

SUR L'UTILISATION D'UN SGBD RELATIONNEL  
 POUR LA GESTION D'UNE BASE DE DONNEES IMAGE  
 ON THE USE OF A RELATIONNAL DBMS FOR THE MANAGEMENT OF AN IMAGE DATA BASE

- Philippe CHASSIGNET

LACTAMME - ECOLE POLYTECHNIQUE - 91128 PALAISEAU Cedex

## RESUME

*Dans cette note nous présentons un projet original pour la réalisation d'une base de données orientée vers la synthèse d'images et d'animations.*

*L'intérêt de ce système est de permettre la gestion conjointe de données iconographiques et sémantiques et donc d'associer finement la forme et le fond.*

*Pour des raisons de performance ce système est réalisé par la connexion de deux machines spécialisées:*

- une Machine Iconographique chargée de la génération et de la gestion physique des images;
- un Système de Gestion de Base de Données Relationnel permettant:

- i) de mémoriser des programmes de synthèse en relation avec d'autres informations,
- ii) d'y accéder par des requêtes variées.

*Une interface utilisateur met la souplesse inhérente au SGBDR au service d'une large interactivité.*

*Une interface de connexion génère, à partir des données extraites du SGBDR, les commandes pour piloter la MI.*

*Enfin, ce schéma peut être étendu pour conférer au système:*

- des aptitudes à une synthèse automatique et
- la possibilité de piloter une machine iconographique à architecture parallèle fonctionnant selon un principe "d'Image Flow".

## SUMMARY

*In this note we present an original scheme for the realization of a database oriented to images and films synthesis.*

*The interest of such a system is to allow the joint management of iconographic and semantic data and thus to finely connect form and substance.*

*Owing to performances this system is work out by the connection of two specialized machines:*

- an Iconographic Machine in charge of physical generation and management of images;
- a Relational Data Base Management System that allows:

- i) to memorize synthesis programs in relation with other informations,
- ii) to have acces to them by various requests.

*An user interface put the flexibility inherent in RDBMS at the service of a large interactivity.*

*A connection interface produces, from data drawn out of the RDBMS, the commands to control the IM.*

*Finally, this scheme may be enlarged to give the system fitness for:*

- automatic synthesis and
- control of a parallel iconographic machine that works on a principle of "Image Flow".



## 1. INTRODUCTION

L'image permet d'utiliser la richesse d'analyse de l'oeil pour appréhender efficacement des données complexes. Aussi, une part de plus en plus importante des données traitées et produites par l'informatique se présente elle sous la forme d'images. Citons le traitement d'images en cartographie ou en médecine, la synthèse d'images en CAO ou EAO.

Il est de plus intéressant de pouvoir traiter ce type de données de manière similaire à d'autres comme des donneurs numériques, du texte, etc... et surtout de pouvoir toutes les associer.

A cet effet les Systèmes de Gestion de Base de Données Relationnels, s'avèrent des outils très appropriés et, de ce fait, ont été utilisés dans plusieurs applications graphiques [ChFu 79, YaKu 82], et en particulier dans le domaine des bases de données cartographiques [SmAl 79, VSHM 82].

Les images ne constituent alors qu'une part des informations et un grand nombre de données sont associées aux diverses entités qu'elles représentent. Ces bases sont bien sûr utilisées essentiellement en consultation, les requêtes combinant des critères sur certaines données et des localisations graphiques.

Selon une approche différente, le système SMC, développé au LACTAMME et orienté vers les applications iconographiques, offre quand à lui une notion intéressante celle de l'image en tant qu'entité transformable interactivement [Colo 84].

Ce système propose pour cela une grande variété de processeurs logiciels permettant une gamme étendue d'opérations iconographiques. Ces processeurs peuvent bien sûr être activés en mode conversationnel, mais également par l'intermédiaire de fichiers de commandes. Une image peut alors être définie par le programme que constitue la liste des commandes qui ont servi à la réaliser (à partir d'autres images).

Il s'agit donc d'un système souple et puissant mais présentant tout de même une lacune importante. Les accès aux programmes se faisant par éditeur, ceux-ci se révèlent très hermétique quand à leur contenu. Par exemple, les relations "génétiques" entre images ne sont pas exploitables simplement.

Le but de ce projet est de combiner la puissance iconographique d'un tel système à la richesse de manipulation permise par un SGBDR afin de définir un système autonome, souple et interactif de synthèse d'images et d'animations numériques.

Nous commencerons par préciser les caractéristiques et la philosophie d'utilisation d'un tel système en détaillant:

- les caractéristiques principales,
- l'usage des commandes de description d'images,
- la création de nouvelles commandes,
- la description des animations,
- les recherches d'informations.

Ensuite nous précisons le schéma retenu pour son organisation générale en décrivant:

- la Base des Images Physiques et les Processeurs Iconographiques,
- la Base des Images Logiques et le Système Relationnel Étendu,
- l'Interface avec l'Utilisateur et l'Interface de Connexion.

Nous évoquons pour finir deux voies d'extension des possibilités du système à savoir l'automatisation et le parallélisme.

## 2. CARACTERISTIQUES ET PHILOSOPHIE D'UTILISATION

### 2.1 Présentation générale.

Ce système doit avant tout, permettre de modéliser, créer, modifier et archiver des images numériques de définition et de qualité esthétique suffisantes et enfin de les reproduire sur divers supports de diffusion (film, vidéo...). Il doit ensuite permettre les mêmes opérations au niveau des séquences.

Les principaux domaines d'application intéressés sont la création artistique, la réalisation de dessins animés, la mise au point d'aides pédagogiques ainsi que la visualisation de résultats de calculs.

Le système offre bien sûr de nombreuses possibilités de création d'images numériques, équipements de numérisation, logiciels standards de définition bi- ou tridimensionnels ou logiciels spécifiques permettent de visualiser des ensembles de données brutes. En outre, il propose une gamme "suffisante" de primitives iconographiques permettant diverses transformations ou combinaisons de ces images.

Une de ses caractéristiques principales est de mettre les fonctionnalités évoquées ci-dessus à la portée d'utilisateurs non-informaticiens. Pour cela, le système permet une description et une manipulation aisées des images sous une forme virtuelle, en particulier indépendante des périphériques.

De plus, ces descriptions virtuelles s'avèrent intéressantes par leur compacité qui donne donc la possibilité de gérer un grand nombre d'images.

Par contre, une image ainsi définie n'est pas visualisable instantanément, sa reproduction pouvant en effet nécessiter un temps relativement long.

### 2.2 Usage des commandes de description d'images.

Le contexte est le suivant : à un instant donné l'utilisateur dispose d'un ensemble d'images et veut, soit modifier certaines d'entre elles, soit les utiliser afin d'en créer des nouvelles.

Il utilise pour cela des commandes dont chacune détermine une action iconographique entreprise sur quelques images dites arguments et se conclue par l'obtention d'une image résultat. Il est bien sûr nécessaire de préciser un certain nombre de paramètres spécifiques au processeur employé.

A chaque processeur sont de plus associées des valeurs et images "par défaut" déterminées pour donner un résultat "généralement satisfaisant".

L'exécution d'une commande est prise en charge par le système dès sa soumission par l'utilisateur et le résultat est affiché sur un moniteur. L'image obtenue, et donc la commande émise, peuvent par conséquent être jugées directement et comparées visuellement à des résultats antérieurs ou obtenus concurremment.

Si le résultat est satisfaisant on passe à la commande suivante, sinon, la commande peut être corrigée ou annulée. Enfin une suite de commandes peut être sauvegardée afin d'être reproduite ultérieurement.

Pour réaliser ces diverses fonctions le système enregistre les commandes émises par l'utilisateur et assure ainsi l'écriture et l'exécution simultanées du programme. Après sauvegarde une image est alors conservée dans le système sous la forme de son programme de génération dans lequel interviennent des images arguments et donc des références à d'autres programmes.

Ce programme peut de plus être optimisé, c'est à dire simplifié et réorganisé afin de réduire son encombrement et d'accélérer les reproductions.

Il est à remarquer que dans ce mode d'utilisation, un programme est conçu en déterminant son résultat et donc automatiquement validé par l'acceptation de ce dernier.

### 2.3 Définition de nouvelles commandes.

L'utilisation systématique des commandes de base s'avère lente de mise en oeuvre et donc fastidieuse pour l'utilisateur qui a dépassé le stade de l'apprentissage.

Aussi le système doit-il permettre de définir de nouvelles commandes plus longues, plus complexes mais d'utilisation plus agréable. Les applications sont nombreuses et voici quelques une des plus intéressantes:

- la définition d'un jeu de commandes standardisées;
- la définition par l'utilisateur d'un environnement de programmation;
- si une image est définie par une seule commande il est aisé de tester quelques paramètres;
- un commande complexe peut être appliquée à des images simplifiées pour évaluer rapidement le résultat et éventuellement la corriger avant de l'utiliser pour obtenir une image mieux définie.

La définition d'une nouvelle commande se fait d'après une description d'image selon le principe suivant.

Une image est complètement définie par le programme la décrivant, à savoir une liste de commandes, et les arguments associés, images et paramètres. Et si l'on change certains de ces arguments l'image résultat se trouve (en principe) modifiée.

Donc, si l'on désigne certains arguments pour indiquer qu'ils sont variables, alors le programme ne définit non plus une image particulière mais toute une classe d'images possibles dont l'une d'entre elles est parfaitement définie dès que l'on attribue des valeurs particulières aux variables. Les valeurs et images associées par défaut à ce nouveau processeur sont alors celles définissant l'image initiale.

### 2.4 Description des animations.

Une animation consiste en une liste d'images distinctes et un ordonnancement temporel de celles-ci et le tout est décrit dans un scénario décomposé en une suite de séquences élémentaires.

Les images numériques sont donc référencées dans ces séquences et plusieurs séquences peuvent, bien sûr, "partager" une même image. Ceci permet en particulier de définir plusieurs scénarios d'une même animation et de les comparer.

Nous retiendrons essentiellement deux types bien distincts de séquences.

La forme la plus rudimentaire est l'énumération explicite d'images auxquelles sont associées leurs durées d'apparition. Une telle séquence doit, bien entendu, être facilement éditable image par image.

D'autre part, des séquences telles les interpolations, les "fondus enchaînés" et autres transitions "continues" sont définies sans expliciter toutes les images qui les composent (dans la pratique on utilise en effet quelques images "clés").

Un exemple intéressant est la définition d'une séquence par une commande où n'interviennent que des variables paramètres. Une image est alors définie par une combinaison particulière de paramètres et la séquence par un chemin dans l'espace des paramètres possibles. Ce chemin est lui-même paramétré par le temps et défini comme passant par les points particuliers associés aux images clés.

Il faut également envisager des opérations sur les séquences qui permettent d'en modifier certaines ou d'en créer de nouvelles.

Un premier cas est celui des modifications de rythme comme les accélérés, ralentis ou retours arrière.

Un second cas consiste en la transformation d'une séquence en définissant le résultat comme la séquence constituée des images obtenues par application d'une même commande à chaque image. On peut de manière similaire définir la combinaison de deux séquences ou d'une séquence et d'une image fixe.

### 2.5 Recherches d'Informations.

Le système doit aussi permettre de retrouver les informations qu'il contient, d'y accéder et de les utiliser. Pour cela, il faut d'abord effectuer des recherches très variées et ce dans un contexte très interactif. Un moyen simple est de formuler les questions en donnant une réponse type sur laquelle des critères de sélection peuvent être ajoutés, modifiés ou supprimés. De plus, n'oublions pas que, les informations étant représentatives d'images (directement ou par associations), celles-ci peuvent être visualisées et faire alors l'objet d'une sélection "manuelle".

Dans un premier cas les critères de recherche portent directement sur des informations relatives aux descriptions des images et processeurs archivés dans le système.

Si de plus on associe aux images des informations ne concernant pas directement leur synthèse mais précisant par exemple leur contenu, alors on étend considérablement la gamme des critères de recherche permettant de constituer une diathèque dans laquelle les images sont accessibles par associations avec des clés.

## 3. ORGANISATION DU SYSTEME

### 3.1 Organisation générale, voir figure ci-dessous.

De l'analyse précédente il résulte que le système doit en fait assurer deux tâches aux caractéristiques très différentes, chacune faisant l'objet de solutions spécifiques. Il est donc intéressant, aussi bien au niveau de la conception qu'à celui de la réalisation, de décomposer ce système en deux parties indépendantes et chargées respectivement de l'une et l'autre de ces tâches.

La première consiste en la gestion des représentations de bas niveaux. Cette tâche est assurée par une "machine" dite iconographique qui utilise des éléments logiciels et (ou) matériels déjà développés et couramment utilisés dans ce domaine [FoDa 82]. Toutefois, nous avons ajouté aux solutions traditionnelles quelques adaptations qui permettent d'assurer un fonctionnement correspondant aux règles énoncées précédemment.

La seconde tâche consiste en la gestion des définitions virtuelles d'images, processeurs et séquences et de leurs associations à d'autres données.

Pour gérer ces données nous avons choisi une Base de Données Relationnelle, [Codd 81, Date 82], qui après quelques extensions s'avère un outil parfaitement adapté à cette tâche.

Le Système de Gestion de cette base supporte deux programmes d'applications dont le plus complexe réalise l'interface avec l'utilisateur, c'est à dire la gestion du dialogue, la présentation des données et la normalisation des requêtes, tandis que le second, indispensable mais plus rudimentaire, assure l'interprétation des descriptions virtuelles afin de piloter la machine iconographique.

La décomposition que nous venons d'évoquer peut très bien s'exprimer au niveau matériel, c'est à dire que le système est alors constitué de deux machines spécialisées et physiquement distinctes, mais elle peut aussi se traduire par deux ensembles de logiciels spécifiques implémentés sur une même machine d'usage général. Dans tous les cas l'interface de communication prévue entre ces deux composantes demeure très simple et suffisamment universelle pour permettre des évolutions séparées tout en assurant la compatibilité.



### 3.2 Base des Images Physiques et Processeurs Iconographiques.

La machine iconographique est constituée des deux parties assurant respectivement l'archivage et le traitement des représentations numériques de bas niveaux.

La Base des Images Physiques assure le stockage sur support externe d'un certain nombre d'images aussi bien permanentes (les images dites de base) que temporaires (les images de travail en cours de calcul ou de visualisation) et de plus gère leur accès.

Les divers Processeurs Iconographiques permettent de réaliser sur ces représentations trois types d'opérations, l'introduction ou la modification d'images de base, les opérations de synthèse qui sont commandées par l'autre composante, ou enfin la production des images obtenues sur les divers périphériques dont, en particulier, le moniteur de travail.

La structure de la BIP et les méthodes d'accès associées sont conçues pour préserver son intégrité et sa cohérence en dépit des aléas de fonctionnement du système.

De plus la gestion des accès à la BIP assure la cohérence en cas d'opérations réalisées concurremment par plusieurs PI.

Ces problèmes sont résolus par le respect des règles données ci-dessous.

Les entités élémentaires de la BIP ne sont pas les représentations d'images mais différentes versions de celles-ci. Pour une image donnée on conserve en effet plusieurs versions successives, deux consécutives partageant leur partie commune.

Les diverses versions existantes sont toutes accessibles en lecture et ce à tout instant. Pour une image donnée, on accède bien sûr en priorité à la plus récente mais on peut également accéder aux autres par ordre d'ancienneté croissante.

Seule la version la plus récente peut faire l'objet d'une nouvelle mise à jour qui s'effectue alors dans le cadre d'une transaction. Une transaction débute par la pose sur l'image d'un verrou d'écriture ayant pour rôle d'interdire toute mise à jour concurrente sur celle-ci. Le corps de la transaction consiste alors en une suite d'opérations assurée par divers processeurs et se termine par une validation de la nouvelle version qui est, alors seulement, accessible en lecture, et par la levée du verrou qui autorisent une nouvelle transaction.

Les versions peuvent être supprimées dans l'ordre, à partir soit de la plus récente soit de la plus ancienne, à condition qu'il n'y ait ni lecture ni mise à jour en cours dessus.

Un verrou d'exclusion au niveau de l'image permet de séquentialiser les phases critiques de début et de fin de ces trois types d'accès lecture, écriture et suppression.

Cette solution présente deux intérêts particuliers.

Les accès aux images, pour mise à jour ou lecture, nécessite un temps relativement long et peuvent donc être difficilement séquentialisés sans dégrader les performances du système. Ainsi par exemple une image de travail doit être accédée en permanence en lecture afin de l'afficher mais doit néanmoins pouvoir être modifiée fréquemment.

Le fait de conserver plusieurs versions améliore également les performances lorsqu'un certain nombre de transactions sont annulées (ce qui se produit fréquemment en contexte interactif), il est alors possible de retrouver une ancienne version sans avoir à effectuer à nouveau son calcul à partir de sa description logique.

### 3.3 Base des Images Logiques et Système Relationnel Étendu.

La Base des Images Logiques contient en fait deux types d'entités logiques:

- les opérations élémentaires qui associent un processeur, des images arguments (elles mêmes résultats d'autres opérations) et des paramètres, il s'agit en fait d'entités complexes faisant appel à des structures relativement évoluées comme des listes d'images ou des enregistrements de paramètres à structures variables;
- les informations qui ne concernent pas directement la synthèse, mais dont nous avons vu la diversité et l'importance.

Bien sûr un certain nombre de relations logiques existent entre ces différentes entités et les définitions d'images, de processeurs et de séquences.

Les accès à la BIL sont gérés par le SGBDR qui permet des recherches selon divers critères, des consultations et des mises à jour; nous avons retenu un système relationnel notamment pour ses principales qualités fondamentales qui sont le dynamisme et l'indépendance des données, le maintien aisé et sûr de la cohérence, la simplicité de présentation et la souplesse d'accès. La plupart des informations s'expriment en effet naturellement selon la formulation relationnelle c'est à dire sous forme de tables dont chaque ligne représente une certaine relation entre les quelques données qui la composent.

Cependant, dans notre cas, du fait de cette conception des données "à plat" et volontairement simplifiée, le cadre relationnel s'avère trop limité. Quelques extensions sont alors nécessaires pour représenter et gérer premièrement les entités déjà complexes que sont les opérations élémentaires, mais également les diverses définitions d'images, de processeurs et de séquences qu'elles composent et enfin les dépendances hiérarchiques qu'elles induisent.

Ainsi, par exemple l'insertion d'une opération nécessite plusieurs mise à jours élémentaires, il faut de plus respecter la cohérence globale c'est à dire vérifier que toutes les images et le processeur référencés sont bien définis. Par contre, pour en supprimer une, il convient de s'assurer que son résultat n'est pas utilisé comme argument d'autres opérations ou pour définir des images ou entités similaires.

Pour cela un certain nombre de requêtes récursives permettent de sélectionner les entités relationnelles associées à un élément donné par certaines règles de chaînage [Knut 73].

La notion d'attribut variable permet en outre la description "incomplète" d'entités définissant des objets paramétrables, des macro-processeurs ou des images génériques servant à décrire des animations.

Bien sûr se pose alors le problème de l'opportunité du choix d'un tel système.

Cependant les extensions envisagées s'inscrivent dans un cadre limité, celui du schéma de description des images qui constitue alors un squelette statique de la base autour duquel, au gré de l'application et de l'utilisateur, vont se greffer dynamiquement les relations porteuses de la plupart des informations et pour la gestion desquelles le schéma relationnel s'avère tout à fait adapté.

Enfin le problème du temps d'exécution des requêtes, qui est l'un des désavantages que l'on reconnaît aux systèmes relationnels, et qui dans cette application s'avère encore plus aigu du fait des extensions, n'apparaît toutefois pas comme un facteur critique, les requêtes du niveau logique étant associées à des calculs physiques qui s'avèrent relativement longs.

SUR L'UTILISATION D'UN SGBD RELATIONNEL POUR LA GESTION D'UNE BASE DE DONNEES IMAGE  
ON THE USE OF A RELATIONNAL DBMS FOR THE MANAGEMENT OF AN IMAGE DATA BASE

- Philippe CHASSIGNET

### 3.4 Interface avec l'utilisateur.

Cette application assure tout d'abord une interface traditionnelle avec le SGBDR pour les consultations et mises à jour des données ne concernant pas directement la synthèse et s'intégrant normalement dans le cadre relationnel.

Elle permet donc d'assurer les recherches d'informations répondant aux diverses requêtes possibles, celles-ci sont formulées par la donnée d'un exemple de réponse et peuvent ensuite être corrigées ou complétées [Zloo 75].

De plus, l'interface gère implicitement une image supplémentaire temporaire qui représente la composition de l'écran et est diffusée et rafraîchie en permanence. Cette image est composée par juxtapositions et superpositions d'images existantes ou appartenant à des séquences. Il peut s'agir d'images de travail en cours d'élaboration, d'images résultats d'une requête ou encore de menus.

L'interactivité est alors assurée par la combinaison des commandes classiques et par l'usage d'un moniteur de visualisation combiné à un dispositif d'entrée graphique qui permettent de déplacer des curseurs de désignation sur cet écran et donc d'effectuer toutes sortes de sélections "manuelles" [ChFu 80].

Mais la tâche prépondérante de cette interface est de saisir, d'enregistrer et de vérifier les opérations de synthèse introduites par l'utilisateur. Elle assure donc pour commencer la mise en application du principe de valeurs par défaut et complète donc automatiquement une commande initialisée. Ensuite, après que l'utilisateur ait apporté les précisions désirées, elle génère à partir de cette commande l'ensemble des mises à jour nécessaires pour la représenter dans la BIL. Il reste alors à activer l'interface de connexion pour calculer et visualiser l'image qui en résulte.

Elle permet ensuite d'optimiser la description obtenue en fonction des règles de simplification physique, et la réalisation de cette optimisation implique la réécriture de tout un ensemble d'opérations.

L'interactivité est mise largement à contribution pour résoudre des problèmes dont la réponse fait intervenir des critères subjectifs qui nécessitent donc de confier la décision à l'utilisateur.

Par exemple, lors de la modification d'une image il faut en effet décider de propager ou non cette modification à toutes les images la référençant.

Et donc, pour chaque image concernée, choisir l'une ou l'autre des deux versions proposées et ce récursivement lorsque c'est la version modifiée qui est choisie.

### 3.5 Interface de connexion.

L'interface de connexion doit essentiellement assurer la cohérence entre les deux systèmes, c'est à dire faire en sorte que les descriptions numériques contenues dans la BIP correspondent aux définitions dans la BIL d'un certain nombre d'images désignées par l'utilisateur. En effet si chaque composante est à même d'assurer sa propre cohérence interne il est nécessaire d'assurer la cohérence de l'ensemble c'est à dire le maintient "en phase" des deux composantes et donc de répercuter sur la BIP toutes modifications intervenant dans la BIL sur les définitions de ces images.

Il faut donc assurer l'accès à la version courante d'une image si elle existe déjà physiquement ou, dans le cas contraire, produire à partir des descriptions logiques l'ordonnancement des opérations élémentaires permettant de réaliser son calcul.

Elle doit en même temps effectuer la mise à jour d'une représentation d'image si sa définition logique est modifiée par l'utilisateur.

L'interface assure de plus une gestion macroscopique de la BIP en tenant compte de l'espace disponible. Rappelons à cet effet que plusieurs transactions (associées à plusieurs images de travail) peuvent avoir lieu simultanément. Cet espace est géré simplement selon une politique d'allocation LRU qui permet de garder le maximum de versions d'images temporaires tout en évitant la saturation.

Notons que les avortements de transactions sont à proscrire à cause de leur coût excessif, aussi est-il nécessaire de les éviter en récupérant et réservant un espace suffisant avant de commencer chaque transaction et en adoptant un ordonnancement qui minimise la place requise.

Enfin il faut se souvenir que cette connexion entre la BIL et la machine iconographique constitue un couplage faible que l'on réalise par l'échange de messages brefs chacun correspondant à l'exécution d'une opération de base. De ce fait, la cadence de production d'images déjà définies est limitée par la rapidité de la machine iconographique. La liaison assurant ce couplage n'est donc pas critique et peut utiliser des fichiers dans le cas d'une solution logicielle ou une ligne type console dans le cas de la connexion de deux machines.

## 4. PERSPECTIVES

### 4.1 Automatisation.

Un moyen d'accroître les possibilités d'automatisation d'un tel système est d'attribuer à chaque image réalisée une note ou une appréciation sur divers aspects esthétiques ou techniques. Ce qui revient alors indirectement à noter l'usage des commandes et les choix des paramètres. Le système est alors à même de proposer une "base d'expériences" (aussi bien positives que négatives) pouvant donc mémoriser un certain savoir-faire en synthèse d'image numérique, aisément exploitable, s'enrichissant automatiquement au fil de l'utilisation et très utile à des fins pédagogiques.

Si de plus on a la possibilité d'effectuer certaines inférences basées par exemple sur la nature des commandes, il est alors possible d'effectuer quelques manipulations sémantiques autour du squelette que constitue le programme. On peut même envisager qu'un système expert utilise cette base de connaissances, concernant aussi bien la synthèse des images que leur contenu, et permette alors une automatisation poussée de la tâche de description des images et scénarios.

On pourrait entre autres étendre la notion de valeur par défaut à celle de valeur la plus appropriée d'après le contexte et les expériences précédentes.

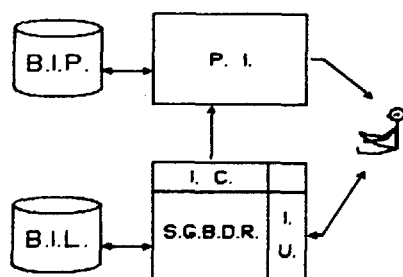


Schéma d'organisation du système.



#### 4.2 Parallélisme.

Rappelons que les définitions d'images sont rangées dans la BIL sous la forme des ensembles d'opérations permettant de les calculer. Ces ensembles sont naturellement structurés en graphes de calcul dont les noeuds sont les opérations et dont les arcs figurent les transmissions d'images arguments. Il s'agit bien là d'une structure pouvant contrôler une synthèse avec synchronisation sur les arguments, en l'occurrence les images intermédiaires; d'où la notion d'image Flow [SyCH 77, McGr 80].

Remarquons alors que la structure et la politique d'accès originales de la BIP permettent une implémentation aisée d'un système fonctionnant selon ce principe, un ensemble de machines graphiques pouvant se partager la même BIP. Cet ensemble peut alors être un ensemble de processeurs matériels ou logiciels soit spécialisés dans les différentes opérations de synthèse, soit d'aptitudes plus étendues.

#### 5. CONCLUSION

Ce papier présentait les principaux aspects d'un système de base de donnée images défini autour d'une machine iconographique et d'un SGBD relationnel.

La répartition des tâches entre ces deux composantes lui confère une structure très simple et largement ouverte vers des évolutions spécifiques dans les deux domaines concernés.

Une extension du SGBD selon les critères énoncés et les différentes interfaces sont actuellement en cours d'élaboration.

Les deux composantes utilisées sont respectivement le système MSC pour la machine iconographique et le SGBDR PEPIN [BCFJS 83].

#### 6. REMERCIEMENTS

Je tiens à remercier le Pr. Erol GELENBE, qui me suggéra ce projet et en guida la réalisation.

Je remercie également Jean Francois COLONNA pour ses commentaires fructueux.

#### 7. REFERENCES

- [BCFJS 83] P. BOUCHET, A. CHESNAIS, J.M. FEUVRE, G. JOMIER, S. SZULMAN :  
"Le SGBD Relationnel pour microordinateurs PEPIN".  
Convention Informatique, juin 1983.
- [ChFu 79] N.S. CHANG, K.S. FU :  
"A Relational Database System for Images".  
Lecture Notes in Computer Science, vol. 80, Pictorial Information Systems, 1979.
- [ChFu 80] N.S. CHANG, K.S. FU :  
"Query By Pictorial Exemple".  
IEEE Trans. on Soft. Eng., vol. SE-6, N° 11, november 1980.
- [Codd 81] E.F. CODD :  
"The capabilities of relational database management systems".  
RJ 3132 Research Report 5/11/81.  
IBM Res. Lab. San Jose (USA).
- [Colo 84] J.F. COLONNA :  
"De la Visualisation de Résultats de Calculs à la création artistique".  
Premier Colloque Image. Biarritz, mai 1984.
- [Date 82] C.J. DATE :  
"An introduction to database systems".  
Third edition, Addison Wesley 1982.
- [FoDa 82] J.D. FOLEY, A. VAN DAM.  
"Fundamentals of Interactive Computer Graphics".  
Addison Wesley 1982.
- [Knut 73] D.E. KNUTH :  
"Fundamental Algorithms".  
Second edition, Addison Wesley 1973.
- [McGr 80] J.R. MCGRAW :  
"Data flow computing software development".  
IEEE Trans. on Computers, vol. C-29, N° 12, december 1980.
- [SmAl 79] B. SMEDLEY, B. ALDRED :  
"Problems with geo-data".  
Lecture Notes In Computer Science, vol. 81, Data Base Techniques for Pictorial Applications, 1979.
- [SyCH 77] J.C. SYRE, D. COMTE, N. HIFDI :  
"Pipelining, parallelism and asynchronism in the LAU system".  
Int. Conf. on Parallel Processing, august 1977, pp. 87-92.
- [VSHM 82] P.D. VAIDYA, L.G. SHAPIRO, R.M. HARALICK, G.J. MINDEN :  
"Design and architectural implications of a Spatial Information System".  
IEEE Trans. on Computers, vol. C-31, N° 10, october 1982.
- [YaKu 82] K. YAMAGUCHI, T.L. KUNII :  
"PICCOLO logic for a picture database computer and its implementation".  
IEEE Trans. on Computers, vol. C-31, N° 10, october 1982.
- [Zloo 75] M.M. ZLOOF :  
"Query By Exemple".  
International Conference on Very Large Data Bases, ACM 1975.