ESB : le liant efficace des applications

vu ici : http://www.01net.com/article/226666.html

ESB : le liant efficace des applications

Facilité de mise en oeuvre, fiabilité, ouverture... l'ESB fournit aux PME un bus applicatif standardisé en remplacement des coûteux produits d'EAI traditionnels.

Julien Lesage , Décision Micro (n° 574), le 15/12/2003 à 00h00

Fin 2002, le terme ESB (Enterprise Service Bus) fait irruption dans le vocabulaire des analystes et désigne une nouvelle approche en matière d'intégration d'applications, exclusivement fondée sur les standards du moment : services web, MOM (middleware orientés messages) au standard JMS (Java Message Service) et connecteurs JCA (Java Connector Architecture). « L'ESB n'invente aucune technologie et se contente d'assembler des solutions existantes, fiables et éprouvées », affirme Marc Delfau, directeur technique de Fiorano France, distributeur de la solution ESB FioranoBIS. De fait, avant la naissance du sigle, plusieurs éditeurs avaient déjà franchi le pas et proposaient des solutions d'intégration plus ou moins complètes exploitant ces standards. Aujourd'hui, l'ESB représente un véritable marché et constitue une alternative de poids aux approches traditionnelles, notamment d'un point de vue économique, puisque les produits se recommandant de cette mouvance sont proposés en moyenne au cinquième du prix des offres traditionnelles d'intégration d'EAI. « L'engouement du marché pour les solutions de type ESB s'explique sans doute par les écueils vécus par les grandes entreprises dans leurs tentatives de mise en oeuvre des offres traditionnelles, onéreuses et complexes à administrer » explique François Peloffy, directeur général de 4Tier, éditeur du middleware OpenMOM. aux solutions ESB, les PME peuvent désormais accéder à une technologie efficace pour créer un véritable bus applicatif standard et remplacer leurs interfaces construites sur un modèle point à point », déclare Florent Lefevre, directeur général de Sonic France, fournisseur de SonicESB. Considéré par certains comme la phase ultime des architectures d'échanges interapplicatifs, l'ESB semble répondre aux attentes d'un marché épuisé par la complexité des projets d'intégration. Mais, quelques questions demeurent : les standards sur lesquels repose l'ESB sont-ils réellement matures ? L'argument économique mis en avant par les défenseurs de l'ESB est-il pertinent sachant que l'acquisition des outils ne représente qu'une faible part du coût global d'une solution d'intégration ? Enfin, les équipements sont-ils vraiment capables de rivaliser avec leurs aînés ou avec les développements spécifiques en mode point à point, largement décriés mais souvent très performants ?

L'utilisation : adapté aux projets non stratégiques

En général, les solutions de type ESB font leur entrée dans les grands comptes par la petite porte. Ces offres sont en effet proposées à des prix attractifs qui permettent de réaliser des maquettes réalistes, dans le cadre de petits projets. « Ces tâtonnements aboutissent souvent à de véritables projets d'intégration, même s'ils sont limités à des ressources no critiques de l'entreprise », affirme Marc Delfau. Toutefois, les grandes entreprises sont confrontées à des contraintes que la technologie ESB ne peut traiter : comment intégrer efficacement une transaction CICS au coeur d'une chaîne EAI standard, tout en garantissant une qualité de services équivalente à celle obtenue avec des connecteurs spécifiques ?

Le prix des solutions ESB permet également aux PME d'accéder à une véritable technologie d'intégration et de relancer des projets gelés pour des raisons économiques. Pour ces entreprises, plus que jamais, l'intégration des applications revêt un caractère stratégique. « La réussite de notre projet d'intégration conditionnait directement la mise en oeuvre de notre stratégie commerciale », commente Cristof Thomas, directeur informatique et marketing produits du loueur de voitures DCS Fleet, avant de poursuivre : « Il s'agissait pour nous de donner la possibilité à nos clients de personnaliser leurs services en fonction de leurs besoins spécifiques. Nous souhaitions également réduire les problèmes naissant de la disparité des informations maintenues au travers de multiples systèmes, sans synchronisation. Concrètement, nos utilisateurs passaient plus de temps à tenter de maintenir la cohérence de nos systèmes plutôt qu'à répondre aux attentes des clients. »

Les gains : simplicité technique et réduction des coûts

L'utilisation du langage XML comme format d'échange de données au coeur des offres ESB recueille tous les suffrages, d'autant que l'extensibilité et la lisibilité de ce langage, associées à la possibilité de valider les documents par des DTD (Data Type Definition) ou des schémas ont déjà séduit de nombreux éditeurs de solutions applicatives. « Parmi les applications que nous devions intégrer, la plupart disposaient de fonctions d'import/export de données au format XML. Le premier intérêt des solutions ESB était donc de nous permettre d'utiliser directement ce langage comme format d'échange, en appliquant les nécessaires opérations de transformation de données » explique Jérôme Kozlowski, directeur adjoint des systèmes d'information de l'hôpital de Longjumeau.

Autre avantage de taille, la traçabilité des opérations. « Lorsqu'un problème survient, se pose la délicate question de s localisation au coeur de la chaîne EAI. Les ESB ne sont pas des boîtes noires parce qu'ils mettent en oeuvre des techniques standardisées. FioranoBIS dispose de fonctions pratiques pour visualiser la chaîne EAI et identifier aisément la cause du problème. Ces fonctions d'administration nous permettent également de mieux piloter la solution et d'adapter dynamiquement notre contexte de production répartition de charges, allocation des ressources systèmes... », poursuit Jérôme Kozlowski.

Naturellement, l'introduction d'une solution de type ESB au sein d'une entreprise doit être économiquement justifiée. Pour Cristof Thomas, de DCS Fleet, les bénéfices résultant de l'utilisation d'un tel produit doivent pouvoir être chiffrés concrètement « Grâce à l'approche ESB nous avons défini un nouveau standard de performance et de qualité en termes de traitement de l'information pour la gestion des flottes automobiles. Nous avons par exemple réduit de 85 nos temps de traitement lors de la création d'une commande, notre production annuelle a augmenté de 30, à effectifs constants, et nous réalisons aujourd'hui des économies substantielles dans divers domaines, notamment plus de 75 000 euros dans la production d'états destinés à nos clients ».

Les écueils : absence de normes

Par définition, l'ESB doit exploiter exclusivement des normes et des standards du marché. Dans le domaine de l'intégration d'applications, l'absence de normes en matière d'orchestration des processus, de qualité de services et de sécurisation des échanges de flux XML retarde la pénétration des ESB au sein des grands groupes. « Le manque de normes relatives à la transmission de flux XML à la fois fiable et sécurisée pénalise fortement le développement des EAI à base d'outils ESB pour les échanges interentreprises », explique François Peloffy, directeur général de 4Tier. Ces limitations cantonnent aujourd'hui les solutions ESB à des projets non stratégiques au sein des grands groupes. Habib Guergachi, cofondateur du cabinet de conseil en architecture Infhotep, va encore plus loin et considère que « l'ESB est un concept liquide, qui noie les clients sous le poids de conjectures marketing, abritées derrière des standards technologiques XML, services web... Les ESB répondent à une problématique d'architecture d'intégration simple, fondée sur les données et le langage XML comme vecteur d'échanges. Mais, pour gérer des transactions complexes, longues et fiables, difficiles à intégrer parce que vieillissantes, il convient de mettre en oeuvre des batteries de connecteurs et des fonctions EAI sophistiquées ».

Tous les produits qui se recommandent de la mouvance ESB implémentent une couche transport qui exploite le standard JMS. Les performances d'une chaîne d'intégration dépendront fortement de la qualité de cette implémentation. « Nous avons testé les performances des moteurs JMS de plusieurs solutions ESB », explique Al Seigneur, professeur d'informatique à l'université Paris V. « Les résultats de nos tests affichent des écarts difficiles à expliquer », conclut-il. Les entreprises intéressées par ce type de solutions devront donc impérativement réaliser des tests de charge réalistes avant de faire leur choix. Enfin, la qualité de la documentation fournie avec les produits n'est pas toujours à la hauteur de l'offre qu'elle décrit.

Mise en oeuvre : projet pilote de rigueur

En théorie, l'utilisation d'un produit de type ESB requiert simplement la maîtrise des souches technologiques sur lesquelles repose cette nouvelle génération de solutions EAI, notamment le langage XML et le standard JMS. « En pratique, il convient de se faire accompagner par l'éditeur et de gérer efficacement un transfert d'expertise des équipements », précise Jérôme Kozlowski. Le manque de compatibilité entre les logiques d'intégration des solutions traditionnelles d'EAI crée une dépendance entre entreprises et éditeurs, imposant aux premières de faire appel à des consultants spécialisés, dont les tarifs sont souvent élevés. Ces prestations d'accompagnement représentent un investissement lourd, susceptible d'atteindre cinq fois le prix des licences de base. « La réduction des tarifs pratiqués par les éditeurs d'ESB, associée à l'utilisation de standards maîtrisés par les sociétés de services, contribue à réduire le coût global d'intégration » explique Florent Lefevre, directeur général de Sonic France.

Les produits de type ESB semblent bien adaptés à la mise en oeuvre de projets EAI de taille moyenne, concernant des chaînes d'intégration qui impliquent des ressources non critiques. Ils intéressent également les PME pour leur faible coût d'acquisition et les standards qu'ils exploitent, notamment XML, de plus en plus utilisé comme format d'import/export de données au coeur des solutions métier. Mais les ESB présentent des faiblesses inhérentes à l'absence de normes concernant l'orchestration des processus et la sécurisation des échanges de flux XML. Avant de faire un choix définitif, il convient de procéder à un test de performances (moteur JMS principalement).

Coût d'acquisition

Le coût global de la mise en oeuvre d'une solution de type ESB reste très inférieur à celui des solutions traditionnelles d'EAI. Cette économie résulte d'une part des prix relativement bas pratiqués par les éditeurs d'ESB et d'autre part de l'utilisation de technologies standards, qui contribue efficacement à réduire le montant de la facture lorsque l'entreprise fait appel à des compétences externes pour développer ses projets d'intégration. Le respect de

Parmi les technologies sur lesquelles repose l'approche ESB, XML fait l'unanimité. L'adoption généralisée de ce langage contribue à simplifier les échanges entre l'entreprise et ses partenaires. L'exploitation des services web facilite

l'intégration de composants applicatifs hétérogènes. Le middleware JMS permet de passer aisément d'une architecture en mode point à point à un bus d'échanges fonctionnant en mode dynamique. Une approche e

L'ESB souffre de l'absence de normes définitives, notamment en matière de description de processus (orchestration, gestion des exceptions et des erreurs...) et de sécurité, surtout lorsque les échanges de données impliquent l'entreprise et ses partenaires. Dans l'attente de l'émergence de normes définitives, les éditeurs utilisent des solutions transitoires, une concession de poids par rapport aux objectifs initiaux de l'approche ESB. L'int

L'ESB repose sur un système d'échange de messages éprouvé : le middleware orienté message (MOM). Les solutions développées en Java, et qui respectent la norme JMS, affichent en général des performances répondant aux attentes des entreprises les plus exigeantes. L'utilisation de JMS en interne par les différents composants de l'outil ESB constitue également un gage de performances et d'ouverture. Le triplet JMS, JCA, services web facilite grandement l'intégration d'un bus applicatif de type ESB avec un existant, la plupart des éditeurs de produits d'EAI fournissant déjà des solutions de connectivité compatibles avec ces normes.

vu sur : http://www.tibco.com/international/france/software/enterprise_service_bus/default.jsp

Solution ESB

Un bus de services d'entreprise (ESB) est une couche de communication normalisée dans une architecture orientée services (SOA) qui permet aux services d'être utilisés par de multiples protocoles de communication. Il simplifie le déploiement et l'administration des services et favorise la réutilisation desdits services dans les environnements hétérogènes. Pour fournir ces fonctionnalités, les ESB prennent en charge les standards ouverts et les technologies propriétaire, notamment les services Web et les registres UDDI, afin de détecter et publier les services, Java Message Service (JMS) ainsi que d'autres protocoles de messaging, les transformations standards (XML) et le routage des messages.

TIBCO ActiveMatrix BusinessWorks™ intègre tous les composants d'un ESB (services Web, XML, SOAP, transformations XSLT et prise en charge UDDI) et bien plus encore. TIBCO offre une prise en charge indépendante de la plate-forme pour les environnements informatiques et SOA hétérogènes, ce qui vous permet de ne pas dépendre de l'environnement d'un seul fournisseur. TIBCO BusinessWorks exploite l'architecture Information Bus™ de TIBCO pour offrir des performances, une capacité d'évolution, une fiabilité et une sécurité de très haut niveau. Depuis 20 ans, TIBCO est la seule société à avoir aidé des milliers de clients à améliorer la flexibilité de leurs processus métier.

Avantages de la solution ESB de TIBCO

- Réduit le coût de la connectivité dans les environnements hétérogènes en supportant les standards ouverts existants et le large éventail de protocoles de communications disponibles, ainsi que les applications et technologies aujourd'hui déployées dans les techniques de l'information.
- Augmente la productivité du développement et des opérations en remplaçant le codage personnalisé et l'administration manuelle par un environnement de services intégrés (ISE) qui offre une approche graphique centrée sur les processus pour l'assemblage de services, le testing, le déploiement et l'administration. L'accroissement de la flexibilité et de la productivité améliore la réactivité face à l'évolution des besoins métier et réduit le temps nécessaire au développement et au déploiement de nouvelles fonctionnalités et leur mise sur le marché.
- Fournit des performances, une capacité d'évolution, une fiabilité et une sécurité éprouvées pour les systèmes critiques. Des milliers de clients de par le monde font confiance à TIBCO pour construire leur infrastructure. Grâce à TIBCO BusinessWorks, les clients peuvent supporter des milliers de messages par seconde et bénéficient d'un taux de disponibilité de 99,999%.

vu sur : http://enterpriseservicebus.blogspot.com/2007/08/prsentation-de-la-solution-esb.html

Présentation de la solution ESB

Voici une présentation de notre architecture SOA développée sur base d'un ESB Open Source, ServiceMix.

Contraintes techniques:

- 2500 applications Client dans un premier temps puis à terme 3500 vont appeler nos services web. En moyenne 1000 applications Client accèderont en simultané à nos services web à valeurs ajoutées.
- Taux de charge: environ 1,5 millions d'appels de 30-50ko de données en in-out dans un premier temps à traiter en 3-5 jours.

- les services web à valeurs ajoutées exposées sur Internet (on-line) doivent pour certains être fortement sécurisés avec non-répudiation.
- certains services web à valeurs ajoutées retournent un nombre de données important.
- les applications métiers seront accessibles via des services web sur l'intranet.
- certains services web à valeurs ajoutées fourniront la réponse en synchrone d'autre en asynchrone suivant les besoins.
- tous les services doivent être impérativement disponibles en 24/7.
- certains back-end - 80 applications back-end aujourd'hui puis 120 à terme - sont peu performants en terme de capacité et donc l'ESB devra être capable de "réguler" la charge sur les back-end.

- dans le cadre de la sécurité, la validité des certificats qui ont permis de signer les messages devra être vérifiée.
- les applications Client sont soit des applications serveurs (application standalone de type "Batch" ou alors des applications web) soit des applications Client lourd ou léger installées sur des postes clients (personal computer).

Architecture générale:

L’ESB est composé de trois entités principales.

- (SMX-CAS) La première a la responsabilité de gérer les aspects de sécurité, de génération d'ID (estampillage les messages) et d'enrichissement pour récupérer les données permettant le contrôle d'accès aux données. Elle est capable de fonctionner en synchrone ou en asynchrone. En asynchrone elle dépose les messages dans une queue et en synchrone elle fait appelle directement à la deuxiéme entité principale.
- (SMX-WORK) La deuxième entité s’occupe de la partie traitement proprement dite : validation du schéma XSD, appels aux différents WS back-end, gestion des erreurs... Les concepts SOA seront implémentés dés cette entité - jusqu'au back-end évidement ;-) .
- (SMX-POLLING) La troisième entité est utilisée par les applications Client pour récupérer les réponses lors des appels asynchrone.

Les deux entités SMX-CAS et SMX-WORK sont connectées via un broker JMS qui permet d'assurer la persistance des messages ainsi que la régulation (notion de bassin d'orage) de la deuxième entité. Remarque: la régulation "joue" sur le nombre de listeners JMS qui démarrent les processus dans SMX-WORK pour régler la charge de manière indirecte sur les back-end.

Remarque: SSL acc = SSL accelerator

Gestion de la charge dans l'infrastructure:

Les load-balancer (LB) en entrée permettent d'assurer la disponibilité des services en répartissant le nombre des appels sur deux SMX-CAS. En cas de dépassement d'un seuil (prédéfini) une ou deux instances SMX-CAS sont automatiquement démarrées pour fournir en tout 4 SMX-CAS au LB.

Le Broker JMS en cluster réceptionne les messages du SMX-CAS et les met à disposition de SMX-WORK. Si cet élément est surchargé alors toute l'infrastructure perdra en performance. D'où en cas de détection d'un temps de réponse trop élevé, une instance du broker JMS sera démarré automatiquement et entrera comme acteur dans le cluster.

Pour avoir une infrastructure performante, il faut que les messages en attente dans le broker JMS doivent rester le moins longtemps possible. De ce fait il est prévu que lorsqu'un nombre de messages en attente dans le broker JMS est dépassé une instance supplémentaire de SMX-WORK est démarrée.

Biensûr toutes les instances en stand-by peuvent être démarrées manuellement dans l'hypothèse d'une anticipation de charge.

Gestion de la disponibilité de l'infrastructure:

En plus d'une architecture redondante nous utilisons les fonctions de disponibilité du logiciel de virtualisation VMWare. Il permet de déplacer des instances d'un serveur physique à un autre (par exemple si une carte réseau est défaillante) avec un léger temps de latence.

Boite à outils:

L’ESB dispose d'outils annexes :

- deux outils de développement graphique : Cimero et Intalio Designer
- une console d’administration, de monitoring et d'audit