

EAI urbanisation comment réussir ?

Cet article constitue une synthèse du document « Interface et urbanisation du système d'information » publié par l'AFAI (Association Française des Auditeurs et conseils Informatiques) et réalisé par les co-auteurs :

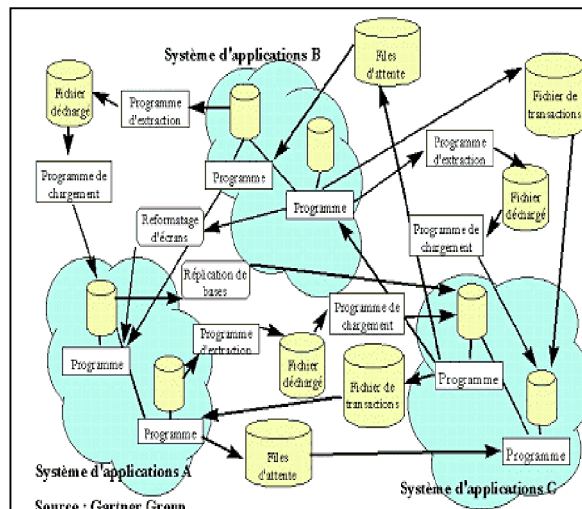
- Fabien Cleuet, Directeur audit et conseil système d'information, DIATHESE,
- Vincent Goujon, Directeur de mission, audit et conseil système d'information, ARTHUR ANDERSEN,
- Henry Peyret, Senior Advisor, Giga Groupe.

1. HISTORIQUE DES BESOINS D'INTERFACE

Durant des années, la communication au sein du système d'information a été réalisée en point à point pour aboutir au « modèle spaghetti » décrit par le Gartner Group.

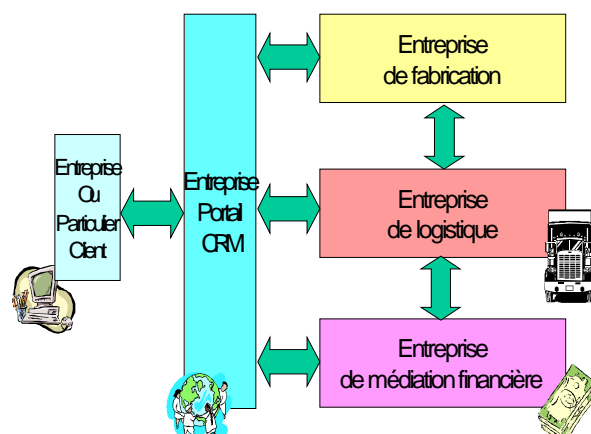
Pour sortir de ce modèle, les urbanistes se sont orientés vers une structuration par les données du système d'information ce qui a abouti à la vague des ERP dans le courant des années 90.

Mais, très vite un nouveau dilemme est apparu, l'ERP ne suffisait pas ou plus à tous les besoins des entreprises.



Les systèmes d'information s'organisaient de nouveau autour de multiples briques, des progiciels fournis par plusieurs éditeurs : ERP (ou PGI), CRM (ou GRC) SCM (gestion de la chaîne logistique).

En parallèle, les entreprises ont cherché à se recentrer sur leur cœur de métier et à développer des communautés d'intérêts économiques. Gartner Group appelle cette capacité de collaborer entre entreprises, et non plus seulement intra- entreprise, le C-Commerce pour « Collaborative Commerce ».



Cette nécessité d'interfacer les entreprises ou des progiciels entre eux, s'est également retrouvée dans les vagues de fusion/acquisition/cession.

Ces différents cas généraient un même besoin : faire communiquer entre eux des systèmes d'information distants et indépendants par volonté ou par nécessité.

L'architecture globale du système d'information est ainsi amenée à répondre de manière plus expresse à des impératifs de flexibilité et d'automatisation (intégration) que Giga Group résume sous le terme E-Ability :

- l'arrivée de Progiciel de Gestion Intégré (PGI alias ERP) impose leur connexion avec des modules métiers existants,
- bon nombre d'entreprises acquièrent des produits d'horizon différents et le système d'information dans son ensemble est concerné par le développement des technologies de l'Internet et leur intégration au système d'information de gestion,
- jamais les organisations n'ont dû être aussi réactives compte tenu des fusions, acquisitions, filialisations, externalisations, évolutions vers de nouveaux canaux de distribution, ...

Ces besoins sont par ailleurs renforcés par le développement de l'e-commerce, qui catalyse l'évolution des pratiques et des comportements des clients, partenaires, et utilisateurs qui demandent un « service temps réel et sécurisé ».

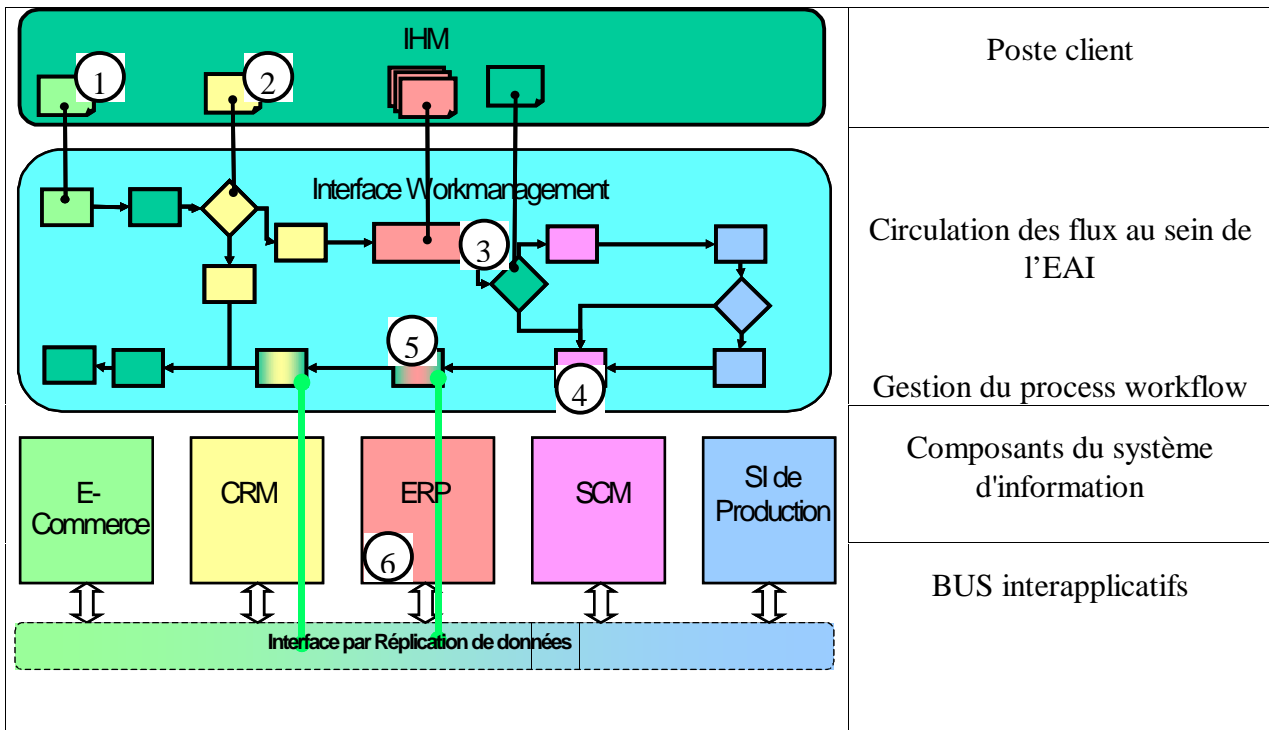
Dans ce contexte la tâche des auditeurs internes pour évaluer les risques, les qualités nécessaires et les bonnes pratiques du système d'information devient de plus en plus ardue car, après les progiciels qui deviennent de plus en plus des boîtes noires, les interfaces à leur tour « s'opacifient ».

2. DÉFINITION DE L'EAI

L'acronyme EAI se traduit littéralement en français par « Intégration d'Application d'Entreprise ». Ce concept, apparu en 1998, rassemble des produits aussi divers que : les middlewares de communication, les serveurs d'applications, les outils de transformation de données, les message broker, les workflow broker, les outils de réplication de données, les outils d'alimentation de datawarehouse, etc. En réalité le seul critère commun à tous les produits dits d'EAI est le fait qu'ils offrent tous un ou plusieurs connecteurs vers des progiciels.

En effet, depuis quelques années, une majorité de progiciels ne permet plus à des programmes d'accéder directement à leurs tables internes stockées dans des SGBD. En contrepartie, ils proposent des APIs, des tables d'interface, des programmes d'injection et d'extraction, ce qui nécessite le développement de connecteurs (appelés aussi adaptateurs, extracteurs ou injecteurs) propres à chaque progiciel.

De plus une nouvelle catégorie appelée IAI (Internet Application Integration) va se développer pour accompagner l'extension des processus de l'entreprise vers l'extérieur. Ces IAI permettront aussi la mise en place rapide de nouveaux canaux de distribution qui accompagnent le recentrage des entreprises vers leurs clients.



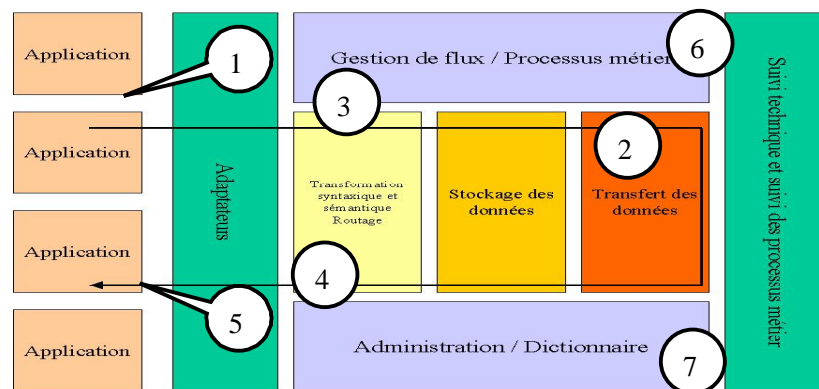
Le schéma ci-dessus illustre un processus de prise de commande client en mode synchrone :

1. l'arrivée d'une commande dans le système d'e-commerce,
2. la saisie des informations clients (compte, adresse de livraison, de facturation) dans le logiciel de CRM,
3. la détermination du délai de livraison suivant le stock dans l'ERP, et une gestion back-office des commandes clients en mode asynchrone :
4. la demande de mise en fabrication si les stocks sont insuffisants,
5. le déclenchement de la livraison,
6. le déclenchement de la facturation (asynchrone) lorsque la livraison a eu lieu.

3. LES COUCHES FONCTIONNELLES DE L'INTERFACE

Un système d'interface peut être représenté par les sept couches fonctionnelles suivantes :

1. Extraction
2. Transport
3. Transformation
4. Dispatching
5. Injection
6. Archivage
7. Administration



4. LES MOYENS NÉCESSAIRES A LEUR MISE EN OEUVRE

Ces EAI tendent à devenir la colonne vertébrale des systèmes d'information. Véritable centre de gestion des données qui circulent dans l'entreprise, ils permettent de développer une meilleure vue d'ensemble et aussi une meilleure gestion des flux. En contrepartie, cette concentration en fait des systèmes présentant un risque accru pour les entreprises.

4.1 Evaluation des risques et critères de qualité

L'importance d'un système d'interface s'accroît avec celle des processus dans lesquels il intervient. Une manière d'évaluer globalement cette sensibilité est d'apprécier les conséquences d'un dysfonctionnement, les travaux qu'il conviendrait de réaliser et leur degré d'urgence : correction des erreurs, mise en place d'une solution de secours pour assurer la continuité du processus... Cette appréciation peut globalement être réalisée en analysant:

- L'importance stratégique du processus auquel le système d'interface contribue (prise de commande sur Internet...),
- le mode de fonctionnement de l'interface (régulier ou occasionnel, synchrone ou asynchrone...),
- La nature des travaux qu'elle réalise sur les données (simple transfert de données ou avec transformation et routage des données en fonction de règles de gestion...).

Pour prévenir et gérer les conséquences d'éventuels incidents, il est nécessaire d'appliquer lors de la conception et la mise en œuvre de ces systèmes un certain nombre de principes regroupés dans les quatre critères de qualité suivants :

| | |
|----------------------------|--|
| Exactitude et exhaustivité | <p>L'information reste intègre durant son traitement c'est à dire qu'elle ne subit pas de détérioration.</p> <p>L'information est « traitée » dans son intégralité (toutes les lignes et colonnes d'une table).</p> <p>Dans le cas d'une interface, chaque événement est diffusé une seule fois.</p> |
| Traçabilité | <p>La traçabilité repose sur le suivi des données dans le système d'information et le suivi des règles appliquées (transformation, « dispatching », ...) tout au long de leur cheminement.</p> <p>Rappelons qu'au-delà des aspects réglementaires la traçabilité est une nécessité de gestion permettant d'expliquer et de recouper l'information. Enfin, elle est indispensable pour mettre en œuvre les mécanismes de réversibilité.</p> |
| Réversibilité | <p>C'est la capacité à reprendre, suite à un incident, un lot, une transaction, un flux pour relancer le processus d'interface à partir de l'une des couches fonctionnelles.</p> |
| Documentation | <p>La qualité de la documentation des interfaces va de paire avec la traçabilité. Elle a pour but de satisfaire au maintien de la piste d'audit. Elle concerne la cartographie globale du système d'information indispensable à sa lisibilité, mais aussi, la description des règles applicables à chaque lot, transaction, flux, données,... à chaque niveau des différentes couches fonctionnelles d'une interface.</p> |

4.2 Les bonnes pratiques

Les notions évoquées ci-après sont fondamentalement des règles d'architecture de système d'information. A ce titre, elles doivent faire l'objet d'une réflexion globale et préalable avant leur mise en œuvre.

4.2.1 L'EVOLUTION DU RESPONSABLE DES DONNÉES VERS LE RESPONSABLE DE PROCESSUS

Que l'interface fonctionne en transactionnel ou en batch, il est nécessaire de connaître de manière certaine le propriétaire d'une information transmise. Ceci ne pose généralement aucun problème lorsque les échanges sont unidirectionnels. Dans le cas contraire, l'analyse doit être faite à un niveau plus fin qui n'est plus la donnée mais l'événement de gestion acheminé.

Dans une approche d'interface classique, ce sont les systèmes émetteurs qui « poussent » l'information vers les systèmes destinataires. Il en découle clairement que toute modification ou annulation doit être faite sur l'application émettrice pour être restituée aux autres systèmes. Ainsi, à titre d'exemple, une facture client annulée ou modifiée doit être retraitée en gestion commerciale mais pas directement dans une application liée.

Cette approche imposée par les techniques traditionnelles évolue et aujourd'hui l'application propriétaire d'une information n'est plus nécessairement celle qui a l'initiative de sa diffusion. Cette évolution est permise par l'arrivée de nouvelles technologies d'EAI, de workflow, ...

4.2.2 LE LOTISSEMENT

La gestion des lots d'interface est un point crucial car il contribue à structurer l'approche globale des interfaces.

A l'origine des interfaces batch par fichiers, le lotissement consistait à identifier les données transmises depuis le système émetteur mais aussi à conserver certains paramètres de ce lot (date de création, d'envoi, nombre de mouvements, ...). Même si cette approche reste fréquemment utilisée en périphérie des ERP, une tendance de fond est constatée en faveur des interfaces temps réel. Dans un tel cas de figure, le lot devient une transaction qui doit être identifiable sans équivoque dans les systèmes émetteurs et récepteurs.

4.2.3 LA SYNCHRONISATION DES APPLICATIONS

L'interfaçage entre deux applications doit être contrôlable a posteriori. Le respect de ce principe repose sur les fonctions décrites ci-après :

- Marquage émetteur et récepteur par la mémorisation du numéro de lot au niveau de chaque événement de gestion interfacé depuis le système émetteur ou inséré ou mis à jour dans le système récepteur
- Blocage après extraction et injection qui interdit une modification des informations essentielles qui pourraient désynchroniser les systèmes
- Existence de mécanismes de contrôle de bouclage entre systèmes (contrôle de flux et de stock)

4.2.4 LA MISE EN ŒUVRE DE LA RÉVERSIBILITÉ

Quelle que soit la qualité d'une interface, il est fort possible d'envisager qu'à un moment donné il faudra revenir en arrière sur un lot déjà interfacé. Parmi les raisons qui peuvent expliquer ce cas de figure, on peut noter les éléments suivants :

- Une règle de transformation des données qui n'est plus conforme à la codification utilisée dans le système récepteur (la réfection ne peut être que partielle).
- Une interface déclenchée par inadvertance dans le système émetteur et qui doit donc être annulée et refaite de bout en bout.
- Un format de données d'injection non conforme mais qui a été accepté.

Rappelons aussi que le contexte de mise en œuvre de la réversibilité est très diversifiée :

- Une organisation batch avec des fichiers classiques ou autour d'une base de données.
- Une interface TP animée par des transactions.
- L'utilisation d'outils d'interfaces.

Le plan comptable 1999 et la réglementation fiscale consacrent la traçabilité du système d'information et l'idée que les applications de gestion deviennent des systèmes comptables auxiliaires. Dès lors, la réversibilité doit être analysée sous un angle réglementaire.

4.2.5 LA GESTION DES DROITS D'ACCÈS ET LA CONFIDENTIALITÉ

Le paramétrage de l'interface doit faire l'objet d'un contrôle d'accès.

Le logiciel retenu doit donc permettre :

- La gestion de groupes d'utilisateurs ayant des droits spécifiques. A ce sujet, il faut considérer l'existence de trois familles d'utilisateurs : les fonctionnels de la maîtrise d'ouvrage, les techniciens administrateurs qui supervisent le paramétrage et les opérateurs d'exploitation devant gérer les opérations et résoudre les difficultés quotidiennes.
- La conservation de l'historique des modifications apportées aux droits d'accès.
- La conservation de la trace des différents accès.
- Cryptage des informations échangées avec l'extérieur (notamment dans le cadre d'EDI ou d'externalisation).
- Authentification des émetteurs.

5. CONCLUSIONS

Bien que les qualités et la couverture fonctionnelle des outils d'EAI arrivent à maturité, leur utilisation dans le Système d'Information doit être précédée d'une démarche volontaire d'amélioration de la qualité des interfaces et donc de diminution des risques liés au système d'information.

Enfin, l'utilisation d'un outil, par essence plus contraignant qu'un développement spécifique, ne permet plus de faire jouer aux interfaces le rôle de joint de dilatation aux interfaces. N'oublions pas l'expérience des premières implémentations des ERP qui ont conduit à de nombreux développements de spécifiques au sein de l'ERP diminuant ainsi leur potentiel d'évolution.