

INFRASTRUCTURE.

Serveur d'applications : Java EE inspire Microsoft

La firme de Redmond vient de dévoiler les caractéristiques de son futur middleware. Qui, incontestablement, ressemble beaucoup aux piles d'infrastructures Java...

LES FAITS

Microsoft a présenté la semaine dernière sa vision des SOA. Il se repositionne dans l'infrastructure avec son serveur d'applications Dublin dont les premières versions bêta arriveraient cette année. Et il a précisé son projet Oslo, centré sur l'exécution de modèles.

L'ANALYSE

Parlez à un puriste de Java du serveur d'applications de Microsoft, et il esquissera un sourire : « Lequel ? Windows Server ? » C'est pourtant lui que la firme de Redmond oppose aux serveurs Java EE. Beaucoup refusent cette comparaison, puisque les services proposés tournent au sein même du système d'exploitation et non sur un conteneur indépendant. D'ici 2010, il faudra peut-être réévaluer la situation. Microsoft va lancer un « vrai » serveur d'applications, Dublin. Adossé au framework .Net 4, prévu à la même date, il copiera les fondamentaux des piles Java actuelles.

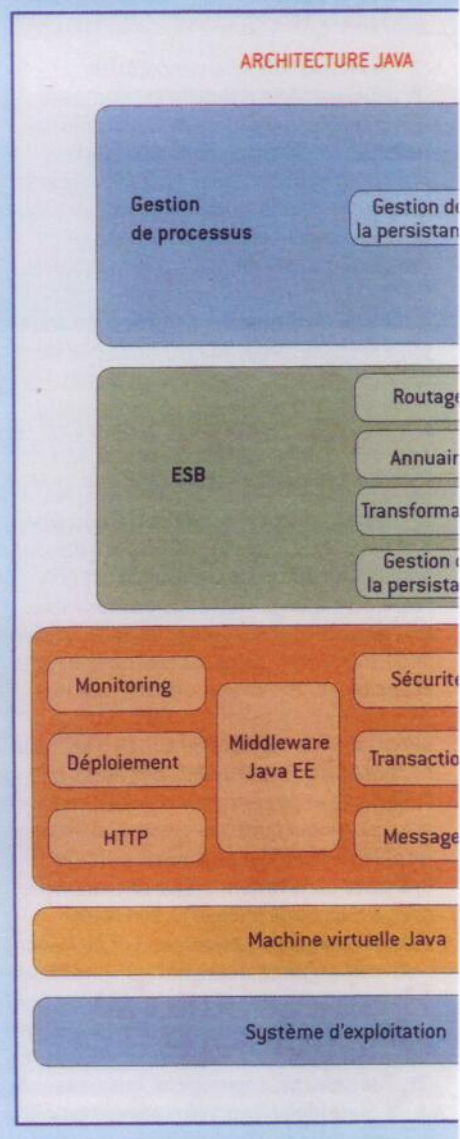
Première similitude : Dublin sera livré avec un conteneur central pour héberger ses services. « Aujourd'hui, les services conçus dans le framework .Net 3.5 sont directement exécutés dans Windows Server, précise Eric

Ortiz, responsable produit Biztalk et SOA chez Microsoft. Ils sont hébergés soit sur le poste client, soit sous la forme d'un service de bas niveau dans Windows, soit encore dans un répertoire sur le disque, soit enfin directement dans le serveur web IIS, sous la forme de services web. » Un éparpillement des services qui présente inévitablement des problèmes de montée en charge : Microsoft sait faire de la répartition de charge sur des frontaux IIS ou des objets COM/DCOM, mais pas sur des services formatés en WCF (Windows Communication Foundation, le modèle de programmation de services web apparu avec .Net 3.0). L'approche par conteneur inspiré de Java permettra de mieux appréhender les enjeux de volumétrie.

La fin des développements en silos

Autre élément comparable, la place névralgique accordée à l'orchestration de services : au cœur du serveur d'application. Outre son conteneur, Dublin repose en grande partie sur Windows Workflow Foundation (WF), un framework d'assemblage de services. « Avec lui, les développeurs de Visual Studio concevront des applications à base de services réutilisables. Finis les développements

Dublin, un conteneur

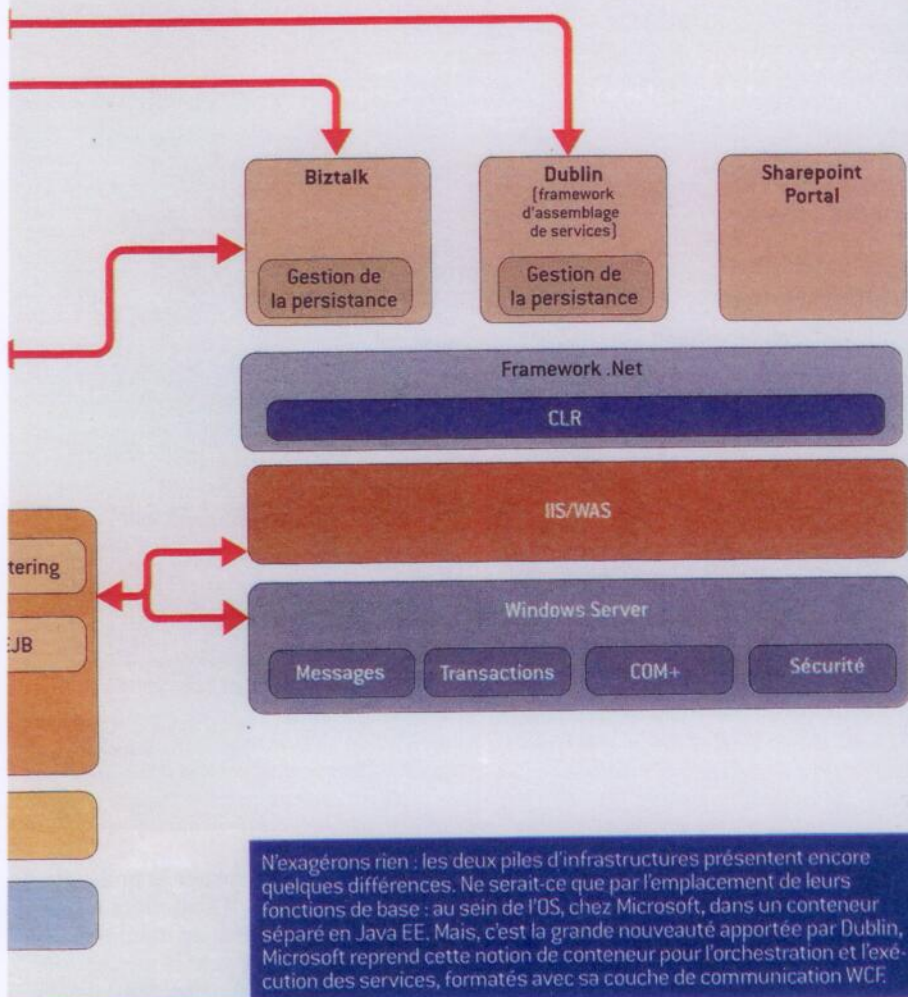


en silos comme ceux pratiqués dans IIS », rapporte Bernard Lenssens, président de Codit, société spécialisée dans les technologies d'intégration de Microsoft.

Avec Dublin, Microsoft entend donc sortir la logique métier du serveur web (IIS), des pages ASPnet ou des postes clients pour la confier à WF. Une démarche déjà entreprise par tous les grands acteurs de Java EE (IBM, Oracle-BEA, Red Hat-JBoss) qui cherchent également à centraliser cette logique métier mais cette fois au niveau de leur moteur d'orchestration : « Historiquement, dans les piles Java, les codes métier et graphiques s'exécutent au niveau du moteur de servlets. Mais de plus en plus, les éditeurs souhaitent

services calé sur la pile Java

ARCHITECTURE MICROSOFT



déporter l'exécution du code métier directement dans le conteneur de leur moteur d'orchestration : Process Server chez IBM, BPEL Manager chez Oracle... », avance Cyrille Le Clerc, consultant chez Xebia.

En matière d'orchestration, la comparaison avec les spécialistes Java ne s'arrête pas là. De la même façon que ces derniers ont rapproché leur moteur de workflow humain et documentaire de leur serveur d'intégration (EAI ou ESB), Microsoft fusionnera sûrement à terme WF et Biztalk, son EAI historique, positionné aussi comme ESB. Pour l'instant, Microsoft tient à cantonner ces deux offres dans des périmètres bien distincts : WF est dédié au développe-

ment d'applications composites à base de services. « Biztalk, en revanche, orchestre des processus à travers différentes applications », précise Eric Ortiz. Il intervient lors des besoins d'intégration à des systèmes non orientés services (mainframe, PGI, application maison) ou pour l'interfaçage avec des protocoles spécifiques tel l'EDI. Bref, pour tout ce qui n'est pas formaté en WCF.

En attendant le rapprochement de Biztalk et de WF, notons que certaines fonctions du premier se retrouvent déjà embarquées dans le second. « Les fonctions de publication et d'abonnement au cœur de Biztalk commencent à percer dans les couches basses de WCF au sein du framework .Net 4, détaille Bernard

Les deux éléments clés de Dublin

Windows Communication Foundation 4.0 (WCF)

- Lancé en 2006, WCF est un modèle de programmation permettant de masquer les contraintes d'invocation de services. Il supporte les protocoles COM+, .Net Enterprise Services, Message Queuing, .Net Remoting, ASP.Net et les services web. Dans la version 4 de .Net, il sera doté d'un environnement graphique (à travers des modèles disponibles dans Visual Studio) qui servira à définir les relations entre le contrat du service, son hébergement et son invocation protocolaire.

Windows Workflow Foundation 4.0 (WF)

- Aujourd'hui embarqué dans Sharepoint ou utilisé par certaines entreprises pour créer leurs applications, ce workflow humain et documentaire sera désormais associé à WCF dans une logique d'assemblage et d'orchestration de services. Des services qu'il n'est pas capable d'appeler avec l'actuel .Net 3.5. Autre nouveauté : il garantira la persistance des flux. Au sein de Dublin, WF devient ainsi la pièce maîtresse pour le développement d'applications composite.

Lenssens. Mais elles restent encore lourdes à manipuler et ne concernent que des besoins spécifiques d'intégration au sein de l'application. » L'autre fonction héritée de Biztalk, concerne la supervision et la persistance d'état du workflow : avec WF, le développeur peut déterminer des points de reprise d'activité en cas de dysfonctionnement du flux. Ce dernier est alors réinitialisé dans le contexte grâce à une base de persistance associée au workflow.

Microsoft adopte MDA

Dernier point de comparaison avec Java : Oslo, un projet qui est la déclinaison dans le monde Microsoft du modèle MDA (Model Driven Architecture). « Poussée par les tenants de Java, cette démarche propose de générer automatiquement du code sur la base de modèles décrits en UML, et indépendants des plates-formes », décrit Cyrille Le Clerc. Oslo vise donc à rendre tous les modèles ●●●

SERVEUR D'APPLICATIONS : JAVA EE INSPIRE MICROSOFT

CE QU'EN PENSE
LE SPECIALISTE JAVA

Mariano Boni,
directeur technique
de Dreamsoft
(filiale de Solucom)

« S'affranchir du système
d'exploitation »

« Avec ces séries d'annonces, Microsoft se rapproche de l'esprit de Java. D'abord à cause des quelques projets open source qui gravitent autour de Dublin. Ensuite, et surtout, en raison de sa volonté, (toute relative) de s'affranchir du système d'exploitation, sur lequel, aujourd'hui, les composants .Net sont directement déployés. Les traitements génèrent alors un exécutable sur lequel il n'y a aucune prise. C'est problématique pour la montée en charge. Le seul recours consiste à rajouter de la mémoire. En adoptant le concept de conteneur, Microsoft dissocie ses briques. Cette approche apporte plus de souplesse. Il est ainsi possible de jouer sur le nombre d'instances déployées et la réutilisation de processus. N'oublions pas que cette nouvelle architecture portée par Dublin est encore cantonnée à des applications web. »

CE QU'EN PENSE L'ÉDITEUR



Kris Horrocks,
directeur technique senior
chez Microsoft

« Windows Server reste
le cœur de notre
serveur d'applications »

« Nous évoquons aujourd'hui Dublin, pour que les utilisateurs éventuels puissent identifier les nouvelles fonctions d'orchestration de services proposées. Mais il ne sera pas positionné comme un produit. Et encore moins comme un serveur d'applications. En soit, les avancées de Dublin ne peuvent pas être comparées aux piles Java EE. La comparaison tient en revanche si l'on rajoute à Dublin notre serveur web Internet Information Services, notre framework .Net et, bien sûr, Windows Server, qui reste le cœur de notre serveur d'applications. Par ailleurs, Dublin se distingue de Java car son conteneur n'héberge que des services et non des composants .Net, par exemple. Les applications seront conçues à base d'assemblage de services et non plus à l'aide d'objets dépendants de méthodes d'invocation propriétaires. »

CE QU'EN PENSE L'INTÉGRATEUR



Bernard Lenssens,
président de Codit, société
spécialisée dans les technolo-
gies d'intégration de Microsoft

« La promesse
d'Oslo : une meilleure
gouvernance »

« La seconde version de Dublin, qui lui succédera très vite, se renforcera sur l'administration des services et présentera de nouvelles fonctions d'intégration. Je pense que Biztalk, le bus d'intégration de Microsoft, convergera nécessairement avec Workflow Foundation. En termes de fonctions, cette fusion n'apportera pas grand-chose. Mais elle suppose que les flux d'intégration de Biztalk soient modélisés dans le référentiel d'Oslo. Un référentiel qui, pour rappel, concentrera et reliera les modélisations de flux éparées de l'entreprise. En théorie, ces flux d'intégration pourront être corrélés avec les modélisations issues des outils d'administration IT, qui tracent le trafic réseau ou l'activité des centres de données. Ce qui devrait donc apporter une meilleure supervision des processus. »

●●● élaborés dans l'entreprise potentiellement exécutables sous la forme d'un workflow WF. Peu importe que ces modèles émanent de profils (architectes, analystes métier, développeurs) ou de produits (outils de modélisation de processus, de flux d'intégration, d'architecture du système d'information...) différents.

Partant du constat qu'il n'existe aujourd'hui aucune passerelle entre ces environnements de modélisation hétérogènes, l'éditeur compte l'établir grâce à Oslo. Et ce à l'aide de trois éléments. L'un, Oslo Repository, est le référentiel, basé sur SQL Server, utilisé pour stocker les différentes modélisations. Un deuxième élément, « M », est un nouveau langage axé sur la description des schémas du référentiel, et de leurs relations, qui propose également une grammaire servant à définir les DSL textuels (Domain Specific Language). Enfin Visual Studio Quadrant, éditeur visuel manipulant les données du référentiel, héberge des sché-

mas représentant des processus, des applications, des workflows, des services et même des environnements (machines, réseaux, centre de données...).

Dans la vision d'Oslo, toutes ces représentations liées les unes aux autres aboutissent à un modèle exécutable. Un modèle qui pourra être exporté depuis le référentiel en XAML (langage de Microsoft servant à décrire le workflow à exécuter) puis interprété

Pour lancer son middleware, Microsoft mise sur ses propres spécifications

par WF. Microsoft va mener le projet Oslo par paliers. D'abord, il s'attellera à la sémantique des services de manière à ce qu'elle devienne partageable par les différents profils de l'entreprise. Suivront ensuite les domaines des applications web ou des environnements techniques (centre de données, virtualisation).

Pour mettre son middleware sur les rails de l'orchestration de processus et l'exécution de modèles, Microsoft mise sur ses propres spécifications. Pour la modélisation des workflows, il privilégie son langage XAML à BPEL, adopté, lui, par tous les grands de Java. Il s'appuie sur WCF lorsque IBM, BEA ou Tibco tablent sur SCA pour l'assemblage et l'invocation de composants. Enfin, il cherche à imposer son « M » face à la multitude d'outils de modélisation du marché. Microsoft a beau s'aligner sur l'architecture Java EE, il ne va pas jusqu'à respecter l'esprit des standards qui caractérise le monde Java. ■ VINCENT BERDOT