



Developing Agility

*Une lettre d'information destinée aux clients
Unisys EAE et Agile Business Suite*

Des questions ?
ABSuite@unisys.com peut vous aider.

Table des matières

- Page 2 **Agile Business Suite 2.0 IC1400 : améliorer la productivité des développeurs** Avec la publication de la correction provisoire (IC) 1400, nous avons modifié la fonctionnalité sous-jacente d'Agile Business Suite 2.0 dans le but de rationaliser et de simplifier le travail de développement, mais aussi de faciliter la transition vers AB Suite pour les utilisateurs d'EAE.
- Page 3 **Rappel important sur la migration à l'intention des clients EAE utilisant Microsoft Windows, Linux ou UNIX** La phase 1 du support pour EAE sur les plates-formes Microsoft® Windows®, UNIX® et Linux® prendra fin le 31 décembre 2011. Il ne reste que six mois environ avant cette échéance ; il est donc important de lancer sans plus attendre vos plans de migration.
- Page 4 **Le coin des ingénieurs : transactions avec et sans état** Les transactions « à mémoire d'état » et « sans état » présentent des applications et des avantages différents dans les univers EAE et AB Suite ; mais les réunir peut vous aider à moderniser vos applications de diverses manières. Lisez la suite de cet article pour savoir comment faire.
- Page 7 **Le coin des partenaires : Information Exchange Group** Réunissant une riche expérience des plates-formes EAE et AB Suite, l'Information Exchange Group propose une multitude de services utiles pour ces outils de développement. Lisez cet article pour découvrir ce que ce forum peut vous apporter.

**Vous aimeriez avoir votre propre exemplaire de Developing Agility ?
Inscrivez-vous sans plus attendre sur le site eCommunity !**

Agile Business Suite 2.0 IC1400 : améliorer la productivité des développeurs

Par Maarten Schneider, responsable marketing international, Enterprise Application Environment (EAE) et Agile Business Suite

Nous souhaitons transmettre nos sincères remerciements aux premiers utilisateurs d'Agile Business Suite (AB Suite) à travers le monde, qui ont soumis de nombreuses suggestions, précieuses et judicieuses, pour améliorer la facilité d'utilisation de la fonctionnalité System Modeler d'AB Suite 2.0, nous avons aujourd'hui le plaisir de vous annoncer, avec la publication de la correction provisoire (IC) 1400, des améliorations importantes à AB Suite 2.0. Avec l'IC1400, AB Suite 2.0 utilise au mieux l'infrastructure sous-jacente de Microsoft Visual Studio® afin d'offrir un comportement cohérent entre System Modeler et les types de projets Visual Studio.



Par exemple, AB Suite Developer va désormais vous fournir plus de détails sur votre application existante en séparant le volet de la vue Class et ses membres en deux volets distincts. Ainsi, chaque fois que vous sélectionnez un objet dans Developer, le volet inférieur affichera automatiquement les informations pertinentes sur cet objet.

Nous avons également ajouté une nouvelle fonction de recherche dans la vue Class ; elle vous propose un affichage filtré qui vous permet d'accéder directement à l'emplacement où vous maintenez votre application et à des informations sur les catégories Owner, Inherits ou Type. La correction IC1400 introduit en outre une fonction « Synchronize Class View » (synchronisation de la vue Class) qui, à partir de n'importe quel élément, vous dirige au bon endroit dans la vue Class.

Nous avons aussi prévu une vue simplifiée des « frames » et du contenu dans les Reports, pour faciliter la maintenance.

Grâce à toutes ces fonctionnalités, et aussi aux nombreuses mises à jour, plus modestes mais néanmoins très utiles, que nous avons incluses dans la correction Agile Business Suite 2.0 IC1400, les utilisateurs d'Enterprise Application Environment (EAE) pourront se rendre compte que l'heure est maintenant venue pour eux de migrer vers AB Suite.

Comme bon nombre des mises à jour contenues dans la correction IC1400 avaient pour but d'intégrer à AB Suite des fonctions familières, les utilisateurs d'EAE n'auront pas de difficultés à gérer leurs applications après avoir migré vers AB Suite. Les nouvelles fonctions de recherche permettront, par exemple, de réduire la courbe d'apprentissage et de préserver une productivité élevée.

Par ailleurs, cette IC contient plusieurs améliorations qui maximisent l'utilisation des fonctionnalités Visual Studio les plus courantes ; les développeurs C#, habitués à travailler avec des fonctionnalités similaires, apprendront donc encore plus facilement à maîtriser AB Suite.

Développeurs EAE : l'heure de la migration a sonné

Comme vous pouvez le voir, cette IC apporte une vraie valeur ajoutée pour les clients EAE qui migrent vers AB Suite. Je vous encourage donc, si ce n'est déjà fait, à préparer vos plans de migration. Consultez la section Documentation du [site Web du support Unisys](#) pour de plus amples détails sur AB Suite IC1400.

Si vous avez besoin d'aide pour intégrer votre migration à votre budget 2012, n'hésitez pas à vous adresser à votre consultant technique Unisys qui vous guidera dans l'élaboration d'un projet de plan de migration. Vous pouvez également contacter Diane McGonigle, responsable de la migration vers AB Suite, par [e-mail](#) ; elle vous orientera vers la bonne personne.

Rappel important sur la migration à l'intention des clients EAE utilisant Microsoft Windows, Linux ou UNIX

Par Diane McGonigle, responsable du département Agile Business Suite Migration, Unisys TCIS



La phase 1 du support d'EAE 3.3 sur les plates-formes Microsoft Windows, UNIX et Linux prendra fin le 31 décembre 2011. À six mois environ de cette échéance, je tenais à vous rappeler de planifier votre migration vers Agile Business Suite et de passer en revue certains des avantages offerts par le passage à ce nouvel environnement.

En migrant vers AB Suite sous Windows, vous disposerez d'un nouvel environnement d'exécution .NET qui vous permettra d'éviter les dépenses induites par le renouvellement des licences COBOL que vous aviez dans EAE. Et ce changement d'environnement est parfaitement transparent : même si AB Suite génère du C#, vous pouvez continuer à développer en LDL (aujourd'hui LDL+ dans AB Suite). Vous pourrez passer à un environnement de développement moderne, sous Windows, sans que l'application ne change ou presque.

Plusieurs entreprises utilisent déjà AB Suite sous Windows au sein d'environnements de production très sollicités. Par exemple, [Thomasville Utilities](#) – auquel un article a récemment été consacré dans *Developing Agility* – a procédé à une migration réussie il y a peu et fait aujourd'hui tourner son application critique de facturation, recouvrement ainsi que la gestion des commandes et des stocks sous AB Suite.

Citons également le cas de [TIES](#), partenaire technologique du secteur de l'éducation, qui exécute des centaines de milliers de transactions par jour sur une application Windows migrée, et intègre chaque semaine de nouveaux districts scolaires. [Harris School Solutions](#), autre éditeur de logiciels pour le secteur de l'éducation, a migré environ 20 % de ses districts vers la version AB Suite de son application SchoolMax.

Si vous utilisez actuellement EAE sous UNIX ou Linux, veuillez [me](#) contacter ou bien vous adresser à votre représentant commercial Unisys pour discuter des possibilités de migration vers une plate-forme d'exécution AB Suite prise en charge et pour en savoir plus sur les économies que cette démarche vous permettrait de réaliser.

Planifiez votre migration

Le 1er janvier 2012 marquera le début de la phase 2 du support d'EAE 3.3 Runtime pour les environnements d'exploitation Windows, UNIX et Linux. Au cours de la phase 2, les clients disposant de contrats de maintenance valides continueront à bénéficier d'une assistance par téléphone et en ligne du Support client Unisys, à obtenir des correctifs pour les problèmes identifiés et, dans la mesure du possible, à recevoir des solutions palliatives aux erreurs qu'ils signalent et pour lesquelles il n'existe pas encore de correctif. Notez que si vous n'arrivez pas à finaliser votre migration d'ici la fin de l'année, vous avez la possibilité de bénéficier d'un support étendu (avec transmission de hot patches pour la correction des nouveaux problèmes) pendant un an maximum, et moyennant facturation.

Si vous des questions sur la migration vers AB Suite, veuillez contacter Diane McGonigle, Responsable de la migration vers AB Suite, par e-mail à l'adresse diane.mcgonigle@unisys.com pour obtenir notre liste de contrôle et des conseils pour commencer le processus de migration.

Le coin des ingénieurs : Les transactions avec et sans état

Par Alan Hood, ingénieur conseil, Agile Business Suite, Unisys TCIS

Dans le vocabulaire informatique, le concept d'état désigne un environnement transactionnel dans lequel un utilisateur se connecte à un serveur à partir d'un lieu donné, d'un terminal ou autre dispositif, et s'authentifie au moyen d'identifiants uniques afin d'avoir accès aux informations saisies au cours de sa session.

La plupart des applications traditionnelles de transactions en ligne s'appuient sur la notion d'état et la conservation des informations d'un utilisateur lambda entre deux transactions. Ce faisant, il n'a pas à s'authentifier ou à entrer des identifiants chaque fois qu'une nouvelle transaction est transmise à l'application. Dans les environnements EAE et Agile Business Suite, ces informations sont généralement conservées dans GLB.WORK, de sorte que toute application utilisant cette fonction puisse exécuter des transactions « avec état » (« stateful »).

Si l'état est très utile à bien des égards, un certain nombre d'environnements transactionnels et de protocoles exigent des interactions sans état (« stateless ») ; c'est le cas de SOA (service-oriented architecture) et de http (Hypertext Transfer Protocol). Dans ce cas, chaque demande est traitée comme une transaction distincte, sans lien aucun avec une demande antérieure. Les transactions sans état fonctionnent comme des paires indépendantes de demandes et réponses : le serveur hôte ne retient aucune information de statut ou de session entre deux demandes, comme il le ferait dans un environnement avec état.

Composants essentiels de toute approche informatique moderne, les transactions sans état sont indépendantes de toute transaction antérieure. On peut leur appliquer l'acronyme « ACID » :

- **Atomicité** : la suite d'opérations est indivisible, en cas d'échec en cours d'une des opérations, la suite d'opérations doit être complètement annulée (rollback) quel que soit le nombre d'opérations déjà réussies.
- **Cohérence** : le contenu de la base de données à la fin de la transaction doit être cohérent sans pour autant que chaque opération durant la transaction donne un contenu cohérent. Un contenu final incohérent doit entraîner l'échec et l'annulation de toutes opérations de la transaction.
- **Isolement** : lorsque deux transactions A et B sont exécutées en même temps, les modifications effectuées par A ne sont ni visibles par B, ni modifiables par B tant que la transaction A n'est pas terminée et validée (commit).
- **Durabilité** : une fois validé, l'état de la base de données doit être permanent, et aucun incident technique (exemple : crash) ne doit pouvoir engendrer une annulation des opérations effectuées durant la transaction.

Bien que très utiles, les transactions sans état ne sont pas la panacée pour toutes les formes de traitement des transactions. Par exemple, beaucoup de transactions ont besoin d'informations complémentaires (d'identité, d'authentification, de sécurité), qu'un environnement sans état ne peut souvent pas fournir, ce qui impose à l'utilisateur de devoir les retaper systématiquement. De même, dans le cas de messages longs véhiculant quantité d'informations entre le serveur et l'utilisateur, un environnement avec état est avantageux puisqu'il évite de devoir échanger de manière répétitive ces données.

Les deux modèles de transactions présentant chacun des avantages, les développeurs ont intérêt à intégrer les deux de manière à profiter au maximum des atouts de chacun, tout en prenant en charge les applications mobilisant les protocoles modernes. EAE et AB Suite intègrent justement des outils en ce sens. >>



Les protocoles sans état dans EAE et AB Suite

Beaucoup de développeurs adeptes d'EAE et d'AB Suite pensent que Component Enabler impose l'utilisation de transactions avec état pour leurs applications. Or, tout dépend de la manière dont l'application a été programmée. Si la présence de GLB.WORK implique souvent l'utilisation de transactions avec état, certains changements spécifiques dans le code de l'application avec état permettent d'en faire une application sans état. De surcroît, vous verrez dans la rubrique suivante que le choix entre avec et sans état n'est pas forcément définitif.

Par exemple, EAE et AB Suite prennent en compte le standard Open DTP (anciennement XOpen DTP ou Open OLTP) et permettent d'exposer/d'utiliser des services entrants et sortants. Pour ce faire, vous pouvez définir l'application EAE ou AB Suite en tant que serveur et chaque Ispec comme un service. Au moment de générer l'application, le View Buffer utilisé par le programme « client » appelant est généré automatiquement. De même, vous pouvez invoquer un service Open DTP externe en important son View Buffer dans un type spécial d'Ispec externe, appelé OLTP View Ispec (OVI). Toutes les transactions sortantes sont invoquées au moyen de l'habituelle commande AUTO External. Cette configuration permet d'exécuter des transactions globales à deux phases (2PC pour 2 Phase Commit) et accepte en parallèle les transactions standard et clients GUI utilisant les mêmes bibliothèques de transactions.

Pour que cette approche fonctionne, il faut des Ispec sans état (sans conservation des informations dans GLB.WORK). Notez que les Ispec COPY.FROM et « big buffer » ne sont pas supportés. Et les types de données doivent aussi être conformes aux définitions de buffer XCOMMON, autrement dit simples, avec des longueurs fixes de caractères ou de nombres.

En plus d'Open DTP, EAE et AB Suite permettent de tirer avantage des interfaces Web Services. Le Component Enabler Client Generator pour ASP .NET Web Services permet de définir un bundle d'Ispec et de générer l'interface client correspondante au moyen d'un générateur drop-in. Le générateur d'interface client ASP .NET Web Services produit les définitions XML et SOAP qui permettent aux utilisateurs d'invoquer les Ispec au moyen des protocoles standard de Web Services. Enfin, comme pour Open DTP, pour que ces interfaces fonctionnent il faut des Ispec sans état, et des types de données simples, avec des longueurs fixes de caractères ou de nombres. Parmi les éléments/options clé qui différencient ces interfaces de l'implémentation Open DTP figurent la prise en charge des Ispec COPY.FROM et la possibilité de définir des champs de données comme entrée, sortie, ou entrée/sortie.

Le juste équilibre entre transactions avec et sans état

Les techniques décrites ci-dessus permettent de créer des applications prenant en charge des protocoles avec et sans état. Tout Ispec (en dehors des restrictions évoquées ci-avant) peut être défini comme service OLTP ou service Web ; sans affecter les autres interfaces avec et sans état.

Avec la possibilité de combiner des transactions avec et sans état dans une même application, vous ferez sensiblement progresser vos travaux de modernisation et pourrez mieux préparer vos applications pour les utiliser en environnement SOA ou à partir de terminaux mobiles.

Examinons comment procéder. Imaginez que vous souhaitiez optimiser une application bancaire à écran vert classique pour son utilisation sur le Web sans pour autant supprimer les caractéristiques d'identité et d'authentification des transactions avec état. Il est important de garder à l'esprit qu'entre les transactions, ce type d'information est généralement conservé dans GLB.WORK. >>

Avant d'effectuer des transactions sans état, afin de conserver les propriétés de mémoire d'état utiles de l'application, vous devez commencer par créer un nouvel Ipec de type output (que nous nommerons « SWORK » dans cet exemple) dans lequel seront stockées les informations de connexion chaque fois qu'un nouvel utilisateur accédera à l'application, ainsi que tout ce qui est habituellement conservé dans GLB.WORK. Lors de la création du SWORK, vous devez créer une clé spéciale qui servira d'index à SWORK pour la session de l'utilisateur. Cette clé sert à accéder aux données de transaction avec état nécessaires contenues dans SWORK, qui conserve les informations entre les transactions au cours d'une session donnée. La clé SWORK est ainsi la seule information qui circule de et vers le client lors de chaque transaction. Quand l'utilisateur se déconnecte, SWORK efface les données jusqu'à la prochaine ouverture de session.

Pour ajouter des contrôles de sécurité, pour la transmission de données confidentielles ou financières par exemple, il suffit de changer la valeur de la clé SWORK à chaque session de l'utilisateur, ou même transaction par transaction. Pour simplifier ces modifications, pensez à créer deux logiques globales, que nous nommerons ici « GET-SWORK » et « PUT-SWORK ». Insérez GET-SWORK au début de la transaction pour récupérer les données d'état de la transaction précédente, puis placez PUT-SWORK à la fin de la transaction pour stocker les données d'état qui seront nécessaires à toutes les transactions ultérieures, jusqu'à la déconnexion ou l'interruption de session pour cause d'inactivité de l'utilisateur.

Dans quel état êtes-vous ?

Avez-vous déjà tenté de faire fonctionner vos applications EAE ou AB Suite dans un environnement sans état ? De quelle façon vous y êtes-vous pris ? [Faites-nous part de votre expérience](#). Nous l'évoquerons peut-être dans un prochain numéro de Developing Agility.

Vous avez des questions ou des commentaires sur les transactions avec et sans état dans EAE ou AB Suite ? Ecrivez-nous à ABSuite@Unisys.com et nous nous ferons un plaisir de vous renseigner.

Le coin des partenaires : Information Exchange Group



Fondé en 1995, l'Information Exchange Group, Inc. (IEG) a été mis en place avec un objectif précis : aider les entreprises à maximiser la valeur des systèmes, des applications et des données supportant leurs activités métier au quotidien.

16 ans plus tard, IEG propose une expérience et une expertise substantielles aux entreprises utilisant EAE et Agile Business Suite et qui souhaitent concevoir et moderniser leurs interfaces utilisateur, modifier la présentation et la cinématique des composants de leurs formats d'affichage, mais aussi simplifier et rationaliser leurs migrations.

Concevez votre interface utilisateur

Grâce à l'Interface Builder (IB) d'IEG et ses fonctionnalités flexibles et automatisées de conception de formulaires, qui facilitent à la fois la modernisation de votre interface utilisateur et l'optimisation d'applications existantes sans modifications conséquentes, votre interface utilisateur sera toujours au goût du jour et en adéquation avec vos exigences professionnelles et informatiques.

Initialement lancée en 1999, la version actuelle d'IB offre une coordination bidirectionnelle avec AB Suite, ce qui permet d'initialiser l'outil comme faisant partie d'un processus Build ou Rebuild d'AB Suite ou de le configurer pour qu'il initialise AB Suite Build.

IB automatise le processus d'ajout de fonctionnalités graphiques et la personnalisation ou les traitements des modules clients et serveurs. L'outil permet de répondre aux besoins de modernisation des interfaces utilisateur, tels que l'adaptation de la taille de l'écran au navigateur et le choix des polices selon les paramètres recommandés qui, s'ils devaient être réalisés manuellement ou page par page, exigeraient un temps et des ressources informatiques importantes.

Personnalisez les formulaires selon vos besoins

Avec Dynamic Web Alteration (DWA), un autre outil puissant proposé par IEG pour les environnements EAE et AB Suite, les développeurs multiplient les possibilités d'amélioration de leurs formulaires Web.

Cet outil peut, par exemple, servir à déterminer dans quelles conditions invoquer le changement de couleur d'un champ ou la désactivation d'un bouton. Et comme DWA change le formulaire ASPx au sein du cycle de vie ASP .NET, il permet d'appliquer dynamiquement quasiment tous les attributs HTML ou les styles JavaScript à n'importe quel élément adressable du formulaire.

Lorsque vous ajoutez le module Dynamic Web Module (DWM) d'IEG à DWA, vous pouvez intégrer des modifications conditionnées par des données, celles d'authentification des utilisateurs par exemple. Ce qui vous donne la possibilité de contrôler l'affichage destiné à un utilisateur final en particulier. DWM faisant partie de votre application, toutes les requêtes DWA sont intégrées à votre référentiel de données applicatives. >>

Simplifiez les migrations

Afin de garantir que tout changement d'environnement se fasse en douceur, le processus « Information Migration » d'IEG gère les transferts de données applicatives et de code source. Idéalement conçu pour les migrations entre plates-formes, identiques ou différentes, ou depuis EAE vers AB Suite, le processus génère des programmes de reporting adaptés à chacune de vos applications, qui automatisent le transfert de données des référentiels existants vers un référentiel cible, d'EAE ou d'AB Suite indifféremment. Et comme le format est indépendant de la plate-forme, vous pourrez migrer aisément d'une plate-forme sous Windows vers un serveur MCP, par exemple.

En outre, IEG peut vous aider à transférer vos applications EAE d'un système UNIX ou Linux vers AB Suite sur des serveurs MCP et/ou Windows. En cours de migration, le processus analyse les spécifications EAE et remplace automatiquement les mots réservés, enlève le surlignage des séquences de contrôle et suggère/modifie les problèmes éventuels des constructions LDL DETERMINE et LOOK.UP. Pour un déploiement sur le Web, le processus de migration vous assiste dans des tâches comme le redimensionnement de la grille utilisée pour le dessin des formulaires dans les deux sens.

IEG est fier de sa réactivité en réponse aux demandes de ses clients. Pour en savoir plus sur ce qu'IEG peut apporter à votre activité professionnelle, rendez-vous sur le [site Web](#) ou contactez-les dès aujourd'hui pour une démonstration de faisabilité de projet.

Les spécifications sont susceptibles d'être modifiées sans préavis.

© 2011 Unisys Corporation.

Tous droits réservés.

Unisys et le logo Unisys sont des marques déposées d'Unisys Corporation. Linux est une marque déposée de Linus Torvalds aux États-Unis et dans d'autres pays. Microsoft, Visual Studio et Windows sont des marques déposées de Microsoft Corporation. UNIX est une marque déposée de The Open Group. Tous les autres noms de produits et marques mentionnés dans ce document sont considérés comme des marques commerciales ou déposées de leurs détenteurs respectifs.