Plein	Pointillés	
1	11	rien d'affiché
2	12	ident affiché au milieu de la figure
3	13	ident affiché à gauche de la figure
4	14	ident affiché à droite de la figure
5	15	ident affiché au-dessus de la figure
6	16	ident affiché au-dessous de la figure

Liens :

Plein	Pointillés	
1	11	lien sans flèche
2	12	lien avec flèche

Remarque :

le fait qu'il y ait ou non une barre est indiqué par BH et BV, suivant la procédure suivante :

si (BH = 0) et (BV = 0) alors lien direct
sinon si (BV = 0) alors barre verticale
sinon barre horizontale

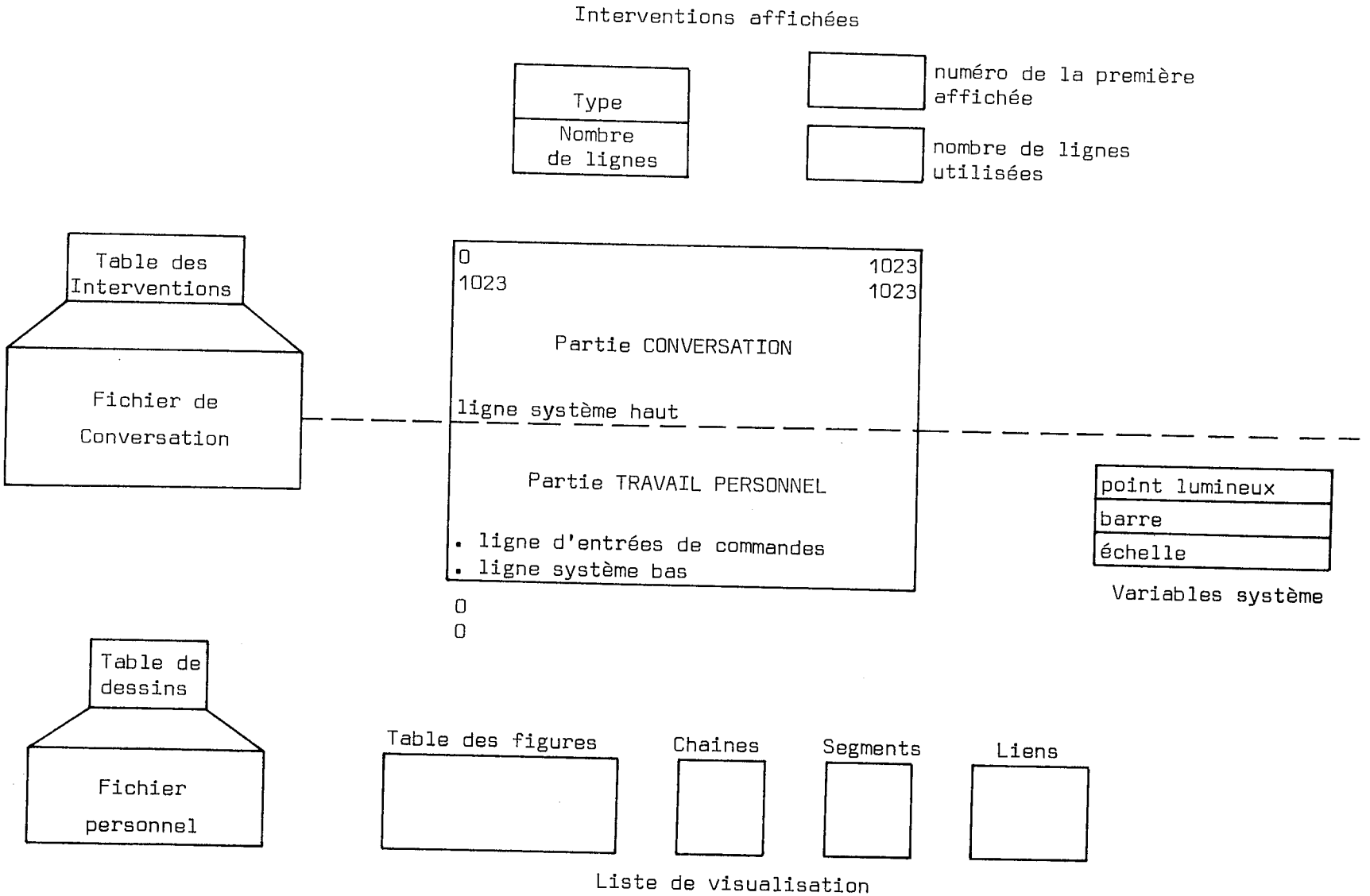


Figure 9

Structure de donnée associée à la console virtuelle

IV.3.2. Interpréteur de commandes INTER

Il peut avoir à interpréter des commandes provenant du module de communication (analyseur lexicographique) ou du SUPLOC. Nous allons étudier dans une première phase l'interprétation des commandes venant du module de communication, puis dans une seconde phase celle des commandes venant du SUPLOC. Nous allons donc voir l'analyseur lexicographique, l'analyseur syntaxique, l'interprétation des commandes utilisateurs, puis l'interprétation des informations venant du SUPLOC.

IV.3.2.1. Analyseur lexicographique

Il lit une ligne complète de commandes (séparées par des blancs).

Il analyse la première commande, passe le contrôle à l'analyseur syntaxique s'il la reconnaît sans erreur, s'arrête et envoie un message dans le cas contraire (et relit la ligne suivante).

Il prend donc le contrôle dans les cas suivants :

. fin d'interprétation d'une commande : il analyse s'il en existe encore :
  - si oui, il analyse la suivante,
  - sinon, il attend une nouvelle ligne ;

. procédures de lecture de paramètres : il peut être appelé par l'interpréteur pour lui fournir les paramètres éventuels donnés entre parenthèses (avec des valeurs par défaut dans la plupart des cas).

IV.3.2.2. Analyseur syntaxique

Il est basé sur un automate d'états. L'analyse est donc simplement : test de la validité de l'état :

- si valide, l'analyseur demande à l'analyseur lexicographique les éventuels paramètres et passe le contrôle à l'interprétation proprement dite ;

- si l'état n'est pas valide pour la commande, l'analyseur syntaxique fait une courte analyse supplémentaire pour sortir un message qui tienne compte de l'état lui-même et de la commande donnée.

Dans l'automate d'état, on peut distinguer essentiellement les environnements suivants :

- DESSIN :

on y entre par la commande DE ; c'est le début de la création d'un dessin ; on en sort par la commande EDessin (effacement de dessin, il est perdu), ou MEmorise (mémorisation de dessin, il est conservé dans le fichier personnel). Dans cet environnement, on peut accéder à trois autres environnements :

. Figure : c'est l'environnement de création de figures et de liens concernant cette figure, dite courante ;

. Texte : c'est l'environnement de création d'un texte ;

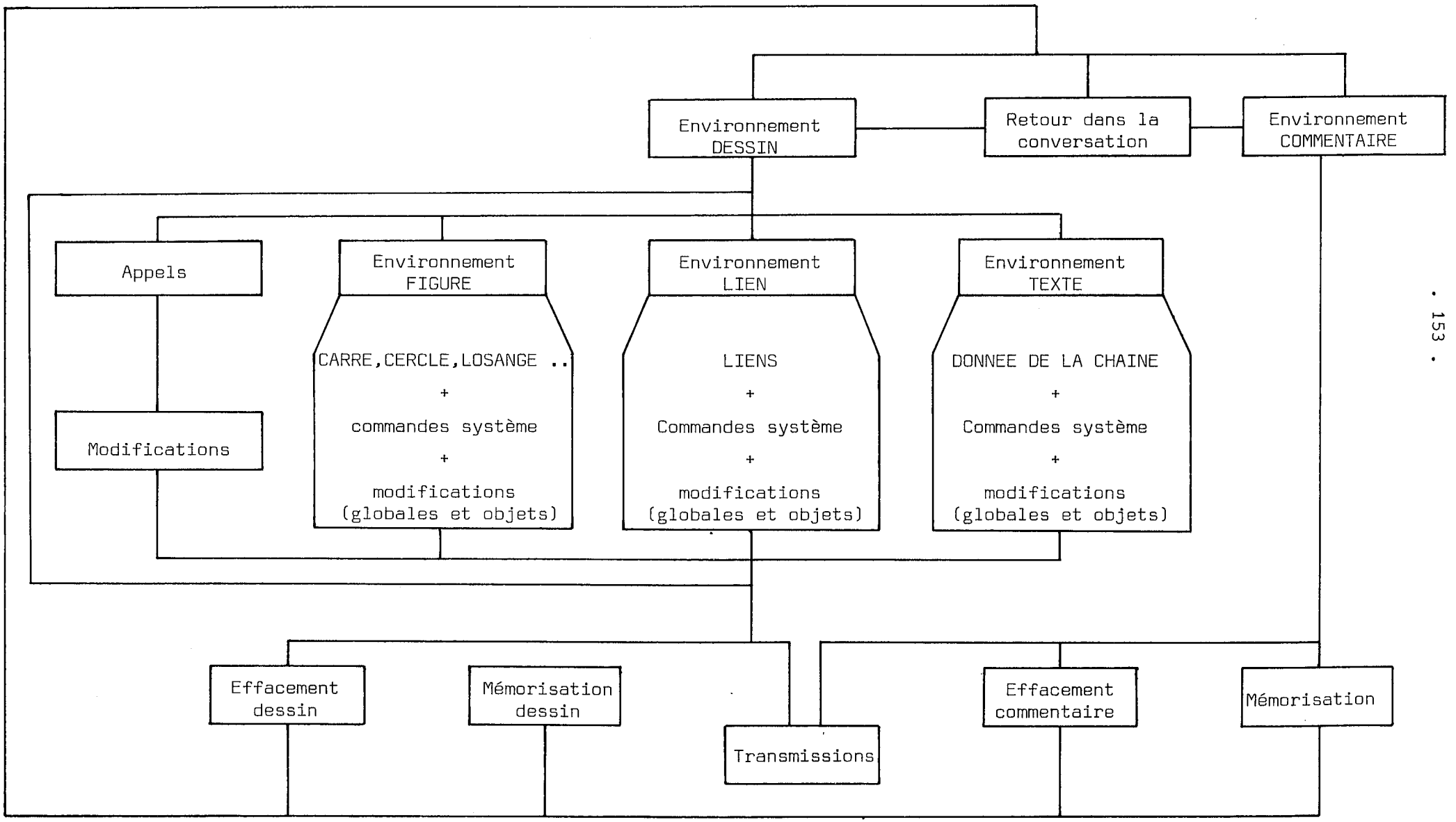
. Lien : c'est l'environnement de création de liens entre des figures quelconques.

- CONV : c'est l'environnement d'accès au fichier de conversation.

- ! : c'est l'environnement de préparation d'un commentaire.

- Commandes de transmission.

Figure 10


Environnement DESSIN
Retour dans la conversation
Environnement COMMENTAIRE
Appels
Modifications
Environnement FIGURE
CARRE,CERCLE,LOSANGE ..
+
commandes système
+
modifications (globales et objets)
Environnement LIEN
LIENS
+
Commandes système
+
modifications (globales et objets)
Environnement TEXTE
DONNEE DE LA CHAINE
+
Commandes système
+
modifications (globales et objets)
Effacement dessin
Mémorisation dessin
Transmissions
Effacement commentaire
Mémorisation

IV.3.2.3. Interprétation des commandes

L'interprétation se fait relativement à la console virtuelle et c'est seulement ensuite que les procédures particulières à chaque matériel interviennent.

Sans décrire explicitement les méthodes employées pour chaque type de commande, on peut donner quelques exemples pour montrer les méthodes employées d'un point de vue général.

1) Dessin :

. initialisation d'un certain nombre de pointeurs (tables de figures, liens, ..),

. initialisation de paramètres système (échelles, curseur, mode, ..),

. appel à une procédure graphique : affichage du curseur.

2) Création d'objets (figures, textes) :

. initialisation de pointeurs de tables,

. vérifications diverses (nombre de figures, etc ..),

. initialisation des tables,

. pour figures et textes, initialisation du nom,

. appel à une procédure graphique, seulement dans le cas de création de textes uniques (c'est-à-dire d'un dessin composé uniquement de textes), affichage du curseur en haut à gauche.

3) Figures prédéclarées

. initialisation de la table des figures et des textes à partir de la position courante du curseur (X, Y), du rayon courant (R), du nom de la commande (NO), du mode de tracé courant (plein ou pointillé) et du paramètre de placement de l'identificateur (M) ;

. vérification que la figure ne sort pas de l'écran,

. appel à la procédure d'affichage de figure qui, en se branchant sur le numéro de forme, code ces informations pour les passer au LGB.

4) Figures déclarées segment par segment

Il peut s'agir du début ou de la suite d'une figure :

. début d'une figure :

initialisation de la table des figures et textes avec la position courante du curseur (X, Y) et le numéro de forme (fixé) ;

le déplacement D, reçoit l'adresse du premier endroit libre dans la table des segments et le nombre de segments R est initialisé à O ;

. suite d'une figure :

on remplit la position suivante dans la table de segments à partir de la position courante du curseur et on incrémente R de 1 ;

appel à la procédure d'affichage.

5) Textes :

. s'il s'agit de la première chaîne créée sous l'environnement TEXTE, on initialise dans la table (X, Y) à partir de la position courante du curseur, sinon à partir de la position courante diminuée de la valeur d'une ligne ;

. dans tous les cas, initialisation de ID (premières lettres de la chaîne), de D (adresse de la première place libre dans la table de chaînes), de R (longueur de la chaîne) et de la table de chaînes ;

. appel à la procédure d'affichage de texte.

6) Liens :

. pour un lien avec la figure courante, teste s'il s'agit de figures multiples pour les lier toutes ;

. recherche dans tous les cas les noms des figures concernées et, si elles existent, on entre leur numéro de la table de figures (nom *interne*) dans la table des liens (le premier nom *externe* trouvé est considéré comme le bon) ;

. entrée des renseignements concernant la barre et le mode (variables système) ;

. appel à la procédure graphique d'affichage de liens.

7) Commande de retour (annulation de la commande précédente) :

. branchement sur le numéro d'état actuel,

. action contraire à celle des commandes valides à ces états.

8) Modification de figures ou de textes

. si pas de paramètres, il s'agit de la figure courante, d'où recherche pour savoir si c'est une figure multiple ;

. si un paramètre, appel à la procédure de recherche de figures sur le nom externe-id :

- premier nom id (s'il est unique, il est considéré comme le bon) : question à l'utilisateur : "est-ce une bonne figure ?" avec procédure graphique d'affichage du curseur au centre de la figure pour la préciser à l'utilisateur ;
- si oui, on a trouvé le nom interne N,
- sinon, on recommence.

. Dans le cas où l'on a trouvé ce qu'il faut modifier sous son nom interne N, la suite des procédures suivantes est appelée :

- procédure graphique d'effacement de figure de nom interne N,
- procédure de réinitialisation dans la table des figures de l'entrée N (remise à zéro s'il s'agit d'un effacement, action voulue de modification dans les cas de translation),
- procédure graphique de réaffichage de la figure interne N dans le cas où ce n'est pas un effacement,
- procédure de recherche dans la table des liens de nom interne N de la figure,
- si on en trouve un, numéro P, le traitement général est le suivant :
  . appel à la procédure graphique d'effacement du lien P,
  . s'il s'agissait d'un effacement de la figure N, remise à zéro dans la table des liens, sinon appel à la procédure graphique de réaffichage du lien P (pas de changement dans la table de liens, car le lien fait appel au nom interne de la figure dont l'entrée a déjà été modifiée si nécessaire),
  . si l'on n'est pas à la fin de la table des liens, on recommence la recherche dans la table des liens sur le numéro de figure N.

9) Effacement des liens :

. appel à la procédure de recherche de la première figure de nom externe id1,

. premier trouvé, s'il n'est pas unique (sinon c'est le bon) : question à l'utilisateur ; on recommence si la réponse est négative (cf. § 8 précédent),

. si on a trouvé un nom interne N1 pour le nom externe id1, on appelle la procédure de recherche sur le nom externe id2 (sinon on arrête),

. si on a trouvé un nom interne N2 pour le nom externe id2, la suite des procédures suivantes est appelée (sinon on arrête) :

. procédure de recherche dans la table des liens d'un lien entre les figures de noms internes N et P ; si on en trouve un K :

. procédure graphique d'effacement du lien K,

. remise à zéro dans la table des liens de l'entrée K,

. si l'on n'est pas à la fin de la table des liens, on recommence la recherche dans la table des liens sur les noms internes N et P.

10) Modification sur un dessin (variation ou translation)

. appel à la procédure graphique d'effacement général,

. modification dans la table des figures sur toutes les figures existantes suivant que c'est :

- une translation (dans les quatre directions),
- une variation de taille,

. modification dans la table des liens sur les renseignements sur l'existence de barres horizontale ou verticale,

exemple : translation de N pas vers la gauche :

si barre verticale alors translation vers la gauche de N pas de cette barre verticale (XBAR := XBAR - NxPAS)

exemple : diminution de N pas :

si barre horizontale alors YBAR := YBAR/N

sinon XBAR := XBAR/N

. appel à la procédure graphique d'affichage de dessin (affichage des figures et des liens.

11) Mémorisation

. remplissage de la table des dessins (nom du dessin, déplacement dans le fichier, nombre de figures, ..),

. appel à la procédure de codage dans le buffer,

. appel à la procédure d'écriture dans le fichier personnel du dessin codé,

. appel à la procédure graphique d'effacement du dessin.

12) Appel

. recherche dans la table des dessins du nom demandé,

. quand on l'a trouvé, le déplacement dans le fichier permet l'appel à la procédure de lecture dans le fichier personnel,

. décodage de l'information dans le buffer et remplissage des tables de la liste de visualisation avec les modifications demandées en cas d'appel translaté ; il s'agit également de réinitialiser convenablement les noms internes de figures qui doivent rester non ambigus ;

exemple : cas d'un appel suivant un autre appel :
le premier dessin contenait N figures, les noms internes des figures venant du second dessin devront donc commencer à N+1 et lorsque le décodage du buffer indiquera un lien entre les figures P et Q, il faudra mettre dans la table des liens (P+N) et (Q+N) ;

. appel à la procédure graphique d'affichage de dessin.

13) Envoi d'une intervention

. appel de la procédure de calcul du nombre de lignes réellement occupées par le dessin à transmettre (calcul sur les figures, les textes et les barres horizontales),

. remplissage de la table locale des interventions, numéro de console, nombre de lignes, nombre de figures, ..),

. codage de l'intervention, précédée d'un en-tête comprenant :

- le numéro de console émettrice,
- le numéro de console destinataire (-1 si publique),
- date et heure d'émission,
- nombre de lignes,
- nombre d'enregistrements,

avec tous les renseignements permettant le remplissage de la table des interventions du ou des destinataires, et la reconstitution de l'intervention ;

. le contrôle est passé au SUPLOC qui se charge de transmettre,

. mise à jour du fichier de conversation et affichage de l'intervention.

14) Retours dans la conversation (cf. § IV.3.4.1.2)

recherche de l'intervention : deux cas peuvent se produire :

. elle se trouve dans le fichier provisoire de conversation de la machine virtuelle :

- on utilise la procédure de chargement d'une zone du fichier de conversation à partir des renseignements trouvés dans la table des interventions,
- on décode l'intervention afin d'obtenir tous les renseignements concernant les figures, les textes et les liens,
- on remplit la liste de visualisation à partir de cette information décodée,
- on appelle la procédure d'affichage correspondant à l'affichage d'un dessin si l'intervention était sous forme de dessin, sinon à la procédure correspondant à l'affichage d'un commentaire,
- on peut ensuite accéder au dessin (ou au commentaire) comme s'il avait été créé par l'utilisateur ;

. elle ne se trouve pas dans le fichier provisoire ; il faut alors faire une demande au SUPGEN :

- l'interpréteur fournit au SUPLOC les renseignements nécessaires (exemple : nom du participant dont on cherche l'intervention) qui fait une demande au SUPGEN,
- il repasse le contrôle au moniteur (et peut très bien interpréter d'autres commandes en attendant que le SUPGEN fournisse l'intervention recherchée),
- si l'intervention est trouvée, le SUPLOC le signale et fournit à l'utilisateur l'information codée,
- le traitement revient alors au traitement précédent.

15) Procédure d'affichage d'une intervention

Nous donnons ci-dessous l'algorithme d'affichage d'une intervention :

```
    si nombre de lignes disponibles est suffisant alors
début
  |       mise dans la table des interventions affichées de la nouvelle
  |       intervention
  |       affichage de l'en-tête
  |       affichage de l'intervention
fin
    sinon
début effacement du 1/2 écran supérieur ;
  |       calcul du nombre minimum de lignes à remonter : N
  |       P := 0 ;
  |       tant que P < M faire
  |           début
  |             |   P := P + nombre lignes (intervention I) ;
  |             |   I := I + 1 ;
  |           fin
  |       commentaire : I pointe sur la première intervention à translater ;
  |       Pour J := I jusqu'à nombre  affichées faire
  |       début
  |         |       TYPE(J - I + 1) := TYPE(J) ;
  |         |       nombre de lignes(J-I+1) := nombre de lignes(J) ;
  |       fin
  |       Mise à jour des différents pointeurs et des différentes variables ;
  |       Mise dans la table des interventions affichées de la nouvelle
  |       intervention ;
  |       tant que il y a des interventions à afficher faire
  |       début
  |         |       affichage de l'en-tête ;
  |         |       affichage de l'intervention ;
  |       fin
fin
```

IV.3.2.4. Interprétation des informations venant du SUPLOC

L'interpréteur reçoit ces informations sous la forme :

. un numéro de commande,

. une suite d'informations dont le décodage se fait en fonction du numéro de commande.

Les cas à traiter, suivant le numéro de commande, sont :

. intervention transmise par un participant présent,

. interventions en retard (permettant de constituer un fichier provisoire de conversation sur la machine virtuelle du participant (cf. § IV.3.4.1.2.),

. retours dans la conversation,

. commandes (arrivées, départs).

a) Intervention transmise par un participant présent

Le SUPLOC fournit à l'interpréteur les renseignements suivants :

. numéro de la console émettrice (numéro de fin de table si anonyme),

. numéro du destinataire (numéro de console ou -1 s'il s'agit d'une intervention publique),

. nombre d'enregistrements (de 80 caractères),

. date et heure d'émission,

. numéro associé à la réunion (cf. § IV.3.4.5 réunions simultanées),

. nombre de lignes qu'occupe l'intervention,

. nombre de figures (0 si c'est un commentaire),

. nombre de caractères,

. nombre de segments,

. nombre de liens,

. codage des renseignements permettant de construire l'intervention.

A partir de ces données, l'interpréteur met à jour la table des interventions (cf. figure 7). Elle range ensuite toute l'information dans le fichier de conversation se trouvant sur la machine virtuelle (fichier provisoire) et appelle la procédure d'affichage de l'intervention courante.

b) Interventions en retard

Il s'agit des interventions qui ont été transmises au participant durant son absence (il les reçoit en début de session).

L'information et le traitement sont les mêmes que pour le cas a), mais il n'y a pas affichage de l'intervention. L'interpréteur affiche le message indiquant le nombre d'interventions en retard.

c) Retours dans la conversation

Il s'agit d'une intervention que le participant veut revoir.

Le traitement est expliqué au § 12) parmi les exemples de l'interprétation des commandes.

d) Commandes

Dans une réunion asynchrone, elles sont au nombre de trois :

. acceptation et liste des participants : le SUPLOC fournit à l'interpréteur les renseignements concernant les participants (nom personnel, date et heure de dernière participation, indicateur de présence). L'interpréteur affiche la liste des participants et met à jour sa table des participants ;

. départ d'un participant : le SUPLOC fournit le numéro de console du participant qui part, ce qui permet à l'interpréteur d'afficher un message à partir du nom trouvé dans la table des participants ;

. arrivée d'un participant : idem départ d'un participant.

IV.3.3. Résolution des problèmes graphiques

Ce paragraphe a trait au passage de la console virtuelle aux différentes consoles réelles. La première partie donne deux exemples de problèmes graphiques, la seconde décrit quelques cas particuliers (IBM 2250, TEKTRONIX 4010, consoles alphanumériques, ..).

IV.3.3.1. Exemples de problèmes graphiques

Il est inutile d'expliquer comment on reconstruit à partir des renseignements de la structure de données de la console virtuelle les figures, puisque, à part pour le segment dont on connaît l'origine et l'extrêmité, elles sont construites à partir d'un centre et d'un rayon (cf. § II.5.1.3. codage). Les problèmes que nous allons expliciter concernent le tracé de liens et la création automatique de flèches.

a) Tracé de liens

Il s'agit de tracer une ligne droite ou brisée, joignant deux figures dont on connaît le centre et le rayon.

• Lignes droites (liens directs) :

la première idée qui a été appliquée est de joindre les figures par une ligne centre à centre, coupée par les contours des figures (cf. figure 11).

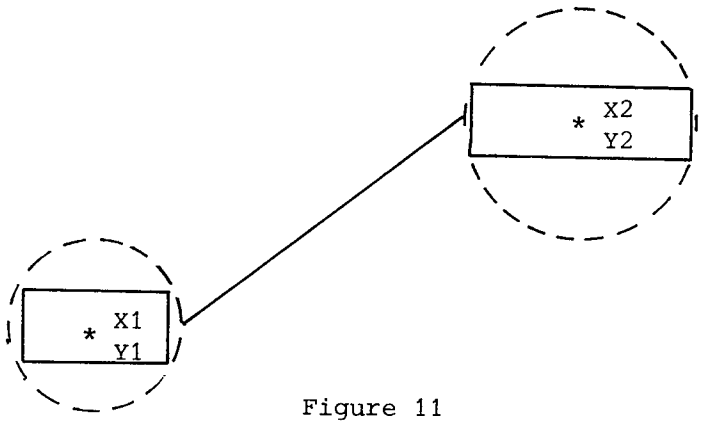


Figure 11

On trace le segment, coupé par des cercles de centre (X1, Y1), rayon R1 et centre (X2, Y2) rayon R2. D'où le calcul suivant :

$$HYPO := \sqrt{(X2 - X1)^2 + (Y2 - Y1)^2}$$

$$COS := (X2 - X1) / HYPO$$

$$SIN := (Y2 - Y1) / HYPO$$

coordonnées de l'origine : $XX1 := X1 + R1 * COS$

$YY1 := Y1 + R1 * SIN$

coordonnées de l'extrêmité : $XX2 := X2 - R2 * COS$

$YY2 := Y2 - R2 * SIN$

Cette méthode, utilisée pendant un certain temps, a été abandonnée au profit de la méthode décrite ci-après, essentiellement à cause de son manque de précision, particulièrement pour les rectangles horizontaux et verticaux. Il aurait fallu faire un calcul de découpage différent pour chaque type de figure.

Finalement, les liens directs sont créés de la façon suivante (cf. figure 12) : les segments sont tracés suivant les positions respectives des figures. On distingue deux groupes parmi les figures prédéclarées :

- premier groupe : carrés, cercles, losanges, triangles,
- deuxième groupe : rectangles.

Les deux exemples suivants montrent comment l'on détermine la place des figures les unes par rapport aux autres.

Pour chaque figure concernée, on cherche dans quel cadran (I, II, III, ou IV) se trouve le centre de l'autre figure. Pour cela, il suffit de calculer le sinus et le cosinus de l'angle de la droite joignant les centres avec l'horizontale. Une fois trouvé le cadran, on prend comme extrêmité du segment le point correspondant (1, 2, 3 ou 4).

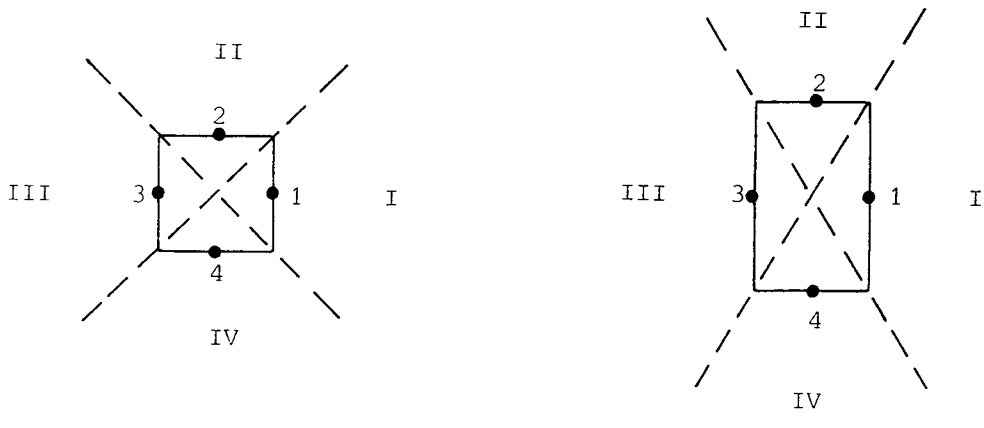


Figure 12

• Liens avec barres :

Nous allons expliquer la méthode de construction de liens avec barre verticale, la construction avec barre horizontale étant similaire.

On commence par placer la première figure par rapport à la barre verticale ; trois cas peuvent se produire :

1er Cas : la figure est située entièrement au gauche de la barre (X1 + R1 < XB). L'origine et l'extrêmité du premier segment sont alors :

origine : $\begin{cases} X1 + R1 \\ Y1 \end{cases}$; extrêmité : $\begin{cases} XB \\ Y1 \end{cases}$

Remarque : en fait, suivant le type de figure, R1 peut être remplacé par R1 divisé par deux (cas des rectangles).

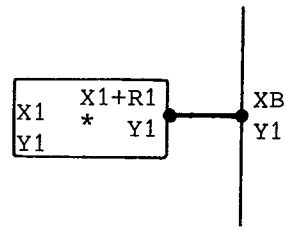


Figure 13

2ème Cas : la figure est entièrement située à droite de la barre (X1 - R1 > XB), d'où l'origine et l'extrêmité, à la remarque près :

origine : $\begin{cases} X1 - R1 \\ Y1 \end{cases}$; extrêmité : $\begin{cases} XB \\ Y1 \end{cases}$

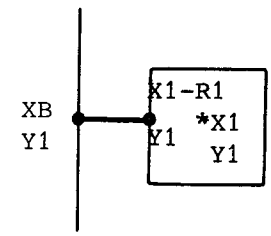


Figure 14

3ème Cas : la figure est coupée par la barre ; il faut alors situer les deux figures par leurs ordonnées respectives. On détermine alors un point :

soit $\begin{cases} XB \\ Y1 - R1 \end{cases}$ (si Y2 < Y1)

soit $\begin{cases} XB \\ Y1 + R1 \end{cases}$ (si Y2 > Y1)

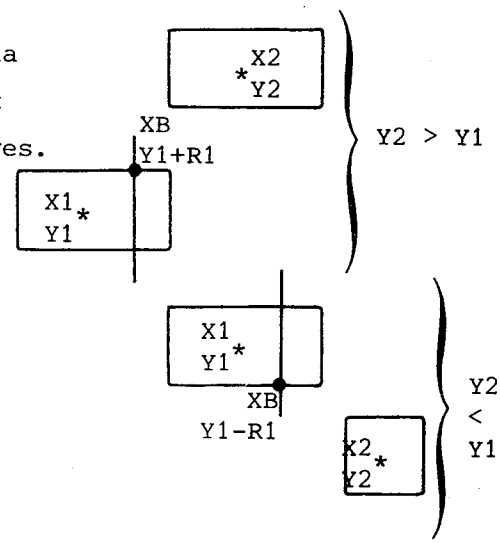


Figure 15

On recommence le même travail sur la deuxième figure et l'on peut donc obtenir une ligne brisée composée de trois segments (la barre ne coupe aucune figure), deux segments (la barre coupe une et une seule figure), ou un seul segment (la barre coupe les deux figures).

Remarque :

Il existe des cas limites où lorsqu'une figure est coupée par la barre, le lien passe par l'intérieur de cette figure. Il n'est pas utile d'ajouter une série de tests (simples mais coûteux) puisque dans tous les cas, l'utilisateur peut annuler l'affichage de ce lien (ou l'effacer).

- Cas particulier d'une figure créée segment par segment

Dans le cas d'une figure créée segment par segment, l'option actuelle dans la version réalisée et testée, est de prendre l'origine du premier segment créé comme point d'attache des liens. Il est évident que cette méthode présente des inconvénients, la ligne de lien pouvant couper la figure. Une solution possible est la suivante :

. Lien direct :

- entre une figure A déclarée segment par segment et une figure B prédéclarée, ayant les points d'attache de liens (1, 2, 3, 4) (cf. figure 16): on cherche alors à minimiser la distance entre chacun des points extrêmités de segments de la figure A et les quatre points d'attache de la figure B. Lorsqu'on a trouvé pour quel couple de points cette distance est minimum, on trace le lien entre ces deux points.

- entre deux figures A et B déclarées segment par segment : on minimise la distance de la figure A et chacun des points extrêmités de segments de la figure B. Quand on a trouvé le couple le points pour lequel cette distance est minimum, on trace le lien entre ces deux points.

. Lien avec barre :

on cherche pour quel point (extrêmité de segment) la distance à la barre est minimum (recherche simple, car elle se fait sur les ordonnées pour une barre horizontale, sur les abscisses pour une barre verticale) et on trace ensuite le segment joignant ce point à la barre (on ne s'occupe pas du cas où la barre coupe la figure, car il faudrait faire une recherche sur tous les segments, le seul inconvénient étant de toute façon d'avoir des liens passant par l'intérieur de la figure).

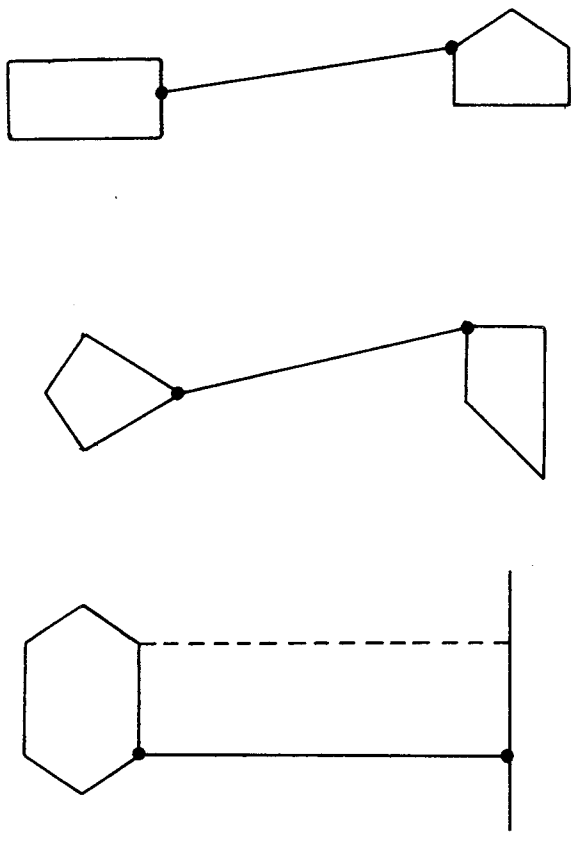


Figure 16

Le segment en pointillés aurait la même longueur que celui choisi. Ce choix dépend de l'ordre dans lequel l'algorithme a balayé les points, donc de l'ordre dans lequel ils ont été définis par l'utilisateur.

b) Création automatique de flèches

Le problème est le suivant : ayant déterminé (cf. a) le lien, il faut tracer automatiquement une flèche. Pour les liens avec une barre, on a choisi arbitrairement de placer cette flèche au milieu du dernier segment, en les prenant dans l'ordre indiqué par la flèche. Pour les liens directs, la flèche est placée au milieu du segment. Cela revient donc à tracer une flèche sur un segment dont on connaît l'origine (X1, Y1) et l'extrêmité (X2, Y2) (la flèche devant indiquer le sens de l'origine vers l'extrêmité).

La flèche sera créée de la façon suivante :

. chaque branche fait un angle de 45 degrés avec le segment considéré,

. le paramètre VAR permet d'obtenir l'ordonnée ou l'origine des extrêmités ne constituant pas la pointe de la flèche (cf. figure 17).

Les calculs sont alors les suivants : calcul des coordonnées du milieu du segment (pointe de la flèche) :

$$X(2) = \frac{X1 + X2}{2}$$

$$Y(2) = \frac{Y1 + Y2}{2}$$

Les quatre cas de figures permettent de trouver les quatre types de flèches possibles :

1er cas :

$0 \leq \theta \leq \frac{\pi}{4}$ $\qquad \alpha = \frac{\pi}{4} - \theta$

$$tg\alpha = \frac{d}{VAR} \Rightarrow d = VAR\left(\frac{1-tg\theta}{1+tg\theta}\right)$$

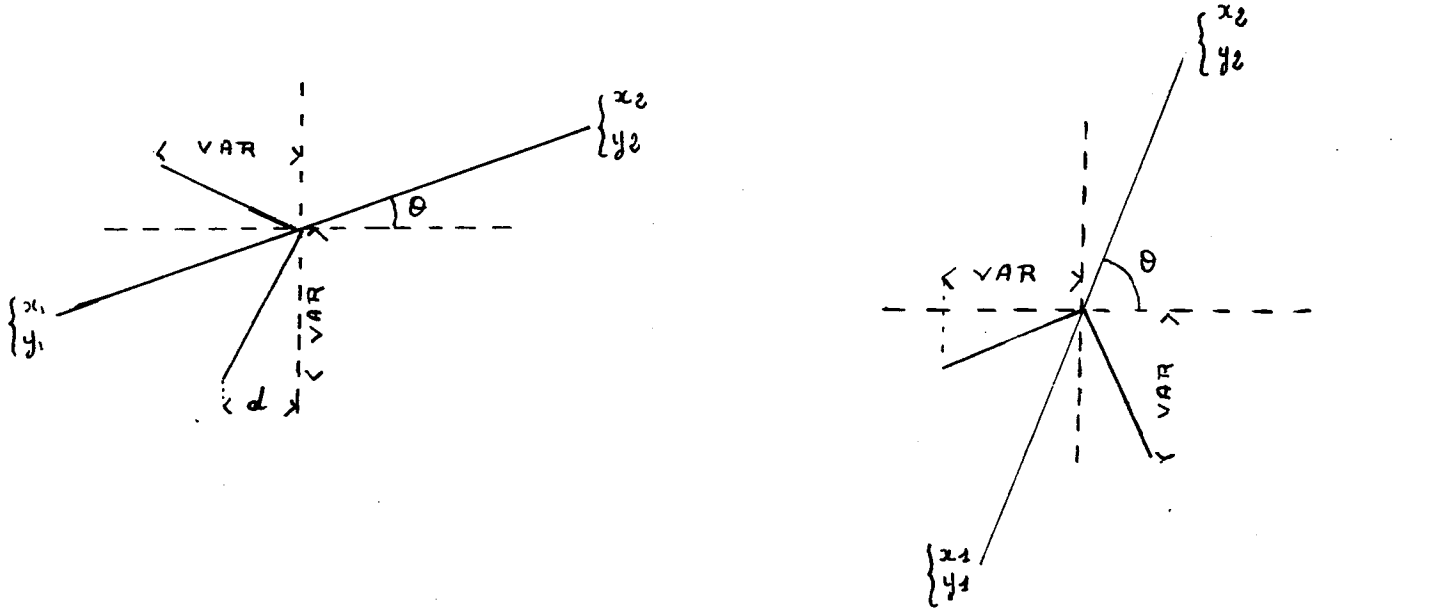


Figure 17

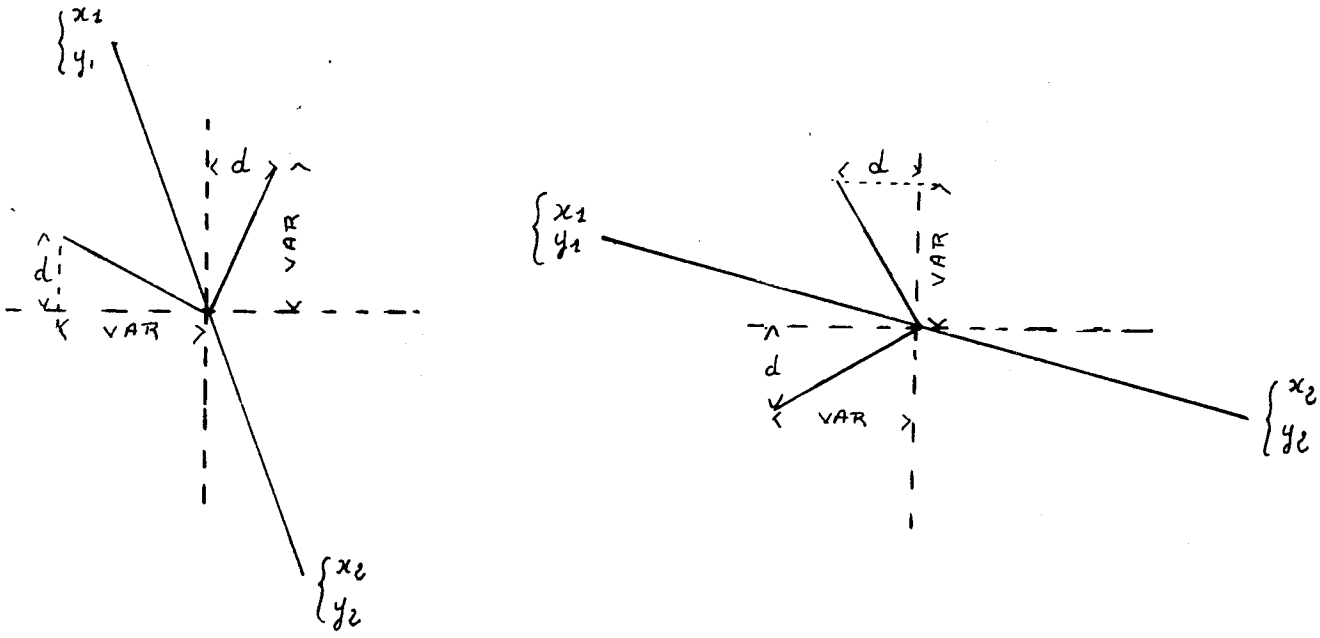


Figure 18

2ème cas :

$\frac{\pi}{4} < \theta \leq \frac{\pi}{2}$ $\quad \alpha = \theta - \frac{\pi}{4}$

$tg\alpha = \frac{d}{VAR} \quad \Rightarrow d = VAR(\frac{tg\theta - 1}{1 + tg\theta})$

3ème cas :

$\frac{\pi}{2} < \theta \leq \frac{3\pi}{4}$ $\quad \alpha = \frac{\pi}{4} - \theta$

$tg\alpha = \frac{d}{VAR} \quad \Rightarrow d = VAR(\frac{1 + |tg\theta|}{1 - |tg\theta|})$

4ème cas :

$\frac{3\pi}{4} < \theta \leq \pi$ $\quad \alpha = \theta - \frac{\pi}{4}$

$tg\alpha = \frac{d}{VAR} \quad \Rightarrow d = VAR(\frac{-|tg\theta| - 1}{1 - |tg\theta|})$

d'où l'algorithme suivant :

```
dX := X1 - X2 ;
dY := Y1 - Y2 ;
si Y1 > Y2 alors DELTA := VAR sinon DELTA := - VAR ;
co détermine le sens de la flèche ;
si dX * dY >= 0 alors
début co la pente est positive ;
   X(1) := X(2) + DELTA *(dX - dY)/(dX + dY) ;
   Y(1) := Y(2) + VAR ;
   X(3) := X(2) + VAR ;
   Y(3) := Y(2) - DELTA *(dX - dY)/(dX + dY) ;
fin sinon co : pente négative ;
   X(1) := X(2) + VAR *(dX + dY)/(dY - dX) ;
   Y(1) := Y(2) + VAR ;
   X(3) := X(2) - VAR ;
   Y(3) := Y(2) + VAR *(dX + dY)/(dY - dX) ;
fin ;
```

La flèche est alors tracée par deux segments joints, partant de (X(1),Y(1)), passant par (X(2),Y(2)) et ayant pour extrêmité (X(3),Y(3)). On ne prétend pas que cet algorithme soit le plus simple et le plus performant, mais il permet d'obtenir des flèches d'aspect correct, l'angle de 45° facilitant et simplifiant les calculs.

IV.3.3.2. Cas de quelques consoles

a) IBM 2250

On a vu (§ IV.2) qu'il était accessible à partir d'ALGOLW. La principale contrainte est que le nombre de figures est limité à 16. Il a donc fallu calculer un numéro réel à partir d'un numéro virtuel. De plus, étant donné que le demi-écran supérieur (NV = 1) constitue un tout du point de vue graphique, il doit continuer à être particularisé. D'où le calcul du numéro réel (NR) à partir du numéro virtuel (NV) (REM = reste de la division entière) :

```
si NV ≠ 1 alors NR := (NV - 2) REM 15 + 2
sinon NR := 1 ;
```

Ceci se justifie encore lorsqu'on analyse la procédure d'effacement d'une figure. En effet, en général, plusieurs figures ont le même numéro réel ; la procédure d'effacement est la suivante :

. effacer la figure réelle NR contenant l'objet NVi que l'on veut effacer,

. régénérer tous les objets contenus dans la figure réelle NR (avec vérification qu'elles étaient affichées), sauf celle de numéro virtuel NVi.

Il apparaît donc important de réserver NR = 1 au demi-écran supérieur, sous peine d'avoir à effacer puis régénérer tout ce demi-écran lors de l'effacement d'un simple objet.

Remarque :

au point de vue partage de l'écran, le 2250 IBM est l'exacte copie de la console virtuelle.

Toutes les possibilités présentées dans le chapitre III sont très intéressantes sur ce terminal grâce aux dimensions de l'écran et à la rapidité de l'affichage.

Signalons qu'une commande permet de sortir au BENSON la copie de ce qui se trouve sur l'écran et que l'utilisateur peut, s'il le désire, se réserver tout l'écran, sans participer à la conversation.

b) TEKTRONIX 4010 et 4015

Comme on se sert du logiciel GRIGRI pour les deux écrans TEKTRONIX, les mêmes contraintes et les mêmes solutions que pour le 2250 IBM se retrouvent.

En revanche, d'autres contraintes liées au matériel posent des problèmes :

. pour effacer un objet, il faut obligatoirement effacer tout ce qui se trouvait sur l'écran et tout régénérer. Ceci a imposé en particulier une optimisation des programmes, en ce sens qu'il a par exemple fallu regrouper autant que possible les effacements ;

. par ailleurs, le TEKTRONIX 4010 a été, pendant un certain temps, connecté au 360/67 au moyen d'une ligne à 110 bauds, ce qui évidemment ajoutait à la lenteur de l'affichage. Maintenant les TEKTRONIX 4010 et 4015 sont connectés au 360/67 via CYCLADES à 1200 bauds.

L'utilisateur peut, comme sur le 2250 IBM se réserver tout l'écran.

c) Consoles alphanumériques

Une version a été implémentée et elle a permis un certain nombre de tests. Les figures sont constituées d'étoiles. Le pseudo-écran est en fait un tableau de caractères que l'utilisateur affiche à son gré grâce à la commande PRINT.

. Terminal papier : IBM 2741

aucun problème de partage d'écran ni de numéro réel ne se pose. Lorsqu'une intervention sera reçue, elle s'affichera immédiatement.

. Terminal écran : SYNELEC, IRISCOPE, ..

le problème du numéro de figure ne se pose pas. Pour le partage de l'écran, la formule adoptée est la suivante :

- l'utilisateur tape la commande PRINT lorsqu'il veut afficher l'état actuel de son dessin ; celui-ci occupe alors tout l'écran ;

- lors d'une réception d'intervention, celle-ci remplace ce qui se trouvait sur l'écran et l'occupe tout entier.

IV.3.4. Tenue d'une réunion

On rappelle que les composants essentiels du système CONGRES, dans le domaine des communications, sont :

. le SUPLOC dont chaque machine virtuelle active pour le système CONGRES reçoit une copie,

. le SUPGEN qui est unique dans le cadre d'une réunion.

Avant de présenter ces deux modules, on va revenir sur les fichiers pour expliquer plus précisément les méthodes d'implémentation et d'utilisation choisies, surtout pour :

. le fichier personnel qui sert en quelque sorte de "bloc note" à un participant ; chacun dispose donc de son propre fichier personnel qui reste en permanence sur sa propre machine virtuelle ;

. le fichier de conversation qui contient toutes les interventions transmises au cours d'une réunion.

IV.3.4.1. Les fichiers et la sécurité

1) Le fichier personnel

Il s'agit du fichier où sont conservés les dessins ou les interventions que désire conserver le participant. Les informations codées contenues dans ce fichier sont accessibles via la table des dessins (cf. figure 19). Il existe sur la machine virtuelle de chaque participant.

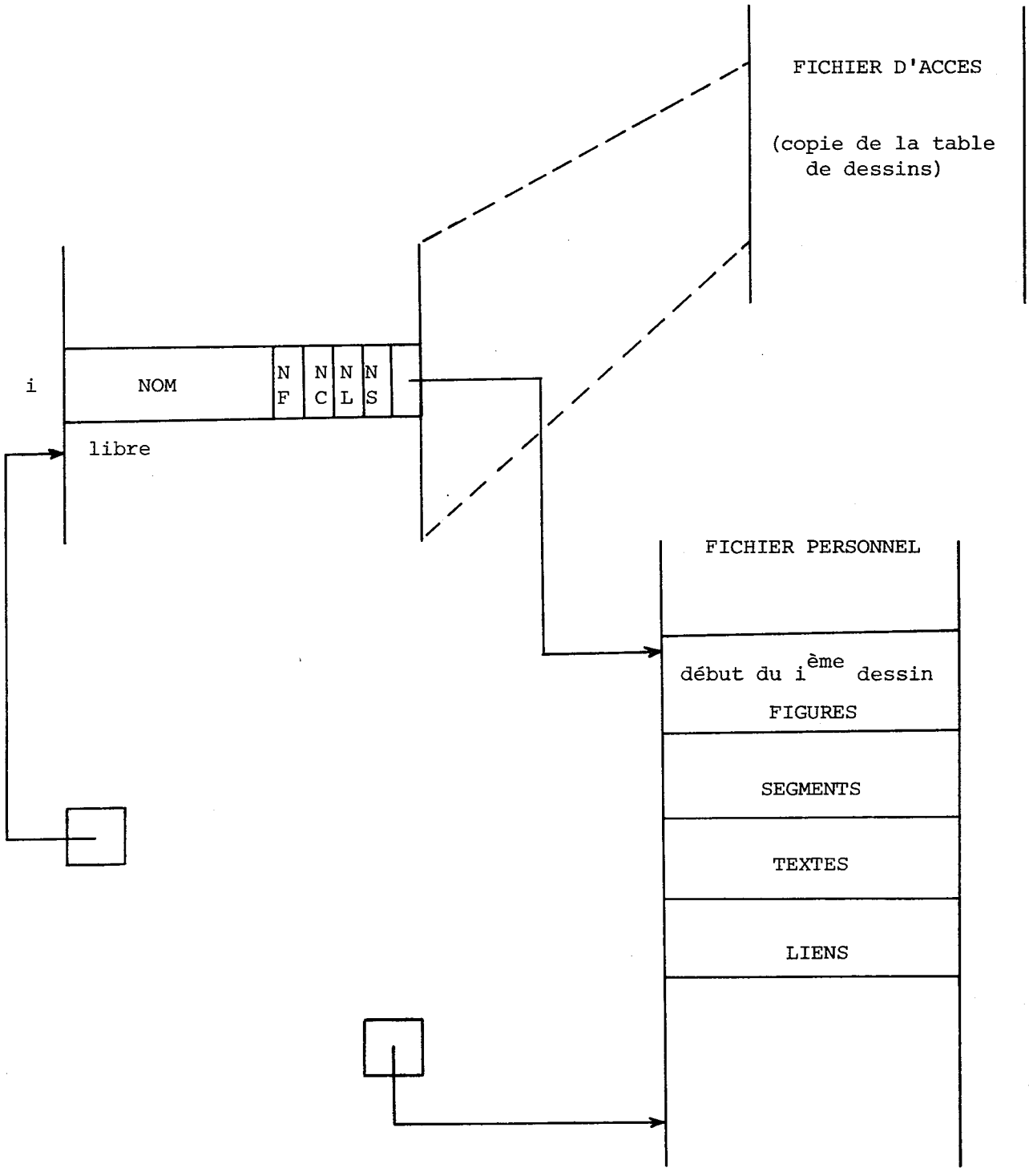


Figure 19

Fichier personnel et accès par nom
via une table de dessins

A chaque fois que l'utilisateur mémorise un dessin ou une intervention, deux fichiers sont mis à jour :

. le fichier personnel où est codée toute l'information nécessaire à la reconstruction du dessin (sur les figures X, Y, R ..),

. le fichier de contrôle qui est en fait une copie de la table des dessins (nom du dessin, nombre de figures, de textes, de liens, déplacement dans le fichier personnel). Il sert en particulier à initialiser la table des dessins à chaque début de session.

Si une erreur système intervient (au niveau CP ou au niveau CONGRES), tout ce qui a été mémorisé jusque là sera accessible à la réinitialisation. Il peut être cependant nécessaire d'effacer du ou des fichiers la dernière mémorisation si l'erreur s'est produite à ce moment-là.

Lors d'un appel de dessin, seule la partie utile du fichier est chargée, puisqu'on a dans la table des dessins toutes les données nécessaires à ce chargement minimum (déplacement, longueur).

2) Le fichier de conversation

A ce sujet, plusieurs solutions étaient possibles :

a) Une copie sur chaque site :

L'avantage essentiel est évidemment un accès rapide, facile et peu coûteux. En revanche, ce fichier pouvant être assez gros, il aurait occupé beaucoup trop de place sur chaque machine virtuelle participante. De plus, cette copie aurait pu être modifiée (volontairement ou non) et il aurait toujours fallu s'assurer de la compatibilité entre les diverses copies.

b) Une seule copie par réunion :

Le gain de place par rapport à la première solution est évident et le fichier ne peut alors plus être atteint par le participant en dehors d'une session CONGRES, puisqu'il se trouve sur la machine SUPGEN. Cependant, pour chaque retour dans la conversation, l'accès est long et coûteux.

c) Solution mixte : une seule copie + une partie courante sur chaque machine virtuelle :

Il existe une seule copie du fichier de conversation située dans le SUPGEN ; cependant, à chaque fois qu'une machine virtuelle entre dans la réunion, elle reçoit toutes les interventions que le participant considéré n'a pu voir. Cette partie forme un fichier partiel de conversation sur chaque machine virtuelle active pour CONGRES, qui est complété pendant toute le temps où elle est active pour les interventions transmises. Ce fichier partiel est détruit à chaque fin de session CONGRES pour une machine virtuelle. D'où un encombrement faible et un accès rapide et peu coûteux à tout ce que n'a pas vu l'utilisateur (donc ce qui l'intéresse le plus a priori). L'accès au reste du fichier sera plus lent en plus coûteux, mais aussi moins courant. Il est également doublé sur le SUPGEN d'un fichier de contrôle contenant, pour chaque intervention, le numéro de l'émetteur (-1 si anonyme), celui du destinataire (-1 si publique), la longueur de l'intervention ..

Ces deux fichiers sont mis à jour à chaque transmission, ce qui minimise le travail à faire en cas de cassure système.

Remarque 1 :

on verra dans la partie consacrée au SUPGEN comment est assurée l'exclusion mutuelle pour ce fichier.

Remarque 2 :

les droits d'accès sont exprimés dans la table des interventions par le numéro de console émetteur et le numéro de destinataire.

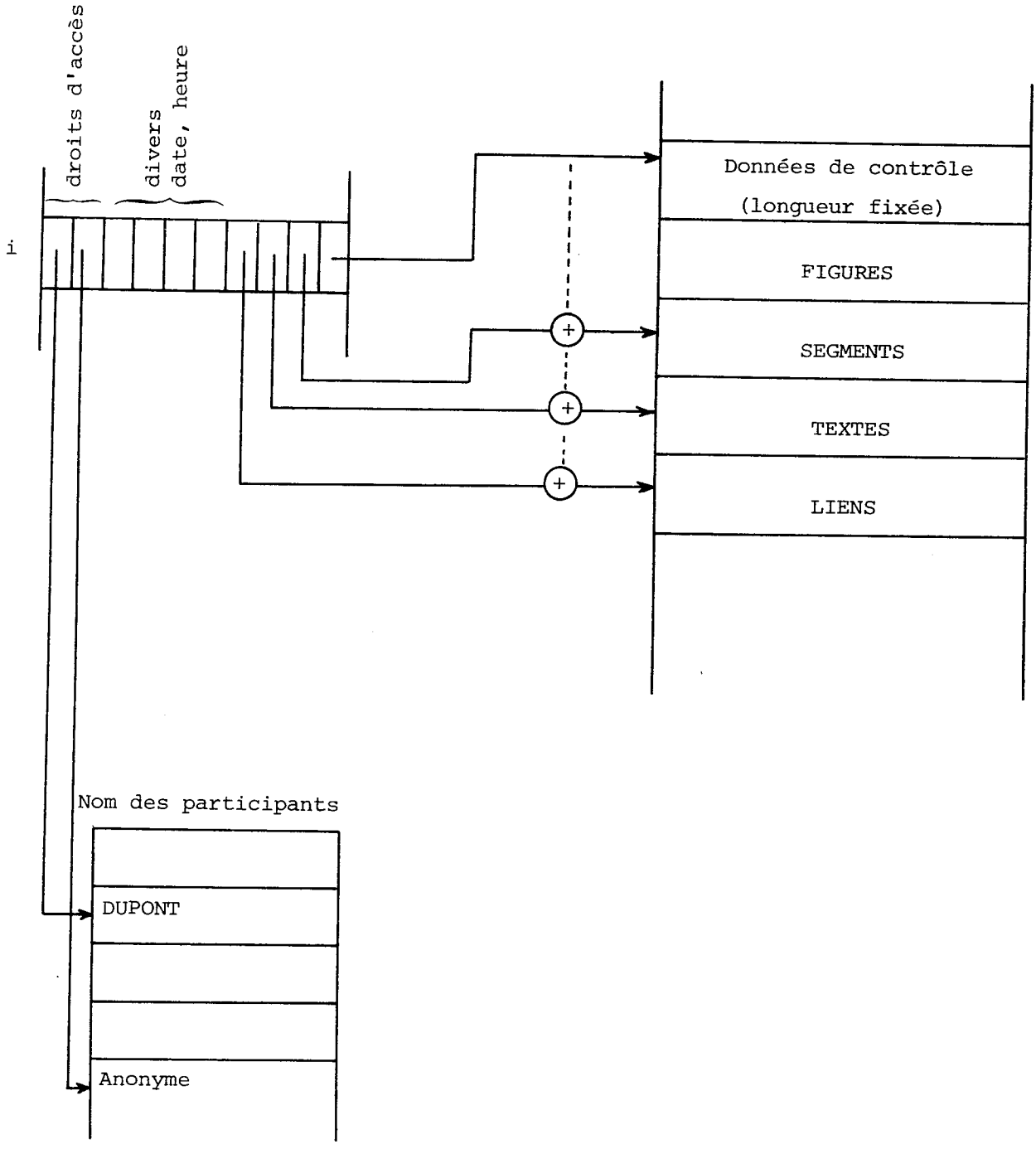


Figure 20

Accès au fichier de conversation

3) Les autres fichiers

Il s'agit de fichiers situés sur la machine virtuelle du SUPGEN et qui ne sont accessibles que par lui. Ils servent essentiellement à conserver un historique de la réunion et à réinitialiser le système, soit en début de session du SUPGEN, soit en cas de cassure système. Expliquons cette seconde nécessité :

Supposons qu'après un arrêt, on relance le programme SUPGEN. Il faut qu'il retrouve :

. la liste des participants pour pouvoir gérer la conversation,

. la date et l'heure de leur dernière participation (renseignements non absolument nécessaires pour le bon déroulement de la réunion, mais utiles pour l'information des participants),

. un état de chaque participant par rapport à la réunion, principalement les interventions qu'il a déjà reçues et celles qu'il n'a pas encore eues l'occasion de voir.

On conservera donc deux fichiers de contrôle :

. un fichier des participants : c'est la seule donnée qui doive être initialisée pour préparer une réunion. Dès qu'une réunion est décidée, il suffit de préciser dans ce fichier les noms personnels et les noms des machines virtuelles utilisées pour chaque participant (et mettre JAMAIS dans la partie réservée à la date et l'heure de leur dernière participation) ;

. un fichier relatant l'historique de la conversation au point de vue réception d'interventions.

4) Durée de vie des fichiers

La durée de vie de la copie complète du fichier de conversation sur le SUPGEN sera au moins égale au temps que dure la réunion, alors que celle des fichiers provisoires de conversation est limitée à la durée d'une session utilisateur (temps entre son arrivée et son départ).

Tous les fichiers relatifs à la réunion (accès au fichier de conversation, liste des participants, ..) auront une durée de vie égale au temps que dure la réunion. Ils sont initialisés au début d'une réunion et effacés lorsque la fin de la réunion est décidée.

On peut considérer que le fichier personnel doit être conservé tout le temps que le participant correspondant est recensé dans une réunion. En fait, on laissera le soin au participant de l'effacer, car on considère qu'il lui appartient en propre et qu'il pourra l'utiliser pour d'autres raisons que la participation à une réunion (par exemple, préparation de dessins et sorties sur BENSON).

IV.3.4.2. Le SUPLOC

Le SUPLOC a pour tâche de gérer les communications entre machines virtuelles, donc en particulier :

- gérer les interruptions sur le lecteur de cartes virtuel,
- s'occuper des transmissions vers les autres machines virtuelles.

Il a été décidé que le SUPLOC, lors d'une transmission d'intervention, serait chargé de transmettre directement cette intervention à tous les participants présents.

Lorsqu'un participant entre dans le système CONGRES, le SUPLOC transmet une demande de participation au SUPGEN (dont il connaît l'adresse puisqu'il est unique). Si cette demande est acceptée, le SUPGEN lui envoie en retour un certain nombre de renseignements, en particulier la liste des participants (avec leur nom de machine virtuelle pour pouvoir communiquer avec eux), la date et l'heure de dernière participation et une indication précisant s'ils sont présents ou non. De ces informations, certaines sont importantes pour le SUPLOC, d'autres non :

. il conserve pendant toute la durée de la session la liste des machines virtuelles participantes et l'indication précisant si elles sont actives pour CONGRES ou non. Ces informations lui sont nécessaires pour savoir à qui transmettre les interventions. En revanche, il n'a pas besoin de connaître les noms des participants car il travaille au niveau des machines virtuelles ;

. il passe à INTER les autres renseignements à qui certains, tels que le nom des participants, vont servir pendant toute la durée de la session, d'autres, tels que la date et l'heure ne lui servant qu'un moment (pour l'affichage de la liste des participants) et étant abandonnés ensuite.

Le SUPLOC reçoit ensuite, éventuellement, les interventions que le participant n'a pas encore pu voir. Ces informations sont passées à INTER qui se chargera d'une part de créer un fichier de conversation (et fichiers et tables associées) et d'autre part d'avertir le participant.

Dans toute la suite, les cas où SUPLOC intervient, sont :

1) envoi d'une commande (fin de session, identique à début de session)

INTER passe au SUPLOC l'enregistrement à transmettre au SUPGEN. Le SUPLOC se masque pour les interruptions, transmet cet enregistrement au SUPGEN, se démasque et rend le contrôle au moniteur.

2) envoi d'une intervention

INTER passe au SUPLOC l'intervention codée et le numéro de la machine virtuelle (identique au numéro du participant) à qui elle est destinée si c'est une intervention privée, -1 si l'intervention est publique ; si l'intervention est privée, le SUPLOC vérifie si la machine virtuelle destinataire est active pour CONGRES :

. si oui, il lui transmet l'intervention directement, puis la transmet également au SUPGEN pour mettre à jour le fichier de conversation,

. sinon, il transmet simplement l'intervention au SUPGEN.

Ensuite, le SUPLOC rend le contrôle au moniteur.

3) interruption sur le lecteur

le programme doit être interrompu et le contrôle passé au SUPLOC (si c'est lui qui est actif, aucune interruption ne peut lui parvenir, car il est masqué). Etant donné que l'on ne peut appeler un programme ALGOLW à partir de l'assembleur, l'interruption n'est réellement prise en compte qu'à chaque fois que l'on revient au moniteur qui passe le contrôle au SUPLOC pour vérifier s'il y a quelque chose dans le lecteur de cartes : si oui, il conserve la main et se charge de lire l'intervention ou la commande (s'il y a plusieurs interventions ou commandes sur le lecteur, il ne lit que la première) et rend le contrôle au moniteur qui s'occupera de la suite. Une fois que cette information (commande ou intervention) est traitée, le moniteur passe une nouvelle fois le contrôle au SUPLOC pour vérifier s'il y a encore autre chose dans le lecteur, et ainsi de suite.

IV.3.4.3. Le SUPGEN

Le superviseur général (SUPGEN) est chargé de gérer la conversation pour les points suivants :

. gérer dynamiquement les arrivées et les départs des participants et les signaler aux différents SUPLOC actifs,

. mettre à jour la version complète du fichier de conversation et à chaque arrivée le fichier provisoire du participant concerné.

1) Initialisation d'une réunion :

Une fois qu'une réunion entre plusieurs participants a été décidée, il suffit d'indiquer le nombre de ces participants, leur numéro de passe, leurs noms et le nom de la machine virtuelle qu'ils utilisent. Il est possible par la suite d'ajouter ou de retirer un participant.

2) Définition et initialisation d'une session SUPGEN :

Le SUPGEN doit être actif à chaque fois qu'il peut y avoir un ou des participants. La machine virtuelle sur laquelle est implémenté le SUPGEN est donc activée, le programme est lancé et la machine virtuelle est déconnectée par programme (pour ne pas monopoliser une console). La session SUPGEN commence à cet instant et se termine soit sans intervention à la fin d'une session CP, soit en reconnectant la machine et en tuant le programme (il n'y a rien à récupérer à la fin d'une session SUPGEN puisque toute l'information est conservée au fur et à mesure dans les fichiers). A chaque début de session, le SUPGEN exécute les procédures suivantes :

. chargement à partir du fichier A de la liste des participants et de la date et l'heure de leur dernière participation,

. initialisation de la table permettant l'accès au fichier de conversation à partir du fichier B,

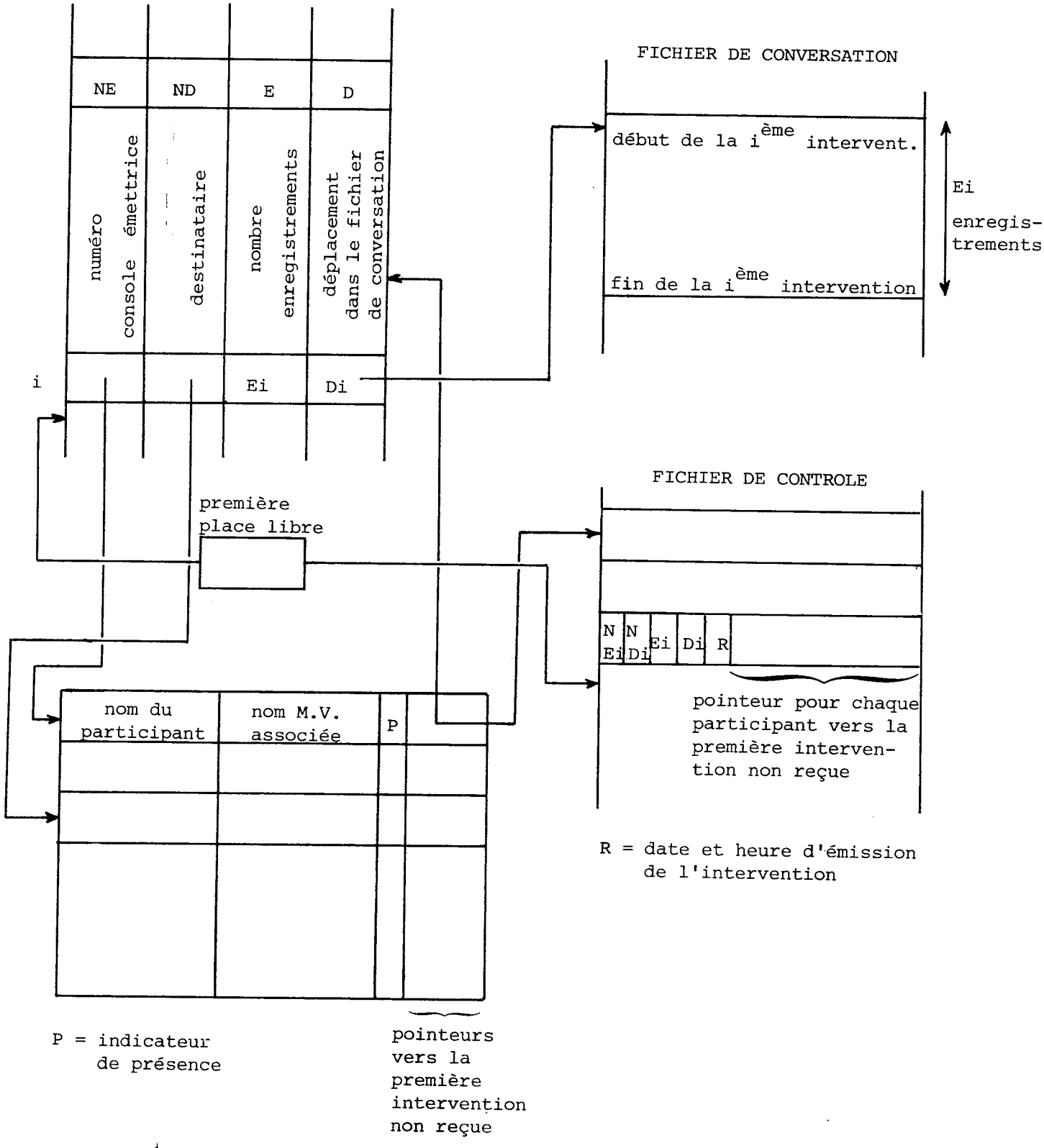


Figure 21 - Accès au fichier de conversation

. initialisation du reste de la structure de données.

Ensuite, le SUPGEN se met en attente d'une interruption sur le lecteur de cartes virtuel. Plusieurs cas d'interruptions peuvent se produire :

1) Arrivée d'un participant

. vérification des renseignements fournis : numéro I, nom MV, nom participant A. S'ils sont inexacts, le SUPGEN envoie un accusé de réception négatif et finit ce traitement (WAIT ou traitement de la commande ou de l'intervention suivante).

. Transmission de la liste des participants à la machine virtuelle qui vient de se connecter (participant A).

. Transmission à chaque participant présent du signal d'arrivée du participant A.

. Mise à jour de la table de présences.

. Mise à jour de la date et de l'heure de dernière participation de A (nécessaire pour avoir une date et une heure assez précises en cas de cassure système).

. Vérification dans TABPERS de la première intervention non reçue par A (numéro I dans la table) :

- si TABPERS(I) = -1 alors rien en retard, donc rien à faire,
- sinon transmission de l'intervention TABINT3(I) à la machine virtuelle M, puis de toutes les interventions suivantes destinées au participant A.

LISTE DES PARTICIPANTS

FICHIER DES PARTICIPANTS

(la date et l'heure sont modifiées à chaque arrivée ou départ pour un participant)

NBR	NOM MV	NOM PAR	DATE	HEURE	. . . . .
. . . . .					

NBR = nombre de participants
NOM MV = nom de machine virtuelle
NOM PAR = nom du participant
DATE = date dernière participation
HEURE = heure dernière participation

TABLE DES PARTICIPANTS

chargée à l'initialisation à partir du fichier des participants

NOM MV	NOM PAR	DATE	HEURE
MV A	PAR A	MM/DD/YY	HH,mm,ss
MV B	PAR B	MM/DD/YY	HH,mm,ss
MV C	PAR C	MM/DD/YY	HH,mm,ss
	ANONYME		

La dernière entrée de la table comprend le participant ANONYME, ce qui simplifie le traitement des interventions anonymes.

Figure 22

2) Départ d'un participant

Les renseignements reçus sont bons car transmis par le logiciel local de la machine virtuelle participante sans intervention du participant (qui tape simplement la commande FINI).

. Mise à jour de la table de présence.

. Transmission à chaque participant présent du signal de départ du participant A.

3) Réception d'une intervention

. Mise à jour de la table des interventions (numéros consoles émettrices et destinataires, taille de l'intervention et déplacement dans le fichier de conversation).

. Mise à jour du fichier de conversation C.

. Recherche dans TABPERS, suivant deux cas :

- intervention publique : on doit rechercher parmi tous les participants absents, ceux qui n'ont pas d'intervention en retard et signaler qu'alors celle-ci est la première, ce qui peut s'exprimer par :

```
pour I := 1 jusqu'à nombre de participants faire
    si ( PRESENT(I)) et (TABPERS(I) = -1)
    alors TABPERS(I) := PINT ;
```

- intervention privée : on doit seulement chercher si le destinataire (numéro DEST) est absent pour vérification (s'il est présent, l'intervention lui a été transmise directement, mais le SUPGEN la reçoit pour mettre à jour le fichier de conversation) et si oui et qu'il n'a encore aucune intervention en retard, on indique que celle-ci est la première. Cela peut s'exprimer par :

```
si ( PRESENT(DEST)) et (TABPERS(DEST) = -1)
alors TABPERS(DEST) := PINT ;
```

. Mise à jour du fichier B d'accès au fichier de conversation (table de conversation + copie de TABPERS).

. Incrémentation des pointeurs pour la table des interventions (PINT) et pour le fichier de conversation (DEPCONV).

4) Retour dans la conversation

(cas d'une intervention ne figurant pas dans le fichier provisoire de la machine virtuelle demanderesse).

. Recherche dans la table des interventions suivant le type de commande (numéro ou recherche sur nom d'émetteur).

. Si l'intervention est trouvée, il faut vérifier que la machine virtuelle demanderesse est destinataire de cette intervention :

- si oui, on la transmet à la machine virtuelle demanderesse en signalant qu'il s'agit d'une intervention "revue",
- sinon, on abandonne et on envoie un accusé de réception négatif, sauf dans le cas d'une recherche sur nom d'émetteur, auquel cas on recommence.

Pour résumer le fonctionnement du SUPGEN, on peut dire qu'il sera généralement en attente d'interruption et que son schéma d'exécution général est le suivant :

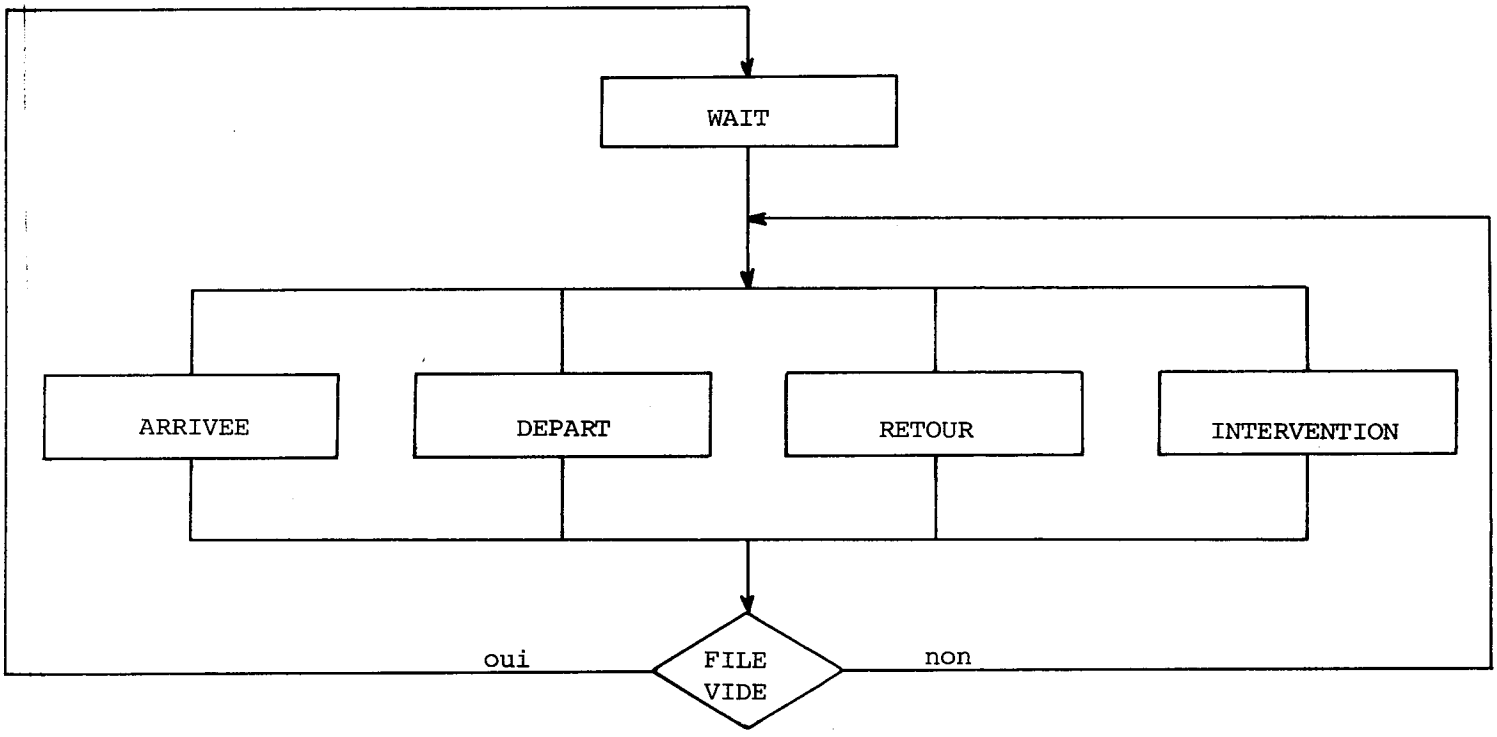


Figure 23

5) Fonctionnement dans le cas de plusieurs réunions simultanées

Actuellement, la version testée ne supporte qu'une seule réunion à un instant donné pour une copie du SUPGEN. Il est évident que l'on peut résoudre le problème de réunions simultanées en activant plusieurs machines virtuelles possédant chacune une copie du SUPGEN et en donnant aux SUPLOC les adresses des machines virtuelles supportant le SUPGEN de chaque réunion. Une solution plus élégante est la suivante :

. chaque réunion porte un titre (par exemple, de huit lettres) :

- lors de sa demande de participation, le participant donne, en plus de son numéro, de son nom personnel et du nom de sa machine virtuelle, le nom de la réunion à laquelle il veut participer (il peut être recensé dans plusieurs réunions) ;

- la même structure de données que pour une réunion unique est implémentée pour chaque réunion, c'est-à-dire que si on recense huit réunions simultanées, le SUPGEN dispose de huit tables de participants ; il vérifie alors dans la table qu'il retrouve, en connaissant le nom de la réunion, les renseignements fournis par l'utilisateur ; il lui fournit en retour la liste des participants et un numéro correspondant à la réunion à laquelle il participe.

. Dans toute la suite, le SUPGEN reconnaît toute transmission par rapport au numéro de réunion que le SUPLOC émetteur lui précise. Il peut alors retrouver les données dont il a besoin.

Le schéma devient alors :

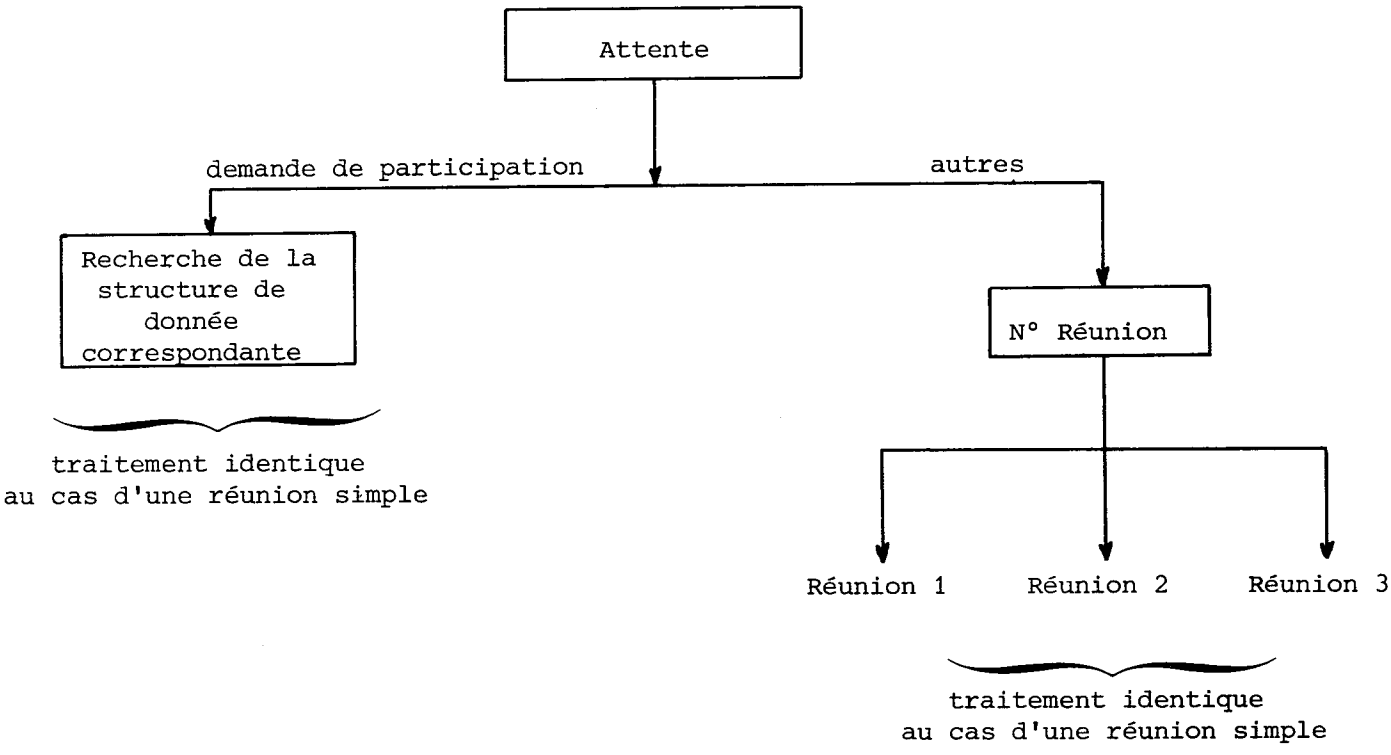

Attente
demande de participation
autres
Recherche de la structure de donnée correspondante
N° Réunion
traitement identique au cas d'une réunion simple
Réunion 1
Réunion 2
Réunion 3
traitement identique au cas d'une réunion simple

Figure 24

La structure de données est la suivante :

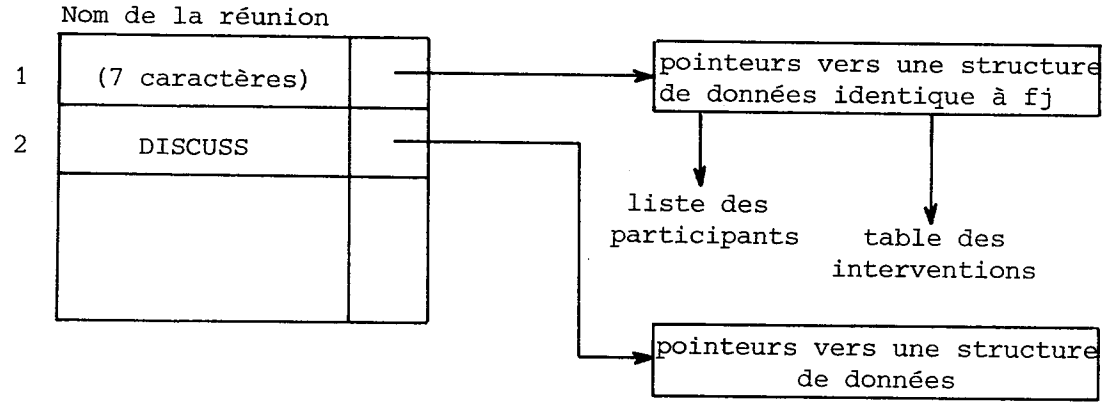


Figure 25

Les noms des fichiers sont générés de façon très simple par :

. nom de fichier de conversation = nom de la réunion,

. nom des fichiers de contrôle = nom de la réunion suivi d'une lettre particulière à chaque type de fichier (exemple : fichier d'accès au fichier de conversation est associé à la lettre A).

Exemple : réunion de nom DISCUSS

le nom du fichier est retrouvé par :

```
TABREUNION (2)
```

le nom du fichier d'accès au fichier DISCUSS est trouvé par :

```
NOM (0/7) := TABREUNION (2) ;
NOM (7/1) := "A" ;
```
} NOM(0/8) = DISCUSSA

le nom du fichier contenant la liste des participants est trouvé par:

```
NOM (0/7) := TABREUNION (2) ;
NOM (7/1) := "L" ;
```
} NOM(0/8) = DISCUSSL

IV.3.4.4. Cas particulier d'une réunion normale

On rappelle que dans une réunion "normale" une partie du logiciel doit tenir le rôle de l'animateur et assurer en particulier l'unicité de parole. En dehors de tout ce que l'on a décrit jusqu'à présent, on définit quatre états fondamentaux dans lesquels peut se trouver la console par rapport à la conversation :

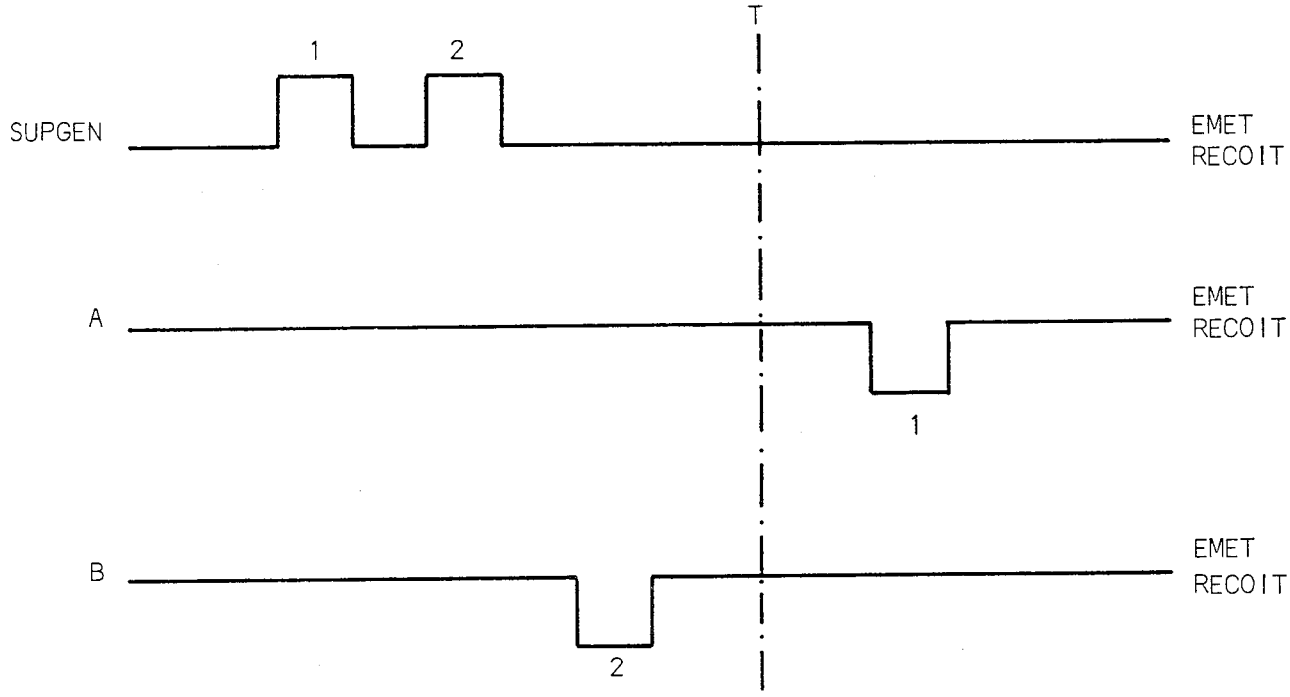
P = état PAROLE (peut émettre),
PA = attente fin de transmission,
A = attente de PAROLE
E = état ECOUTE.

Ainsi, le SUPLOC de chaque site participant doit gérer un automate à quatre états. Le SUPGEN doit gérer n automates à trois états fondamentaux : P, A, E. En effet, il n'a pas besoin de savoir si la console émettrice est en attente de fin de transmission d'INTERVENTION, car c'est géré directement par le SUPLOC.

On a donc des automates liés deux à deux. Etant donné que les SUPLOC gèrent eux-mêmes l'envoi et la réception d'INTERVENTIONS, les n automates à quatre états des SUPLOC doivent être compatibles entre eux. En particulier, un seul automate devra à chaque instant être en état P ou PA. Or, le problème est qu'il n'y a pas simultanéité entre un changement d'état dans le SUPGEN et le changement correspondant dans un SUPLOC.

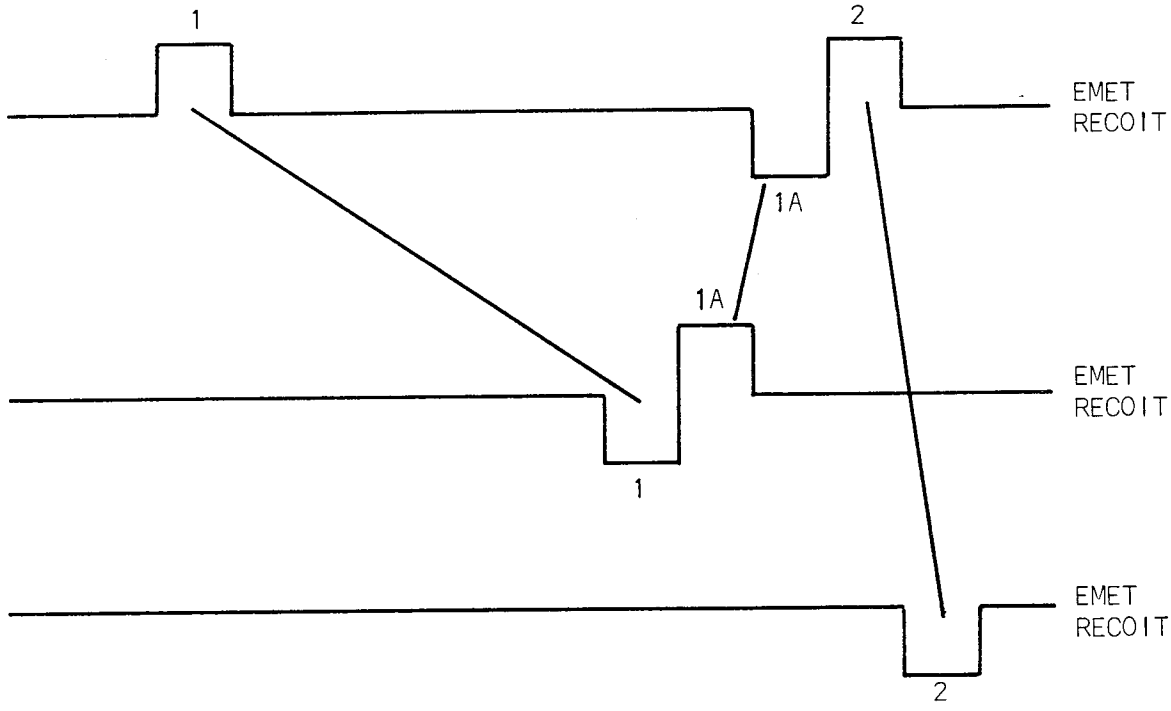
Expliquons ce problème sur un exemple : on considère deux consoles A et B. A est en état PAROLE (dans SUPGEN et dans SUPLOC(A)). B est en état ATTENTE DE PAROLE (dans SUPGEN et dans SUPLOC(B)). A dépasse le temps maximum de PAROLE qui lui est imparti. Alors le SUPGEN lui envoie une commande de fin de PAROLE et donne la PAROLE à B. On peut ainsi avoir le cas de la figure si B reçoit la commande 2 avant que A ne reçoive 1, les consoles A et B sont en état PAROLE en même temps. C'est donc un cas de *conflit*. Une solution possible est que le SUPGEN attende les accusés de réception de ses commandes. Le cas précédent est décrit dans la figure 26 . Ceci implique l'existence d'états intermédiaires. La figure 27 décrit la solution adoptée.

Figure 26 - Sans accusé de réception



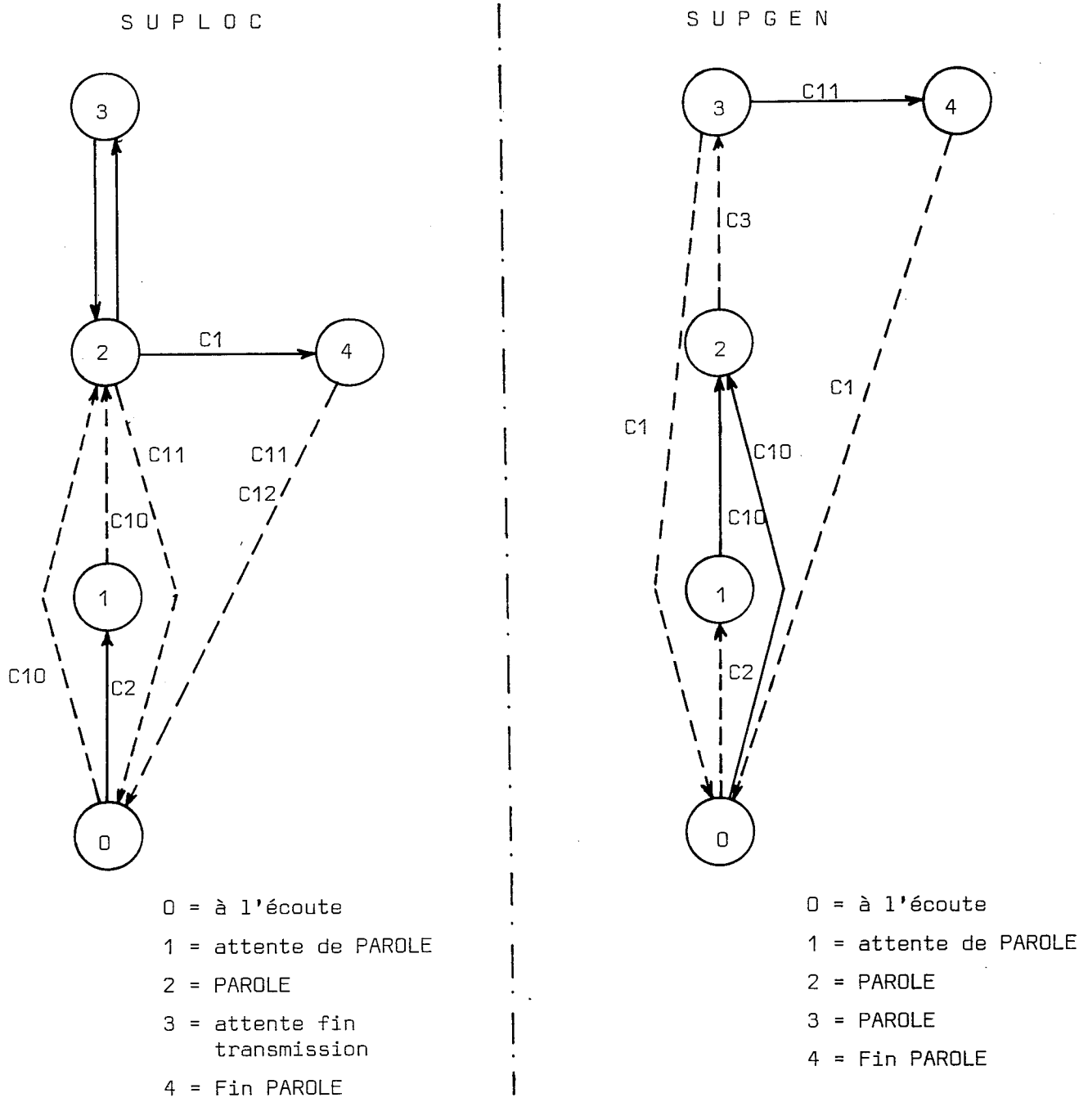
1 = retire la PAROLE
2 = donne la PAROLE

Figure 26 bis - Avec accusé de réception



1 = retire la PAROLE
2 = donne la PAROLE
1A = accusé de réception de 1

Figure 27 - Automates réseau



Signification des commandes :

. Commandes SUPGEN :

C10 : donne la PAROLE

C11 : retire la PAROLE

C12 : accusé de réception de C1

. Commandes SUPLOC :

C1 : retrait de PAROLE

C2 : demande de PAROLE

C3 : accusé de réception de C10

C4 : accusé de réception de C11

Les tableaux de validité et d'envoi d'accusés de réception sont les suivants :

SUPGEN

	C1	C2	C3	C4
0	0/C12	1	-	0
1	-	1	-	-
2	0/C12	2	3	-
3	0/C12	-	3	-
4	0/C12	-	4	0

SUPLOC

	C10	C11	C12
0	2/C3	0/C4	0
1	2/C3	0/C4	1
2	2/C3	0/C4	-
3	Mise en attente commande		
4	4/C3	0/C4	0

Remarque :

plusieurs commandes identiques peuvent être envoyées à la suite (avec un time-out). Pour éviter qu'une commande très attardée ne vienne tout perturber, on envisagera de numéroter les commandes. A la réception de commandes, on ne prendre donc pas en compte celles ayant un numéro inférieur au dernier numéro reçu.

Les commandes sont décrites dans la figure 27 . Un SUPLOC peut traiter les commandes suivantes :

. réception	. Emission
C10 donne la PAROLE	C1 retrait de PAROLE
C11 retire la PAROLE	C2 demande la PAROLE
C12 accuse réception de C1	C3 accuse réception de C10
C13 signale l'arrivée de participant	C4 acuse réception de C11
C14 signale le départ de participant	C5 demande pour participer
C15 donne liste des participants	C6 signale départ de participant

Le SUPGEN fait exactement l'inverse.

Il est évident que cette liste n'est pas limitative. Par ailleurs, les commandes C5, C6, C13, C14, C15, n'influent pas sur les automates réseau, étant entendu qu'à leur arrivée à la réunion, les participants sont en état ECOUTE.

IV.3.4.5. Déroulement d'une réunion

On peut considérer qu'il existe trois niveaux d'utilisation pour une réunion :

. niveau global d'une réunion : il s'agit de tout ce qui se passe entre le moment où la tenue d'une réunion est décidée et le moment où l'on décrète sa fin ;

. niveau session SUPGEN : il s'agit du déroulement des programmes SUPGEN entre le moment où on les lance et le moment où on les arrête (coïncidant en général avec une session CP) ;

. niveau session utilisateur : il s'agit de toute ce que peut faire un participant entre le moment où il se connecte à CONGRES et celui où il se déconnecte.

La figure 28 explique ces notions sur un exemple :

1) Niveau global d'une réunion :

il faut initialiser cette réunion et la détruire en fin de réunion (on conserve une copie du fichier d'interventions). Pour l'initialisation, il suffit de donner un nom à la réunion et la liste des participants et des machines virtuelles qu'ils utiliseront. On pourra, en dehors d'une session SUPGEN ajouter ou retirer des participants, simplement en les supprimant ou en les ajoutant en fin de liste. A la fin d'une réunion, on efface tous les fichiers, en conservant une copie du fichier de conversation et de son fichier d'accès, sur bande par exemple.

2) Niveau session SUPGEN :

le SUPGEN doit être actif à chaque fois qu'un participant est actif pour CONGRES. On l'initialise donc à chaque fois qu'il risque d'y avoir une participation. Les tables nécessaires sont mises à jour par l'intermédiaire de fichiers dont il connaît le nom (ou dont il peut reconstituer aisément le nom). Une fois cette initialisation faite, le SUPGEN se met en attente d'une interruption qui peut signifier :

. une commande (arrivée, départ ..),

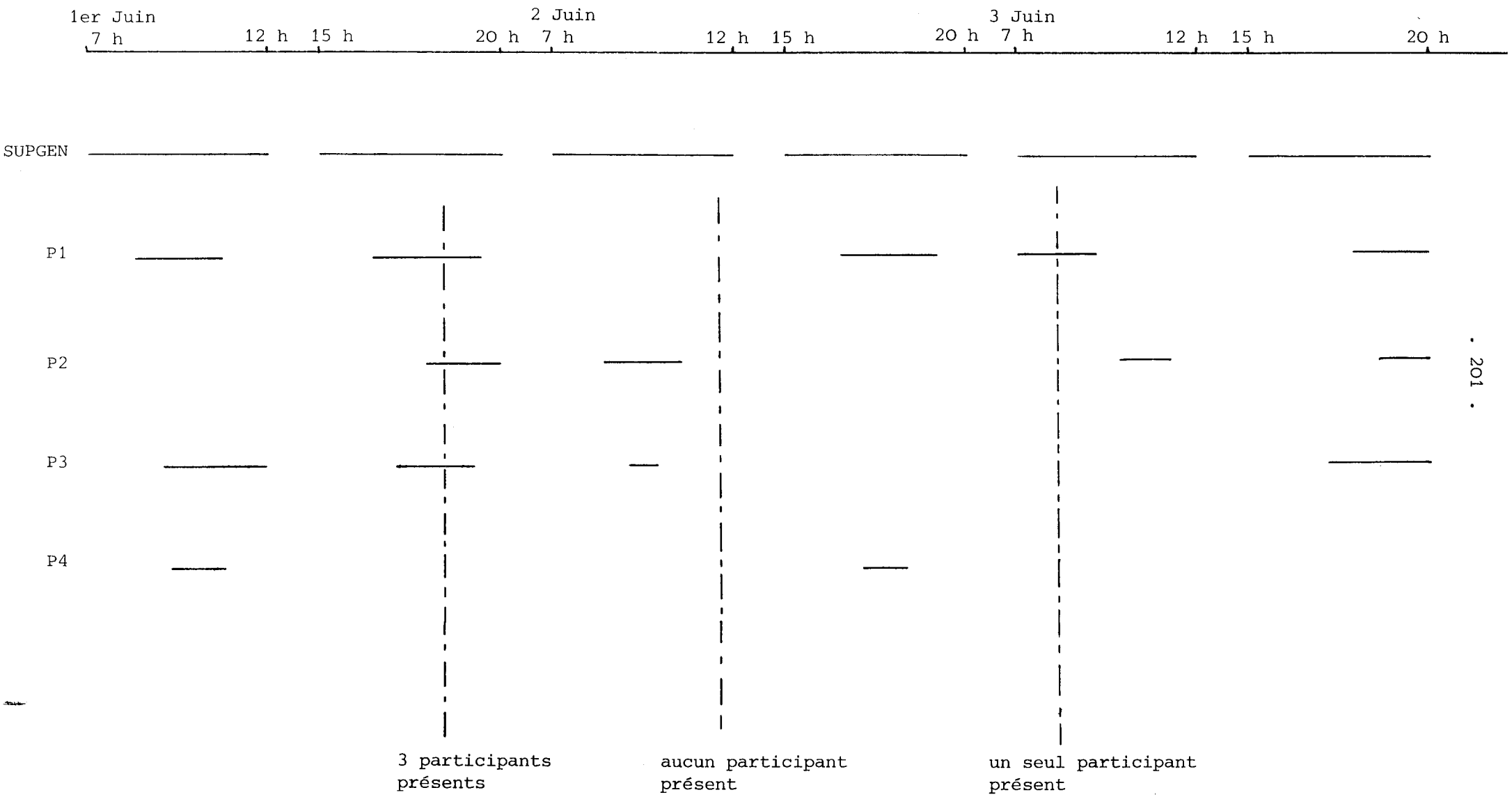
. une intervention.

Il traite ces interruptions suivant les critères exprimés au § IV.3.4.3. Pour terminer une session SUPGEN, il suffit de lui imposer une interruption externe, aucune sauvegarde n'étant nécessaire puisqu'elles sont faites au fur et à mesure des besoins pendant le traitement.

3) Niveau session utilisateur :

une session utilisateur commence au moment où il lance le programme local de CONGRES. Il reçoit alors un message lui indiquant les précisions qu'il doit donner pour participer à une réunion ou pour travailler uniquement en local. S'in participe à une réunion, la demande est faite par le SUPLOC qui reçoit en retour tous les renseignements nécessaires au bon fonctionnement du système local (liste des MV, interventions en retard, ..).

Figure 28 - Exemple de réunion sur trois journées avec quatre participants (P1, P2, P3, P4)



IV.3.5. Conclusion

Tout ce qui a été décrit dans ce chapitre a été implémenté et testé. Seule la partie concernant les réunions normales n'a pas été utilisée, surtout parce qu'elle ne peut avoir d'intérêt que dans le cadre d'une étude socio-psychologique.

On peut ajouter que nous avons laissé de côté quelques utilitaires qui peuvent être utilisés hors du cadre d'une réunion, en particulier :

. SORFICH : permet de sortir,sur fichier que l'on peut lister, les dessins (de type assez simple),

. EFDES : permet d'effacer du fichier personnel un dessin. Les fichiers (personnel et d'accès) sont alors recompactés.

Enfin un fichier de traces (historique) est mis à jour automatiquement par le SUPGEN. Il contient :

. un historique des départs et des arrivées,

. pour chaque intervention :

- le numéro de console émetteur,
- le numéro de console récepteur,
- la longueur de l'intervention.

CHAPITRE V

CONCLUSION

Les différentes étapes dans la conception, la réalisation et l'utilisation de ce système ont été les suivantes :

• Dans un premier temps (1975), on pensait concevoir un système de réunions calqué sur les réunions en face à face, le logiciel tenant le rôle d'animateur (cf. réunions normales). On développait en parallèle un interpréteur permettant en particulier la construction de dessins.

• Au début de l'année 1976, un certain nombre d'options sur le plan construction de dessins avaient été prises et un interpréteur expérimental était mis à la disposition des utilisateurs. Ceci a permis des tests d'utilisation du système en local (constructions et modifications de dessins).

• Vers la fin de l'année 1976, un prototype permettant des réunions entre deux consoles graphiques était implémenté et testé. Une période d'utilisation, particulièrement entre un IBM 2250 et un TEKTRONIX 4010, a suivi cette implémentation. A noter que le TEKTRONIX 4010 était à l'époque connecté au 360/67 par une ligne à 110 bauds ce qui, associé au fait que le 4010 ne possède pas de mémoire d'entretien, en rendait l'utilisation assez fastidieuse. Cependant, à partir de ces expériences d'utilisation, on a pu construire le système CONGRES.

• CONGRES permettant des réunions à n consoles (n limité par le nombre d'accès à CP via CYCLADES), a été terminé en mai 1977. Considérablement amélioré par rapport à ses prédecesseurs, tant du point de vue construction et modification de dessins que du point de vue indépendance par rapport au matériel ou possibilités de discussion, il a été utilisé en local et pour quelques expériences de réunions. C'est ce prototype qui a été décrit dans les chapitres III et IV.

Dans ce dernier chapitre, on dresse un bilan de cette expérience et on fait un certain nombre de propositions qui semblent valables pour des systèmes de ce type. On exprime ensuite le coût d'utilisation d'un tel système en coûts élémentaires. Enfin, on expose les conclusions que l'on peut tirer de cette étude.

V.1. BILAN ET AVENIR

On ne peut dresser un bilan définitif de ce qui n'était qu'une expérience. Il faudrait maintenant, à partir de ce prototype, mettre sur pied un ensemble de réunions de longue durée (plusieurs mois).

Un certain nombre d'outils de mesure sont en place qui devraient permettre de mesurer :

- les durées de participation moyennes,
- le nombre de messages transmis par un participant,
- la longueur moyenne des interventions,
- les proportions d'interventions publiques et privées, signées et anonymes,
- etc ...

A partir de ces observations, on pourra décider en connaissance de cause de l'utilité des dessins, de leur complexité, de ce qui est utile, de ce qui au contraire manque, etc .. Nous pensons que toutes les remarques permettraient d'élaborer un système répondant effectivement à des besoins.

Nous pensons d'ores et déjà pouvoir avancer un certain nombre de propositions.

V.1.1. Propositions techniques

Ces propositions sont en particulier celles qui ont été faites dans le deuxième chapitre de cette thèse. Nous n'y reviendrons donc pas, mais nous ajouterons que de tels systèmes pourraient être réalisés sur des mini-calculateurs, à condition qu'ils puissent supporter une mémoire secondaire importante (fichiers).

V.1.2. Propositions générales

• Réaliser un système dont au moins un sous-ensemble (transmissions de textes) soit aisément compréhensible par des non-informaticiens (rendre les aspects techniques aussi invisibles que possible à l'utilisateur).

• Lors des premières utilisations d'un tel système, prévoir la participation d'un "aide". Cet aide est une personne connaissant parfaitement le système (ayant si possible participé à son élaboration) qui aidera chacun à bien se servir de ce moyen de communication en répondant aux questions des participants. Son rôle sera très important, car il évitera aux utilisateurs de se trouver devant une "machine" (le logiciel joue de toutes façons un rôle d'aide par des messages d'erreurs explicites, etc ..).

• Les principales caractéristiques à prendre en compte avant d'élaborer un tel système sont :

. les facilités d'accès aux terminaux (graphiques et alphanumériques),
. la fiabilité du réseau et de l'ordinateur utilisés,
. les possibilités d'accès aux autres services qu'offrent les ordinateurs du réseau (bases de données, ..),
. le coût.

• Comme tout moyen nouveau de communication, il faudra prévoir quelque peu les réactions des participants sous peine de voir ce type de système rejeté à court terme. Les facteurs suivants semblent particulièrement importants :

. taille du groupe :

il semble que ce mode de communication ne soit intéressant que pour des groupes de taille moyenne, au moins dans l'état actuel des moyens d'entrée et de sortie d'information ;

. attitude du participant envers l'ordinateur :

deux cas peuvent se produire lorsqu'une personne n'ayant aucune expérience dans l'utilisation des ordinateurs va utiliser un tel système : ou bien elle se décourage rapidement devant quelques obstacles techniques, ou bien, intéressée par l'aspect nouveauté, elle trouve amusant ce moyen d'expression. Un informaticien, en revanche, ne butera pas sur les problèmes techniques éventuels.

. attitude individuelle dans la réunion :

on a vu que ce moyen de communication laisse l'utilisateur libre de participer lorsqu'il le désire. Si tous les membres du groupe restent passifs, les communications risquent d'être très réduites. D'un autre côté, il est très facile d'ignorer un membre du groupe, tout simplement en ignorant ses questions ou en ne lui transmettant qu'un minimum d'information. Il faudra donc que le groupe soit suffisamment homogène pour que la réunion puisse se dérouler librement.

V.2. ÉVALUATION DU COÛT

On s'intéresse essentiellement aux coûts d'utilisation d'un tel système. On ne discute donc pas, dans cette partie, les coûts de développement et de maintenance qui sont cependant non négligeables.

On peut décomposer le coût d'utilisation en cinq coûts élémentaires :

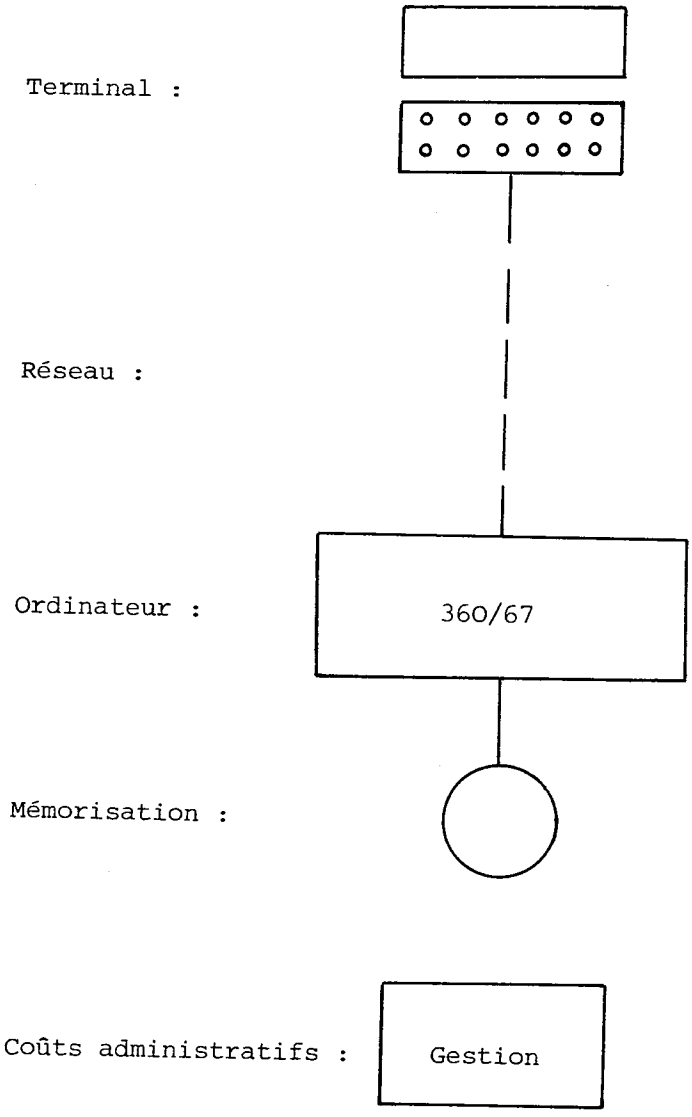
1) Coût du terminal

il s'agit du coût de location ou d'amortissement des terminaux utilisés. Il varie fortement suivant le type de terminal (graphique ou alphanumérique). On peut estimer ce coût comme assez faible dans le cadre d'un Laboratoire Universitaire par exemple, puisque ces terminaux existent et sont utilisés parallèlement pour d'autres applications. Cependant, dans le cadre d'une distribution par une firme, ce coût serait pris en compte. Toutes les observations tendent à prouver que ce coût va diminuer dans les années à venir.

2) Connexion à l'ordinateur

On comprend dans cette rubrique tous les coûts imposés par la connexion du terminal à l'ordinateur où sont implémentés les programmes. Cette connexion se fera en général par un réseau et comprend donc en particulier, le coût de la connexion du terminal au réseau.

Décomposition des coûts d'utilisation



3) Utilisation de l'ordinateur

on comprend dans cette rubrique les coûts en points comptables sur l'ordinateur hôte des programmes. Dans le cas du système CONGRES, ils se décomposent en :

. pour chaque utilisateur, coût d'utilisation des programmes locaux (essentiellement interpréteur et superviseur local). Ce coût est facilement comptabilisé puisqu'il entre dans la comptabilité de la machine virtuelle propre à l'utilisateur. Il sera donc compris dans la facture générale que reçoit l'utilisateur pour cette machine virtuelle ;

. pour l'ensemble des utilisateurs, partage du coût d'utilisation de la machine virtuelle centrale (SUPGEN). Ce partage peut être effectué soit de façon égale entre chaque participant, soit, ce qui semble plus logique, en fonction du travail imposé au logiciel SUPGEN par chacun.

4) Mémorisation

il s'agit du coût d'immobilisation imposée par les fichiers qui peuvent être de grande taille.

5) Coûts administratifs

Il est évident que ces coûts dépendent du type de réunion, de l'attitude des participants, du nombre et de la complexité des messages transmis (en particulier des dessins) etc ..

Remarque :

Cependant, cette décomposition en cinq points permet de se faire une idée des coûts d'utilisation et de fournir des informations de base pour la facturation de l'utilisation d'un tel système.

Pour le système PLANET dont on a parlé dans le premier chapitre et qui consiste essentiellement en un système de réunions asynchrones avec uniquement des transmissions de textes alphanumériques, une évaluation du coût pour une réunion effectuée dans le cadre de la NASA a donné les chiffres suivants :

- coût moyen par message : 2,07 dollars
- coût moyen par heure et par participant : 16,45 dollars

Une étude a également été faite sur l'évaluation du coût horaire par participant dans un système de réunions sans transmission de dessins (donc sans coût graphique), dans le cadre de la NASA (donc sans profits commerciaux). Le coût des terminaux a été négligé parce qu'ils existent sur tous les sites et sont déjà utilisés pour d'autres tâches :

- coût réseau (2) : 1,50 dollars
- coût ordinateur (3) : 2,50 dollars
- autres coûts (4 + 5) : 1 dollar

d'où le coût moyen par heure et par participant : 5 dollars.

Même en considérant que ce n'est qu'une évaluation faite dans le cadre particulier de la NASA, cela ouvre des perspectives intéressantes pour les systèmes de réunions assistées par ordinateur.

Enfin, pour donner un exemple pratique du coût des conférences assistées par ordinateur, nous donnons ci-dessous un bref résumé des travaux de M. Turoff [TU2] sur ce sujet. Tout ce qui suit est relatif à un système basé sur un mini-ordinateur (INTERDATA 7/32).

M. Turoff prend comme formule de base :

$$\text{coût} = a + bN$$

où a représente le coût de base (équipement et développement) et bN le coût d'utilisation en fonction du nombre N de portes d'accès à l'ordinateur.

L'étude du coût donne alors les résultats suivants (équipement acheté et amorti sur cinq ans) :

. coût unité centrale :
  coût UC = 100 000 + 625.N dollars,

. coût du sous-système de disques :
  coût disque = 40 000 + 800.N dollars,

. coût des lignes d'accès :
  coût portes = 1 000.N dollars,

. coûts annexes (bandes, fournitures, ..) :
  coûts annexes = 600.N dollars

. coûts de développement :
  coût développement = 100 000 + 800.N dollars.

D'où le coût total d'investissement :

coût = 240 000 + 3 825.N dollars

M. Turoff en déduit alors le résultat intéressant suivant (pour son système) en supposant un amortissement sur cinq ans avec cinquante semaines de cinq jours de quinze heures d'utilisation :

amortissement :

. coût par heure (en dollars) :	0,61	0,41	0,34	0,31
. nombre de portes :	32	64	96	128

A ces coûts d'investissement s'ajoutent les coûts d'utilisation que M. Turoff évalue comme suit (l'utilisation de l'équipement n'entre plus dans cette rubrique, puisqu'on l'a déjà recensé précédemment) :

. coût en personnel :
100 000 + 1 575.N dollars

. coût de maintenance :
16 800 + 300.N dollars

. coût des locaux :
25 000 + 100.N dollars

. coût réseau (TELNET) :
14 400 + 300.N dollars

d'où le coût total (matériel + développement + utilisation) :

coût annuel : 204 200 + 3 050.N dollars

Sans entrer plus avant dans les détails, signalons cependant que M. Turoff fait également une étude du coût pour l'utilisateur, suivant que le système est peu utilisé (50 %) ou très utilisé.

V.3. CONCLUSION

Le système CONGRES représente une expérience intéressante dans le cadre des systèmes de réunions assistées par ordinateur, d'une part du fait que dans un cadre d'étude assez nouveau il comprend des transmissions de dessins, d'autre part parce qu'étant opérationnel, il a pu servir comme expérience d'utilisation.

Nous pensons qu'à partir du logiciel CONGRES, deux grands axes de recherche peuvent être définis :

- mettre sur pied des réunions expérimentales :

  . asynchrones de longue durée (plusieurs mois), afin d'étudier tous les aspects du système, en particulier du point de vue utilisation ;

  . synchrones (réunions normales) et se demander si ce type de réunions peut avoir un intérêt pour des études de comportement de groupe, par exemple ;

- construire, à partir de ces expérimentations, particulièrement grâce aux réunions asynchrones qui représentent le type de réunions à développer (des réunions normales n'ayant qu'un intérêt expérimental), un système plus complet et plus performant que l'on pourrait ainsi adapter aux besoins des utilisateurs.

En se reportant par exemple à la comparaison entre ce type de communication et les moyens plus connus de communication (cf. II.2), on constate que certains arguments plaident en faveur des communications assistées par ordinateur et que d'autres jouent en sa défaveur.

En fait, nous pensons que les systèmes de réunions assistées par ordinateur seront adaptés à la résolution de certains problèmes et pas à d'autres. Il faut d'ailleurs insister sur le fait qu'elles ne doivent pas *remplacer* les moyens connus jusqu'alors, mais les *compléter*.

Sans revenir sur les caractéristiques d'un système de réunions assistées par ordinateur, on peut faire deux remarques qui concernent deux aspects :

. le coût de tels systèmes va tendre à diminuer et devenir, selon les cas, de un tiers à une demie fois moins important que celui de réunions en face à face (voyages + séjours etc ..) ;

. on a constaté aux Etats Unis, dans une expérience de réunions assistées par ordinateur, que les transmissions entre la côte ouest et la côte est s'étendaient bien en dehors des heures habituelles des communications téléphoniques.

On peut donc affirmer que ce domaine de recherche ne doit pas être abandonné, alors qu'il peut apporter des solutions originales à certains problèmes de communications. Des recherches pourraient d'ailleurs être faites dans des cadres plus restreints, par exemple l'étude d'un système informatique de communications au sein d'une entreprise.

R É F É R E N C E S

AN ANSART J.P.
Système interactif dans un environnement réseau ; connexion d'une machine à mémoire virtuelle IBM 360/67 au réseau CYCLADES
Thèse de Docteur-Ingénieur, Grenoble, Février 1976.

BE BEAL, BOHLEN & RAUDABAUGH
Les secrets de la dynamique de groupe
(Livre). Chotard et Associés, 1969.

CH1 CHUPIN J.C. & SEGUIN J.
M A D R E
Centre Scientifique CII-HB de Grenoble, Rapport de contrat DRME, Avril 1974.

CH2 CHUPIN J.C.
Répartition d'applications et de banques de données sur un réseau général d'ordinateurs
Thèse de Doctorat d'Etat, Grenoble, Octobre 1977.

CR CROCUS
Systèmes d'exploitation des ordinateurs
(Livre). Dunod Editeur.

D DAKIN A.
Computer assisted planetary thinking
ICCC 1976.

DA DAY L.
The future of computer and communication services
AFIPS, NCCE 1973.

FE FETTER W.
Computer graphics in communication
(Livre). McGraw Hill Engineering graphics monograph, 1964.

FL FLAMENT C.
Réseaux de communication et structures de groupe
(Livre). Dunod, 1965.

FR FREESE F.
Computer communication and growing pains
ICCC 1976.

GA1 GARDAN Y.
Système de réunions avec des consoles graphiques ou alphanumériques reliées à un réseau (version expérimentale)
Rapport de recherche, ENSIMAG, Décembre 1975.

GA2 GARDAN Y.
Un système de réunion, version expérimentale sous CP pour deux consoles
Rapport technique, ENSIMAG, 1976.

GA3 GARDAN Y.
Notice utilisateur de CONGRES. Construction de dessins en local
ENSIMAG, Juin 1977.

GA4 GARDAN Y.
Notice utilisateur de CONGRES. Participation à une réunion.
ENSIMAG, Juin 1977.

GA5 GARDAN Y.
A terminal based conference system permitting sketch communication
Proposé à ICCC 1978.

GR GROUP COMMUNICATION THROUGH COMPUTERS
FORUM : pragmatics and dynamics
Institute for the future, Menlo Park, California, R.35, 1975.

HA HALL T.W.
Implementation of an interactive conference system
AFIPS SJCC 1971.

HI HILTZ R.
A social scientist looks at computer conferencing
ICCC 1976.

IB IBM
CP 67, Program logic manual

LA LAUGIER, LEDUC LEBALLEUR, LUCAS, MARTINEZ
GRIGRI, logiciel de base pour l'utilisation des consoles de visualisation du CICG
Grenoble, T.29, Novembre 1975.

LB LE BIHAN J.
La répartition des données dans les réseaux informatiques
Congrès AFCET, Panorama de la Nouveauté Informatique en France, Novembre 1976.

LE LEDUC LEBALLEUR A.
Conception et réalisation d'un logiciel graphique de base indépendant de son contexte. Application au logiciel GRIGRI.
Thèse de Docteur-Ingénieur, Grenoble, Septembre 1977.

LI LIPINSKI H. & MILLER R.
FORUM : a computer assisted communication medium
ICCC 1974.

LL LEDUC LEBALLEUR, LUCAS, MARTINEZ
Conception et réalisation d'un logiciel graphique interactif indépendant du contexte d'utilisation : le logiciel de base GRIGRI
Séminaire de Programmation, ENSIMAG, Juin 1977.

LU1 LUCAS M.
La programmation des consoles de visualisation : les techniques de base
Polycopié, Grenoble, Décembre 1974.

LU2 LUCAS M.
Techniques de programmation et d'utilisation en mode conversationnel des terminaux graphiques
Thèse de 3ème Cycle, Grenoble, 1968.

LU3 LUCAS M.
Contribution à l'étude des techniques de communication graphique avec un ordinateur ; éléments de base des logiciels graphiques interactifs
Thèse de Doctorat d'Etat, Grenoble, Décembre 1977.

MA MACCIO C.
Animation de groupes
Chronique sociale de France, 1972.

ME MENARD
Interface display 2250 IBM - BENSON
Note technique, Grenoble, 1975.

MI MIHALASKI J.
Psychic transmission of information - fact or fancy
ICCC 1976.

MO MORVAN P. & LUCAS M.
Images et ordinateur : introduction à l'infographie interactive
(Livre). Larousse, Septembre 1976.

PA PAIR C.
Les structures de l'information
Polycopié, Cours Alès, 1971.

PO POOLE M.
Fundamentals of display systems
Spartan Books, Washington, Mac Millan and Co Ltd., London, 1966.

RI RINDE J.
TYMNET 1. An alternative to packet technology
ICCC 1976.

SE SEGUIN J. & SERGEANT G.
Interrogation, mise à jour et cohérence d'un fichier à copies multiples réparties sur un réseau d'ordinateurs
Rapport de recherche ENSIMAG, 1977.

SR SHERIDAN T.B.
Technology for group dialogue and social choice
AFIPS FJCC 1971.

SH SHANNON C.E.
A mathematical theory of communication
Key papers in the development of information theory, premier article, 1974.

SU SUTHERLAND I.E.
SKETCH PAD : a man-machine graphical communication system
MIT Lincoln Laboratory Technical Report, n° 296, Janvier 1963.

TU1 TUROFF M.
Session on views of the future : chairman's introduction opposing views
AFIPS NCCE 1973.

TU2 TUROFF M.
The costs and revenues of computerized conferencing
ICCC 1976.

TU3 TUROFF M
DELPHI and its potential impact on information systems
AFIPS FJCC 1971.

UM UMPLEBY S.
Structuring information for a computer based communications medium
FJCC 1971.

VA1 VALLEE J., JOHANSEN R., LIPINSKI M., WILSON T.
Pragmatics and dynamics of computer conferencing. A summary of findings from the FORUM project
ICCC 1976.

VA2 VALLEE J. & WILSON T.
Computer based communication in support of scientific and technical work
NASA, CR. 137-879, 1976.

VAN VANOYE F.
Expression. Communication
Armand Collin, Collection U; 1973.

ZI1 ZIMMERMANN H.
Virtual terminal protocol (VTP)
Bibliothèque CYCLADES, TER 503.

ZI2 ZIMMERMANN H.
High level protocols standardization. Technical and political issues
ICCC 1976.

dernière page de la thèse

AUTORISATION DE SOUTENANCE

VU les dispositions de l'article 3 de l'arrêté du 16 Avril 1974,

VU les rapports de présentation de Messieurs :

- L. BOLLIET, Professeur à l'U.S.M.G.
  etM. LUCAS, Maître-Assistantà l'U.S.M.G.

- Cl. BOKSENBAUM, Maître de Conférences à l'I.U.T. de MONTPELLIER

Monsieur Yvon G A R D A N

est autorisé à présenter une thèse en soutenance pour l'obtention du titre de DOCTEUR-INGENIEUR, spécialité "Génie Informatique".

Grenoble, le 30 Novembre 1977

Ph. TRAYNARD
Président
de l'Institut National Polytechnique