

# Open ESB : l' 'open source' commercial selon Sun

**Quel est l'intérêt majeur d'Open ESB, respectant les spécifications JBI ?** Open ESB respecte les spécifications JBI (Java Business Integration) et est composé de trois éléments essentiels. Au cœur du dispositif se trouve le routeur de messages à la norme NMR (Normalised Messages Router). Sur ce module viennent se greffer deux types de composants : les moteurs de services (BPM, BAM, Workflow, moteur d'orchestration ou de transformation, scripts, etc.) et les composants de liaison (binding) comme les Services Web, le (S)FTP, les modules Com/DCom, Rosetta Net, ebXML, etc. Alors, le NMR assure le lien entre ces divers composants dont un plug-in a été écrit (en général par l'éditeur du composant) pour une compatibilité avec la spécification JSR 208 (JBI). Avantage par rapport aux services Web : il suffit d'écrire un unique plug-in pour ouvrir le composant (ou l'application) à la communication avec tous les autres composants adaptés à l'ESB. Avec les services Web sans ESB, il faut écrire autant d'interfaces que de liens à établir? **Selon quel processus les contributions open source deviennent-elles commerciales ?**



En fait, nous proposons Open ESB en open source (licence GPL me semble-t-il) et nous en assurons l'animation, l'évolution, et l'hébergement. Par ailleurs, nous testons sur des dizaines de scénarios tous les nouveaux modules ou fonctions proposés avant de les intégrer. Ensuite, nous intégrons les modules à nos offres payantes afin d'élargir notre éventail de composants, si les développeurs l'acceptent. Bien entendu, nous mettons alors en place un système de royalties. Aujourd'hui, Open ESB propose 35 composants (services ou binding ?liaison-) dont seulement 11 ont été développés par Sun. Évidemment, il est toujours possible de télécharger gratuitement Open ESB. **Alors, Sun a cessé d'être l'apôtre de l'open source ?** Nous restons des évangélistes de l'Open Source, mais nous n'en faisons pas non plus une religion. Il faut rester économiquement pragmatique. C'est pourquoi nous faisons de l'Open source un modèle de développement pour Sun, qui bénéficie et du savoir-faire et de l'expérience de milliers de développeurs à travers le monde. Ces derniers y trouvent leur intérêt, et nous pouvons commercialiser nos propres produits avec trois fois plus de composants en limitant l'effort de développement. Nos clients profitent ainsi de l'expérience de milliers de développeurs à travers le monde. Si les développeurs participent nombreux à Open ESB, c'est aussi parce qu'ils apprécient cette initiative open source qui s'appuie sur le standard JBI. **Avec Open ESB en open source, toutes les SSII donc peuvent se passer de vous ?** Seulement 1 % des indépendants et entreprises téléchargent Open ESB pour l'utiliser directement dans leurs projets SOA ou ceux de leurs clients. Les 99 % autres préfèrent passer par nous ou les spécialistes que nous formons. Par ailleurs, toutes les entreprises ne disposent pas des millions de dollars nécessaires pour développer des briques de bases SOA et préfèrent collaborer à ce type de projet et/ou passer par Sun. Prenons un exemple. Si un éditeur développe une solution BPM, mais sans moteurs d'orchestration ou de transformation, et se limitant à des échanges basiques en services Web. Pour une implémentation complète, avec une forte intégration, il peut utiliser Open ESB et fournir ses propres produits avec 3 fois plus de composants sans aucun développement. Pour la partie commerciale, Sun propose des packages professionnels, avec une installation automatisée et

assistée, une documentation détaillée, du support avancé, des formations, des prototypages et proof-of-concepts, des services spécifiques? Par ailleurs, le rythme des mises à jour est un peu plus rapide dans nos solutions commerciales. **Et dans votre offre, comment s'articule l'offre entre Open ESB et Java Caps ?** Depuis 2003, nous commercialisons nos solutions Java Caps [Composite Application Platform Suite : plateforme complète SOA, d'intégration et de développement]. Datant de quatre ans, ce produit peut être considéré soit joliment mature ou très ancien, selon le jury? Toutefois, son design est identique à celui d'Open ESB, à la différence qu'il utilise un protocole propriétaire (et très efficace) Seebeyond Communication au lieu de NMR. Cette proximité permet aux composants Java Caps (BPM, BAM, Signe Costumer View, Single Patient View? ou les composants binding) d'interagir avec Open ESB. De même, les 85 adaptateurs eWays peuvent se connecter à Open ESB. Enfin, avec la version 5.2 de Java Caps, prévue au second trimestre 2008, les clients pourront utiliser librement soit Java Caps, soit Open ESB (version commerciale), ou les deux.