

**François GUENA**  
**Jean-Pierre LEININGER**  
**Khaldoun ZREIK**

CIMA - Paris

---

**Problématique  
de l'intégration  
des concepts de l'intelligence  
artificielle dans un système  
de CAO en architecture**

## ABSTRACT :

-----

This paper relates the problems encountered while developing and integrating modules for technical and cost evaluation into the CAAD system "X2A".

Two major components were used to allow the system to evolve : the modularity of its architecture and its construction around a data base management system working simultaneously with a set of utilities dealing with the geometrical model.

When the designer modifies his project, in turn the DBMS modifies all the existing links on the objects, independently of their nature, thus allowing to enhance the conceptual model.

Allowing development of new applications, or tailoring such a system to a specific design process, implies adding new types of objects to the basic model that handles the data describing the building; this in turn substantially increases the set of utilities needed to control the behaviour rules of objects, and the coherence and consistency of the data stored in the DB.

Such evolution affects also the interaction level (i.e. controlling the process of manipulation and positioning objects).

A.I. techniques seem to be promising for the evolution of such systems. Declarative knowledge are used at three levels:

- in the DBMS in order to define the behaviour and generation rules of the objects.
- in a command language acting upon the logic of the system (i.e. upon the controller), and allowing to define rules which govern the activation of a process or utility, and control their parameters.
- in the technical evaluation modules, to allow specifying the conditions for the activation of an algorithm or object filtering, and also to allow the user to define new rule driven applications.

Many papers deal, upstream, with object modelling techniques and integration of expert systems with DBMS systems; downstream, many others investigate the use of expert systems in the handling of new applications. To study possible modifications of the architecture of the CAD systems seem to open promising research directions.

## RESUME :

-----

L'article qui suit expose les problèmes rencontrés au cours des développements d'un système de CAO en architecture "X2A", pour y intégrer des modules d'évaluation technico-économique.

La démarche utilisée pour assurer l'évolutivité du système se fonde sur deux caractéristiques: La modularité de son architecture et l'existence d'un système de gestion de base de données associé à un ensemble d'utilitaires gérant un modèle géométrique de départ.

Lors des modifications du projet effectuées par le concepteur, le SGBD répercute l'ensemble des liens existants sur les objets sans préjuger de leur nature, autorisant ainsi l'enrichissement du schéma conceptuel.

L'adaptation du système à de nouvelles applications ou à d'autres logiques de conception, entraîne l'adjonction au modèle de départ décrivant le bâtiment, la création d'autres types d'objets. Il implique cependant un accroissement substantiel du nombre des utilitaires gérant les lois de comportement des objets et contrôlant la cohérence et la validité des informations stockées dans la base de données. Les répercussions se situeront aussi au niveau des interactions, c'est à dire du contrôle des processus de positionnement et de manipulation des objets du concepteur.

Les techniques de l'intelligence artificielle représente une voie d'évolution de tels systèmes.

Les connaissances déclaratives interviennent à trois niveaux:

- Au niveau de la base de données pour définir les règles de comportement et de génération des objets.

- Au niveau d'un langage de commande agissant sur la logique de fonctionnement du système (action sur le contrôleur) et permettant de définir les règles d'activation des différents processus ou utilitaires, ainsi que les contrôles effectués sur leurs paramètres .

- Au niveau des modules d'évaluations techniques, permettant de préciser les conditions d'activation d'un algorithme, de filtrer les objets sur lesquels il travaille, et enfin d'offrir à l'utilisateur la possibilité de définir de nouvelles applications gérées déclarativement.

Si de nombreux travaux portent en amont, sur la modélisation d'objets et l'intégration de techniques relevant de l'intelligence artificielle aux bases de données; en aval, sur la gestion d'applications à l'aide de systèmes experts; l'étude des modifications à apporter à l'architecture proprement dite des systèmes de CAO constitue une voie de recherche très prometteuse.

## I. INTRODUCTION.

---

Le système "PROJET" (ref. 1), construit autour d'une base de données de type réseau, outil de conception assistée par ordinateur (C.A.O.) en avant-projet sommaire de bâtiment, a constitué l'esquisse du système "X2A" présenté ici.

Les hypothèses de travail qui ont conduit cette première recherche sont fondées sur une série de spécifications d'un tel outil se formulant en termes d'exigences .

Un système de CAO doit pouvoir :

1. Prendre le relais de tout un ensemble de tâches de manipulation de l'information.
2. Intégrer des programmes spécialisés sans reconstruire le système.
3. S'adapter aux diverses manières de travailler du concepteur.
4. Repérer, sinon proposer, des éléments de solutions susceptibles d'amélioration.
5. Permettre à l'utilisateur de choisir son propre mode de travail pour un projet donné.
6. Apporter au concepteur un ensemble d'évaluations ou de constats relatifs à la cohérence de son projet.

La première maquette du système, restreinte à l'approche architecturale de la conception, nous avait permis d'étudier les trois premiers points ainsi que de définir une modélisation générale du bâtiment; restait à confronter nos hypothèses de travail au sein d'une équipe pluridisciplinaire et d'aborder les autres points.

Le projet "X2A", regroupant quatre équipes (CIMA, Mines St-Etienne, INSA Lyon, Université Savoie) et les prolongements que nous lui avons donnés (introduction de l'approche systèmes experts au niveau des applications) a constitué l'occasion recherchée.

Cependant, la démarche employée pour assurer l'évolutivité du système et son adaptation à des modèles particuliers de données (une architecture modulaire organisée autour d'un système de gestion de base de données), conduit à multiplier le nombre des utilitaires associés au SGBD permettant de gérer la cohérence et le comportement des objets .

Les limitations de cette gestion procédurale d'un système de CAO, nous amène à en reconsidérer l'architecture en introduisant des mécanismes experts au sein du SGBD et du contrôleur du système.

Cette solution semble intéressante, et vient compléter les travaux de recherche existants, centrés en amont sur l'intégration de techniques relevant de l'intelligence artificielle au niveau des bases de données, et en aval, sur l'emploi de systèmes experts au niveau des programmes d'applications.

"X2A" est donc un prototype de CAO intégrant des modules d'évaluations techniques et économiques (calculs de structure, thermiques, acoustiques : INSA, LGCH Chambéry) et de visualisation du projet (images numériques, ENMSE).

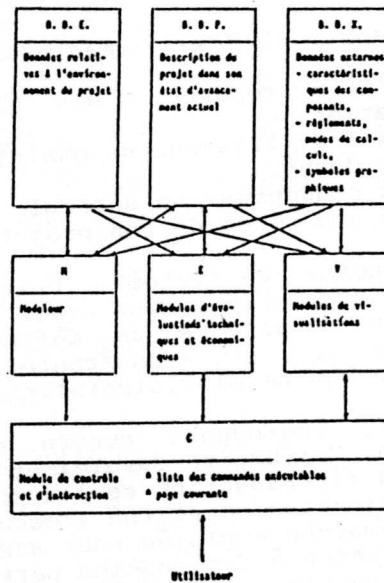
Pour cette étude le CIMA a :

- . conçu et administré la base de données
- . développé des programmes permettant à l'utilisateur de modéliser son projet
- . réalisé les interfaces nécessaires aux modules cités plus haut.

II. ARCHITECTURE DU SYSTEME.

Le schéma suivant donne l'architecture du système.

E-2-A : Schéma statique des composants logiciels



Les flèches indiquent les flux de données

La Base de Données du Projet en cours de conception (B.D.P) contient une description du projet dans son état d'avancement actuel. Elle gère la modélisation du projet sous ses aspects géométriques, structuraux, économiques et fonctionnels. C'est une base dynamique, autorisant des mises à jour de l'information du point de vue de son contenu et de sa forme, par une évolution du schéma conceptuel et des occurrences d'objets mémorisés. Cette approche est induite par les reformulations itératives du projet effectuées par le concepteur et par l'action des modules d'application sur la base de données.

La Base de Données Environnement (B.D.E) contient les informations relatives au site : connaissance du sous-sol, environnement climatique, acoustique, position et masse des bâtiments voisins.

La Base de Données Externes (B.D.X) regroupe des informations stables, correspondant à la description et représentation des ouvrages (bibliothèque de symboles et composants) ou objets intervenant dans un type de projet donné.

B.D.X. et B.D.E. sont utilisées en consultation par les modules de calculs au cours d'une session de travail.

La composante de modélisation ou modeleur (M) structure les informations provenant de la composante de contrôle et d'interaction (C). Elle active les mécanismes de gestion du modèle (règles de comportement et génération d'objets) stockés dans le schéma conceptuel du système et renvoie à l'utilisateur l'écho des modifications effectuées. Elle est constituée d'un ensemble d'outils (éditeurs 2D ou 3D) pouvant, lors des premières phases de saisie travailler dans un mode "off line" du système global.

(E) est le module d'évaluation technique et économique; (V) le module de visualisation.

La page courante est en particulier une image graphique, textuelle et numérique des parties de la B.D.P, B.D.E et B.D.X qui sont intéressantes à un stade donne d'une session d'utilisation de "X2A".

La composante de contrôle et d'interaction (C) joue un rôle essentiel dans la mesure où elle est l'unique interface avec l'utilisateur. Elle gère le vecteur d'état du système et l'ensemble des commandes activables à un instant donné correspondant au niveau de description du projet.

### III. LA BASE DE DONNEES, CONCEPTS ET PRINCIPES DE FONCTIONNEMENT.

---

La B.D.P. est construite sur un SGBD de type réseau. utilisant les concepts suivants :

- . Classe d'objets :
  - Est identifiée par un nom.
  - Possède des attributs.
  - Possède des procédures permettant, au moment de la génération d'un objet de la classe, de créer automatiquement des "sous-objets" induits par l'objet "père".
- . Objet :
  - Possède un nom.
  - Possède des attributs.
- . Attribut :
  - Est identifié par un nom.
  - Est de type : variable ou tableau, réel, entier, caractère.
  - Peut être implicite (procédural, ex: surface d'une pièce), interne (la valeur est effectivement stockée dans la B.D.P.), externe (sa valeur est stockée dans B.D.X.).
- . Relation (N-M) :
  - Est identifiée par un nom.
  - Est caractérisée par le nom des types d'objets "pères", types d'objets "fils".
  - Peut être explicite (elle correspond à des liens existant dans la base), ou implicite (procédures déduisant des liens à partir des liens stockés).
  - Peut posséder des attributs.

Le schéma de la base mémorise l'ensemble des caractéristiques des types d'éléments manipulés à un instant donné. Il est géré à l'aide d'un langage de description des données et est modifiable dynamiquement. Les occurrences de classes d'objets, objets, relations sont gérées par un langage de manipulation des données. Ces deux langages sont constitués de primitives appelables à partir des programmes d'application (La B.D.P. et la B.D.E. sont gérées à l'aide du SGBD décrit ici. Pour plus de détail et pour la description de la B.D.X., voir ref. 2).

### IV. MODELISATION DU BATIMENT.

---

Le modèle employé pour décrire un projet utilise les notions d'objets et de relations. Il constitue un schéma minimum nécessairement complété par les sous-schémas propres aux applications. Lors des modifications effectuées par le concepteur sur le projet, le SGBD et ses utilitaires répercutent l'ensemble des liens existants sur les objets du modèle, sans préjuger de leurs natures, permettant ainsi de compléter le schéma conceptuel de départ. La géométrie et la topologie d'un bâtiment sont définies par une structure "fil de fer" tridimensionnelle à base de facettes. Elle est "habillée" par un ensemble de composants ou ouvrages.

On peut définir schématiquement ce squelette à l'aide des type d'objets suivants :

- . PROJET
- . BATIMENT : les bâtiments constituant le projet
- . ETAGE, SUR-ESPACE : la notion de "sur-espace" correspond au regroupement fonctionnel ou spatial de "pièces" ou de "sur-espace" ( appartement, ensemble d'étages identiques du point de vue structure ...)
- . UVE, UVI : unités volumiques internes externes correspondant à un contour polygonal fermé (notion de pièce).
- . GROUPE-FACETTE : contour polygonal ouvert.
- . FACETTE-LATERALE, FACETTE-PLAFOND
- . PLANS AXIAUX : trame spatiale définie sur le bâtiment.

Définissons sommairement les relations reliant ces objets (pour plus de détail ref. 2).

. PROJET	"FORME DE"	BATIMENT
. BATIMENT	"FORME DE"	PLAN-AXIAUX
	"FORME DE"	ETAGE
	"FORME DE"	SUR-ESPACE
. ETAGE	"SUR"	ETAGE
	"FORME DE"	SUR-ESPACE
	"FORME DE"	UVE / UVI
. UVI	"SUR"	UVI
	"FORME DE"	FACETTE-LATERALE
	"CONTIENT"	UVI / GROUPE-FACETTE
. GROUPE-FACLAT	"FORME DE"	FACETTE-LATERALE
. FAC-LAT	"LIEE A"	COMPOSANT
	"SUR"	PLAN-AXIAL

Un composant peut être un objet ou ouvrage plus ou moins défini ou un symbole graphique qui sémantise une relation entre UVI ou UVE.



Il est décrit à l'aide d'attributs permettant de mémoriser des caractéristiques numériques, alpha-numériques (texte, caractéristiques techniques), mais aussi géométriques ou graphiques (ex: points d'attaches, enveloppe, espace servant/servi ...) et alpha-graphiques (le/les symboles attachés). Ces attributs sont soit statiques (stockés dans la B.D.X.) soit dynamiques (paramètres définissant la position de l'occurrence, attributs manipulés par les modules de calcul... stockés dans la B.D.P.).

#### V. LES PROGRAMMES D'APPLICATIONS, EXEMPLE DU CALCUL DE STRUCTURE.

---

Une fois la volumétrie "filaire" du bâtiment créée dans la base soit à partir du modèleur tri-dimensionnel ( saisie de la volumétrie générale du bâtiment et création des facettes des UVE/UVI) soit en deux-dimensions et demi (étage par étage, les facettes latérales sont alors perpendiculaires au plancher), l'utilisateur peut habiller cette géométrie "fil de fer" à l'aide des composants répertoriés dans la bibliothèque gérée par le système (B.D.X.).

Il s'agit donc maintenant de préciser trois points essentiels:

1. Les objets du modèle filaire manipulés par les programmes d'application.
2. Le lien entre les composants implantés par l'utilisateur et les objets propres au modèle de calcul.
3. La liste des objets et relations propres au modèle de calcul (sous-schema) pour chaque application.

La solution retenue consiste à définir pour chaque application un sous-schema incluant le modèle filaire et complété par des objets spécifiques.

Deux scénari d'utilisation sont ensuite retenus: soit la manipulation directe des objets propres à l'application par l'utilisateur connaissant le modèle de calcul, soit la génération automatique des objets du modèle à partir de composants généraux et des règles attachées aux classes d'objets dans le schéma de la base.

#### Exemple du calcul de structure.

---

le concepteur manipule les composants suivants :

- MUR-PORTEUR, POTEAU, POUTRE, CLOISON, objets de type PERCEMENT

Toute demande de calcul de descente de charge provoque alors une vérification de cohérence verticale ainsi que l'activation des procédures de génération des objets du gros oeuvre et de leurs liens respectifs :

- DALLE, PORTION-DALLE, APPUI, VOILE, TRAVEE-POUTRE, EXTREMITÉ-POUTRE

La description du fonctionnement du système soulève alors un certain nombre de questions, notamment par rapport aux modifications ultérieures du projet par l'utilisateur :

- Quand activer les vérifications de cohérence?
- Peut-on faire intervenir des niveaux de cohérence, qui vont impliquer une certaine hiérarchie au niveau des objets?
- Quelle stratégie utiliser lors de la détection d'une incohérence? Dans quelle mesure peut-on la corriger automatiquement, ou doit-on plutôt poser une question à l'utilisateur?

La solution retenue est de mémoriser au niveau de la base, l'ensemble des UVE ou UVI dont les objets ont été modifiés, si ces mises à jour impliquent des recalculs, par la création d'un objet spécifique à chaque application. Les mécanismes de vérification de cohérence sont activés ensuite localement à chaque commande faisant appel au schéma spécifique d'une application.

La mise en oeuvre d'une telle solution implique donc la multiplication des procédures gérant les règles de comportement des objets et la cohérence des informations décrivant le bâtiment par rapport aux modèles de calculs.

Les techniques de l'intelligence artificielle, et notamment des systèmes experts, semblent être une solution aux problèmes évoqués, permettant d'extraire du système la connaissance nécessaire à la gestion des informations décrivant un projet.

## VI. LES SYSTEMES EXPERTS COMME METHODE D'EVOLUTION DU SYSTEME.

---

Il est possible de situer l'intervention de connaissances déclaratives à trois niveaux dans le système:

- Au niveau du schéma de la base de données, pour définir les règles de comportement et de génération des objets, et les règles de vérification de cohérence.
- Au niveau d'un langage de commande permettant d'agir sur le contrôleur et donc sur les interactions et les commandes.

- Au niveau des modules d'évaluation technique, permettant de préciser les conditions d'activation d'un algorithme, mais aussi d'offrir à l'utilisateur la possibilité de définir de nouvelles applications gérées déclarativement.

### 1. Les mécanismes de la génération d'objets.

---

#### A. Génération "père --> fils".

---

Reprenons l'exemple du calcul de structure :

Lorsque l'utilisateur implante un composant de type PERCEMENT (fenêtre, porte ...) sur un mur porteur, le système génère automatiquement une poutre, si la largeur de l'ouverture est supérieure à une portée critique (règle "plein pour vide", permettant de négliger en première approximation les petites ouvertures, ref. 3).

Enfin, si le composant implanté est du type FENETRE, le système va générer une allège correspondant à la hauteur d'implantation de la fenêtre.

La description de ces règles au niveau du schéma sous forme déclarative va permettre à l'utilisateur d'intervenir au niveau des mécanismes internes du système.

Ces règles seront alors employées indifféremment par les mécanismes de génération d'objets et de vérification de cohérences.

#### B. La génération "fils --> père".

---

Les objets poutres manipulés par l'utilisateur sont gérés par le système comme des objets à "géométrie dépendante", autrement dit leurs caractéristiques s'obtiennent à partir du contexte dans lequel l'objet est situé. Plus précisément, si une poutre est supportée par une facette "fil de fer", sa longueur est déterminée par la longueur de la facette; et si cette poutre est appuyée sur deux autres poutres le calcul va s'effectuer à partir des points d'attache de la poutre.

Pour créer un tel type d'objet il faut donc mémoriser les liens possibles de cet objet avec son environnement, mais aussi des prescriptions de saisie (nombre de points à saisir, désignation implicite d'objet ...).

On comprend donc maintenant l'influence des règles stockées dans le schéma au niveau des interactions utilisateur. Elles doivent donc agir au niveau du contrôleur.

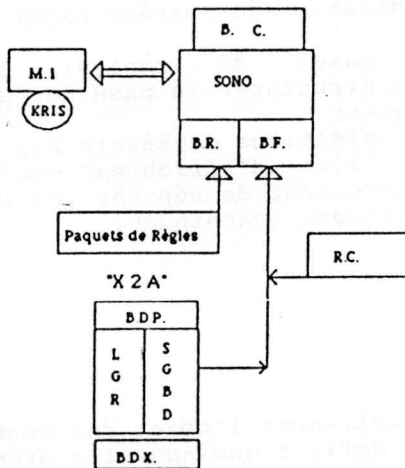
### 2. Les systèmes experts dans les programmes d'application.

---

La gestion des programmes d'application à l'aide de systèmes experts permet de dépasser le simple constat relatif à la cohérence du projet en cours de conception et de mieux aborder les points 4 et 6 cités au début de notre exposé, quant aux exigences relatives au fonctionnement d'un système de CAO.

Le système "SONO", développé parallèlement à "X2A" par K. ZREIK (ref. 6), nous en fournit un exemple.

Il utilise le moteur d'ordre 1, "KRIS", développé au CIMA (C. MANAGO, ref. 4, 5) et son fonctionnement peut être décrit selon le schéma suivant:



M.1 : MOTEUR D'INFERENCE D'ORDRE 1  
 KRIS: M1  
 B.D.P.: BASE DE DONNEES DU PROJET  
 B.D.X.: BASE DE DONNEES EXTERNE  
 L.G.R.: LANGAGE DE REQUETES  
 B.C.: BASE DE CONNAISSANCES (B.F. + B.R.)  
 B.R.: BASE DE REGLES  
 B.F.: BASE DE FAITS  
 R.C.: REGLES DE COMMANDES

La base de Fait est chargée à l'aide du langage de requête de la base de données, par un ensemble de "règles de commande".

Le système effectue alors une évaluation acoustique du bâtiment, repère, sinon propose, les éléments d'une solution susceptible d'améliorer les performances du projet.

La réalisation d'un tel système pose un certain nombre de problèmes dus à la taille de la Base de Faits (chargement du modèle entier) et de la Base de Règles (grand nombre de règles) et à l'utilisation interactive des systèmes experts.

La communication entre l'utilisateur et le système expert suppose enfin (problème de désignation, écho...) une interface graphique.

Il faut donc pouvoir:

- charger le modèle partiellement à partir du contexte d'utilisation repéré par la base de données, et donc régler le problème de la communication du système expert avec des structures externes.
- Segmenter la bases de règles; la solution retenue actuellement étant de structurer la base par des paquets de règles indépendants partiellement.
- Activer des primitives de dessin par des règles, ou/et utiliser la notion de "liste d'affichage" employée par le système "PROJET" (ref. 1) permettant de séparer les informations logiques des informations strictement graphiques.

## VII. CONCLUSION.

Elles font essentiellement l'objet des recherches en cours:

Nous devons donc définir une nouvelle architecture de système fondée sur les hypothèses suivantes:

- Rendre cohérente la logique fonctionnelle du système avec sa logique purement informatique.
- Donner au concepteur un langage de haut niveau, qui lui permette de créer et manipuler les objets de son domaine.

Le système doit se définir du point de vue utilisateur comme un ensemble d'outils (utilitaires) dont la cohérence logique d'enchaînement est contrôlée par des règles accessibles en modification et agissant sur une base de connaissance.

Définissons pour conclure un scénario d'utilisation d'un tel système, en précisant les différents modes de travail:

1. Définition/ modification du modèle du bâtiment (utilisation des utilitaires 2D, 3D de saisie).

2. Evaluation/ lancement d'applications (appel à la base d'algorithmes et aux règles spécifiques).
  3. Accès à la logique du système:
    - Logique de fonctionnement
    - logique de description des objets
    - logique de gestion du modèle
  4. Accès aux données statiques du système (B.D.X. et B.D.E.).
- 

REFERENCES:  
-----

REF. 1 : "PROJET, une base de données réseau en conception assistée par ordinateur, pour le projet d'architecture", CIMA  
Rapport PLAN CONSTRUCTION, septembre 1982.

REF. 2 : "X2A, pour un système de conception assistée par ordinateur en avant-projet sommaire de bâtiment", CIMA  
Rapport PLAN CONSTRUCTION, février 1985.

REF. 3 : "Un modèle d'évaluation technique et économique de gros-oeuvre de bâtiment, intégré dans un système de CAO utilisant un SGBD réseau", H. GALLEY  
Thèse de doctorat en génie civil en préparation  
Laboratoire de Génie Civil et Habitat, Université de Savoie

REF. 4 : "LEGO, un système expert pour la résolution de quelques problèmes en architecture", C. MANAGO  
Thèse de III cycle, université de PARIS VI, janvier 1984

REF. 5 : "LEGO, un système expert pour la conception de bâtiment par composants compatibles", C. MANAGO  
MICAD, novembre 1983.

REF. 6 : "Techniques des systèmes experts, travaux et applications", K. ZREIK  
Séminaire d'informatique "intelligence artificielle et conception" PLAN CONSTRUCTION, janvier 1985.

AUTEURS : François GUENA  
Jean Pierre LEININGER  
Khalidoun ZREIK

ADRESSE : CIMA 9, rue Barbanègre, 75010 PARIS

TELEPHONE : 42.09.08.00