

INTRODUCTION AUX EIA ERP

SOMMAIRE

INTRODUCTION AUX EIA ERP	1
SOMMAIRE.....	i
INTRODUCTION	1
CHAPITRE I LES ERP ET LES SI DES ENTREPRISES.....	3
I Définition et caractéristiques	3
I.1 Définition.....	3
I.2 Caractéristiques des solutions ERP.....	3
II Description technique des ERP	4
III Architecture Modulaire des ERP	4
III.1 Le module finance	5
III.2 Le module logistique	5
III.3 Le module e-commerce.....	5
III.4 Résumé des Composants d' une RP	6
IV Avantages et inconvénients des ERP	6
IV.1 Avantages des ERP	6
IV.2 Inconvénients des ERP	7
V Le marché des ERP	7
V.1 Les offres commerciales	8
V.2 Les solutions Open Source.....	8
V.3 Types d'offre ERP	8
VI Mise en place des solutions ERP	9
VI.1 Gestion des projets ERP	9
VI.2 Méthodologie de choix	9
CHAPITRE II LES EAI	13
I INTRODUCTION	13
II PROBLEMATIQUEDEL' INTEGRATION.....	13
III DEFINITION.....	14
IV Architectures des EAI	16
IV.1 L' architecture Hub and Spoke	16
IV.2 L' architecture Bus Adaptatif	16
V Architectures fonctionnelle des EAI	17
V.1 Transport de données	17
V.2 Courtier de messages (Message Broker) ou Moteur d'Intégration	17
V.3 Adaptateur applicatif(Connecteurs)	17
V.4 Référentiel (Repository)	18
V.5 Gestion des processus.....	18
V.6 Supervision et Administration	18
VI Les ESB.....	18
VII Quelques produits d' intégration	19
WEBOGRAPHIE.....	1

INTRODUCTION

La maîtrise des flux qui traversent une entreprise a été depuis toujours la préoccupation première des entreprises. L'informatique et plus particulièrement les logiciels ou progiciels sont devenus un outil incontournable dans cette optique. Ces logiciels sont classés en :

- **Les progiciels horizontaux** : Logiciels standards utilisables par une entreprise quel que soit son secteur d'activité.
- **Les progiciels verticaux** : Logiciels spécifiques à une profession ou un secteur d'activité précis. Ces logiciels permettent de prendre en charge les spécificités de chaque métier.

Au début des années 70-80 les données gérées par une entreprises se présentaient comme suit.

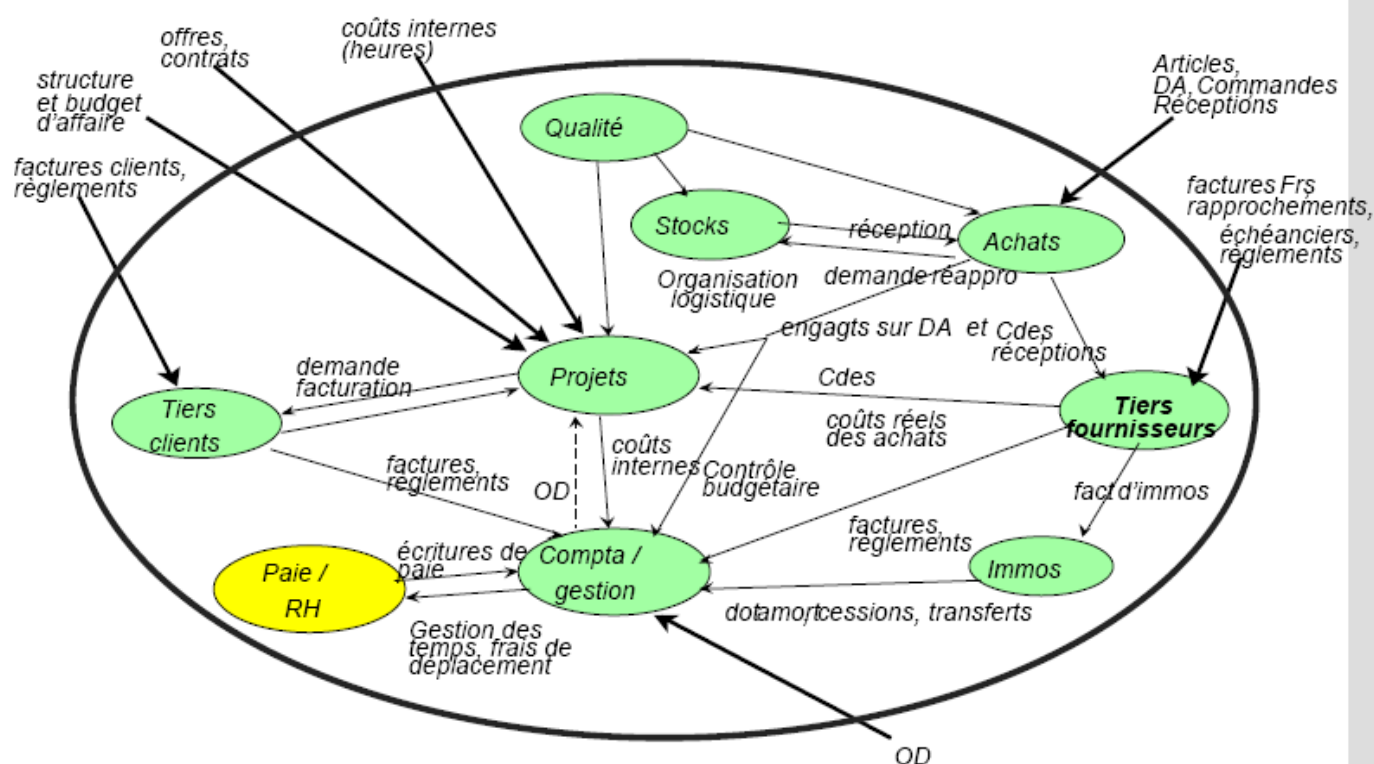


Figure1: Jungle des données dans un SI dans les années 70-80

Le système d'information (SI) est en ce moment constitué d'un ensemble hétérogène de logiciels utilisés de manière monofonctionnelle et tournant sur des gros systèmes. La conception de ces logiciels est conduite par le département informatique.

A partir des années 90, avec le besoin pressant des informations, les SI sont tournés vers le client. La mise en place du SI est alors conduite par la direction générale avec les informaticiens. Il est constitué d'un ensemble de systèmes flexibles, facile à modifier, organisés de façon multifonctionnelle autour de processus métiers, organisés autour des réseaux, sous des systèmes Unix et de l'Internet.

L'intégration des données se fait ressentir. Cette intégration vise comme objectif :

- accélérer la prise de décision stratégique (EIS/SIAD);
- étendre l'entreprise à ses clients et ses fournisseurs ;

- améliorer le partage et l'apprentissage des connaissances au sein de l'entreprise ;
- améliorer le partage de la connaissance et de l'information au sein de l'entreprise ;

De nos jours, vu le nombre élevé de domaines que couvre une entreprise, les entreprises recourent de manière générale à des techniques pour couvrir les besoins logiciels de l'ensemble des domaines. Elle peut :

- **Effectuer la démarche «Best of BREED»** pour une fonctionnalité donnée qui consiste à sélectionner, pour construire son système d'information, le progiciel le plus adapté, fonction par fonction (comptabilité, paie, production...). Les solutions «best of BREED» doivent être sélectionnées sur des critères d'interopérabilité. Les briques dédiées supporteront avec une plus forte probabilité des standards ouverts, facilitant cette communication. Le risque de construire son système d'information sur des «Best of BREED» est lié au manque de cohérence du système et à l'obligation de développer en spécifique la communication entre les différents modules. Multiplier les briques complexifie également le paramétrage et l'administration (plusieurs interfaces donc plusieurs compétences), ainsi que les relations avec les éditeurs (plusieurs interlocuteurs).
- **Adopter un logiciel couvrant l'essentiel des domaines** (fonctions) de celles-ci. Cette approche lorsqu'elle vérifie certaines conditions est qualifiée d'ERP
- **Intégrer l'ensemble des applications existantes via des EAI** (Entreprise Application Intégration)

Il est question dans ce cours de présenter les solutions ERP et EAI.

CHAPITRE I LES ERP ET LES SI DES ENTREPRISES

I Définition et caractéristiques

I.1 Définition

L'acronyme ERP (Anglo Saxon) signifie «Enterprise Ressource Planning » et se traduit en français par Progiciel de Gestion Intégré ou PGI. ERP est le terme le plus couramment utilisé. Plusieurs définitions dont certaines très minimalistes permettent de définir un ERP. Parmi ces définitions on peut retenir :

- «Un ERP/PGI dispose d'au moins deux fonctionnalités applicatives reposant sur une base de données commune ».
- «l'ERP est un progiciel qui couvre au moins trois des grandes fonctions de l'entreprise et partage ses données dans une base commune à tous les modules».

De manière générale, et pour des raisons historiques, le système ERP/PGI traite les fonctions de production, de vente, d'achat, de stock et de comptabilité, etc. Le module de gestion des ressources humaines est également souvent assimilé à un module du système ERP/PGI. Ce flou sur la composition exacte d'un système ERP/PGI est d'ailleurs entretenu par les définitions des différents organismes référents en la matière. Il n'est donc pas rare de voir encore aujourd'hui des soi-disant ERP qui ne fonctionnent que sur AS400, qui ne couvrent que la fonction financière ou qui couvrent très mal certaines fonctions

De manière détaillée on peut définir un ERP comme :

- «Un **système ERP/PGI**, est un progiciel qui vise à couvrir et optimiser la (quasi) **totalité** des **fonctions** et des **processus** de gestion d'une organisation. Il est composé de modules interconnectés partageant le même référentiel fonctionnel et technique. Il s'appuie sur une **couche standard** pour traiter les besoins génériques et répondre aux besoins spécifiques par des paramétrages dont la réalisation peut échoir à des utilisateurs fonctionnels. Enfin, il peut fonctionner indifféremment sur plusieurs serveurs de données, systèmes d'exploitation et SGBD».
- « Ensemble d'unités logicielles intégrant les principales fonctions nécessaires à la gestion des flux et des procédures de l'entreprise (comptabilité et finances, logistique, paie et ressources humaines, etc.). Toutes ces unités accèdent à des ressources communes, en particulier des bases de données».

I.2 Caractéristiques des solutions ERP

Les ERP se caractérisent par l'usage systématique de ce qu'on appelle un moteur de workflow (qui n'est pas toujours visible de l'utilisateur), qui permet, lorsqu'une donnée est entrée dans le système d'information(à travers un module), de la propager dans tous les modules du système qui en ont besoin, selon une programmation prédéfinie.

En définitive, pour qu'une solution logicielle soit qualifiée d'ERP, elle doit :

- Couvrir la (quasi) totalité du S.I de l'entreprise
- Emaner d'un éditeur unique
- Permettre l'adaptation rapide aux règles de gestion
- Permettre une gestion effective des domaines de l'entreprise et une collaboration des divers processus
- Permettre une mise à jour en temps réel des informations modifiées de tous les modules concernées par la modification

- Garantir à l'utilisateur l'unicité de l'information
- Disposer des IHM uniformisées
- Permettre l'existence des d'outils de développement de compléments applicatifs

Le passage vers les solutions ERP s'explique à travers des raisons environnementales (avancée de la technologie, mondialisation, multinationalisation) ou interne (développement de nouvelles activités, fusion acquisition, ...)

II Description technique des ERP

Un ERP est un logiciel client/serveur. L'architecture de déploiement d'un ERP est donnée par la figure ci-dessous.

Un ERP est donc un serveur. La base de données utilisée par la plus-part des ERP est ORACLE. Ils sont compatibles avec les outils du pack office et ceci pour les besoins de personnalisation du bureau, d'import/export de données. Ils sont dans la majeure partie des cas compatibles avec la plupart des outils de reporting.

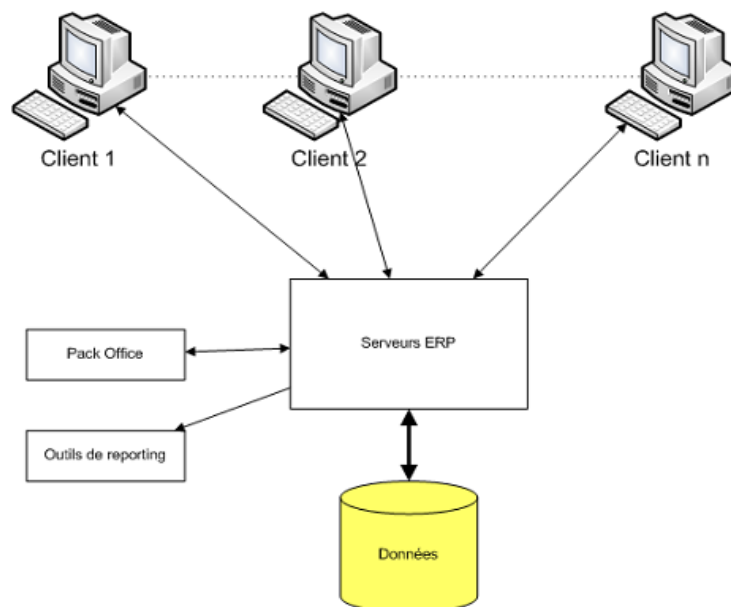


Figure 2: Architecture d'un ERP

III Architecture Modulaire des ERP

Un ERP est un ensemble dont toutes les parties fonctionnent les unes avec les autres d'où l'ergonomie et l'unicité des informations et donc la cohérence du SI.

Un ERP est modulaire dans le sens où il est possible de n'avoir qu'une ou plusieurs applications, en même temps, ou peu à peu. Les applications modulaires telles que les ERP permettent d'être sûr de la compatibilité des modules entre eux, ils s'imbriquent comme des blocs de Lego et fonctionnent ensemble (pas de vérification de compatibilité à effectuer).

Voici un exemple d'architecture modulaire qui tend à représenter tous les ERP :

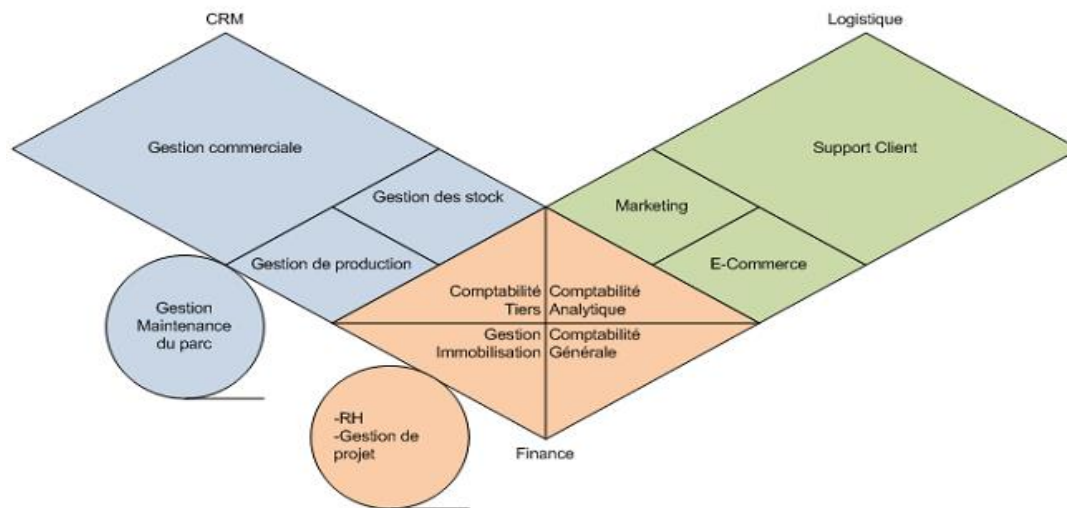


Figure3: Architecture modulaire des ERP

III.1 Le module finance

Le module finance se divise en 4 voire 5 sous-modules. Il rassemble les données pertinentes (en particulier les données comptables de l'entreprise) pour établir sur un plan international des comptes annuels. Ce module effectue du contrôle de gestion et des prévisions concernant les objectifs de l'entreprise.

De même, il permet de faire de la comptabilité tiers, analytique, générale, de gérer les immobilisations (gestion des investissements), les ressources humaines, soit l'administration du personnel, des frais de déplacement et du temps de travail ainsi que la gestion de paie et les besoins en postes.

Ce module est complexe car il faut avoir de bonnes connaissances en comptabilité.

III.2 Le module logistique

Le module logistique ou encore appelé Négocier, est le module le plus convoité par les entreprises car il permet de gérer tout ce qui se rapporte aux ventes/achats, en particulier la gestion des stocks qui coûtent chers aux entreprises.

Ce module gère les commandes clients et les livraisons. Il assure une liaison directe avec le compte de résultats et le système de production. Il permet l'optimisation des processus de workflow, la gestion précise des stocks, des contrôles qualités et factures.

Suite au contrôle qualité, il y a coordination et déclenchement des mesures correctives.

Enfin, il est possible de planifier/gérer/suivre la maintenance du matériel.

III.3 Le module e-commerce

Ce module permet de faire du e-commerce. C'est un logiciel de Gestion Relation Client (GRC) plus communément appelé Customer Relationship Management (CRM). Il permet d'effectuer les statistiques voulues (sous forme de requêtes) sur tous types de données (en particulier clients, vendeurs, production, fournisseurs,...). Un requêteur est disponible et il n'est pas utile que l'utilisateur connaisse le SQL.

Mais sachant le nombre très important de tables, celui-ci doit donc impérativement savoir "lire" un Modèle Conceptuel de Données (MCD) que les éditeurs mettent à disposition. Ensuite ses résultats sont analysés en fonction des besoins de l'entreprise. En particulier pour ce module,

un outil de reporting est associé. Enfin, ce module permet, suite aux résultats, d'effectuer de l'e-mailing, des offres marketing.

III.4 Résumé des Composants d' une RP

En résumé un ERP couvre les domaines suivants :

- la gestion comptable et financière (problème de double comptabilité, gestion de trésorerie) ;
- la gestion de la production : Gestion à l'ordre de fabrication (PROCESS MANUFACTURING), etc. ;
- La gestion des ressources humaines et la paie ;
- La gestion commerciale (dont les intégrations avec les systèmes de caisse, la configuration à la commande, etc.) ;
- La gestion des achats et logistique (appels d'offres, suivi qualité, gestion d'entrepôts, etc.) ;
- La gestion et suivi des projets (suivi à l'affaire, ordonnancement, etc.).

Les nouveaux domaines de déploiement des ERP/PGI :

- La gestion de la relation client ou Customer Relationship Management (CRM) ;
- La gestion de la chaîne logistique ou Supply Chain Management (SCM) ;
- La gestion des cycles de conception de produits ou Product Data Management (PDM) ;
- Les applications décisionnelles (BI, OLAP) et CORPORATE PERFORMANCE MANAGEMENT (CPM) ;
- E-APPLICATION
- E-PROCUREMENT, places de marché, applications en libre-service.
-

IV Avantages et inconvénients des ERP

IV.1 Avantages des ERP

Les ERP/PGI (opposés aux applications dédiées) présentent plusieurs avantages

- optimisation des processus de gestion ;
- cohérence et homogénéité des informations ;
- intégrité et unicité du Système d'information (un seul fichier articles, un seul fichier clients, etc.) ;
- mise à disposition d'un outil multilingue et multidevises (très adapté aux multinationales) ;
- communication interne et externe facilitée par le partage du même système d'information ;
- meilleure coordination des services et donc meilleur suivi des processus (meilleur suivi de commande ou meilleure maîtrise des stocks par exemple) ;
- normalisation de la gestion des ressources humaines (pour les entreprises gérant de nombreuses entités parfois géographiquement dispersées) ;
- maîtrise des coûts et des délais de mise en œuvre et de déploiement ;
- mise à disposition, des cadres supérieurs, d'indicateurs nettement plus fiables que lorsqu'ils étaient extraits de plusieurs systèmes différents ;
- partage du même système d'information facilitant la communication interne et externe ;
- minimisation des coûts (formation et maintenance) ;

- maintenance corrective simplifiée car assurée directement par l'éditeur et non plus par le service informatique de l'entreprise (celui-ci garde néanmoins sous sa responsabilité la maintenance évolutive : amélioration des fonctionnalités, évolution des règles de gestion, etc.) ;
- globalisation de la formation (même logique, même ergonomie) ;

Ce dernier point est essentiel et la mise en œuvre d'un ERP/PGI dans une entreprise est fréquemment associée à une révision en profondeur de l'organisation des tâches et à une optimisation et standardisation des processus, en s'appuyant sur le « cadre normatif » de l'ERP/PGI.

Les ERP/PGI vont pouvoir gérer et prendre en charge :

- plusieurs entités ou organisations (filiales, etc.) ;
- plusieurs périodes (exercices comptables par exemple) ;
- plusieurs devises ;
- plusieurs langues pour les utilisateurs et les clients (cas des multinationales) ;
- plusieurs législations ;
- plusieurs plans de comptes ;
- plusieurs axes d'analyse en informatique décisionnelle.

IV.2 Inconvénients des ERP

- Les ERP/PGI ne sont cependant pas exempts d'inconvénients :
- coût élevé (cependant, il existe des ERP/PGI qui sont des logiciels libres, les seuls coûts étant alors la formation des utilisateurs et le service éventuellement assuré par le fournisseur du logiciel) ;
- périmètre fonctionnel souvent plus large que les besoins de l'organisation ou de l'entreprise (le logiciel est parfois sous-utilisé) ;
- lourdeur et rigidité de mise en œuvre ;
- difficultés d'appropriation par le personnel de l'entreprise ;
- nécessité d'une bonne connaissance des processus de l'entreprise (par exemple, une petite commande et une grosse commande nécessitent deux processus différents : il est important de savoir pourquoi, de savoir décrire les différences entre ces deux processus de façon à bien les paramétrer et à adapter le fonctionnement standard de l'ERP/PGI aux besoins de l'entreprise) ;
- nécessité parfois d'adapter certains processus de l'organisation ou de l'entreprise au logiciel ;
- nécessité d'une maintenance continue ;
- captivité vis à vis de l'éditeur : le choix d'une solution est souvent structurant pour l'entreprise et un changement de PGI peut être extrêmement lourd à gérer.

V Le marché des ERP

Les ERP sont principalement destinés aux grandes entreprises ou multinationales du fait d'un coût important. Cependant, le marché des ERP tend à se démocratiser vers les PME/PMI. Certains éditeurs conçoivent un ERP uniquement pour ce type de structure.

Enfin, il existe des ERP open source ce qui revient moins cher, puisqu'il n'y a pas de coût de licence (ils sont gratuits). En revanche, il faut inclure dans le calcul du coût d'acquisition total, les frais de maintenance et l'assistance technique.

L'offre en matière d'ERP/PGI s'est considérablement étoffée mais aussi restructurée- ces toutes dernières années. Désormais, les principaux types d'offre, pouvant cumuler tout ou partie des caractéristiques des solutions alternatives sont les suivantes :

- une offre généraliste permettant de traiter « a-minima » de tous les besoins de l'organisation ;
- une offre généraliste mais également sectorisée par métier et/ou par fonction ;
- une offre de « niche », se distinguant par des caractéristiques fonctionnelles et/ou techniques complètement sectorisées.

Les solutions ERP disponibles sur le marchés sont classées en offre commerciale et en offre Open source.

V.1 Les offres commerciales

Les principales offres Cinq grands éditeurs dominent le marché des ERP/PGI :

- SAP avec SAP Business One, SAP S/4HANA,.
- Oracle Corporation avec : Oracle E-business Suite, Oracle NetSuite ERP, PeopleSoft.
- Microsoft avec en Europe avec Microsoft Dynamics Nav, Microsoft Dynamics GP.
- Sage avec : SAGE 100c, SAGE Intacct.
- LAWSON INTENTIA avec : M3 et S3

V.2 Les solutions Open Source

Les offres open source du marché comprennent:

- COMPIERE (Oracle, Java) ;
- Idempiere;
- Dolibarr ERP/CRM
- Apache OFBiz;
- Open ERP/Odoo (PostgreSQL, Python) ;
- Blueseer;
- Easy ERP
- ERP Next;
- WebERP
- OpenSI.

V.3 Types d'offre ERP

Type	Avantages	Inconvénients
ERP Standard	<ul style="list-style-type: none"> • Permet de formaliser et standardiser les processus • Marché concurrentiels • Proposé par de grands éditeurs 	<ul style="list-style-type: none"> • Nécessité des s'adapter aux processus définis dans l'ERP
ERP Spécifique	<ul style="list-style-type: none"> • Permet d'innover au niveau des processus • Permet de personnaliser et adapter les interfaces de l'ERP • Marché de niche 	<ul style="list-style-type: none"> • Développé par de petits éditeurs
ERP commerciaux	<ul style="list-style-type: none"> • Fiabilité • Pérennité • Assistance, support • Fonctionnalité, puissance, standardisation 	<ul style="list-style-type: none"> • Dépendance éditeur • Flexibilité, adaptabilité • Coûts, mise en œuvre

ERP Libre	<ul style="list-style-type: none"> • Agilité, flexibilité • Spécificité • Coûts • Maintenance • Mise en œuvre 	<ul style="list-style-type: none"> • Assistance • Support • Puissance • Pérennité de l'éditeur
ERP résident	<ul style="list-style-type: none"> • Maîtrise • Indépendance 	<ul style="list-style-type: none"> • Compétences nécessaires • Gestion des évolutions • Maintenance • Gestion opérationnelle • Rigidité (montée en puissance)
ERP hébergé	<ul style="list-style-type: none"> • Gestion des évolutions • Maintenance • Maîtrise des coûts • Disponibilité continue • Gestion opérationnelle • Ouverture 	<ul style="list-style-type: none"> • Intégration avec les S.I locaux • Contrat solide nécessaire

Un ERP est dit résident s'il est déployé chez le client. Certaines offres ERP sont fournies via le réseau. Il est disponible en mode SAAS (Software As A Service). Le logiciel est disponible en Cloud.

VI Mise en place des solutions ERP

VI.1 Gestion des projets ERP

Une étude du SYNTEC informatique a démontré que 62,5 % des entreprises ou des établissements considèrent leurs projets de renouvellement ou d'acquisition d'ERP/PGI comme plutôt insatisfaisants. Les principales raisons de cette importante insatisfaction sont les suivantes (elles peuvent se cumuler) :

- Des besoins, des attentes ou des objectifs mal définis (18%);
- Le déficit d'implication du management adéquat (11%);
- Des projets mal préparés ou mal conduits (8%);
- Des fonctionnalités inadaptées, sur ou sous dimensionnées (16%);
- Des utilisateurs insuffisamment associés au projet (9%);
- Des engagements mal définis, partiellement ou pas du tout respectés par les fournisseurs (fonctionnalités absentes,...) (13%);
- Des ressources humaines et/ou budgétaires qualitativement et/ou quantitativement insuffisantes (12%);
- Des technologies manquant de fiabilité ou difficiles à maîtriser (10%);
- La dimension stratégique du projet insuffisamment qualifiée et perçue (4%).

Pour ce faire un projet d'implémentation ERP est un projet d'entreprise et non un simple projet informatique qui nécessite une méthodologie. la mise en place d'une solution ERP

VI.2 Méthodologie de choix

La mise en place d'un système se fait en 10 étapes

1. **Créer la vision de l'entreprise** : il est question à ce niveau de définir la vision de l'entreprise sur une durée de 03 à 05 ans, les rôles et les responsabilités de chaque acteur, les priorités du projet, ses objectifs et le budget alloué
2. **Créer la liste des fonctionnalités recherchées**
3. **Classifier les fonctionnalités** : il sera question de classer les fonctionnalités en fonctionnalités classiques et en fonctionnalités critiques
4. **Créer une liste de candidats potentiels** : la sélection se fait sur les critères ci-dessous
 - Fonctionnalité et base technologique du système
 - Base d'implantation
 - recommandations faites par des amis ou une entreprise qui a déjà implémenté un ERP/PGI ;
 - Pérennité du fournisseur ;
 - Présence de l'éditeur du progiciel dans le pays ;
 - Prix, l'adéquation entre le progiciel et les processus métier de l'entreprise ;
 - L'ergonomie de l'ERP/PGI.

Réduire la liste des candidats à 04 voire 06 candidats sérieux à travers une présélection sur la base des critères ci-dessus.
5. **Créer le cahier de charges et envoyer l'appel d'offres**
6. **Analyser les soumissions**
7. **Sélectionner 03 finalistes**
8. **Démonstration des solutions par les finalistes**
9. **Sélectionner le système ERP**
10. **Négocier le contrat et planifier l'implantation**

La sélection et l'évaluation de solutions ERP/PGI est un processus complexe à cause, entre autres, de :

- la difficulté à définir les besoins précis de l'entreprise et à les traduire en termes de fonctionnalités et de performances ;
- la difficulté à évaluer l'adéquation d'un P.G.I. et des services associés avec les attentes réelles de l'entreprise : cela requiert l'analyse approfondie de centaines ou même de milliers de critères fonctionnels et techniques ;
- la difficulté à comparer les avantages des différentes solutions ERP/PGI envisageables ;
- la difficulté à appréhender les risques pris en cas de compromis (car finalement aucune des solutions n'est réellement idéale) ;

Ce processus de choix d'un ERP/PGI induit dans tous les cas la nécessité :

- de définir clairement les objectifs stratégiques et opérationnels du projet ;
- d'exprimer précisément et objectivement les vrais besoins fonctionnels ;
- d'évaluer méthodiquement toutes les solutions possibles ;
- de se doter des moyens adéquats de conduite de la démarche ;
- de s'assurer du soutien fort des niveaux appropriés de management ;
- de rester indépendant et objectif quelle que soit l'étape du processus ;
- d'appréhender les enjeux politiques et de gérer les conflits d'intérêts.

VII Les phases d'étude d'un projet ERP/PGI

Les 5 phases du projet ERP comprennent :

- 1) La phase préliminaire ;
- 2) La phase préparation ;
- 3) La phase cadrage ;
- 4) La phase mise en œuvre ;
- 5) La phase support & évolution.

VII.1 Phase préliminaire (opportunité)

- 1) L'analyse existant ;
- 2) La définition de la cible ;
- 3) La structuration des objectifs et gains envisagés ;
- 4) La détermination des stratégies d'évolution ;
- 5) L'évaluation des scénarii ;
- 6) La 1^{ère} évaluation budgétaire.

VII.2 Phase préparation

- 1) L'étude préalable ;
- 2) Le choix de l'ERP/PGI ;
- 3) L'urbanisation du Système d'information ;
- 4) Le cahier des charges des intégrateurs ;
- 5) La sélection de l'intégrateur ;
- 6) L'organisation du projet.

VII.2.1) L'étude préalable

- 1) La définition des objectifs ;
- 2) La définition du périmètre projet ;
- 3) La définition des besoins ;
- 4) La définition des principales interfaces ;
- 5) L'enveloppe budgétaire ;
- 6) Le ROI (Return On Investment) ;
- 7) L'approche de mise en œuvre ;
- 8) L'organisation de projet interne.

a) La définition des objectifs

- Rentabilité (Maîtrise des coûts) : obsolescence application, externalisation, gestion des risques ...
- Efficacité (Soutien à l'activité) : alignement SI avec processus métiers (développement produit, gestion client, logistique, REPORTING financier ...), externalisation ...
- Différenciation (Avantage concurrentiel) : stratégie de développement/acquisition/intégration (Exemple : Oracle), internationalisation, globalisation des marchés, externalisation.

b) 8.2.1.2 Le calcul du ROI

Des composantes quantitatives et qualitatives interviennent dans la phase d'étude préalable :

1) ROI Financier

- L'amélioration de la rentabilité par réduction des coûts à périmètre égal.

2) ROI Performance

- L'accroissement efficacité et flexibilité par la mise en oeuvre d'un SI «agile».

3) ROI Stratégique

- La création de différenciation en sélectionnant des axes stratégiques qui permettront au SI de devenir le levier de la compétitivité de l'entreprise.

Lors des choix d'évolution du SI, il faut effectuer une analyse de ROI.

Terminologie

SCM : Supply Chain Management, ou en français GCL (Gestion de la Chaîne Logistique) ;

CRM : Customer Relationship Management, ou GRC (Gestion de la Relation Client) ;

MRP : Manufacturing Resource Planning, ou GPP (Gestion et Planification de la Production) ;

CMS : Content Management System, ou SGC (Système de Gestion de Contenu).

SCPM: Supply Chain Process Management.

SCV : Supply Chain Visibility (Visibilité de la chaîne logistique).

CMMS : Computerized Maintenance Management System, ou GMAO (Gestion de la Maintenance Assistée par ordinateur).

SCEM: Supply Chain Event Management (Gestion événementiel de la chaîne d'approvisionnement).

ADD-ON : Composant logiciel permettant d'ajouter des fonctionnalités supplémentaires à un autre logiciel.

BI : Business Intelligence.

E-commerce : Commerce Électronique ;

ERP : Entreprise Ressource Planning ou Progiciel de Gestion Intégré (PGI).

CHAPITRE II LES EAI

I INTRODUCTION

Certaines entreprises ne peuvent abandonner en une seule étape leurs systèmes informatiques pour se convertir aux solutions intégrées de type ERP (risque trop élevé pour la continuité des opérations quotidiennes). D'autres part, les solutions ERP se révèlent très souvent très coûteuses, long à déployer, et entraînent souvent une refonte complète de systèmes d'information, la réalisation des objectifs économiques de l'entreprise passant parfois au second plan derrière l'accomplissement technique du projet. Enfin, il existe toujours des domaines de l'entreprise qui ne sont pas couverts par les ERP.

Les systèmes d'information complexes sont donc forcément hétérogènes en termes d'architectures (centralisée, client-serveur, Internet, etc.) de plates-formes (Mainframe, Windows, UNIX, etc.), de Bases de Données (Oracle, DB2, Sybase, SQLServer...), d'environnements de développement (C/C++, Java, ASP, Java), et ils utilisent des applications répondant à des besoins fonctionnels précis, certaines ayant fait l'objet d'un développement spécifique, et d'autres étant des solutions packagées qui ont pu nécessiter des investissements importants (ERP, CRM). De plus, les rachats, fusions ou réorganisations d'entreprises sont des facteurs qui vont accentuer cette tendance...

Il existe alors une solution intermédiaire: les EAI ou Enterprise Application Intégration. Ces logiciels intermédiaires (middleware) créent une interface ou un pont entre des applications différentes via des échanges de données restructurées selon un format de type XML.

II PROBLEMATIQUE DE L'INTEGRATION

Les systèmes d'information ayant atteint un certain stade de complexité sont confrontés à un problème classique: comment intégrer les applications entre elles?

Les solutions traditionnelles n'abordent le problème de l'intégration entre applications que par les données: transferts périodiques de fichiers, partage de base de données, réplication et transformation des données utilisées par les applications...

Ainsi sont développées des solutions d'intégration spécifiques capables de répondre rapidement au besoin d'intégration: les applications se parlent alors en face à face (on dit encore en "point à point") via des interfaces qui doivent être paramétrées et maintenues une à une: c'est l'approche «spaghetti». On obtient alors le schéma ci-dessous:

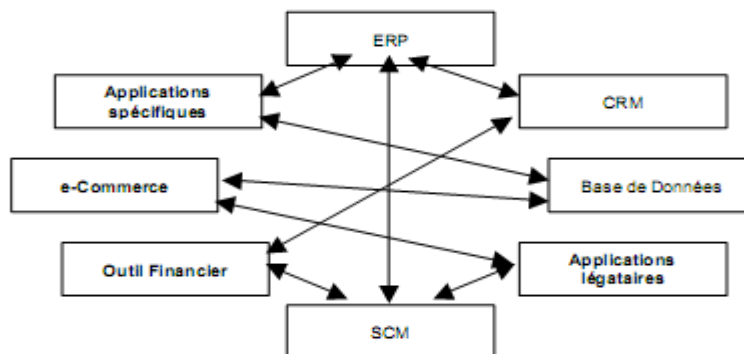


Figure4: Intégration nœud de spaghetti

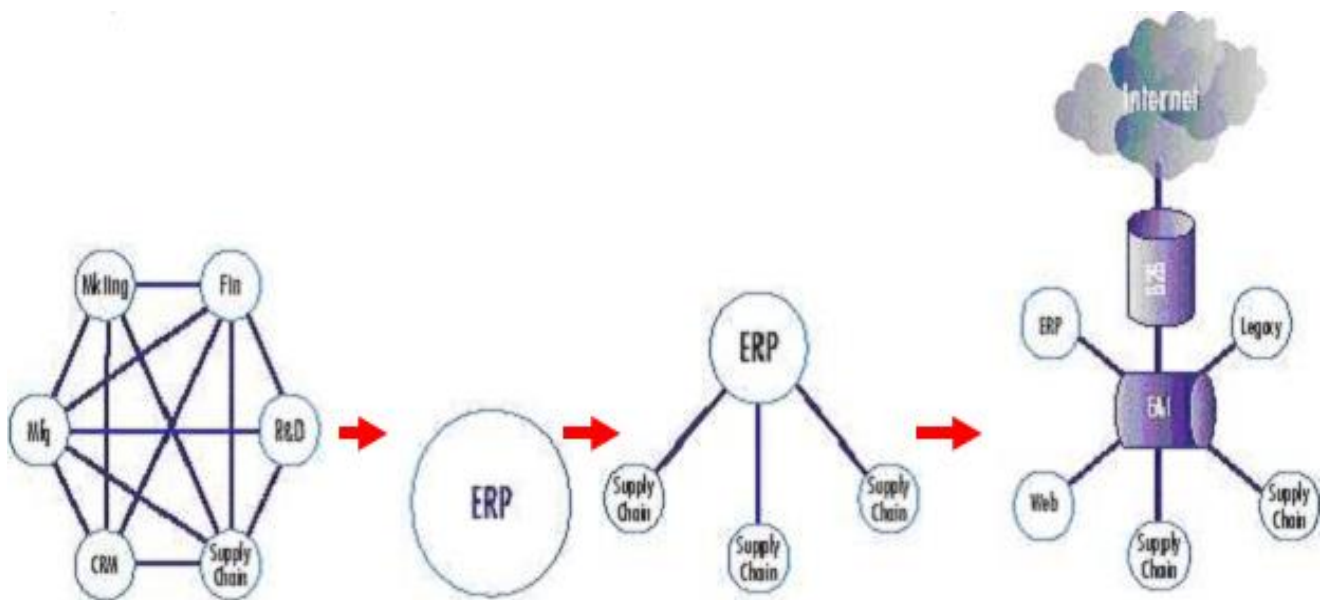
Par rapport à la logique de développement d'un nouveau système, cette approche est initialement peu coûteuse et rapide à mettre en œuvre, et a l'avantage de s'appuyer sur l'existant.

En revanche le nombre d'intégration point à point augmente de manière exponentielle lorsque de nouveaux systèmes doivent être intégrés ce qui alourdit l'administration et surtout la maintenance.

Une telle stratégie exige alors les programmeurs maîtrisant les divers protocoles de transmissions, des langages de programmation, les différentes plateformes de base de données, et débouche sur des coûts de programmation élevés, entraînant souvent des retards de projet et dépassement de coûts, une redondance du code d'intégration, quand des modifications de proses métier exigent les modifications correspondantes des applications, des coûts accrus et les inefficacités opérationnelles se développant comme le carré du nombre de systèmes intégrés.

La plupart de ces interfaces s'exécutent en mode batch, généralement pendant la nuit ceux-ci ne sont plus adaptés surtout quand on connaît le contexte commercial concurrentiel qui réclame de plus en plus de réactivité et une fluidité accrue de l'information.

D'après le cabinet Forrester Research, les entreprises confrontées à un environnement en constant changement consacrent jusqu'à 40% de leur budget informatique annuel aux passerelles entre leurs différents systèmes et applications. Le chantier d'intégration est donc un poste important du budget informatique des entreprises, et il doit privilégier la flexibilité et la pérennité du système intégré. Répondre simplement à cette problématique de réactivité, de productivité, de volumétrie, de sécurité et de fluidité de l'information est l'objet de logiciels classés dans la catégorie des outils liés au marché de l'EAI.



Figures: Evolution de l'intégration des solutions

III DEFINITION

Un EAI (entreprise Application Intégration) ou encore Intégration d'Application d'entreprise est un composant technique du système d'information destiné à industrialiser et à normaliser les échanges entre applications et de minimiser les interfaces. Plus simplement, les flux de données organisés de "point à point", c'est à dire directement d'une application à une autre, sont remplacés par des échanges dans des formats pivots entre chaque application et un médiateur: l'EAI.

L'EAI regroupe un ensemble de solutions techniques permettant à des systèmes informatiques de nature différente d'échanger des informations selon un processus normalisé. Elles vont prendre en charge les échanges entre des applications développées indépendamment et qui n'ont jamais été conçues pour s'entendre, de telle façon qu'elles fonctionnent comme une

seule (Ces applications peuvent utiliser des technologies incompatibles et rester indépendamment contrôlées).

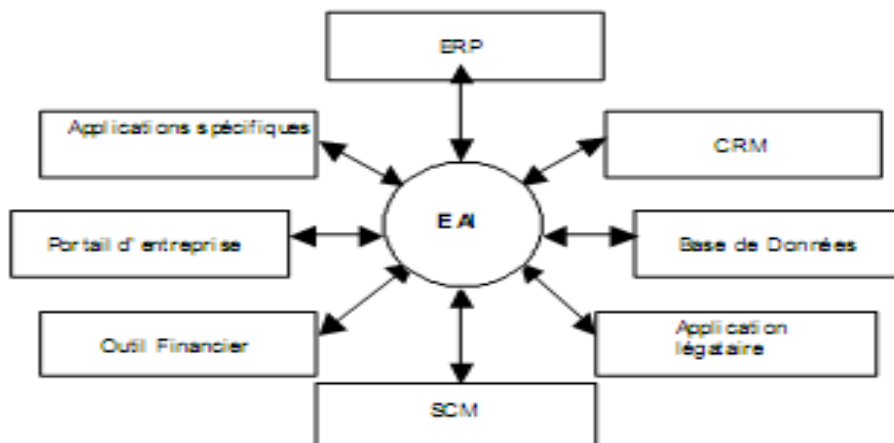


Figure6: Schéma d'interconnexion des éléments dans une EAI

Le nœud central, gère les interactions entre les applications, apporte une notion de découplage: grâce à l'utilisation d'un format intermédiaire de communication les liens (connecteurs) tissés entre chaque application sont maintenant remplacés par une liaison unique partant de l'application vers la solution d'EAI. Ainsi, pour cinq applications, il suffit de disposer de cinq passerelles, contre dix dans la version précédente. Ce nœud central assure ensuite la communication avec les autres applications.

L'EAI déporte et mutualise la problématique d'interfaçage (La logique métier est bien traitée par l'application dédiée qui la concerne, mais toutes les traitements tels que: Ordonnancement, Extraction, Transformation, Emission, Routage, Suivi, Réplication, Synchronisation, Remontée d'alertes... sont pris en charge et ont leur interface déporté dans l'EAI)

Il est à noter que L'EAI n'est pas un middleware. L'EAI n'est pas un ETL. L'EAI n'est pas un workflow. L'EAI est plutôt une combinaison des technologies utilisées dans ces différents produits. Une solution complète d'EAI utilise des services de connectivité fournis dans les produits de middleware, des services de transformation de données fournis dans les produits d'ETL, et des services de gestion de processus fournis dans les produits de workflow.

En effet:

- Les EAI diffèrent des middlewares orientés message (MOM) car ils intègrent la gestion de processus métier et la capacité à transformer des messages. Ils reposent sur les MOM existants pour faire collaborer les applications tout en respectant les processus définis par l'entreprise.
- Les EAI diffèrent des ETL qui sont une réponse à la mise en cohérence (harmonisation des référentiels, nettoyage, traitement des rejets et constitution d'indicateurs et d'agrégats) des flux provenant de systèmes source hétérogènes, permettant un traçage logique des informations (piste d'audit). Ils sont particulièrement indiqués au traitement de volume de masse en mode batch.
- Les EAI diffèrent des moteurs de workflow, qui ciblent principalement les processus longs impliquant des actions humaines et reposent sur une base de données pour faire progresser ces processus. Les moteurs de workflow peuvent être utilisés pour automatiser la collaboration avec les partenaires, mais pas entre les applications. Ils ne fournissent pas une gamme de connecteurs aussi large que les progiciels d'EAI.

Un EAI vise les objectifs suivants:

- Intégration du front-office et back-office
- intégration des nouvelles applications à l'existant
- synchronisation des données dispersées
- maîtrise de l'évolution du système d'information
- baisse des coûts et délais d'insertion de nouvelles applications au sein du SI
- baisse de coûts de maintenance
- baisse des coûts de traitement de l'information

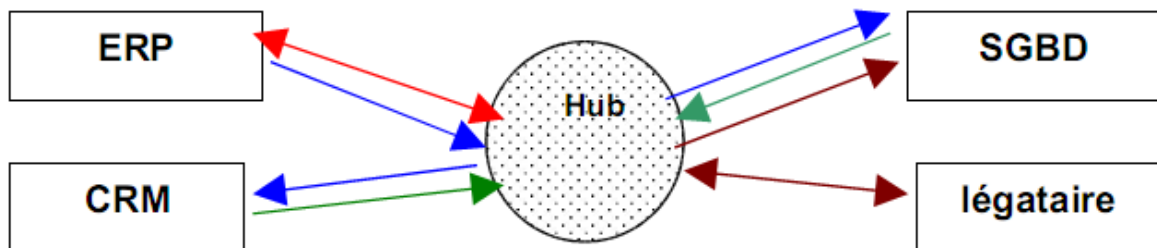
La mise en œuvre d'un EAI est une solution de la démarche d'urbanisation du système d'information permettant de structurer et normaliser les échanges de données.

IV Architectures des EAI

IV.1 L'architecture Hub and Spoke

C'est le modèle centralisé de l'EAI. Ici, tout passe par un "hub" central qui concentre les services sur un seul serveur. Aucun flux n'est possible sans l'entremise de ce hub. Quand une application envoie un message, ce dernier est expédié à destination du hub. Le référentiel (la base où sont stockées les règles de routage et de transformation) est donc lui aussi centralisé.

L'avantage d'une telle architecture saute aux yeux: l'administration est grandement facilitée. En revanche, la gestion de la charge s'avère complexe dans ce type d'environnement: la seule solution consiste en effet à multiplier les hubs sur les différents segments du réseau, sachant qu'il faudra veiller à synchroniser les règles stockées sur ces différents nœuds.



IV.2 L'architecture Bus Adaptatif

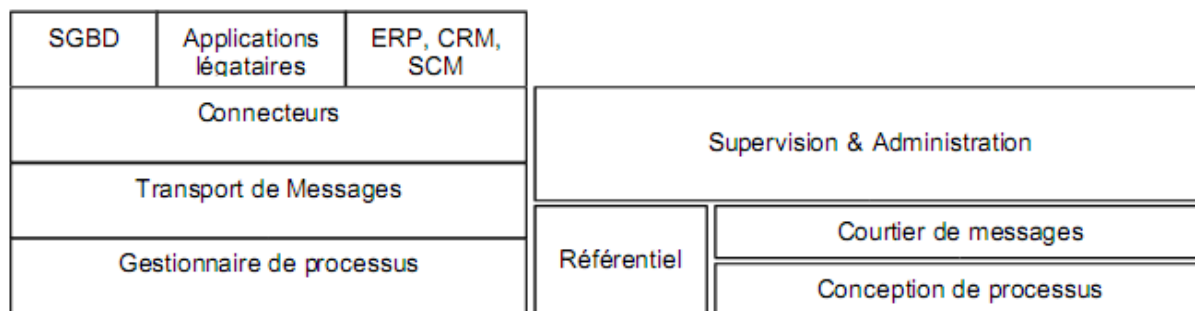
Il s'agit cette fois de la version décentralisée de l'implémentation de l'EAI : l'architecture « bus applicatif » distribue les services sur plusieurs serveurs. Des référentiels de règles et des gestionnaires de messages sont disséminés sur l'ensemble des nœuds (point de connexion à une application). Quand une application émet un message, ce dernier est traité par le référentiel du nœud correspondant afin que les applications abonnées à ce type de messages le reçoivent. Avec ce type d'architecture, la charge est donc répartie sur l'ensemble des nœuds.

Avec l'accroissement de l'environnement (systèmes, applications, règles, utilisateurs, volume de transactions, etc.), le modèle « Bus » offre potentiellement de meilleures performances que le modèle « Hub », mais sa mise en œuvre est plus complexe, et est plus difficile à administrer.



V Architectures fonctionnelle des EAI

L'architecture fonctionnelle d'un EAI est la suivante :



V.1 Transport de données

Cette couche repose sur un middleware généralement asynchrone classifié comme MOM (Message Oriented Middleware). En effet il est constitué de files d'attente de messages (message queues), et offre un certain niveau de tolérance de panne (ou de tolérance d'applications lentes) : un message n'est pas perdu lorsqu'une application n'est pas prête à le recevoir. De plus c'est une couche logicielle non bloquante : l'application émettrice du message redevient immédiatement disponible. Il permet la communication par publication / abonnement.

Des EAI proposent aussi des canaux de communications synchrones (requête-réponse), bas sur des ORB (Object Request Brokers) comme Corba, DCOM, RMI-Java, voire sur des protocoles TCP/IP comme http, qui assurent les appels de procédures à distance d'une machine vers une autre en utilisant la sémantique des technologies objet.

V.2 Courtier de messages (Message Broker) ou Moteur d'Intégration

La couche Transport de Données MOM a besoin pour fonctionner d'un outil d'administration des messages transportés : le Message Broker. Dans une solution d'EAI, ce dernier tient le rôle de moteur (serveur) d'intégration.

C'est le chef d'orchestre de l'EAI : son rôle est d'administrer les règles de routage des données, les règles de transformation et de traitement.

Il permet de conserver les informations pour assurer les échanges asynchrones entre applications. Il comprend un gestionnaire de files d'attente qui gère les abonnés des applications aux messages, un gestionnaire de message qui gère les messages échangés entre les applications, et un moteur de transformation utilisé pour convertir les messages d'un format d'application à un autre. Ces modules permettent d'effectuer des traitements de manière transparente sur les informations échangés à travers l'EAI.

V.3 Adaptateur applicatif (Connecteurs)

L'extraction d'informations depuis les applications hétérogènes et l'insertion de nouvelles informations dans ces applications est réalisée par l'utilisation de connecteurs, qui se « branchent » sur les applications.

Il s'agit d'un composant logiciel qui offre la connectivité nécessaire à l'interfaçage avec les applications et les sources de données, avec ou sans intelligence métier. Cette couche logicielle masque à l'utilisateur la complexité de la communication entre l'API (interface de programmation) du moteur d'intégration et celle de l'application.

Ces connecteurs peuvent fournir des services complémentaires tels que la gestion des exceptions ou des mécanismes de remontée d'erreurs.

V.4 Référentiel (Repository)

C'est une base de données qui contient toutes les définitions des structures des données ou métadonnées échangées, ainsi que les règles de transformation et de routage.

Les formats de messages et les règles de transformation de ces messages sont stockés dans des référentiels uniques au niveau de l'outil d'EAI pour faciliter leur maintenance.

V.5 Gestion des processus

Le gestionnaire de processus pilote les processus d'intégration. Il contrôle l'exécution et le cadencement des processus métiers au sein d'un moteur de workflow associé à une base de données. Il permet aussi les modéliser et de les faire évoluer. La logique de gestion des flux inter-applicatifs est alors extraite des applications qui peuvent évoluer chacune indépendamment avec un minimum d'impact sur le reste du SI.

Il communique avec les connecteurs à travers une couche de transport de message qui gère les communications entre les connecteurs.

V.6 Supervision et Administration

Ce service fait le lien entre la solution d'EAI et les outils d'administration du système. Il suit les flux et s'assure de la qualité du service. Il gère aussi les utilisateurs, paramètre le système et assure la sauvegarde.

Les outils d'administration aident le service informatique à superviser et configurer le niveau de service des connecteurs et des gestionnaires de processus.

VI Les ESB

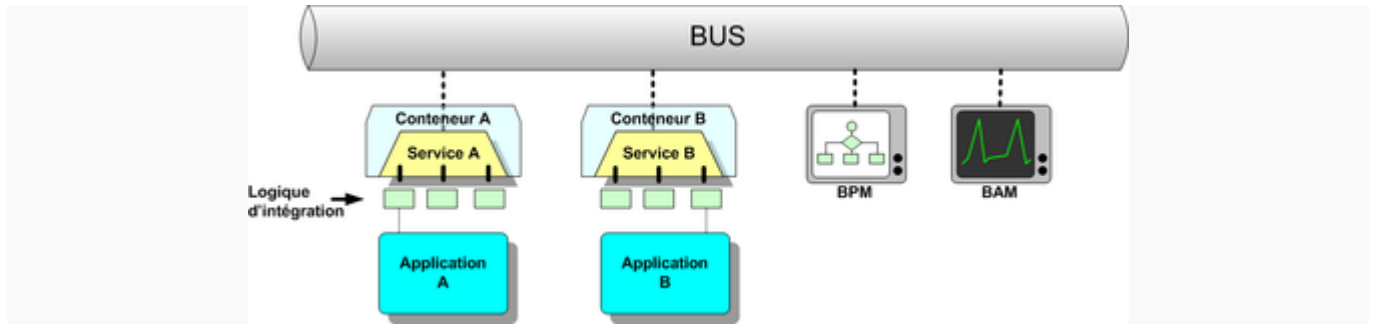
L'ESB peut être considérée comme une nouvelle génération d'intégration d'applications d'entreprise (EAI) construite sur des standards comme XML, Java Message Service (JMS) ou encore les services web. La différence majeure avec l'EAI est que l'ESB propose une intégration complètement distribuée grâce à l'utilisation des conteneurs de services. Ces « mini-serveurs » contiennent la logique d'intégration et peuvent être déposés n'importe où sur le réseau.

L'ESB en tant que médiateur entre les clients et les fournisseurs de services s'appuie sur les principes suivants :

- La découverte dynamique : les services ainsi que la sémantique associée sont enregistrés dans un annuaire partagé.
- La chorégraphie des processus métiers et l'orchestration des services associés : un outil permet d'orchestrer automatiquement les services nécessaires à l'implémentation des processus collaboratifs représentés graphiquement.
- La distribution forte : les services sont distribués sur le réseau de l'entreprise ou sur Internet.
- La communication par messages : les services s'échangent des messages représentés par des documents textuels.

D'une certaine façon les ESB constituent une évolution de l'EAI, notamment pour des raisons de performance en cas de grosses volumétries (les traitements sont potentiellement distribués) et de sécurité (éviter le 'Single Point of failure' potentiel d'une plate-forme centrale), bien que les EAI/ESB les plus récents se soient améliorés sur ces deux problématiques.

L'entreprise service bus possède comme son nom l'indique une architecture en bus qui l'oppose à l'architecture hub and spoke des premiers EAI. Ceci fait de l'ESB une solution hautement distribuée. Les composantes de cette architecture sont illustrées sur la figure suivante.



L'ESB possède quatre fondements :

- Le message-oriented middleware (MOM) qui permet l'échange de messages de manière asynchrone. Ainsi chaque message est déposé sur une file d'attente avant d'être consommé par le destinataire.
- Les services web qui permettent d'interfacer les applications avec le bus. Chaque service contient une logique d'intégration des messages (transformation, routage, etc.).
- Les transformations qui concernent les messages circulant sur le bus, elles sont essentielles dans un ESB car leur rôle est de permettre à des applications de converser même si elles définissent différemment leurs données.
- Le routage intelligent qui découple l'expéditeur du message de son destinataire. C'est en fait l'ESB qui va déduire la destination du message. Pour cela il se fonde sur le contenu du message et les règles qui ont été définies.

Peuvent venir se greffer à cette architecture :

- Le business activity monitoring (BAM) qui permet de suivre l'activité d'un processus métier.
- La procédure d'entreprise (BPM) qui a pour but de maîtriser l'orchestration des processus métier, c'est-à-dire l'enchaînement des échanges entre applications.

VII Quelques produits d'intégration

- | | |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <ul style="list-style-type: none"> • Apache Camel • Apache Synapse • Blackbird ESB • ChainBuilder ESB • FUSE ESB - Enterprise • ServiceMix • Jitterbit • Mule ESB • Openadaptor | <ul style="list-style-type: none"> • Microsoft : BizTalk Server 2000 • NetFish (racheté par Iona) : XDI System • Neon (racheté par Sybase): eBiz Integrator • Sopra : eTalk • STC : e*Gate Integrator • Tibco : Active Enterprise • Vitria : BusinessWare • webMethods : webMethods Enterprise • IP Net, XML Solutions, Extricity ... |
| <ul style="list-style-type: none"> • Active Software : ActiveWorks Integration System • BEA Systems : eLink • Bowstreet : Business Web Factory • CrossWorlds : Interchange Server • IBM : EAI MQSeries Integrator et WebSphere B-to-B Integrator • Mercator : Mercator E-Business Integration Broker | |

WEBOGRAPHIE

- [1]www.seralia.com
- [2]http://proxy.siteo.com.s3.amazonaws.com/www.intraland.fr/file/pourquoiEAI_1.pdf
- [3]<http://solutions-entreprise.developpez.com/solutions-entreprise/cours/>
- [4]http://cigref.typepad.fr/cigref_publications/RapportsContainer/Parus1999/1999_Retour_d_experience_ERP.pdf
- [5]<http://cernierg.free.fr/CV/projets/INPG%202006-2007%20-%20Cas%20ERP%20-%20complement%20de%20cours.pdf>
- [6]<http://fablain.developpez.com/tutoriel/presenterp/>
- [7]<ftp://ftp-developpez.com/fablain/pdf/presenterp.pdf>
- [8]<http://www.iut2.upmf-grenoble.fr/UserFiles/File/formation%20continue/modules/informatique/proiciel-gestion-integre-pgi-erp.pdf>
- [9]http://www.univ-angers.fr/docs/etudquassi/MPo8_03.pdf
- [10]<http://solutions-entreprise.developpez.com/solutions-entreprise/cours/#erp>
- [11]<http://www.trivadis.com>