



Developing Agility

Lettre d'information trimestrielle pour les clients EAE et Agile Business Suite d'Unisys

Juillet 2012

Table des matières

- 1 Comment améliorer la productivité des développeurs ?
- 3 Nouveau ! L'intégration d'ePortal au service de la mobilité des solutions EAE et AB Suite
- 4 Coup de projecteur sur le centre de support client : Grant Paine
- 6 Le coin des ingénieurs : Trucs et astuces concernant Debugger
- 9 Les migrations vers AB Suite vont bon train en France
- 10 Info Center

Comment améliorer la productivité des développeurs ?

Par Maarten Schneider, responsable marketing international, Enterprise Application Environment et Agile Business Suite, Unisys TCIS

« Faire plus avec moins », voilà un objectif en tête des priorités de tout le monde aujourd'hui, en particulier dans le domaine informatique. Naturellement, cela implique d'être plus productif. Mais comment mesurer la productivité et savoir si elle progresse ou non ? C'est-là une question à laquelle les équipes de développement logiciel essaient tant bien que mal de répondre depuis des années et qui se pose aujourd'hui avec plus d'acuité que jamais.

Les responsables du développement logiciel avec lesquels j'ai pu en discuter estiment pour la plupart avoir une bonne idée de la productivité de leur équipe mais admettent qu'il est difficile de la mesurer précisément. Certains calculent les « heures par point de fonction » une fois le projet bouclé et quelques-uns utilisent ces données pour estimer l'ampleur de leurs prochains projets de développement. D'autres relèvent le nombre de lignes de code écrites sur une période donnée.

Je pense pour ma part que mesurer la productivité est plus important que la méthode utilisée. Il faut trouver le bon point de départ et affiner progressivement votre approche. Commencez par exemple par « X heures par point de fonction » et évaluez si cette hypothèse reflète le travail effectivement fourni.

Lorsque je repense aux nombreux projets de clients Enterprise Application Environment (EAE) et Agile Business Suite (AB Suite) que j'ai pu voir, un dénominateur commun ressort : Plus l'équipe est nombreuse, plus ses membres passent de temps à communiquer et plus la productivité par développeur chute. J'ai même vu certains cas où l'arrivée de nouveaux développeurs en renfort sur un projet en retard ne faisait que ralentir encore les choses. >>



La leçon à tirer ? Mieux vaut miser sur une petite équipe de très bons développeurs, suivant une procédure rigoureuse, étayée par le meilleur outil possible.

Ma conclusion est donc la suivante : pour améliorer votre productivité, vous devez vous concentrer sur trois éléments clés :

1. L'éventail des compétences techniques et les connaissances métiers de vos développeurs
2. La méthode de développement
3. L'outil de développement

La formation continue de vos développeurs est forcément payante. Et cette formation doit porter sur trois facettes essentielles :

1. Les objectifs métiers de votre organisation, les processus et les besoins des utilisateurs finaux de l'entreprise
2. Les méthodes de développement de nouvelles applications
3. Les nouvelles fonctionnalités/améliorations de votre outil de développement d'applications

Comblent le fossé de communication entre les fonctions métiers et informatique

Regardons les choses en face : les cahiers des charges détaillés sont complexes à élaborer, difficiles à lire et quasi impossibles à faire approuver. Par contre, de courtes descriptions de « cas d'usage », des projets au périmètre bien délimité et une approche de prototypage systématique sont d'excellents moyens d'accélérer le développement pour une mise à disposition plus rapide. Par ailleurs, en connaissant vos processus métier, vos développeurs pourront plus facilement communiquer avec les utilisateurs de l'entreprise... au final, la solution développée aura plus de chances de correspondre aux attentes.

Exploiter au mieux les nouvelles approches de développement

Les méthodes de développement des applications ont évolué ces dernières années. La plupart des entreprises utilisent aujourd'hui des approches à mi-chemin entre la traditionnelle méthode dite en « cascade » ou « Waterfall » et le Rational Unified Process (RUP – processus unifié rationnel) ou le Scrum.

RUP et Scrum sont des approches agiles et itératives, utilisant toutes deux le principe du timeboxing (découpage en tranches de temps) pour définir une date butoir, la communiquer à tous les acteurs impliqués

et veiller au respect de cette échéance. Les équipes RUP organisent des réunions de projet régulières ; les équipes Scrum vont encore plus loin, se réunissant chaque jour pour un petit briefing informel d'une quinzaine de minutes. Dans les deux cas, il est essentiel de ne pas être coupé des besoins de l'entreprise et des exigences des utilisateurs et d'appliquer des techniques de gestion de projet rigoureuses.

Bon nombre d'entreprises enregistrent de meilleurs résultats avec des groupes de développeurs plus restreints appliquant les méthodes RUP, Scrum ou une combinaison des deux. Et bon nombre des développeurs que je peux rencontrer confirment que les projets de petite ampleur parviennent plus souvent à un résultat positif que les grands.

Utiliser des outils de développement hautement productifs

Le dernier domaine à considérer pour améliorer la productivité concerne l'outil de développement. Les lignes de code que doit écrire le développeur sont importantes, non seulement pour le développement du logiciel, mais aussi pour sa maintenance à long terme. Plus le code est volumineux, plus l'environnement devient complexe, et plus les opérations de support et d'amélioration sont difficiles.

À cet égard, les utilisateurs d'EAE et d'AB Suite disposent du meilleur des outils de productivité, un langage de haut niveau qui réduit de manière spectaculaire le nombre de lignes de code. L'environnement AB Suite présente un avantage supplémentaire, le framework Microsoft® Visual Studio® permettant aux développeurs de voir plus clairement l'application et d'utiliser un outil de débogage plus simple.

Donc, la prochaine fois que vous chercherez le moyen d'en faire plus avec moins, pensez à tout ce que vous pourriez faire de plus grâce aux toutes nouvelles capacités d'EAE et d'AB Suite. Parcourez par exemple l'article consacré à ClearPath ePortal proposé dans ce numéro ; vous verrez comment il a évolué dans la prise en charge d'EAE et d'AB Suite. Pensez également aux moyens d'enrichir les compétences de vos développeurs, en leur proposant à la fois des formations techniques et métiers. Enfin, étudiez votre processus de développement : peut-être pourriez-vous adopter certains éléments des approches RUP et Scrum pour réduire encore les cycles.

Et vous, quelles sont vos recettes pour en faire plus avec moins ? Votre expérience m'intéresse. Écrivez-moi à l'adresse ABSuite@unisys.com.



Nouveau ! L'intégration d'ePortal au service de la mobilité des solutions EAE et AB Suite

Déjà proposé en standard par tous les nouveaux systèmes ClearPath, le co-processeur spécialisé ClearPath ePortal est un outil très précieux pour rendre les applications ClearPath accessibles par tout type d'appareils mobiles et compatibles Web.

En Juin 2012, nous avons mis à disposition une série d'améliorations qui ont rendu ClearPath ePortal encore plus simple à utiliser et plus puissant avec les applications EAE et Agile Business Suite.

Cette nouvelle version vous permet d'importer votre interface client actuelle dans le générateur ePortal et d'étendre en toute efficacité l'utilisation de vos applications EAE et AB Suite sur le Web et les appareils mobiles, et ce sans rien modifier aux applications de base. Et comme tout ceci est possible en natif dans votre environnement ClearPath, vous pouvez exécuter ces projets tout en bénéficiant des niveaux de sécurité, de flexibilité et d'évolutivité que vous avez déjà pu apprécier dans d'autres projets de développement.

Comment démarrer ?

La première étape de la création d'une application client ePortal consiste à définir le projet de source de données. Il vous suffit de choisir « AB Suite EAE Data Source » dans les modèles de projets de sources de données ClearPath de Visual Studio et de laisser l'assistant faire le reste. Ensuite, importez les définitions Ispec (notez qu'un assistant vous aide là encore) en cliquant sur les "output" désirés de votre générateur client ePortal EAE/AB Suite.

Il faut ensuite définir les types de clients qui auront accès aux transactions Ispec en créant un projet ClearPath Presentation à partir d'un des modèles fournis. Chaque projet client peut utiliser soit les mêmes définitions de sources de données, soit des définitions différentes, en fonction des accès que vous souhaitez autoriser aux utilisateurs.

Les modèles de projets ClearPath Presentation actuellement disponibles concernent les applications mobiles, Web et Services Web.

La mobilité en toute simplicité

Si vous choisissez « Mobile » comme type de ClearPath Presentation, la solution se charge de créer automatiquement l'interface mobile pour les

smartphones Apple® iOS, Android™ et BlackBerry®, et les tablettes iOS ou Android, sans que vous ayez à aucun moment à vous soucier d'apprendre un langage de programmation spécifique. Le type de présentation ("look and feel") sera déterminé au moment de la connection en fonction du modèle d'appareil mobile.

Vous pourrez éventuellement adapter plus finement la présentation pour en éliminer les champs de données inutiles dans une interface mobile ou bien utiliser des commandes GUI (interface utilisateur graphique) spécifiques.

Tout l'intérêt des solutions mobiles réside dans leur rapidité et leur efficacité ; donc, moins votre application mobile a de transactions à réaliser, mieux c'est. Plutôt que d'obliger les utilisateurs à naviguer sur une page de connexion et dans plusieurs menus, une application mobile doit les amener là où ils le veulent en une étape ou deux.

La fonctionnalité d'orchestration d'ePortal est extrêmement précieuse à cet égard. Pour faire simple, l'orchestration crée une macro qui présente à l'utilisateur de multiples transactions comme une seule et même action. Elle peut en outre rationaliser encore l'expérience de l'utilisateur en vous aidant à importer sélectivement les écrans et les images que vous souhaitez intégrer à l'application mobile.

Si l'interface utilisateur de l'application principale est modifiée, ePortal dispose d'une fonction de réconciliation que vous pouvez invoquer pour assurer la synchronisation entre ce que voient les utilisateurs mobiles et ce qui se passe en arrière-plan. Sachez par ailleurs que si vous commencez à utiliser ePortal avec EAE mais prévoyez de migrer vers AB Suite, vous n'avez pas à craindre de perdre tout le travail déjà accompli. Votre interface utilisateur migrera vers le nouvel environnement en toute transparence et vous serez paré.

Pour tout savoir sur ce que vous permet de faire ClearPath ePortal, consultez les listes de lecture sur la chaîne YouTube de ClearPath.

Coup de projecteur sur le centre de support client : Grant Paine



Voici un nouvel article dans la série présentant les analystes du CSC d'Unisys en charge du support d'EAE et d'Agile Business Suite. Vous aimeriez qu'un article soit consacré à un analyste du support en particulier ? Envoyez-nous votre dossier à l'adresse : ABSuite@unisys.com.

Grant Paine a rejoint Unisys (alors Burroughs) en 1983. Presque trente ans plus tard, Grant est l'un des piliers de l'équipe CSC ; il assure le support des logiciels ClearPath MCP, EAE et Agile Business Suite pour les clients de la zone Asie-Pacifique (APAC).

Developing *Agility* s'est récemment entretenu avec Grant à propos de son rôle d'analyste CSC, de ce qu'il préfère dans ce travail et de tout ce qu'il a pu apprendre en travaillant aux côtés d'utilisateurs d'EAE et d'AB Suite depuis près de 30 ans.

Developing Agility : Dites-nous en un peu plus sur les services que vous proposez et les clients que vous aidez.

Grant Paine : Depuis toutes ces années, j'ai eu l'occasion d'intervenir sur un large éventail de services de support : mises à niveau de systèmes d'exploitation MCP, installation de nouveaux systèmes MCP, formation des utilisateurs, assistance à la migration vers AB Suite par exemple. Je propose également des démonstrations aux clients et des services de conseil.

Je travaille essentiellement dans la région APAC, mais j'interviens aussi pour des clients aux États-Unis, au Royaume-Uni et en Europe plus régulièrement depuis que le CSC fonctionne comme une équipe globale.

DA : Que préférez-vous dans vos fonctions d'analyste CSC ?

GP : J'aime tout particulièrement le contact avec le client, chercher à en savoir plus sur son environnement et travailler avec lui à la résolution des problèmes. J'apprécie vraiment les rencontres en face à face ; cela me permet de mieux cerner

les besoins, et donc de mieux y répondre. Dans le même esprit, j'apprécie également la diversité des clients pour lesquels nous intervenons et de leurs activités. Je suis heureux de constater que la majorité des clients sont toujours prêts à adopter de nouvelles technologies et, aussi, de nouvelles manières de travailler.

DA : Quels ont été les moments marquants de l'année passée ?

GP : L'un des temps forts de l'année 2011 a été mon intervention pour le compte du groupe Alliance en Nouvelle-Zélande pour le déploiement d'une version Microsoft Visual Basic® .NET de Client Tools, en remplacement d'un produit de workbench graphique vieillissant, sur plus de 700 postes de travail.

J'ai également eu l'occasion de me rendre à Manille, aux Philippines, pour aider un client à migrer d'EAE 3R2 vers EAE 3R3 sur Microsoft Windows® et à implémenter un système frontal ASP.Net avec Client Tools. Cette interaction a permis de renouveler notre relation avec ce client et a débouché sur un contrat pour la migration vers AB Suite.

DA : À votre avis, y a-t-il certaines choses que les clients devraient savoir concernant le centre de support client et qu'ils ignorent ?

GP : Les clients ne savent pas nécessairement que tous les sites CSC travaillent désormais comme une équipe globale et que les analystes du monde entier sont souvent amenés à collaborer et à s'entraider sur une demande de support client (SR). Une telle structure nous permet de mutualiser nos connaissances et donc d'améliorer le service au client. >>

DA : Que pourraient faire les clients pour améliorer votre efficacité ?

GP : Fournir autant de renseignements que possible lorsqu'ils créent des SR électroniques et penser à noter les versions logicielles/les niveaux des corrections provisoires (IC) utilisées. Plus nous avons d'informations, mieux c'est ; cela nous permet d'évaluer la situation et d'apporter une réponse beaucoup plus rapidement.

DA : Avez-vous un autre message à faire passer à nos lecteurs ?

GP : La page « Documentation » de notre site de support propose une multitude d'informations, notamment les derniers manuels, livres blancs et tutoriels disponibles concernant AB Suite. Pour y accéder, connectez-vous à l'adresse support.unisys.com, choisissez « Documentation » en haut à gauche de l'écran et cliquez sur le lien AB Suite qui s'affiche sous « Application Development Solutions ». Pour des informations sur EAE, cliquez sur le lien EAE.



Le coin des ingénieurs : Trucs et astuces concernant Debugger

Par Suresh Ananthan, Chef d'équipe, Agile Business Suite Debugger, Unisys TCIS, et Paul Bourke Expert, Agile Business Suite Debugger, Unisys TCIS

Le module Debugger d'Agile Business Suite Developer recèle d'une multitude de fonctionnalités très pratiques, dont vous ignorez peut-être même l'existence. Pour vous aider à tirer le meilleur parti de Debugger, nous avons donc décidé de compiler cette liste de conseils et astuces utiles.

Conseil n°1 : Utiliser une copie de votre base de données de production avec Host Database Access (HDBA)

Il est important d'utiliser des données valides pendant le débogage puisque c'est le meilleur moyen d'identifier les problèmes liés aux données. Nous vous recommandons néanmoins d'éviter le débogage en utilisant HDBA avec votre base de données de production. Outre le risque de corruption des données de production, Debugger pourrait par inadvertance déclencher des situations de blocage susceptibles d'immobiliser le système de production.

C'est pourquoi il vaut toujours mieux utiliser une copie de votre base de données de production dans la fonction HDBA. Vous travaillerez ainsi à partir de données valides mais sans risquer de modifier accidentellement un quelconque élément actuellement en production.

À noter : dans les systèmes MCP, il faut définir la variable « Enable Host Database Access » sur « true » pour accéder à la base de données hôte dans Debugger. Pour savoir comment configurer HDBA sur les systèmes MCP, consultez ce [guide pratique](#).

Vous devrez également l'activer dans les environnements Windows, mais aussi faire pointer Debugger vers une base de données d'exécution Windows ou une copie de cette dernière.

Conseil n°2 : Utiliser plusieurs fenêtres d'observation

Le module Debugger propose un certain nombre de fenêtres d'observation permettant de surveiller les valeurs de certaines variables. AB Suite vous permettant d'avoir de multiples variables portant le même nom, copier la variable entièrement qualifiée

et la coller depuis la logique est la meilleure façon d'être sûr de surveiller la bonne variable.

Par ailleurs, la fenêtre d'observation « autos » présente toutes les variables de votre méthode actuelle. Elle se met automatiquement à jour à mesure que votre session Debugger se déroule ; ainsi, chaque fois que vous changez de méthode et que le jeu de variables utilisées change, ces changements sont répercutés dans cette fenêtre.

Conseil n°3 : Ajouter « GLB.STATUS » à la fenêtre d'observation

Un bon moyen de déceler les erreurs inattendues consiste à garder un œil sur GLB.STATUS pendant la session de débogage. En effet, en ajoutant « GLB.STATUS » à votre fenêtre d'observation, vous serez en mesure de détecter les éventuelles erreurs dans le code LDL+ pendant le débogage, par exemple l'ajout d'un enregistrement qui aurait échoué en raison d'un enregistrement dupliqué dans la base de données ou bien une tentative de suppression d'enregistrements qui n'existent pas dans la base de données.

Conseil n°4 : Ne pas oublier le verbe LDL « LOG »

Certainement l'un des verbes LDL les plus utiles, « LOG » vous permet d'enregistrer différents types de messages, comme les infos, avertissements ou erreurs dans la fenêtre de sortie Visual Studio. Il propose en plus la possibilité d'interrompre l'exécution du programme tout en laissant la session de débogage se poursuivre. Lorsque « If LOG DEBUG halt <message> » est utilisé, Debugger interrompt l'exécution de la logique après la ligne LOG LDL+ et affiche un message dans la fenêtre de sortie. Cela déclenche l'enregistrement d'un message uniquement dans la session de débogage. >>

Si vous souhaitez enregistrer les informations dans le runtime et Debugger, il faut remplacer la valeur « ALWAYS » par « DEBUG ». Cette fonctionnalité est particulièrement utile si vous souhaitez identifier et éliminer des conditions spécifiques qui ne devraient jamais se produire.

Conseil n°5 : Penser à l'option des points de trace (tracepoint)

Nouveauté de Microsoft Visual Studio 2005, l'option des points de trace permet une nouvelle utilisation des points d'arrêt. Le point de trace est en fait un point d'arrêt **associé à une action personnalisée**. Contrairement aux commandes LOG, qui deviennent un élément permanent du code, les points de trace ne sont visibles que par le développeur qui les crée. À l'instar des points d'arrêt, ils restent actifs jusqu'à ce que vous les supprimiez de la fenêtre des points d'arrêt.

Leur particularité est de ne pas interrompre l'exécution du programme (même si cela reste possible), mais d'effectuer une action spécifique. Donc, au lieu de provoquer l'arrêt, le point de trace envoie des informations dans la fenêtre de sortie Visual Studio pour vous permettre de visualiser votre message en laissant le débogage se poursuivre.

Vous avez par ailleurs la possibilité de personnaliser le message qui s'affiche dans la fenêtre de sortie et d'y faire figurer les informations de votre choix, comme le nom des variables, le nom des fonctions ou tout autre bloc de texte. Cette option s'avère donc extrêmement utile pour le débogage d'une boucle importante, où vous devez contrôler des milliers de valeurs.

Pour rappel, l'insertion d'un point de trace doit se faire depuis l'option points d'arrêt du menu contextuel.

Conseil n°6 : La fenêtre de la pile des appels indique la méthode appelée

Lorsque vous vous retrouvez avec une méthode qui en invoque une autre, la fenêtre de la pile des appels vous permet de retrouver dans quelles circonstances un élément est invoqué.

C'est particulièrement utile lorsque vous arrivez à un point d'arrêt loin dans votre session de débogage et que vous souhaitez voir comment vous en êtes arrivé là. Dans la fenêtre de la pile des appels, vous verrez que la première entrée correspond à la méthode en cours d'exécution, la suivante à la méthode qui l'a appelée, et ainsi de suite. Pour accéder à la partie du code depuis laquelle provient l'appel, il suffit de double-cliquer sur une entrée de la pile des appels.

La fenêtre de la pile des appels est également pratique pour toutes les méthodes dans le langage AB Suite qui sont indirectement invoquées. Par exemple, si vous choisissez d'imprimer un cadre (Frame), la fenêtre de la pile des appels indiquera quelle méthode a invoqué la méthode impression sur votre cadre, sous la méthode principale de ce dernier.

Conseil n°7 : Utiliser la bonne configuration pour les bonnes raisons

Si vous pouvez créer différentes configurations pour des objectifs spécifiques, nous vous recommandons, en ce qui concerne Debugger, d'utiliser la même configuration que vous auriez utilisée dans le déploiement du runtime. Dans le cas contraire, vous devrez répercuter tous les changements au niveau du runtime dans la configuration de Debugger. Alors que si vous utilisez la même configuration, vous serez toujours synchronisé.

Il existe toutefois quelques cas où cela n'est pas possible. Lorsque plusieurs utilisateurs de Debugger travaillent en même temps dans un environnement de serveur terminal, vous ne pouvez pas utiliser la même configuration. Et si vous voulez déployer le runtime Windows sur la machine sur laquelle tourne aussi le système de production, vous devez apporter des changements pour éviter les conflits entre les deux. >>

Pour en savoir plus sur les configurations et savoir comment les utiliser pour accélérer les cycles de démarrage et de construction de Debugger, reportez-vous à notre [guide pratique](#).

Conseil n°8 : Utiliser la fonction Automate pour automatiser les étapes de routine

Un peu comme la fonction Animate dans EAE Developer Test, la macro Automate de Debugger se charge d'effectuer pour vous des opérations à répétition ; vous pouvez croiser les bras et regarder le cheminement dans le code à mesure que chaque ligne est exécutée, jusqu'à atteindre un point d'arrêt. Cette possibilité est vraiment pratique quand vous avez besoin d'analyser une boucle et ne parvenez pas aux résultats voulus. Vous pouvez activer la macro Automate et la regarder faire sans avoir à intervenir.

Attention : la macro Automate n'est pas directement disponible. Vous devez d'abord créer la macro, puis configurer une barre d'outils. Pour connaître les pointeurs rattachés à cette opération, consultez notre [guide pratique](#).

Conseil n°9 : Utiliser l'option Edit and Continue pour permettre les changements à la volée

La fonction Edit and Continue (Modifier et continuer) permet de mener de front les tests et le développement ; vous avez ainsi un retour immédiat et la confirmation que la logique s'exécute comme vous le souhaitez. Comme son nom l'indique, cette fonction vous permet de corriger toute logique LDL+ non valide identifiée, de reprendre le test et de valider la correction à la volée, sans avoir à interrompre votre session Debugger.

Vous avez vous aussi un conseil à donner ? [Envoyez-le nous](#) et nous l'ajouterons à notre prochaine liste de [conseils et astuces](#).



Les migrations vers AB Suite vont bon train en France

Une nouvelle tendance prend la France d'assaut et, cette fois, il ne s'agit ni de mode ni de cinéma d'art et d'essai. Il s'agit des migrations vers Agile Business Suite qui font véritablement fureur auprès des clients français d'Unisys.

Nous nous sommes récemment entretenus avec les membres de notre équipe française pour comprendre les raisons de cet engouement pour les migrations ; leur réponse se résume en quelques points simples.

D'abord et avant tout, chaque migration suit une procédure bien définie, étape par étape, qui découpe le projet en différentes phases logiques de manière à garantir une progression continue. Tout commence par une analyse de l'environnement EAE utilisé par le client. Notre équipe française passe en revue le code source de l'application, réalise une migration vers un environnement AB Suite de test et remet au client un rapport contenant ses conclusions et recommandations.

Une fois tous ces points étudiés et validés, l'équipe élabore un plan formel pour effectuer la migration de l'environnement du client vers AB Suite.

Ce plan suit une trame éprouvée, dans laquelle chaque phase s'inscrit dans le prolongement de la précédente, pour que l'entreprise arrive au terme de cette migration bien préparée à tirer le meilleur parti de son investissement dans AB Suite. Voici le déroulement :

- Former les développeurs à l'utilisation de Visual Studio, aux nouveaux concepts de développement et à AB Suite
- Installer AB Suite sur le système de développement du client
- Migrer chaque modèle d'application
- Tester et valider les applications migrées
- Élaborer un plan pour faire passer les applications migrées en production

- Organiser des formations pour les développeurs plus approfondies pour qu'ils soient parfaitement familiarisés avec le nouvel environnement
- Accompagner le passage en production, avant et après que les applications soient rendues accessibles
- Contrôler l'utilisation par le client après la migration pour s'assurer qu'AB Suite fonctionne comme prévu

Notre équipe française recommande habituellement deux formations spécifiques à toute entreprise prête pour la migration. Ces cours portent sur le développement et la construction d'applications avec AB Suite et intègrent des modules consacrés à Visual Studio et aux concepts de l'orienté objet (OO).

Et pour les clients qui passent à AB Suite sur Windows, l'équipe propose une formation spéciale de deux jours à l'intention du personnel en charge de l'administration et de l'exploitation. Ce cours est particulièrement important car l'environnement AB Suite Runtime for Windows est très différent de celui d'EAE.

À projets uniques, enseignements uniques

Chacun des clients avec lesquels notre équipe française travaille a sa spécificité ; ainsi, chaque migration AB Suite offre de nouvelles perspectives et alimente notre base de connaissances. L'équipe a déjà accompagné sept clients dans leur migration vers AB Suite ; elle peut donc prodiguer une multitude de conseils précieux à toute entreprise qui envisagerait de sauter le pas : n'hésitez pas. >>

Si l'équipe a bien conscience des inquiétudes que peuvent éprouver les entreprises à l'idée d'une telle migration, les avantages sont bien trop intéressants pour être ignorés. Le fait est que, si les développeurs de l'organisation envisagent dans un premier temps de continuer à travailler comme auparavant dans EAE, AB Suite introduit de nouvelles capacités qu'ils pourront exploiter plus tard.

Et même si l'utilisation d'AB Suite exige un minimum d'apprentissage, notre équipe a pu constater que les entreprises étaient complètement à l'aise avec leur nouvel environnement de développement en à peine un mois ou deux. Parmi les nouvelles capacités que les développeurs ont le plus hâte de tester, citons les concepts de programmation OO.

Une seule et même approche utilisée dans le monde entier

Alors, que faut-il retenir du succès rencontré par notre équipe française dans la migration des entreprises vers AB Suite ? Le fait que la procédure de migration qu'elle applique n'est en aucun cas exclusive aux clients d'Unisys en France.

Il s'agit au contraire d'une méthodologie normalisée, appliquée par l'ensemble des équipes d'Unisys dans le monde entier. Donc, où que vous soyez implanté, votre migration – de l'analyse initiale à la formulation du plan de migration en passant par le programme de formation – suivra la même trame éprouvée que celle que nous venons de présenter dans cet article.

Vous avez des questions concernant le processus de migration ? Vous aimeriez commencer à planifier votre propre projet ? N'hésitez pas à contacter [Diane McGonigle](#).

Info Center

Voici quelques-uns des ajouts récents aux bibliothèques de guides pratiques, de livres blancs et autres informations utiles dans la section [Public Information](#) du site du support Unisys :

- **Guide pratique** : How To Automate Logic Execution in AB Suite Debugger (Comment automatiser l'exécution de la logique dans AB Suite Debugger) (**NOUVEAU**)
- **Guide pratique** : Use MULTI in Windows Runtime (Utiliser MULTI dans le runtime Windows) (**NOUVEAU**)
- **Guide pratique** : Use Non-Phased SQL in Windows Runtime (Utiliser le Non-Phased SQL dans le runtime Windows) (**NOUVEAU**)
- **Livre blanc** : External Classes in AB Suite 2.0 (Classes externes dans AB Suite 2.0) (**NOUVEAU**)
- **Utilitaire** : ABSLogAnalysis (pour l'analyse des performances transactionnelles en ligne des systèmes d'exécution) (**MIS À JOUR**)
- **Démonstration** : Business Integrator (**NOUVEAU**)

Les spécifications sont susceptibles d'être modifiées sans préavis.

© 2012 Unisys Corporation.

Tous droits réservés.

Unisys, le logo Unisys et ClearPath sont des marques déposées d'Unisys Corporation. Android est une marque de Google Inc. Apple est une marque déposée d'Apple Inc. aux États-Unis et dans d'autres pays. BlackBerry® et les marques, noms et logos associés sont la propriété de Research In Motion Limited et sont déposées et/ou utilisées aux États-Unis et dans d'autres pays à travers le monde. Microsoft, Visual Basic, 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.