

Architecture distribuée de méta-bases de données dans un environnement d'intégration évolutif

Gilbert Babin, Zakaria Maamar
Département d'informatique
Université Laval
Ste-Foy, Québec, CANADA, G1K 7P4

Cheng Hsu
Department of Decision Sciences and Engineering Systems
Rensselaer Polytechnic Institute
Troy, N.Y, USA, 12180-3580

Novembre 1994

Résumé

Le projet de recherche décrit dans cet article prolonge les résultats du projet d'intégration [Bab93, HB93] d'un environnement avec une seule méta-base de données vers un environnement avec plusieurs méta-bases de données. Plus précisément, si chaque méta-base de données contient l'information décrivant les différentes unités opérationnelles d'une ou plusieurs entreprises, la création de liens entre ses méta-bases de données réalise l'intégration de ses unités, tout en préservant leur autonomie. Le besoin d'intégration provient de la tendance actuelle de globalisation des entreprises et de la coordination accrue entre vendeurs, manufacturiers et même consommateurs, comme reconnu chez certaines compagnies tel que *General Electric* et *General Motors* [HBR⁺92]. Les résultats de ce projet contribuent à la résolution des problèmes du contrôle concurrentiel et de la gestion distribuée des connaissances dans de multiples bases de données et environnement de connaissances. Certainement, la solution proposée va permettre aux multiples entreprises de conclure des alliances commerciales, assurées automatiquement grâce à leurs systèmes intégrés d'informations. Plusieurs approches de développement peuvent être suggérées, Inter-Entreprise, Intra-Entreprise et Hybride; leur utilisation dépend du type d'intégration à accomplir.

1 Introduction

Considérons une grande entreprise manufacturière avec ses multiples divisions, chacune ayant probablement ses propres applications. De plus, ses applications distinctes peuvent utiliser des modèles d'informations hétérogènes dûs essentiellement aux: (1) divers besoins

des utilisateurs, (2) diverses plate-formes matérielles et logicielles, et (3) diverses structures logiques et physiques. Pour que cette entreprise puisse être performante, elle doit être capable de gérer et d'intégrer toutes ses ressources informationnelles. Le travail proposé constitue un cadre de gestion et d'intégration des applications pour les grandes entreprises, à mesure qu'elles grandissent et s'associent à de nouveaux partenaires économiques. Brièvement, les besoins des multiples systèmes dans la gestion d'information d'une entreprise peuvent être résumés comme suit:

- **Évolution:** l'environnement de l'entreprise doit être extensible et doit aussi permettre un développement incrémental. L'intégration peut commencer avec des parties réduites de l'entreprise, pour s'étendre graduellement au reste de l'organisation (même à d'autres organisations), sans perte de continuité opérationnelle des applications, ni d'intégrité des structures.
- **Adaptabilité:** les systèmes qui utilisent des technologies de l'information standards ou non, peuvent être incorporés dans l'environnement d'intégration de la même façon, ceci sans causer de perturbations à l'architecture existante.
- **Parallélisme:** les multiples systèmes doivent être capables d'opérer sous une forme concurrente, sans sérialisation globale ou mécanisme de synchronisation global imposé sur les transactions au niveau des instances de données.
- **Autonomie:** les systèmes locaux doivent être flexibles dans leur conception, construction et administration, ceci sans se conformer ou se convertir aux caractéristiques du schéma global.

Le travail sur la méta-base de données à *Rensselaer Polytechnic Institute* a mis l'accent sur la création d'un environnement d'intégration répondant aux besoins cités ci-dessus. La solution proposée comprend les éléments suivants:

- **Un modèle d'information de l'entreprise:** ce modèle représente globalement tous les modèles locaux de données et leurs connaissances contextuelles dans l'entreprise. Le modèle global permet aux méta-données des systèmes locaux ou autres d'être ajoutées, supprimées ou modifiées par le biais de transactions ordinaires de méta-données (comme des transactions base de données);
- **Une méta-base de données accessible (indépendante mais partageable):** cette méta-base de données implante le modèle d'information de l'entreprise. Elle comprend une hiérarchie (évolutive) de mini-méta-bases de données, à la façon d'une hiérarchie client-serveur.
- **Une architecture concurrente pour l'exécution:** cette architecture supporte le traitement concurrentiel des systèmes locaux grâce à un contrôle distribué et localisé de la connaissance.

Conjointement, ces éléments constituent une solution au problème de la gestion des ressources informationnelles de l'entreprise. Les résultats précédents ont mis l'accent sur l'autonomie, le parallélisme et l'adaptabilité des systèmes [Bab93]. Ce travail présente de nouvelles approches, qui considèrent les concepts d'évolution et de l'Intelligence Artificielle Distribuée (IAD).

Actuellement, les applications existantes sont devenues de plus en plus ouvertes en terme de topologie, de plate-forme et d'évolution, créant ainsi un besoin d'intégration [JCR93]. Traditionnellement, les approches d'intégration assurent l'inter-fonctionnement des applications grâce à un schéma global ou à un langage de manipulation de données [CHS92]. De plus, la plupart de ces approches fournissent un support pour les niveaux de spécification et de représentation, avec peu d'intérêt, sinon aucun au concept d'évolution. Le concept d'évolution a des implications dans plusieurs champs de recherche, allant des systèmes distribués à l'inter-fonctionnement des bases de données via les services fournis dans un réseau à grande échelle (WAN). Les systèmes évolutifs sont des systèmes qui fonctionnent normalement à grande et petite échelle.

Le développement de machines super puissantes et la prolifération des réseaux de communication ont permis aux chercheurs d'avancer que la majorité des problèmes peuvent être résolus en les distribuant sur une collection d'agents, comme c'est le cas dans l'approche IAD. Un système multi-agents s'intéresse au comportement de coordination entre une population d'agents intelligents et autonomes, plus exactement, comment ils combinent leurs connaissances, buts, états, et comment ils planifient ensemble la prise de décisions ou la résolution d'un problème [HG88]. Notre objectif principal est d'intégrer un ensemble de méta-bases de données et de construire un système coopératif. Un système coopératif à base de connaissances est considéré dans [Dee92] comme un ensemble de bases de connaissances autonomes et hétérogènes, appelées agents, capables d'interagir. Une base de connaissances est une collection de données et de règles; par conséquent, une base de données peut aussi être vue comme un agent. La coopération est la capacité d'un agent de travailler avec un autre agent pour résoudre un problème; ceci implique une cohérence totale du système. Ceci implique que le système reste cohérent [Dee92], et s'occupe:

1. de la consistance de la connaissance entre les agents,
2. de la fiabilité de la totalité du système,
3. de l'intégration des sous-solutions,
4. de l'amélioration de la performance globale.

L'article est organisé comme suit. Dans cette section, nous avons fait un survol de la problématique. La Section 2 présente l'architecture concurrente de l'approche méta-bases de données. Utilisant cette architecture comme cadre de base, on suggère deux schémas de coopération permettant l'évolution, à la Section 3. La Section 4 propose le *Rule-Oriented*

Programming Environment (ROPE) [Bab93, HB93] comme solution de développement de nouvelles composantes de l'architecture. Finalement, on conclut à la Section 5.

2 Architecture concurrente de la méta-base de données

L'architecture concurrente est présentée à la Figure 1. La méta-base de données (collection rigoureusement construite des méta-données de l'entreprise) fournit un modèle d'entreprise intégré pour les multiples systèmes d'informations, leurs bases de données et les interactions entre leurs différentes composantes c.à.d. le contenu informationnel et les connaissances contextuelles. L'approche méta-base de données:

1. utilise le modèle de l'entreprise pour assister les utilisateurs dans l'exécution de requêtes globales indépendamment des détails techniques et de la hiérarchie des schémas intégrés;
2. distribue la connaissance contextuelle pour renforcer les capacités des systèmes locaux en matière d'opérations de mise à jour et de communication entre eux sans un contrôleur central de base de données;
3. incorpore les modèles locaux hérités, nouveaux ou modifiés dans sa structure générique de méta-données pour supporter l'évolution, sans reconception ou recompilation du système.

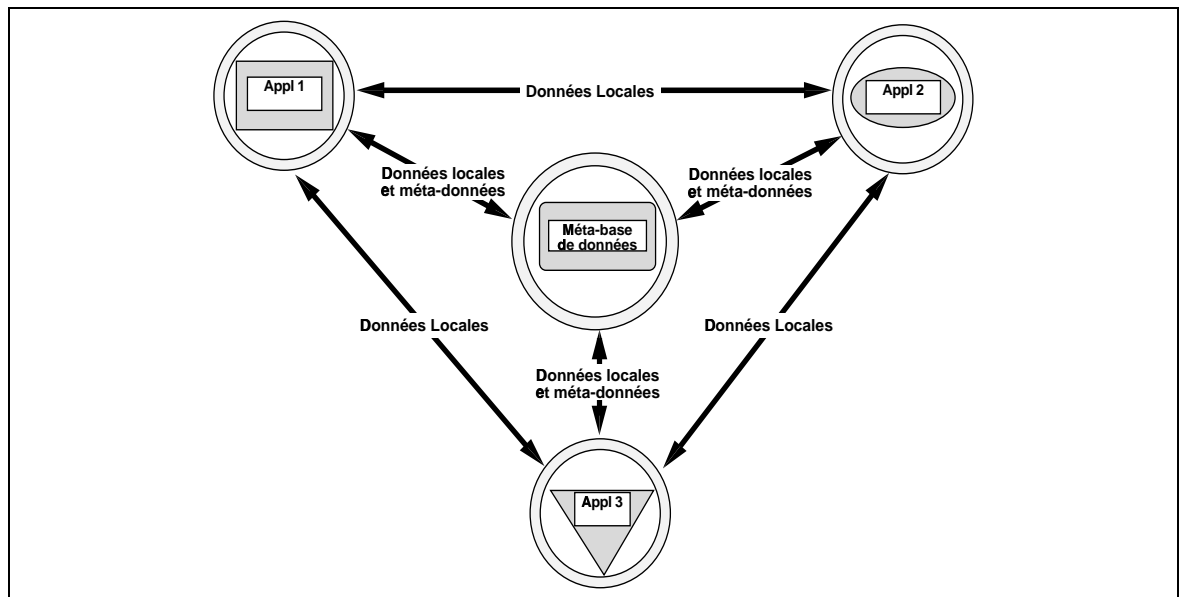


Figure 1: Architecture concurrente de la méta-base de données

Les shells dans l'architecture concurrente implantent les connaissances distribuées, gérées par la méta-base de données. Le schéma de la méta-base de données est un modèle générique appelée *Global Information Resource Dictionary* (GIRD) [Bab93], qui est utilisé pour représenter le modèle de données global et intégrer la connaissance des applications locales.

L'approche méta-base de données ne décrit pas d'implantation particulière de l'architecture concurrente; cependant, une instance quelconque de l'architecture concurrente doit fournir un environnement autonome, parallèle et adaptable pour l'intégration des systèmes de l'entreprise. Une telle implantation de l'architecture concurrente a été développée à *Rensselaer*; il s'agit de *Rule-Oriented Programming-Environnement* (ROPE) [Bab93, HB93]. ROPE définit la structure des shells de l'architecture concurrente, leur comportement et leur gestion par la méta-base de données. Les principaux critères dans la conception de ROPE étaient de minimiser le lien avec l'environnement local (autonomie des applications), tout en maximisant son adaptabilité.

3 Composantes de base de l'architecture concurrente

Au fur et à mesure que les entreprises évoluent et qu'elles s'associent à de nouveaux partenaires, le modèle de l'architecture concurrente pourrait suivre l'évolution de l'approche proposée. Par conséquent, plusieurs architectures devraient être définies pour permettre l'adaptabilité et l'évolution de l'environnement de gestion des informations de l'entreprise.

La majorité des approches d'intégration utilise un mécanisme central pour le contrôle du système des transactions globales. Ce type d'architecture présente les inconvénients suivants:

1. possibilité d'un *dead-lock* des applications locales, si le contrôleur central venait à tomber en panne;
2. possibilité de création d'un goulot d'étranglement par le contrôleur central, si le nombre de transactions qui lui sont adressées est important.

L'approche que nous considérons utilise le concept de méta-base de données et la méthode de développement *Two-Stage Entity-relationship* (TSER) comme moyen d'intégration, préservant les mêmes modèles de données et les mêmes règles de distribution de l'information. Plus précisément, nous voulons considérer un environnement global avec des environnements locaux distincts; chacun est représenté par sa propre méta-base de données et doit être intégré avec ceux qui restent. La nouvelle approche suggère:

1. le développement d'une nouvelle méta-base de données globale, résultant de l'intégration de méta-bases de données et d'applications locales (Approche Intra-Entreprise);
ou

2. la conception d'un processus de communication entre les méta-bases pour la coopération et la collaboration (Approche Inter-Entreprise).

Les deux architectures sont utilisées dans le but d'obtenir une stratégie d'intégration globale, tout en permettant aux méta-bases de données de bas niveau de stratégiquement grouper les systèmes fonctionnels locaux. En se basant sur ces approches, on définit deux architectures de base:

1. architecture hiérarchique de méta-base de données (Approche Intra-Entreprise)
2. architecture coopérative de méta-base de données (Approche Inter-Entreprise).

3.1 Architecture hiérarchique de méta-bases de données

Dans les grandes entreprises, l'architecture concurrente est étendue pour inclure une hiérarchie de méta-bases de données, comme illustré à la Figure 2. Une architecture hiérarchique de méta-bases de données présente des similarités avec l'approche développée en [Bab93]; il y a une correspondance directe entre elles: d'un côté les méta-bases de données locales et les applications, et de l'autre côté, les applications locales intégrées. La même correspondance est faite entre la méta-base de données globale et la méta-base de données résultat de l'intégration locale.

En utilisant cette architecture, plusieurs applications sont représentées dans une mini-méta-base de données, qui, à son tour, est représentée dans la méta-base de données principale. La mini-méta-base devient la porte d'accès aux applications locales qu'elle représente. Dans cette architecture, le nombre de méta-données stocké dans la méta-base n'est pas réduit; il va aussi contenir le modèle de toutes les applications localisées en-dessous (récursivement). Cette architecture est utile pour partitionner les applications en des ensembles où la connectivité est forte (flots d'informations dans un même segment du réseau). L'architecture hiérarchique présente les caractéristiques suivantes:

- intégration entre MBDs et applications;
- échange mixte d'informations:
 - données locales entre applications;
 - données locales et méta-données entre applications et leurs MBDs correspondantes;
 - données globales et méta-données entre applications, MBDs locales et la MBD globale.
- MBD globale résultat de l'intégration va:
 - fournir un modèle intégré pour les multiples applications et modèles de MBDs;
 - permettre de nouvelles fonctionnalités du système global;

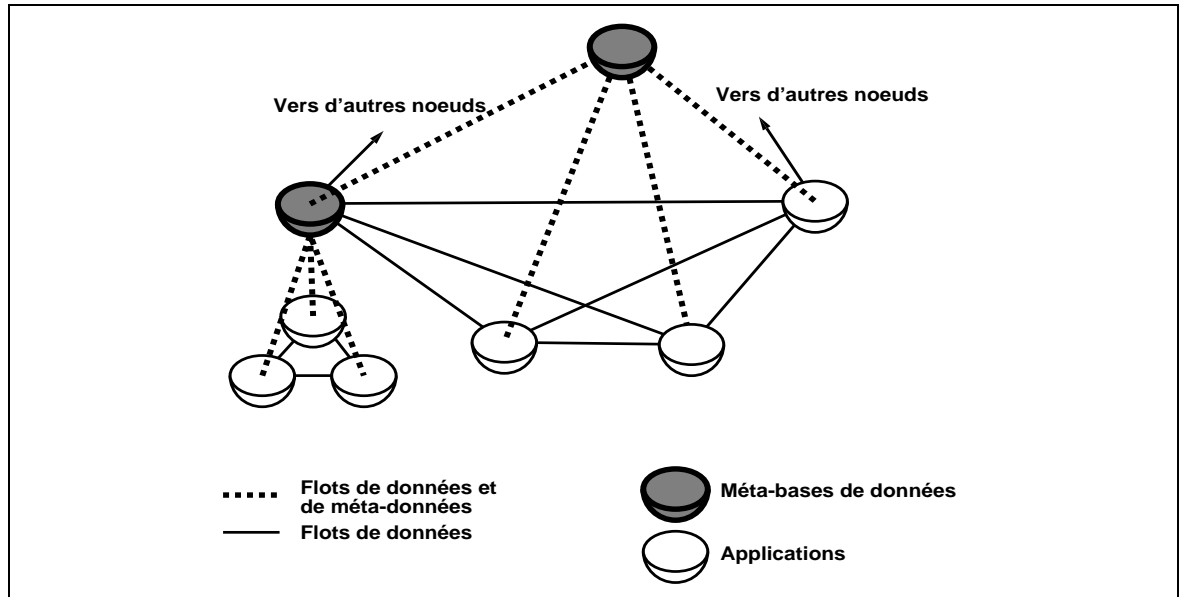


Figure 2: Architecture hiérarchique de méta-bases de données

- intégrer un ensemble de MBDs et applications sans causer de dérangement au système.

Deux méta-bases de données soeurs (c.à.d. sous la supervision d’une même MBD) ne partagent aucune information. Leur contenu de méta-données est limité à la description des applications et/ou MBDs localisées sous leurs hiérarchies. La gestion des données et méta-données globales est assumée par la MBD globale. L’approche hiérarchique présente les avantages suivants:

- méthode similaire à celle développée en [Bab93];
- fonctionnalités du système bien définies:
 - intégration locale assurée par les MBDs locales;
 - intégration globale assurée par la MBD globale.
- spécifications des schémas des MBDs locales nécessitant un minimum de changements;
- possibilité de formuler des requêtes de trois niveaux différents: applications, MBDs locales et MBD globale.

et les inconvénients suivants:

- fonctionnement global du système dépendant du fonctionnement de la MBD globale;
- différents types de données et de méta-données échangées;

- intégration hétérogène:
 $Intégration_{locale} = \{App_1, \dots, App_n\};$
 $Intégration_{globale} = \{App_1, \dots, App_p, MBD_1, \dots, MBD_m\};$

3.2 Architecture coopérative de méta-bases de données

Lorsque la sécurité des informations devient importante et/ou seulement une partie des informations stockées dans un système peut être accessible, l'architecture coopérative de méta-bases de données est proposée (cf. Figure 3).

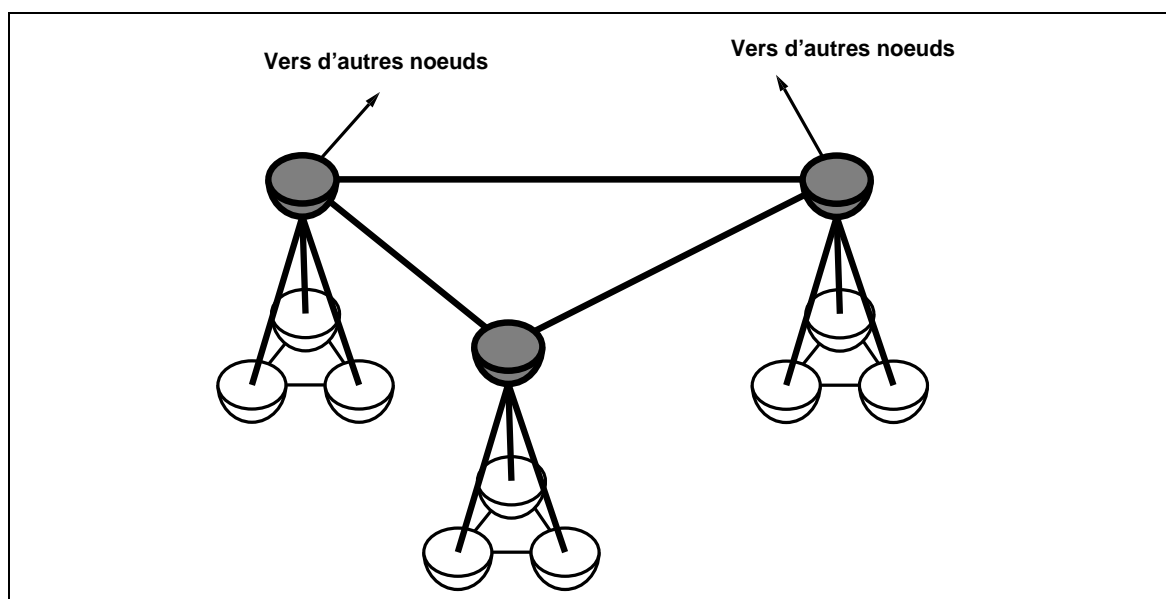


Figure 3: Architecture coopérative de méta-bases de données

L'approche coopérative peut être considérée comme un système multi-agents, où les MBDs sont assimilées à des agents autonomes. L'objectif est d'établir une coopération entre elles. Ceci peut être atteint par un dispositif de communication, permettant un flot d'échange de méta-données. Les actions des agents et leurs interactions sont les bases d'un système coopératif. Chaque méta-base de données est supposée contenir un modèle des applications qui lui sont rattachées. Cependant, elle a seulement des informations partielles des MBDs voisines. Ceci permet un partage sélectif d'informations. Une MBD va transmettre ses méta-données à d'autres MBDs à la base des besoins de chacune d'elles. De cette façon, son modèle d'information est composé de l'intégration des systèmes locaux et des modèles partagés des autres MBDs. L'intégration peut être faite en utilisant l'approche Intra-Entreprise, c.à.d. utilisation d'une MBD globale. Cependant, celle-ci n'est pas adaptée à un contexte de coopération:

1. la confidentialité des méta-données de chaque entreprise n'est pas garantie;
2. la localisation de la MBD globale crée un problème (responsabilité et localisation physique);
3. la perte d'autonomie pour les MBDs locales;

D'où l'intérêt d'une approche coopérative. La coopération dans un système multi-agents doit assurer la cohérence entre les composantes de la structure (agents) en insistant sur:

1. cohérence de la connaissance contextuelle entre les agents;
2. fiabilité de la totalité du système;
3. capacité d'intégrer les résultats partiels.

Toutes les MBDs doivent développer des vues partielles et globales [Dee92]. Les vues partielles correspondent à l'intégration locale et sont obtenues en utilisant ROPE [Bab93], tandis que les vues globales correspondent à l'intégration globale et sont le résultat de la coopération. Cependant, pour chaque vue, on devra posséder différents types de connaissance: (1) connaissance locale correspondant aux vues locales, et (2) connaissance globale correspondant aux vues globales.

L'approche coopérative présente les avantages suivants:

- approche utilisée pour l'intégration de plusieurs entreprises;
- nouvelle approche de conception proposée en considérant les MBDs comme des agents intelligents;
- possibilité d'utiliser des ressources informationnelles externes à l'entreprise;
- réduction de la complexité du problème d'intégration en utilisant l'IAD.

et les inconvénients suivants:

- difficulté de réaliser l'intégration si le nombre de MBDs à intégrer est important:
 $L = n*(n-1)/2$;
L: nombre de liens; n:nombre d'agents;
Le nombre de liens augmente proportionnellement au nombre d'agents;
- MBDs ayant double fonctionnalités: intégration locale et globale;
- risque de conflicts entre les agents, nécessitant ainsi le développement d'un protocole de négociation.

3.3 Architecture mixte de méta-bases de données

Les architectures hiérarchiques et coopératives peuvent être utilisées conjointement créant ainsi une architecture mixte de méta-bases de données. Les mêmes règles de distribution des méta-données seront utilisées, dépendant des liens entre les différentes MBDs. Le but de l'architecture mixte est de réaliser une intégration interne entre les composantes d'une même entreprise (directions, départements, ...), tout en essayant d'atteindre une intégration externe entre plusieurs entreprises. L'approche présente pratiquement les mêmes avantages et inconvénients vus précédemment. Un nombre important de scénarios peut être associé à l'approche hybride; cela est fonction des structures des entreprises et de l'importance des flux de données entre les applications. On peut suggérer comme scénarios:

- les deux entreprises développent des MBDs globales en utilisant l'approche hiérarchique. Elles seront intégrées en utilisant l'approche coopérative;
- les deux entreprises développent des MBDs globales en utilisant l'approche classique [Bab93]. Elles seront intégrées en utilisant l'approche coopérative;
- etc.

Ces scénarios montrent la variété des cas d'intégration et insistent sur les besoins de la gestion des systèmes d'informations vus auparavant.

4 ROPE: une approche d'implantation

La méthode ROPE développe la technologie des shells nécessaire pour l'architecture concurrente des méta-bases de données [Bab93]. Elle définit:

1. comment les règles sont stockées dans les shells locaux;
2. comment les règles sont traitées, distribuées et gérées;
3. comment les règles interagissent avec leur application correspondante;
4. comment les différents shells interagissent entre eux;
5. l'architecture des shells.

ROPE assure l'intégration des systèmes en traitant les requêtes et les règles globales au niveau des applications locales. Cinq types de déclencheurs sont définis permettant aux shells d'exécuter leurs opérations:

1. déclencheur de type temps;
2. déclencheur de type données;
3. déclencheur de type programme;

4. déclencheur de type règle;
5. déclencheur de type utilisateur.

Dans le but de supporter les fonctionnalités de la MBD [HR93], la structure dynamique de ROPE présente les classes de langage suivantes:

- langage de définitions des shells permettant leur création;
- langage de règles et de modélisation aidant la description de la MBD et la connaissance des shells;
- protocole et langage de messages utilisés pour la communication entre les applications et MDBs.

Ces langages supportent les transactions entre la paire (ROPE, applications locales) pour l'exécution des règles et les opérations de gestion de la connaissance, et la paire (ROPE, système de gestion de la MBD) pour la maintenance du schéma de la MBD. Actuellement, dans le contexte de ROPE et de l'architecture concurrente, l'utilisation du système de gestion de la métabase de données MDBMS est limitée à deux tâches:

1. traitement des requêtes globales pour les applications utilisateurs en utilisant le MDBMS ou les shells locaux;
2. gestion de la connaissance pour la création d'un nouveau shell.

Les transactions définies dans ROPE sont liées au *traitement des règles par les shells, aux requêtes sur les données locales, à la modification de la structure du shell et aux requêtes sur les données globales*. En plus de ces différents types de transactions, les architectures coopératives et hiérarchiques requièrent de nouveaux mécanismes, qui devront assurer l'autonomie locale et globale des applications de l'architecture concurrente. Ces mécanismes vont permettre:

1. l'échange de données à l'intérieur et l'extérieur de l'entreprise;
2. l'établissement de liens de communications entre les méta-bases de données;
3. la définition de nouvelles fonctionnalités pour les shells.

5 Conclusion

Les différentes approches d'intégration présentées dans ce rapport sont seulement des moyens qui rendent possible l'intégration de ressources informationnelles externes. Ces ressources sont utilisées pour établir un contact entre des partenaires économiques. Le concept de MBD réduit la complexité du problème d'intégration en: (1) suggérant une conception simplifiée de l'architecture des systèmes et (2) intégrant les concepts des systèmes à base

de connaissance à un gestionnaire de données distribuées.

On a présenté deux approches qui assurent ce type d'intégration. Une suggérant une architecture hiérarchique et une autre suggérant une architecture coopérative de méta-base de données. L'architecture globale résultante est une combinaison entre les MBDs, les applications et les liens de communications. À l'intérieur d'une entreprise, l'architecture hiérarchique permet une intégration graduelle des différents systèmes d'informations. De plus, elle permet aux applications fortement coopératives d'être groupées sous la même méta-base de données. Entre les entreprises, l'architecture coopérative permet aux entreprises de partager leurs ressources informationnelles pour réaliser la collaboration.

Bibliographie

- [Bab93] G. Babin. *Adaptivness in Information System Integration*. Thesis, Department of Decision Science and Engineering Systems, Rensselaer Polytechnic Institute, Troy, New York, USA, Août 1993.
- [CHS92] Christine Collet, Michael N. Huhns, et Wei-Min Shen. Resource integration using a large knowledge base in carnot. *Computer*, pages 55–62, Decembre 1992.
- [Dee92] S. M. Deen. A general framework for coherence in ckbs. *Intelligent Information Systems*, 1(3), 1 septembre 1992.
- [HB93] Cheng Hsu et Gilbert Babin. A rule-oriented concurrent architecture to effect adaptiveness for integrated manufacturing enterprises. Dans *International Conference on Industrial Engineering and production Management*, pages 868–877, Mons, Belgique, Juin 1993.
- [HBR⁺92] Cheng Hsu, Gilbert Babin, Laurie Rattner, Lester Yee, M'Hamed Bouziane, et Waiman Cheung. Metadatabase modeling for enterprise information integration. *Journal of Systems Integration*, 1:5–37, 1992.
- [HG88] Alan H. Bond et Les Gasser. *Readings in Distributed Artificial Intelligence*. Morgan Kaufman Publishers, INC, San Mateo, California, 1988.
- [HR93] Cheng Hsu et Laurie Rattner. Metadatabase solutions for enterprise information integration problems. *Database, A Quarterly Publication of Sigbit*, 24(1):23–35, hiver 1993.
- [JCR93] Franchitti J.-C et King R. Amalgame: A tool for creating interoperating persistent, heterogenous components. *Advanced Database Systems*, pages 313–336, 1993.