



Developing Agility

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

Contenidos

- Página 2 **Preguntas y respuestas de clientes con la aseguradora holandesa Movir**
El ingeniero de sistemas, Ron Thakoer, habla de la migración de su organización a Agile Business Suite.
- Página 4 **Engineering Corner: Llevar las aplicaciones migradas a un nuevo nivel modelándolas para su reutilización**
Grant McCauley revela cuatro formas de aumentar la capacidad de reutilización en los sistemas existentes una vez finalizada la migración a Agile Business Suite.
- Página 9 **Silverlight presenta una nueva opción de interfaz de usuario**
Vea una breve descripción de cómo se puede usar esta nueva tecnología en una aplicación EAE o Agile Business Suite.
- Página 12 **En recuerdo**
Conmemoramos el fallecimiento de dos reconocidos expertos en EAE.
- Página 12 **Calendario**
Consulte nuestro calendario para obtener la última información acerca de los próximos eventos.

Preguntas y respuestas de clientes con la aseguradora holandesa Movir

Uno de los objetivos clave de *Developing Agility* es mantener informados a los clientes de EAE sobre cómo les ha ido a sus colegas con la planificación e implementación de Agile Business Suite. Este artículo prosigue con ese esfuerzo mediante un análisis sobre las actividades y logros de Movir, una compañía de seguros especializada ubicada en Holanda que ofrece cobertura por discapacidad a profesionales holandeses de las áreas del comercio y la salud.

Hace poco hablamos con Ron Thakoer, ingeniero de sistemas de Movir, sobre la reciente migración de la empresa a Agile Business Suite que se focalizó en el entorno operativo ClearPath MCP. Siga leyendo para saber qué requirió el proceso de migración de Movir de EAE a Agile Business Suite y para conocer la opinión de Ron sobre los principales beneficios que espera hacer realidad como nuevo usuario de Agile Business Suite.

Developing Agility (DA): Háblenos del entorno de TI en Movir. ¿Cuántas aplicaciones tienen? ¿A qué funciones de negocio brindan apoyo? ¿Más o menos qué magnitud tienen? Además, ¿cuál es el tamaño de su equipo de desarrollo?

Ron Thakoer (RT): Movir opera tres aplicaciones de EAE principales, dos de las cuales apoyan directamente el negocio, mientras que la otra es una herramienta personalizada de control de versiones relativamente pequeña. Nuestra principal aplicación, MOSYS, contiene 120 ISPECS y 200 INFORMES, mantiene una base de datos de producción de alrededor de 10 GB y hay aproximadamente 110 usuarios finales que trabajan con ella.

La segunda es una aplicación mucho más pequeña que se parece mucho al sistema MOSYS, pero que está específicamente diseñada para apoyar servicios de seguros para pequeños empresarios.

Tenemos cinco desarrolladores de Agile Business Suite.

DA: ¿Qué beneficios comerciales espera concretar con la ayuda de Agile Business Suite?

RT: Empezamos a funcionar en junio de 2009, por lo que el primer paso después de la migración fue darle tiempo a nuestro equipo de TI para familiarizarse con el nuevo entorno. Después de este período de introducción, esperamos aprovechar Agile Business Suite para optimizar ciertos procesos, por ejemplo, ajustar la integración entre Agile Business Suite y nuestras aplicaciones para atención al público.

Asimismo, hemos aplicado planes para usar Agile Business Suite para ampliar el alcance de una capacidad que desarrollamos en Microsoft® Word para automatizar la producción de correspondencia. Además, usamos Infolmage, un sistema de administración de documentos de Unisys, y esperamos simplificar también el proceso de integración de esta herramienta con las aplicaciones de Agile Business Suite.

DA: ¿Qué nos puede decir de los beneficios técnicos?

RT: Por supuesto, esperamos que Agile Business Suite nos ayude a mejorar la productividad de los desarrolladores, pero lo que es más importante, estamos deseosos de trabajar en un entorno orientado a los objetos (OO). Con OO, podemos optimizar la estructura de nuestras aplicaciones, lo que mejorará nuestra capacidad de mantenerlas y mejorarlas con rapidez y eficiencia.

DA: Antes de la migración a Agile Business Suite, ustedes migraron el front-end gráfico en su aplicación EAE de PowerClient (Graphical Interface Workbench) a Microsoft Visual Basic® .NET. ¿Por qué eligieron Visual Basic .NET? ¿Los usuarios finales notaron algún cambio después de que implementaron Agile Business Suite?

RT: Habíamos estado usando una interfaz de usuario final integrada de PowerClient y Visual Basic 6 para nuestra aplicación EAE. Sin embargo, con el soporte principal para ambos productos llegando a su término, sabíamos que era el momento de actualizar las tecnologías por las más recientes. Y como ya estábamos usando Visual Basic 6, tenía sentido desde una perspectiva operacional migrar a Visual Basic .NET usando el generador incluido en EAE y Agile Business Suite Client Tools. También agregamos código personalizado mediante el uso de Visual Basic .NET a la interfaz de usuario final generada. >>

Hemos logrado grandes aumentos de eficiencia por el lado del usuario final. Por ejemplo, cuando se invoca una función de Visual Basic .NET, nuestros usuarios ahora pueden permanecer en la misma ventana. Al modernizar esta interfaz primero, nuestros usuarios pudieron usarla con las aplicaciones de EAE durante algunos meses antes de que migráramos a Agile Business Suite. Igualmente, pudimos migrar nuestras interfaces gráficas de usuario actualizadas de Visual Basic .NET a Agile Business Suite sin ningún problema y los usuarios no percibieron ninguna diferencia.

DA: *¿Qué características de Agile Business Suite tuvieron un papel fundamental al influir en su decisión de migración?*

RT: El factor más determinante fue la sólida funcionalidad de integración de Agile Business Suite. Sin duda que estas capacidades son valiosas ahora y van a ser más y más importantes para nuestra empresa cuando enfrentemos desafíos futuros.

DA: *Ustedes partieron con Agile Business Suite 1.2 en junio de 2009. ¿Qué experiencias ha tenido en su uso de Agile Business Suite en un entorno de producción? ¿Qué opinan los desarrolladores de Agile Business Suite Developer?*

RT: La migración a Agile Business Suite nos ha permitido mantener la calidad de nuestras operaciones sin poner en riesgo el rendimiento. Todos los miembros de los diversos equipos de Unisys con los que trabajamos nos brindaron una valiosa asistencia durante el proyecto. Agradecemos de manera especial la capacidad de respuesta de los ingenieros de Agile Business Suite, que realizaron las correcciones rápidas que nos ayudaron a evitar retrasos importantes.

Desde un punto de vista de desarrollo, algunos de nuestros desarrolladores ya están listos y conocen Agile Business Suite Developer, y esperamos que los demás comiencen a experimentar todos los beneficios bastante pronto.

DA: *¿Qué aprendieron del proyecto de migración que en su opinión puede ser útil para otras organizaciones de TI? ¿Cuáles fueron las actividades más importantes durante el proyecto de migración?*

RT: Tratamos la migración del mismo modo que haríamos con todos los demás proyectos, y creo que ese es un importante modo de pensar. Pensar de este modo nos ayudó a reunir los recursos internos, preparar el caso de negocios y contratar al personal de soporte de Unisys necesario para dirigir el proceso.

En general, la actividad más importante fue permitir a nuestros usuarios finales probar las aplicaciones migradas. Además usamos los servicios y las herramientas de Unisys Business Application Test Manager (BATMan) para reproducir días completos de transacciones de la aplicación EAE en la aplicación Agile Business Suite migrada, lo que nos brindó la tranquilidad de haber cubierto la mayoría de las áreas de la aplicación antes del lanzamiento. Obviamente, las pruebas toman tiempo, pero es un paso extremadamente crucial para garantizar una migración sin problemas.

DA: *¿Cuáles son sus planes para nuevo desarrollo con Agile Business Suite?*

RT: Como nuestra aplicación interna más importante, MOSYS se actualiza y amplía constantemente para satisfacer los requerimientos emergentes de los usuarios finales. Como tal, nuestra primera actividad con Agile Business Suite es usar Agile Business Suite Developer para crear nuevas funciones innovadoras para MOSYS.

BATMan garantiza una transición sin problemas a Agile Business Suite

Entregado como servicio, BATMan automatiza el desarrollo de secuencias de comandos de prueba, que capturan transacciones de la vida real de EAE o Agile Business Suite, para ayudarle a probar aplicaciones en línea antes de su lanzamiento. BATMan reproduce estas transacciones en un entorno de prueba de destino y ofrece la capacidad de confirmar una reproducción exitosa al validar la coincidencia de los resultados con las transacciones originales. Al hacerlo, BATMan entrega una medida de confianza adicional cuando migra sus aplicaciones de entornos de prueba a entornos de producción.

Llevar las aplicaciones migradas a un nuevo nivel modelándolas para su reutilización

Por Grant McCauley, jefe técnico de modelos de Agile Business Suite, GTC Australia, Unisys Systems & Technology

Los beneficios de las capacidades de modelado de UML, como las que ofrece Agile Business Suite, son muy claros cuando desarrolla una aplicación nueva. Pero hay un valor que puede obtenerse para las aplicaciones existentes, en especial para quienes migran de EAE a Agile Business Suite. Este artículo es una revisión de maneras simples y directas de usar las funciones de modelado en Agile Business Suite para facilitar el soporte de sus aplicaciones EAE migradas y facilitar la reutilización del código EAE migrado. Al aprovechar estos consejos, no sólo aumentará la productividad a largo plazo, sino que obtendrá experiencia práctica en modelado al mismo tiempo que trabaja con sus sistemas EAE conocidos (anteriores).

Lo primero es lo primero. Cuando migra a Agile Business Suite, su modelo EAE se convierte en un modelo Agile Business Suite, y muchos de los ladrillos básicos se crean automáticamente para usted. Objetos, clases, métodos, atributos, todos los buenos elementos orientados a los objetos, estarán ahí para usted cuando migre. Hemos tratado los aspectos básicos de la orientación a los objetos en anteriores secciones Engineering Corner, por lo que no entraré en detalles en esta ocasión. No obstante, sí le recomiendo que vuelva a consultar esos artículos cuando tenga tiempo.

Comenzando con el modelo de Agile Business Suite de su aplicación, hay muchas formas de facilitar su comprensión, mantenimiento, uso y reutilización. Veo las oportunidades divididas en las siguientes áreas generales:

1. Analizar
2. Organizar
3. Documentar
4. Desarrollar

Demos un vistazo a cada una con algo más de detalle.

Analizar

Una vez que su modelo migra a Agile Business Suite, un buen primer paso es de hecho retroceder, y ver lo que tiene. No es necesario hacer cambios simplemente por hacer un cambio, así que dese algo de tiempo para analizar su modelo desde “todos los ángulos”. Use las fichas (propiedades, dependencia, etc.) para explorar las distintas vistas del modelo disponibles a través de Agile Business Suite Developer. Revise las dependencias que surgen, además de cómo se definieron las propiedades, como tipo, propietario, herencia y multiplicidad, para los constructores de los objetos. También puede probar las funciones de cuadro de diálogo Search (Buscar) (Alt+S) y Quick Navigator (Navegador rápido) (Alt+G) para analizar más a fondo el sistema y modelo migrados.

Organizar

Cuando tenga una buena idea de cómo se ve el modelo de su aplicación en Agile Business Suite, es hora de organizar. Por organizar, me refiero a los pasos que facilitan la comprensión del modelo, y a la larga su reutilización y reestructuración. Nuevamente, no comienza desde cero porque sus áreas funcionales y actividades de EAE se transfieren a Agile Business Suite durante la migración.

Hay tres herramientas que le recomiendo que use para organizar: carpetas, diccionarios y diagramas. >>

Actualización en orientación a los objetos

Para obtener más información sobre conceptos de orientación a los objetos, dé un vistazo a los artículos de Engineering Corner:

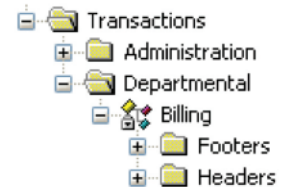
- [Cuando aplicar un estereotipo es algo bueno](#)
- [Encapsulación](#)
- [Polimorfismo](#)



Carpetas

La metáfora de carpeta es un excelente medio de ordenar más su modelos y debe ser el primer paso que tome para organizar. Use las carpetas de Agile Business Suite del mismo modo que usaría una carpeta física: para agrupar elementos relacionados. Por ejemplo, podría desear:

- Colocar todas las transacciones relacionadas con un área del negocio en particular en una carpeta
- Crear una carpeta que contenga todas sus clases externas o todos los informes o transacciones administrativas
- Usar carpetas dentro de un informe para organizar los marcos de encabezado y pie



Las carpetas son tan flexibles que la misma entidad se puede usar en varias carpetas y éstas se pueden organizar dentro de otras carpetas.

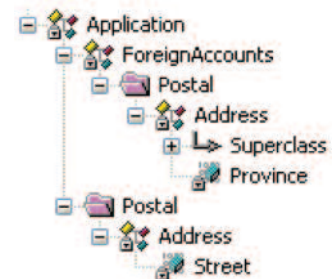


Diccionarios

En Agile Business Suite, el diccionario es un tipo especial de carpeta que se usa específicamente para organizar objetos que sirven de definición para otros objetos, como clases que desee reutilizar fácilmente en muchas áreas de la aplicación. A diferencia de EAE, usted puede tener tantos diccionarios como desee con Agile Business Suite, y pueden existir en varios niveles en el modelo (cuando se aplican a todos los objetos definidos bajo ese nivel).

¿Por qué querría tener más de un diccionario? Bueno, por exactamente las mismas razones que una carpeta, pero con el beneficio adicional de que los diccionarios facilitan la reutilización estableciendo automáticamente la relación “heredada” cuando el nombre de un atributo o una variable coincide con un objeto en un diccionario.

Por ejemplo, tener varios diccionarios podría tener sentido si combina la funcionalidad de dos o tres aplicaciones o desea poder redefinir ciertos términos o definiciones en un área local del modelo. En un lugar podría tener un diccionario que defina una dirección con calle, departamento, ciudad, estado y código postal. En otra parte del modelo, que se ocupa de otros países, podría redefinir la dirección para incluir un código postal largo, un país y una provincia. En cada lugar sería legítimo hacer referencia a una dirección como un solo objeto y automáticamente adjuntarla al que le corresponde.



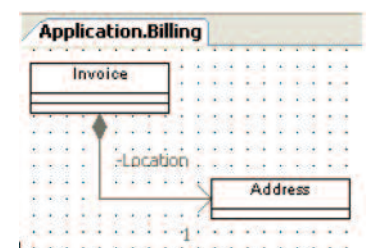
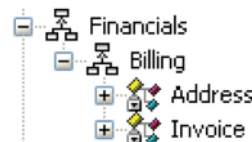
Con EAE, las definiciones en un diccionario estaban limitadas a sólo tipos primitivos, como cadena y número. En Agile Business Suite también puede haber clases que incluyen atributos, lógica y presentación uniforme. Al hacer entradas en un diccionario, usted proporciona un patrón para los elementos comúnmente usados en su modelo, lo que puede terminar siendo su primer paso hacia una reutilización real.



Diagramas

Los diagramas son otra herramienta para organizar. Como una carpeta, puede arrastrar elementos relacionados a un diagrama, lo que crea una imagen UML de los contenidos. Aunque no haya líneas (dependencias/relaciones) que inicialmente conecten los elementos en el diagrama, todavía tiene una ventaja porque tiene una agrupación lógica de elementos relacionados. Como se dice, “una imagen vale más que mil palabras”.

Por último, usar cualquiera de estos consejos de organización no le hará daño a su aplicación. Simplemente son herramientas para ayudar a los desarrolladores a traer algo de orden a su mundo. >>



Documentar

Documentar es simplemente eso: acciones que ayudan a aclarar a cualquiera que entre en contacto con su sistema cuál es el propósito de todos los objetos, la lógica y así sucesivamente. Los diagramas UML son sólo una forma de describir un sistema, crear vistas de una aplicación y mostrar las relaciones estructurales entre clases.

Agile Business Suite también tiene unas excelentes funciones para “documentar sobre la marcha” que facilitan mucho ingresar comentarios y descripciones cuando diseña y desarrolla. Las mismas funciones se pueden usar con el sistema migrado, véalas como “documentar al descubrir”, para formalizar su conocimiento institucional de las diversas funciones en un sistema que ya conoce muy bien. Y puede agregar documentación para diversos públicos. Información sobre cuál función de negocios apoya un objeto y los planes futuros de extender una función podrían ser útiles para los diseñadores. Los usuarios finales se benefician de la documentación que explica cómo usar un Ispec o la fuente de datos de lista.

Comience por ingresar la documentación en el nivel de carpeta y ábrase paso hacia más detalles según lo permita el tiempo y el interés. O si está trabajando en un área del modelo y descubre algo interesante, abra la ventana de documentación en ese punto y anote sus hallazgos. Una función similar a WordPad hace que el proceso sea muy sencillo.

Finalmente, Agile Business Suite tiene una nueva capacidad de extraer documentación de la aplicación. Páginas de comentarios, que son guías de referencia navegables basadas en HTML, proporcionan una solución más sólida y fácil de usar en comparación con la producción estándar de especificación de impresión de EAE. La documentación en línea es fácil de mantener cuando se genera automáticamente desde su sistema cuando es necesario.

Desarrollar

Hasta aquí, cada acción que he recomendado no ha modificado realmente el modelo de su aplicación. Analizar, organizar y documentar simplemente facilita su comprensión. Desarrollar un sistema existente tiene que ver con refinar y extender la aplicación, lo que lleva al centro del modelado para reutilización. Y los siguientes son cuatro pasos que puede tomar para hacerlo: eliminar, encapsular, descomponer y generalizar.

Eliminar

Eliminar tiene que ver con refinar los roles de las entidades en una aplicación: eliminar las capacidades innecesarias o redundantes que ya no tienen ningún propósito. Debido a que EAE ofrece sólo entidades de alto nivel, forzosamente habrá oportunidades para refinar. Por ejemplo, pueden haberse usado Groups (Grupos) para definir las estructuras sin requerir de la capacidad de serialización de cadenas o se puede haber usado Insertables para formularios sin la necesidad de la inserción de código de macro especial. Eliminar las capacidades sin utilizar no sólo simplifica la definición de una entidad, sino que mejora su capacidad de ser reutilizadas.

Esté alerta a los objetos en Agile Business Suite que no usen su comportamiento estereotipado, y configúrelos de nuevo en una clase (no estereotipo), que le permita a su objeto aprovechar totalmente la orientación a los objetos.

Encapsular

La encapsulación facilita la reutilización al agrupar la funcionalidad común y controlar el acceso a ella, lo que elimina los efectos secundarios no deseados. Cuando un objeto está bien encapsulado, se vuelve una “caja negra”, que proporciona una interfaz para usarlo, pero que mantiene ocultos en su interior los detalles de la implementación. Con este enfoque, puede reutilizar el objeto en cualquier parte de su modelo y saber exactamente cómo se comportará. Con más exactitud, todo cambio interno en el objeto no tendrá ningún impacto externo. >>

Puede comenzar a encapsular los objetos cambiando la visibilidad de sus miembros. De forma predeterminada, la mayoría de los elementos de datos migran de EAE como atributos “públicos” a los que se puede tener acceso desde cualquier otra parte de la aplicación. Si desea proteger los datos en una clase Ispec para que no sean modificados fuera de esa clase, configure su propiedad de visibilidad a “privada”. Cambiar la visibilidad identificará rápidamente la lógica fuera de esa clase que esté intentando acceder a los que ahora son datos internos, porque obtendrá un error en cuanto valide el modelo. El recorrido de la encapsulación continúa al trasladar esta lógica a un método dentro del mismo propietario que los datos.

Comience por hacer esto con uno o dos atributos para ver el efecto. A medida que examine el modelo, podrá ver que emergen patrones que originan menos métodos, la eliminación de la duplicación de código y el crecimiento de clases más poderosas.

El siguiente es un ejemplo: Si su modelo tiene una clase Ispec denominada ACCT y establece el atributo BALANCE como privado, cuando valide el modelo podrá encontrar que hay muchos lugares fuera de ACCT en los que tiene lógica similar a lo siguiente:

Subtract TRANTOTAL ACCT.BALANCE : Restar el total del saldo de la cuenta

También podrá ver que tiene cerca una lógica para revisar el saldo e informar un error si el saldo baja de cierto límite. Las condiciones que determinan si se permite cambiar el saldo de la cuenta deben registrarse por su “propietario”, la clase de la cuenta. Al colocar esa lógica dentro de la clase y ponerla a disposición a través de un único método, no sólo protege el saldo para que no sea actualizado incorrectamente, sino que puede simplificar el código en otras partes del modelo y facilitar el cambio de la lógica de actualización si las necesidades del negocio cambian en algún momento en el futuro.

Descomponer

Con esto no me refiero a las caries. En la terminología informática, la descomposición es el proceso de crear objetos más pequeños y más manejables a partir de objetos más grandes. En EAE, era tan sencillo como copiar la lógica de un Ispec a otro, y de una aplicación a otra. Con Agile Business Suite y los conceptos orientados a los objetos, el enfoque es distinto. Las funciones comunes se deben definir como clases y métodos, los que se pueden compartir entre una o más aplicaciones. Esto reduce la complejidad y fomenta la reutilización. La descomposición combina lógica repetitiva en una pequeña cantidad de métodos, y agrupa estos métodos en clases desde las que pueden reutilizarse.

Considere el ejemplo de administrar inventario. Cuando su sistema está procesando una venta en efectivo, una venta a crédito o una devolución, se actualiza el inventario para el artículo del mismo modo (incluido un signo más o menos). Por lo que tiene sentido definir una nueva clase llamada StockLevel (Nivel de existencias) que tenga un método UpdateStock (Actualizar existencias) que haga la suma (signo más) o resta (signo menos) cada vez que se deben modificar los niveles del inventario. Además con este enfoque, si se produjera un cambio fundamental en cómo se administran los niveles de existencias, hay sólo un lugar en el que cambia esa lógica, no obstante el impacto de la funcionalidad se percibirá en toda la aplicación.



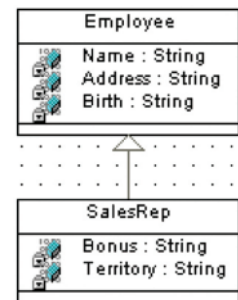
Puede ver pistas para patrones de uso común si se da el tiempo de organizar su sistema en un diagrama UML. La relación en él le mostrará los elementos de datos comunes, la lógica y más, todas oportunidades para descomponer. >>

Generalizar

La última técnica que se debe usar es generalizar, la que se trata de buscar formas de usar la herencia para simplificar su modelo existente. Busque características y comportamientos comunes que comparten más de una clase y defínalos en una nueva superclase. A continuación, puede permitir que sus objetos existentes se conviertan en subclases que heredan y extienden su clase primaria. Y, cuando cambien los requerimientos, simplemente cambie la superclase y, gracias a la magia de la herencia, esos cambios se aplican en las subclases.

El ejemplo clásico de la generalización es un empleado. Los empleados tienen un conjunto definido de características y comportamientos. nombre, dirección, fecha de nacimiento, etc. A partir de ahí, puede definir ciertos tipos de empleados, como ejecutivos de ventas. Ellos pueden heredar todas las características de los empleados, pero también tienen sus propios atributos especiales, como la estructura de bonificación o territorio.

Buscar oportunidades para generalizar le permite agrupar elementos de datos comunes, lógica y presentación uniforme en un grupo generalizado, y después definir especializaciones para crear subclases debajo de él.



Llevar su modelo EAE a nuevos niveles

Migrar a un entorno de desarrollo orientado a los objetos promete entregar beneficios de productividad para su organización. Seguir los pasos descritos en este artículo es una excelente forma de desarrollar su conocimiento de Agile Business Suite, a la vez que migra sus modelos existentes más lejos en el continuo UML.

Silverlight presenta una nueva opción de interfaz de usuario

En julio de 2008, dedicamos una [parte importante de Developing Agility](#) al tema de las interfaces de usuario, con artículos sobre las muchas opciones que tienen los desarrolladores para facilitar el aprendizaje y uso de sus aplicaciones de EAE o Agile Business Suite. En esa ocasión, sus opciones incluyeron a los pintores de GUI en EAE y Agile Business Suite, Client Tools, como ASP.NET Generator, y controles personalizados, como Web Form Renderer (Representador de formularios Web). Además destacamos la capacidad de crear un generador personalizado con Client Tools Customization Kit, o con uno construido por un tercero experto.

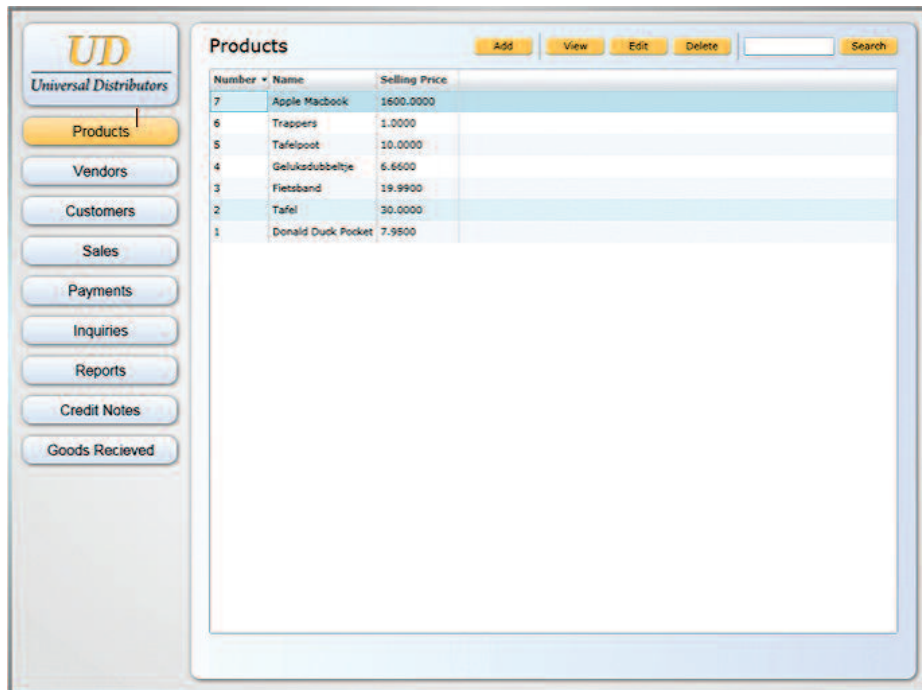
Ahora tenemos una cara nueva, la herramienta de desarrollo de plug-in de explorador Microsoft® Silverlight™, lo que significa que hay otra herramienta que puede usar para desarrollar una mejor GUI.

Puede leer todos los detalles sobre [Silverlight](#) en el sitio Web de Microsoft e incluso descargar una copia del kit de desarrollo. Silverlight ofrece muchísimas y estupendas capacidades para enriquecer aún más la experiencia del usuario final. Es importante que los desarrolladores de EAE y Agile Business Suite sepan que no se necesita nada más para posibilitar las mejoras de los Specs mediante Silverlight que lo que se necesita para otras opciones de GUI. Simplemente genere las transacciones que desea mejorar como servicios Web y póngase a trabajar. Silverlight es un plug-in de Visual Studio, por lo tanto, si está desarrollando con Agile Business Suite, los dos entornos están estrechamente alineados.

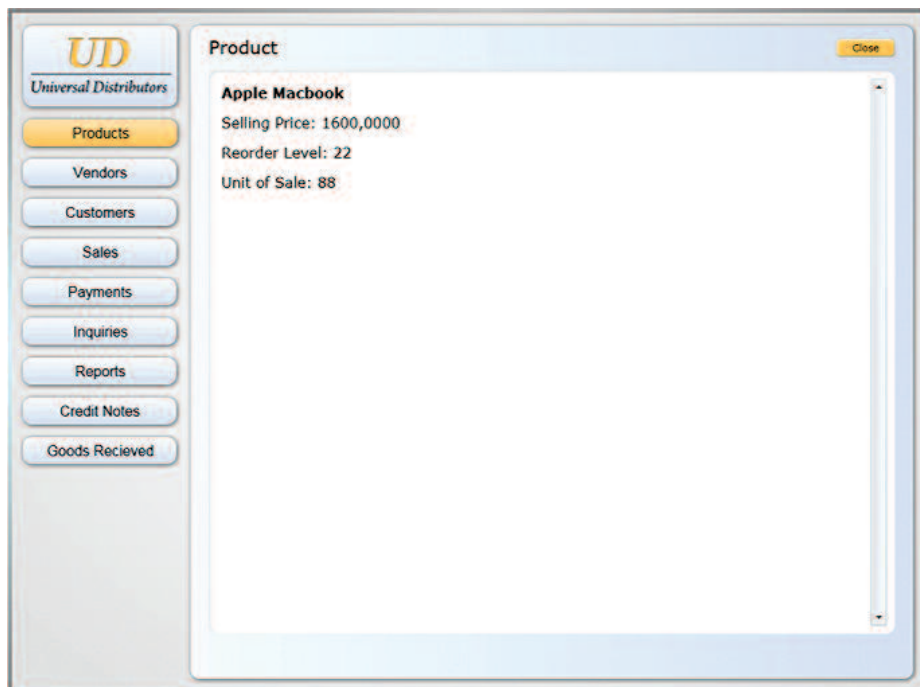
Curiosamente, un grupo de estudiantes universitarios de Holanda hace poco finalizó un proyecto que demostró el uso de Silverlight con nuestra vieja amiga la aplicación “de ejemplo”. Mediante Agile Business Suite y Silverlight 2.0, los estudiantes pudieron pasar de una presentación básica en pantalla a un excelente nuevo front-end gráfico en cuestión de semanas, y sin conocimiento acabado de ninguna de las dos herramientas (pero con algo de conocimiento de Visual Studio).

El equipo empezó con el pie derecho al evaluar la antigua interfaz y probar ideas en pantalla virtual para actualizar la presentación. De especial importancia es cómo usaron Silverlight para combinar las pantallas Product List (Lista de productos), Details (Detalles) y Maintenance (Mantenimiento) en un solo formulario. >>

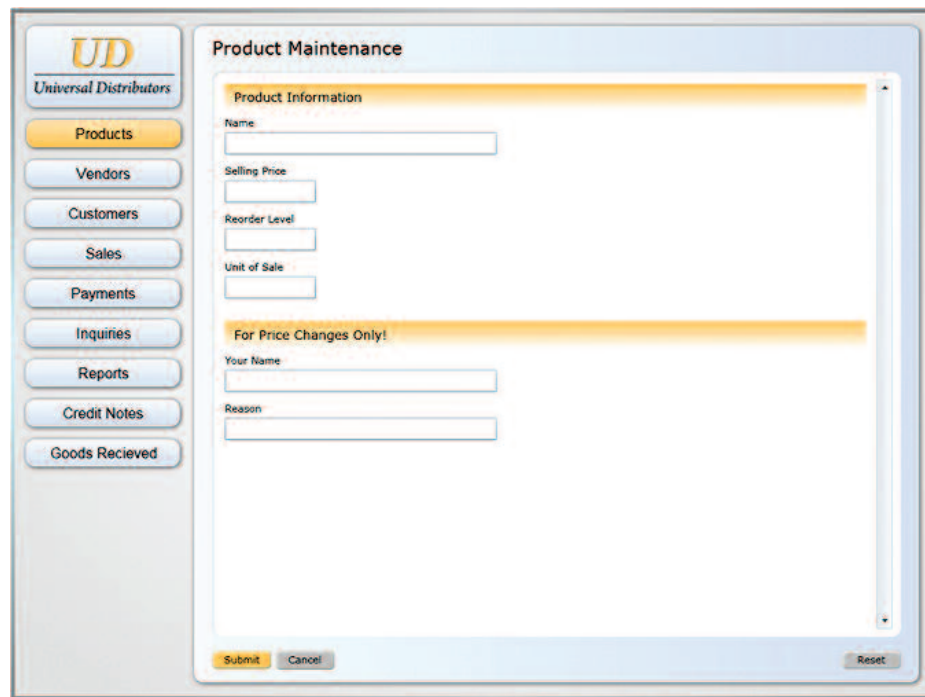
En la siguiente secuencia de capturas de pantalla, el usuario hace clic en el botón Product (Producto) para obtener una lista de todos los productos gracias a un agradable control de cuadrícula.



Otro clic en un producto específico proporciona los detalles, tal como se muestra en la siguiente captura de pantalla. >>



Por último, los usuarios autorizados pueden mantener la información de ese producto usando el mismo formulario.



The screenshot shows a web application interface for 'Product Maintenance'. On the left is a vertical navigation menu with the 'UD Universal Distributors' logo and buttons for 'Products', 'Vendors', 'Customers', 'Sales', 'Payments', 'Inquiries', 'Reports', 'Credit Notes', and 'Goods Received'. The 'Products' button is highlighted. The main content area is titled 'Product Maintenance' and contains two sections: 'Product Information' and 'For Price Changes Only!'. The 'Product Information' section has input fields for 'Name', 'Selling Price', 'Reorder Level', and 'Unit of Sale'. The 'For Price Changes Only!' section has input fields for 'Your Name' and 'Reason'. At the bottom of the form are 'Submit', 'Cancel', and 'Reset' buttons.

En cada uno de los pasos anteriores, sólo se actualiza la parte de la pantalla que se ve directamente afectada, en lugar de toda la pantalla. Además, los estudiantes usaron la misma metáfora para el mantenimiento de proveedor y cliente, lo que ofrece a los usuarios una experiencia muy coherente.

Al término del proyecto, los estudiantes concluyeron que Silverlight era rápido y sencillo de usar y ofrece una buena manera de actualizar la apariencia de aplicaciones más antiguas.

Cualquiera sea la herramienta que elija aprovechar su organización, Silverlight o una de las muchas otras opciones, lo instamos a que mejore la facilidad de uso de sus aplicaciones de EAE y Agile Business Suite con nuevas interfaces de usuario. Es una inversión que seguro dará frutos a la larga.

En recuerdo

Nos entristece transmitir la noticia del trágico fallecimiento de André Paridaens, que perdió la vida en un accidente de motocicleta el 1 de mayo de 2009 en su Bélgica natal. André aportó entusiasmo inagotable y experiencia sin igual en muchas áreas tecnológicas, incluidos EAE y Agile Business Suite, para beneficio de los clientes y sus colegas empleados de Unisys. Su estilo y enfoque únicos lo diferencian de los demás y quienes lo conocieron lamentan profundamente su pérdida. André tenía 55 años, y le sobreviven su esposa y sus dos hijos.

Hemos sido negligentes en comunicar a la comunidad de EAE el fallecimiento de Colin Zealley después de una breve enfermedad el 13 de octubre de 2007. Colin trabajó para Unisys por 36 años, comenzando su carrera profesional con Sperry. Fue un arquitecto técnico altamente respetado, un excepcional defensor de EAE y un prolífico colaborador de grupos de noticias en una amplia gama de temas. Colin aportó sinceridad a todo lo que hizo, tal vez en forma más especial cuando representó a sus colegas como miembro de Unisys UK Works Council. Todavía se echan de menos la personalidad franca de Colin y su humor poco convencional.

Calendario

Existen muchas oportunidades valiosas de aprendizaje disponibles para usted. Asegúrese de revisar la sección Webcasts y eventos de la [eCommunity](#) para conocer la información más reciente.

Qué	Dónde	Cuándo
Talleres de mejores prácticas de OS 2200	Bruselas, Bélgica	24 de septiembre de 2009
Talleres de mejores prácticas de MCP	Houten, Holanda, Bruselas, Bélgica Milton Keynes, Reino Unido París, Francia	29 de septiembre de 2009 30 de septiembre de 2009 7 de octubre de 2009 13 de octubre de 2009
Reunión de usuarios de EAE/Agile Business Suite	París, Francia	13 de octubre de 2009
Presentación de ClearPath y Taller de mejores prácticas de OS 2200	Estocolmo, Suecia	15 de octubre de 2009
Evento de usuario DACH y Taller de mejores prácticas de OS 2200	Basilea, Suiza	22 y 23 de octubre de 2009
Conferencia anual de tecnología UNITE	Hyatt Regency Minneapolis, Minneapolis, MN	Del 8 al 11 de noviembre de 2009
Simposio sobre Business Information Server (BIS)	Unisys Roseville, MN	Del 12 al 13 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 y Visual Basic son marcas registradas y Silverlight es una marca comercial 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.