



Developing Agility

Un boletín trimestral para clientes de EAE y AB Suite de Unisys

Contenidos

- Página 2 **Tesoros ocultos**
Alan Hood y sus amigos revelan funciones poco conocidas de EAE y Agile Business Suite que todos los desarrolladores deben conocer.
- Página 6 **Engineering Corner: Algo más que una bonita imagen**
Nuestra sesión de preguntas y respuestas con Ankur Kotwal de ACUS resalta la utilidad de los diagramas UML para desarrollar sus aplicaciones de Agile Business Suite.
- Página 9 **Caso de estudio de migración en empresas de telecomunicaciones**
Unisys Messaging Solution se ha migrado a Agile Business Suite y se encuentra en proceso de producción en varias empresas de telecomunicaciones. Obtenga más detalles en nuestro caso de estudio ampliado.
- Página 10 **Acompáñenos en Minneapolis este noviembre**
Conozca los programas especiales de incentivo de inscripción que se ofrecen para la Conferencia de Tecnología UNITE 2009.
- Página 10 **Calendario**
Consulte nuestro calendario para obtener la última información acerca de los eventos de 2009.

Tesoros ocultos

Por Alan Hood, administrador de programas de ingeniería, Unisys S&T, con la colaboración de Gary Taylor, Ching Lam, Andy Wardle y Nigel Tunnicliffe

Enterprise Application Environment (EAE) y Agile Business Suite ofrecen una experiencia de desarrollo de aplicaciones de increíble riqueza... todos sabemos eso. Sin embargo, como resultado hay varios aspectos de los productos que no son conocidos por mucha gente. Estos “tesoros ocultos” son cosas realmente asombrosas que se pueden hacer con EAE o Agile Business Suite ... con relativamente poco trabajo. En muchos casos, se logran aprovechando productos o funciones que ya existen en sus sistemas ClearPath o en otros entornos operativos. Las siguientes son sólo algunos de los tesoros ocultos que hemos descubierto con sólo explorar un poco.

Integre sus aplicaciones internas de EAE o Agile Business Suite con los productos de Microsoft mediante servicios Web

Prácticamente todas las interfaces de transacciones Ispec se pueden generar como un servicio web, y es muy fácil integrar esas interfaces de servicio web con los productos de productividad para atención al público, como Microsoft® Word y Excel® mediante su capacidad de generar secuencias de comandos de macros. Por ejemplo, al ingresar un número de cliente en un formulario en Word, se podría iniciar una macro que ejecute una serie de transacciones Ispec que busquen el nombre y dirección del cliente, y llene los espacios correspondientes de la carta de forma automática. Incluso puede recuperar las transacciones pasadas de la cuenta, saldo actual u otra información pertinente. Además, esto se produce de forma transparente: los usuarios de Word no necesitan saber que están conectados a un mainframe.

Mejore la integración de Excel mediante el formato de archivo de extracción “MAPPER”

Para las aplicaciones implementadas en el entorno ClearPath, el calificador de formato “MAPPER” o “MP” en una definición de archivo de extracción produce un archivo sin formato con un encabezado de Business Information Server (antiguamente conocido como MAPPER) y que contiene registros en columnas delimitadas por tabulaciones. ¿Por qué esto es un “tesoro oculto”? Porque resulta muy fácil importar datos delimitados por tabulaciones en una hoja de cálculo, como las de Microsoft Excel. En realidad, este método es mucho más fácil y menos vulnerable a los errores que derivar las columnas de las posiciones de caracteres en los registros del archivo. También puede usar el formato MAPPER para importar datos de archivos de extracción delimitados por tabulaciones.

Los archivos delimitados por tabulaciones son más flexibles que la información y producción tradicionales orientadas por posiciones. Por ejemplo, cuando un usuario final cambia el tamaño de una columna en la hoja de cálculo, no existe impacto sobre la capacidad de importar los datos si usa el formato de archivo de extracción MAPPER (siempre y cuando el orden de los datos siga siendo el mismo, y el tamaño de cada dato no supere el máximo permitido en su informe).

Proporcione varios usuarios para una sola aplicación

Probablemente ya saben que EAE y Agile Business Suite Client Tools les ofrecen una tremenda flexibilidad en la forma en que las interfaces de aplicación se presentan a los usuarios finales. Según su plataforma de tiempo de ejecución y las necesidades de sus usuarios, puede elegir usar una interfaz terminal basada en caracteres, de “pantalla verde”, aplicaciones Active Server Pages o ASP .NET basadas en web, interfaces gráficas del usuario de Visual Basic o Visual Basic .NET, servicios web, clientes personalizados y mucho más. He aquí el tesoro oculto: muchas personas no se dan cuenta de que pueden hacer TODAS estas cosas al mismo tiempo, desde la misma aplicación, con pocos o ningún cambio en la aplicación principal.

El secreto está en el “paquete”. Un paquete es una colección de Ispecs que se desean incluir en un tipo específico de interfaz de usuario. Al definir el paquete, se copian o arrastran los Ispecs que desea tener disponible en esa colección de interfaces y se especifica el nombre del generador cliente (que puede ser algo como GenerarFormularioASPpuntoNET). La mayoría de los desarrolladores crean un solo paquete y generan una interfaz de cliente. Sin embargo, se pueden tener muchos paquetes y generar múltiples tipos diferentes de clientes, para diferentes clases de usuarios, todo al mismo tiempo y sin hacer ningún cambio en la lógica de Ispec. >>

Piensen en el caso de una aplicación para bancos. En primer lugar, podría definir un paquete que contenga todos los Ispecs y genere una interfaz cliente de Visual Basic .NET para implementarse internamente para uso de los empleados del banco. Ahora, puede definir otro paquete que incluya sólo los Ispecs que permiten que los clientes se conecten, vean saldos de cuentas y transfieran fondos entre cuentas, y especificar un generador cliente ASP .NET para ese paquete. De esta manera, la aplicación genera e implementa automáticamente el proyecto de banca en línea ASP .NET, así como también los clientes internos Visual Basic .NET. Además, no hay que preocuparse ante la posibilidad de que un usuario de web se meta por accidente en las funciones administrativas, porque el paquete usado para definir su interfaz cliente basada en web no incluye esos Ispecs.

Llevándolo un paso adelante, puede ofrecer a los comerciantes la capacidad de acceder directamente a su sistema para validaciones de tarjetas de crédito en puntos de venta, agregando un paquete para generar interfaces de servicios web para los Ispecs que manejan esas transacciones. Nuevamente, la interfaz de servicios web de la tarjeta de crédito se genera automáticamente y de forma independiente de las demás interfaces de cliente, y no hay posibilidad de interferencia con los clientes de la banca en línea o internos de Visual Basic.

Este tesoro oculto, de mostrar varias interfaces de cliente para el mismo conjunto principal de transacciones puede agilizar a su organización y le ayuda a responder rápidamente a los cambios en las necesidades empresariales.

Ahorre tiempo y reduzca el gasto de las transacciones con varias “instancias” de Ispecs

Con frecuencia, cuando los desarrolladores definen interfaces cliente personalizadas, navegan por los Ispecs usando el mismo flujo de transacciones que usarían desde una terminal o emulador, y dejan que la aplicación sea la que determine cuál es el Ispec que se debe presentar. Cuando quieren ir explícitamente a un Ispec diferente, los desarrolladores usan una función de “Página 2” para que el Ispec actual recupere el Ispec objetivo. Sin embargo, la interfaz del Habilitador de Componentes le permite especificar el nombre del Ispec directamente: un “tesoro oculto” que evita los gastos adicionales de las transacciones.

Los beneficios potenciales incluyen:

- Una carga de transacciones significativamente reducida en el host: hasta 50 por ciento menos llamadas
- Costos menores para los usuarios finales cuando se cobran tarifas por transacción
- Tiempos de respuesta más rápidos... generalmente se invoca sólo un Ispec en lugar de dos, para ejecutar una función específica

La letra chica: Esta técnica por lo general sólo funciona con una aplicación “sin estado”, o como mínimo, en la que el Ispec inicial en una misma secuencia de transacciones sea sin estado.

¿A qué nos referimos cuando decimos “sin estado”?

En resumen, el estado de una transacción describe la necesidad de información sobre la transacción o el usuario que se debe pasar de uno al otro. En términos de EAE, esto por lo general se refiere al uso de GLB.WORK. Si sus transacciones dependen de datos que se pasan en GLB.WORK entre transmisiones, entonces su aplicación no es sin estado.

Dado que las aplicaciones basadas en servicios web y la arquitectura orientada al servicio (SOA) tienden a ser sin estado, cada vez más aplicaciones se están escribiendo de esta manera. En los casos de aplicaciones heredadas que originalmente se basaron en terminales o clientes internos de GUI (es decir, en los casos en que una aplicación “sabe quién eres”), éstas pueden convertirse para ser por lo menos parcialmente sin estado con relativamente poco esfuerzo. >>

Facilite la distribución de clientes “fat” con ClickOnce

Hoy en día, muchas organizaciones están usando de forma ventajosa las tecnologías de cliente delgado, que normalmente usan los exploradores de web. Una de las ventajas es la capacidad de eliminar la distribución e instalación de aplicaciones de cliente “fat”, que pueden necesitar una inversión en infraestructura significativa. Después de todo, en un explorador el usuario final simplemente hace clic en un vínculo y accede a la versión más actualizada de la aplicación. De manera que los métodos basados en exploradores que usan ASP y ASP.NET, entre otras tecnologías, evolucionaron fácilmente usando esta tecnología.

Sin embargo, si bien las tecnologías basadas en explorador siguen expandiéndose y proporcionando más capacidades, no pueden ofrecer la profundidad de las funciones o el desempeño de las aplicaciones “fat” o locales más tradicionales. Un buen ejemplo es Microsoft Outlook®, que permite el acceso basado en web pero no ofrece la misma gama de funciones, ni, lo que es más importante, la velocidad de una aplicación local.

Cliente “thin” o “fat”: cada método tiene sus ventajas y desventajas. Por lo tanto, puede ser conveniente ofrecer dos versiones del cliente: un método basado en explorador o “thin” para los usuarios móviles o a distancia y una versión local o “fat” para el personal interno o el que necesite un conjunto de funciones más amplio.

ClickOnce: La tecnología de implementación de Microsoft Visual Studio

Una forma de agilizar el proceso de la distribución de clientes “fat”, y que además es muy fácil de implementar para el cliente Client Tools Visual Basic .NET, es el uso de la tecnología de implementación ClickOnce de Microsoft. ClickOnce ofrece la capacidad para distribuir, descargar e instalar a nivel local de forma automática una aplicación cliente y archivos de respaldo desde un servidor web, es decir, un vínculo http link o servidor de archivos. Cada vez que se inicia la aplicación, también puede verificar y descargar una versión actualizada de forma opcional. ClickOnce resuelve la dificultad de la distribución y mantenimiento de clientes “fat”, sin tener que escribir código o emplear personal administrativo.

Se pueden encontrar detalles adicionales sobre la tecnología en [http://msdn.microsoft.com/en-us/library/142dbbz4\(VS.80\).aspx](http://msdn.microsoft.com/en-us/library/142dbbz4(VS.80).aspx)

Con ClickOnce, obtiene los beneficios de una aplicación cliente compleja de Visual Basic .NET, con la facilidad de implementación de un cliente basado en web.

Integre EAE o Agile Business Suite con Enterprise Output Manager

Si bien algunos desarrolladores de EAE y Agile Business Suite pueden estar usando Unisys Enterprise Output Manager para administrar, almacenar y distribuir la producción de informes, pocos conocen su capacidad de producir formularios con buen formato que incluyen elementos tales como imágenes y códigos de barras. Todavía son menos los que saben que con Enterprise Output Manager, pueden distribuir informes por correo electrónico, publicar informes generados en una página web protegida o convertirlos en PDF.

Otro tesoro oculto es la capacidad de integrar el diseñador de formularios de Enterprise Output Manager con EAE o Agile Business Suite Developer para crear un informe básico, y luego usar un DDA (atributo de datos dinámicos) para producir “formularios” realmente excelentes. ¡Es como tener el Habilitador de Componentes para informes!

Puede tener Enterprise Output Manager... y no saber esto. Con cada servidor ClearPath se incluye una licencia de usuario limitada, y muchos de nuestros clientes de EAE y AB Suite que se ejecutan en plataformas Windows, Linux® y UNIX® ya tienen Enterprise Output Manager. ¡Conózcalo!

Facilite la habilitación de producción móvil y de web mediante el software ClearPath ePortal

Quizás hayan oído hablar sobre las soluciones ClearPath ePortal Enterprise y ClearPath ePortal Business, que son motores especializados usados para desarrollar y administrar una gama de interfaces gráficas del usuario (GUI) para aplicaciones COBOL en el entorno operativo ClearPath MCP. >>

Ahora bien: aquí está el tesoro oculto: ClearPath ePortal también se puede usar para implementar y administrar interfaces de usuario basadas en web diseñadas con EAE o Agile Business Suite. Se pinta el formulario GUI en el pintor de EAE o Agile Business Suite y se define el paquete para la generación de cliente. En cuanto haya generado la aplicación, ClearPath ePortal puede asumir la administración del entorno IIS Web Server e implementar las interfaces móviles basadas en web o ASP .NET. ClearPath ePortal maneja lo que puede ser un proceso de mantenimiento bastante intensivo, y le permite usar la misma herramienta para administrar sus aplicaciones EAE, y también aplicaciones COBOL, en ClearPath.

Éste es otro caso en el que los clientes que ya tienen ClearPath ePortal seguramente dirán: “¡No sabía que podía hacer eso!”

¿Cuál es su “tesoro oculto”?

Las ideas presentadas en este artículo se refieren a formas de obtener un uso más intenso, mejor o más fácil de sus aplicaciones EAE o Agile Business Suite ampliando las interfaces del usuario. Además, cada una de las técnicas que les hemos contado están siendo usadas por uno o más clientes de EAE o Agile Business Suite... pero no por la mayoría.

Estamos seguros de que debe haber otros tesoros ocultos, quizás sugerencias para usar el Entorno de Desarrollo de forma más eficaz o de aprovechar al máximo las pruebas o generaciones parciales.

¿Tiene un tesoro oculto para mostrarnos? Si lo tiene, escríbanos a ABSuite@Unisys.com. Nos encantará conocerlo.

Estos “tesoros ocultos” son cosas realmente asombrosas que se pueden hacer con EAE o Agile Business Suite... con relativamente poco trabajo. En muchos casos, se logran aprovechando productos o funciones que ya existen en sus sistemas ClearPath o en otros entornos operativos.

Engineering Corner: Algo más que una bonita imagen

Recientemente hemos tenido la oportunidad de hablar con Ankur Kotwal acerca del desarrollo de diagramas de clase con Agile Business Suite. Ankur es el jefe técnico de Importaciones y Exportaciones en el Australia Centre for Unisys Software (ACUS). Nuestro objetivo era sencillo: averiguar más sobre las capacidades de diagramación de Agile Business Suite, incluso sobre la forma en que se diferencian de EAE y el valor que aportan al proceso de desarrollo de aplicaciones. Siga leyendo para conocer un excelente resumen de esta útil capacidad.

Developing Agility (DA): Empecemos por el principio. En Agile Business Suite, los diagramas de clase UML reemplazan a los gráficos en EAE. ¿Qué es un diagrama de clase UML? ¿En qué se diferencia y en qué es mejor que un gráfico de EAE?

Ankur Kotwal (AK): El lenguaje de modelado de objetos (UML, por sus siglas en inglés) es un lenguaje de modelado estándar de la industria que se usa para especificar, visualizar, desarrollar y documentar la estructura y el diseño de un sistema de software orientado a los objetos (OO). Los diagramas de clase, que representan uno de los 13 tipos estándar de diagramas UML, muestran las colaboraciones y relaciones entre una serie de clases e interfaces.

Un gráfico EAE es una herramienta personalizada para visualizar un sistema EAE que usa notación no estándar. Puede crear objetos en un gráfico EAE, asignar relaciones y ver cuándo las relaciones se han codificado (la letra pasa de minúscula a mayúscula cuando se agrega la lógica).

Los diagramas de clase UML de Agile Business Suite ofrecen muchas más funciones que los gráficos EAE porque son más que una simple vista de la aplicación. Los diagramas de clase UML se pueden usar para modelar un sistema desde la base. Además, los cambios realizados a las clases en el diagrama se aplican al sistema en tiempo real, una poderosa capacidad de “diagramación activa” que existe exclusivamente en las herramientas de Agile Business Suite.

El desarrollador puede incluir en un diagrama dado mucho o poco de un sistema, según lo desee. Con los diagramas UML y de Agile Business Suite, se desarrollan prototipos y se trabaja a nivel de modelo, y se ven los cambios incorporados inmediatamente en la aplicación de trabajo en su entorno de desarrollo.

DA: ¿Un diagrama de clase puede ayudar a realizar mejoras en una aplicación existente... una que se haya importado a Agile Business Suite desde EAE? Si se puede hacer, ¿de qué manera puede lograrse esto?

AK: Ante todo, los diagramas de clase permiten que los desarrolladores visualicen los componentes de la aplicación y sus interacciones. Al hacerlo, un usuario puede ver claramente las dependencias que existen entre las clases. De manera que los diagramas de Agile Business Suite ofrecen una buena descripción visual de alto nivel de la aplicación EAE migrada. Algunos clientes nos han dicho que desean aprovechar los conceptos OO, pero que esto se hace difícil porque no ya no hay desarrolladores en la planta de personal que realmente entiendan la aplicación EAE. Los diagramas UML pueden ayudar a superar este problema, al proporcionar una documentación visual de la aplicación migrada.

Además, los diagramas pueden ayudar a los desarrolladores a hacer mejoras en la aplicación. Por ejemplo, los diagramas de clase pueden ayudar a identificar atributos duplicados en aplicaciones EAE migradas. Cuando se detectan, estas redundancias se pueden eliminar creando una nueva clase para la funcionalidad común que se reutiliza en varios lugares. Esto reduce el mantenimiento permanente y mejora el diseño general del sistema. >>

DA: *Si alguien está desarrollando una nueva aplicación en Agile Business Suite, ¿debe empezar por dibujar un diagrama de clase? Si esto no es así, ¿cómo debe empezar?*

AK: En Agile Business Suite, las clases son los ladrillos básicos de todas las aplicaciones. Por lo tanto, todas las aplicaciones de Agile Business Suite empiezan como un conjunto de clases. Los diagramas de clase ayudan a los desarrolladores a modelar rápidamente clases y relaciones complejas en forma de diagrama, lo que permite el diseño de un modelo de alto nivel para su aplicación en un plazo muy breve. Dado que el diagrama de clase AB Suite es un editor real, todos los elementos creados o modificados en el diagrama de clase realmente se modelan en el sistema.

Se debe tener en cuenta que mientras está creando el modelo con el diagrama Agile Business Suite, al mismo tiempo también está desarrollando su aplicación. Muchas otras herramientas de modelado se limitan a crear el modelo, y luego los desarrolladores deben usar otras herramientas para desarrollar la aplicación. Sin embargo, con Agile Business Suite, la creación del modelo y el desarrollo de la aplicación se producen simultáneamente. Debido a la forma en que AB Suite desarrolla la aplicación directamente desde el modelo, podemos decir que el modelo es la aplicación.

DA: *La mayoría de los clientes están acostumbrados a escribir su aplicación definiendo pantallas y usando estructuras LDL para definir la lógica. ¿De qué manera los diagramas de clase cambian este proceso?*

AK: Antes de diseñar pantallas y lógica, un desarrollador debe tener clases y atributos. Las entidades de datos en Agile Business Suite se modelan como atributos persistentes, que son miembros de clases. Todas las clases y atributos (junto con sus propiedades, como la persistencia) se pueden modelar a través de un diagrama de clase. El diagrama de clase no necesariamente cambia la secuencia del desarrollo de la aplicación, sino que más bien permite que el desarrollador complete la fase de modelado más rápido.

DA: *¿Cuáles son los beneficios reales de usar un diagrama de clase?*

AK: La utilidad de los diagramas de clase consiste en que proporcionan una herramienta de desarrollo visual rápida. También hacen que sea muy fácil ver o visualizar el diseño de una aplicación determinada. Se pueden usar para visualizar de forma diagramada la parte del sistema que uno desea ver. De esta manera, puede haber sistemas grandes compuestos por varios diagramas de clase.

DA: *¿Los diagramas de clase se usan principalmente para el diseño de aplicaciones? ¿Para documentación?*

AK: Los diagramas de clase de Agile Business Suite cumplen varias funciones. No sólo ayudan en el diseño de la aplicación, sino que también constituyen la base de la documentación de una aplicación. Como lo he mencionado antes, dado que los diagramas de clase de Agile Business Suite realmente son un modelo de la aplicación misma, ofrecen verdadera capacidad de seguimiento entre el diseño y la implementación de la aplicación. A diferencia de las herramientas tradicionales de modelado y desarrollo, el diseño y la implementación nunca pueden estar fuera de sincronización con Agile Business Suite. De manera que no habrá más falta de coincidencias entre el código y la documentación: Agile Business Suite se autodocumenta.

DA: *¿Los diagramas de clase ayudan a depurar una aplicación Agile Business Suite?*

AK: No, los diagramas de clase no se usan en la etapa de depuración del ciclo de desarrollo de las aplicaciones. Por otro lado, si un desarrollador está intentando detectar la causa de un problema, los diagramas de clase pueden hacer que sea mucho más fácil identificar los elementos afectados por el tema y posiblemente puedan proporcionar un indicio de dónde está la raíz del problema.

DA: *¿Los diagramas de clase son útiles para crear una solución distribuida o basada en arquitectura orientada al servicio (SOA)?*

AK: Los diagramas de clase representan una sola aplicación, no un ecosistema. Sin embargo, se los puede usar para documentar las interfaces externas de una aplicación, lo que es necesario en un entorno de SOA.

DA: *¿Los diagramas de clase se pueden usar para planificar un entorno de red?*

AK: No, no sirven para el modelado de la interacción de redes. >>

DA: *¿Qué es eso que llaman “XMI”? ¿Cómo puedo usar otras herramientas, como Rational RSM, con Agile Business Suite?*

AK: XML Metadata Interchange (XMI) es una notación estándar de la industria para el intercambio de información de UML. Rational RSM es una aplicación popular de modelado de UML que utiliza XMI como una forma de intercambiar modelos UML con Agile Business Suite, y también con otras herramientas. Como tal, una aplicación se puede modelar en base a Agile Business Suite y se puede transferir a RSM mediante XMI (y viceversa). Para conocer más sobre la importación de modelos a Agile Business Suite a través de XMI, pueden leer mi [artículo](#) en una edición de archivo de *Developing Agility*.

DA: *¿Qué es lo que más cuesta acostumbrarse al usar diagramas UML cuando un desarrollador pasa de EAE a Agile Business Suite?*

AK: Bien, la herramienta de diagramas de clase de Agile Business Suite es muy similar a otras herramientas de diagramación. De manera que yo recomendaría a los desarrolladores de EAE que simplemente se atrevan a probarla. Probablemente lo más difícil sería la falta de familiaridad con las notaciones de UML. Sin embargo, es asombroso lo rápido que las personas pueden aprender una vez que se acostumbran.

Además, vale la pena aprender más sobre los conceptos OO y UML. Los desarrolladores aprovechan mejor los diagramas de clase cuando se aplican los principios OO al proceso de desarrollo de aplicaciones. Además se deben tener en mente los beneficios de los diagramas: la capacidad de mantener el diseño y la aplicación totalmente sincronizados y la velocidad con la que se pueden desarrollar e implementar cambios trabajando al nivel de modelo.

Responder rápidamente ante los cambios siempre ha sido una característica distintiva de las herramientas de desarrollo de Unisys... y Agile Business Suite continúa con esa tradición. Después de que se desarrolla el modelo y se implementa la aplicación, el modelo sigue siendo útil, porque sigue usándose para implementar cambios y mejoras.

Agradecemos a Ankur por proporcionarnos una descripción excelente de los diagramas UML de Agile Business Suite. Si tienen preguntas, pueden enviarnos un mensaje de correo electrónico a ABSuite@unisys.com.

Algunos clientes nos han dicho que desean aprovechar los conceptos OO, pero que esto se hace difícil porque no ya no hay desarrolladores en la planta de personal que realmente entiendan la aplicación EAE. Los diagramas UML pueden ayudar a superar este problema, al proporcionar una documentación visual de la aplicación migrada.

Caso de estudio de migración en empresas de telecomunicaciones

Recientemente publicamos un caso de estudio completo que describe cómo el Grupo Industrial de Comunicaciones Globales de Unisys migró su Messaging Solution a Agile Business Suite. El proyecto fue una colaboración entre equipos de Unisys del mundo entero y el caso de estudio incluye algunos detalles jugosos acerca del proceso de migración en sí.

Unisys Messaging Solution es utilizado por más de 90 empresas de telecomunicaciones del mundo entero, entre las cuales se cuentan varios operadores de Nivel 1, para ofrecer servicios de mensajería de voz a clientes residenciales y comerciales. En total, la solución admite un total estimado de más de 200 millones de buzones de correo y más de 350 millones de transacciones de mensajería por día. Lo que es más, cada instalación empieza con el producto principal y se adapta para satisfacer las necesidades de cada cliente. Esto significa que se ejecuta una migración por cada implementación.

Tal como ocurre con muchas organizaciones de desarrollo, el grupo de Comunicaciones Globales quería preservar el valor y la estabilidad de su solución de mensajería de voz, que era de alto desempeño y rica en funciones, modernizando al mismo tiempo la tecnología de implementación y desarrollo subyacente. Pudo hacerlo trasladando Messaging Solution a Agile Business Suite, y aprovechando la migración segura.

Espero que puedan leer el [caso de estudio](#) completo en la eCommunity para conocer más sobre el proceso y sobre la forma en que el Grupo Industrial de Comunicaciones Globales de Unisys pudo trasladar el desarrollo a la plataforma moderna de Agile Business Suite, realizando al mismo tiempo una transición libre de problemas para sus clientes.

Los beneficios principales incluyen:

- Un mayor valor comercial gracias a la capacidad para obtener un rendimiento continuo sobre la inversión de Messaging Solution
- Estabilidad permanente de la solución mediante la migración segura a Agile Business Suite sin interrupciones observables para los clientes
- Mejor productividad, que incluye una reducción del 80% en el tiempo transcurrido de migración de código en comparación con el enfoque manual
- Mayor agilidad de desarrollo a largo plazo y menor tiempo de salida al mercado mediante el uso de herramientas modernas, basadas en modelos

Para conocer consejos sobre la planificación de la migración a Agile Business Suite, consulte la edición [3Q2008](#) de *Developing Agility*.

Acompáñenos en Minneapolis este noviembre

La Conferencia UNITE 2009 se realizará del 8 al 11 de noviembre de 2009 en el Hyatt Regency Minneapolis. Éste es el momento para aprovechar uno de los [programas especiales de incentivos de inscripción](#) de UNITE. Hay incentivos para todos: desde los que vienen asistiendo desde hace años hasta los novatos. Visite el sitio web de la UNITE (unite.org) para conocer todos los detalles. No se demore: algunas ofertas exigen que se inscriba para el 31 de julio de 2009.

¿Por qué asistir a la UNITE? Para aprender cosas nuevas y reunirse con sus colegas, por supuesto. Tal como ha ocurrido en años anteriores (y si se obtiene la aprobación de la UNITE), los organizadores de sesiones de EAE ofrecerán muchas oportunidades prácticas de probar Agile Business Suite 2.0 en un laboratorio. Simplemente no hay mejor manera de obtener montones de información con sólo invertir unos días de su tiempo.

Cuando asiste a la UNITE, está participando en una organización dedicada en un 100 por ciento a proporcionar un foro para que los usuarios de Unisys intercambien ideas e información, algo que va mucho más allá de la conferencia anual. Haga oír su voz presentando sus requisitos, trabajando como voluntario en un consejo de asesoramiento o fuerza de tarea, o participando en el proceso de elecciones. ¡Incluso puede ser funcionario de la UNITE!

Todos los años, las sesiones de EAE y Agile Business Suite en UNITE tienen gran cantidad de asistentes. Si se tiene en cuenta la situación económica, no hay mejor valor. ¡Haga entonces sus planes hoy mismo!

Calendario

Existen muchas oportunidades valiosas de aprendizaje disponibles para usted. No deje de consultar la sección Webcasts y eventos de la [eCommunity](#) para conocer la información más reciente.

Qué	Dónde	Cuándo
Novedades de tecnología : reuniones de usuarios de Agile Business Suite y EAE	Amsterdam, Holanda Milton Keynes, Reino Unido	16 de junio de 2009, 18 de junio de 2009
Reunión de dos días de usuarios de Agile Business Suite y EAE	St. Paul de Vence, Francia	Del 2 al 4 de noviembre de 2009
Conferencia anual de tecnología UNITE	Hyatt Regency Minneapolis, Minneapolis, Minnesota	Del 8 al 11 de noviembre de 2009

Las especificaciones están sujetas a cambios sin previo aviso.

© 2009 Unisys Corporation.
Todos los derechos reservados.

Unisys y el logotipo de Unisys son marcas registradas de Unisys Corporation. Microsoft, Excel y Outlook son marcas registradas de Microsoft Corporation. Todas las demás marcas y productos a los que se hace referencia en este documento se reconocen como marcas registradas o marcas comerciales de sus respectivos dueños.