



Developing Agility

Um boletim informativo trimestral para os clientes do Unisys EAE e do AB Suite

Conteúdo

- Página 2 **Entrevista com o cliente com a seguradora holandesa Movir**
O engenheiro de sistemas Ron Thakoer fala sobre a adoção do Agile Business Suite.
- Página 4 **Canto da engenharia: Aprimorando aplicativos migrados por meio da modelagem para reutilização**
Grant McCauley revela quatro maneiras de aumentar a reusabilidade de sistemas existentes após a migração para o Agile Business Suite.
- Página 9 **Silverlight oferece nova opção de interface de usuário**
Obtenha uma breve visão geral de como essa nova tecnologia pode ser utilizada em um aplicativo EAE ou Agile Business Suite.
- Página 12 **In Memoriam**
Comunicamos o falecimento de dois conhecidos especialistas no EAE.
- Página 12 **Calendário**
Confira nosso calendário para obter informações atualizadas sobre os próximos eventos.

Entrevista com o cliente com a seguradora holandesa Movir

Uma das principais finalidades do *Developing Agility* é permitir que clientes do EAE saibam como outras organizações estão planejando e conduzindo a migração para o Agile Business Suite. Este artigo dá continuidade a esse esforço com uma discussão sobre as atividades e conquistas da Movir, uma seguradora holandesa que fornece cobertura de invalidez para profissionais holandeses que atuam nas áreas comercial e de assistência médica.

Conversamos recentemente com Ron Thakoer, engenheiro de sistemas da Movir, sobre a recente migração da empresa para o Agile Business Suite com o objetivo de vir a utilizar o sistema operacional ClearPath MCP. Leia esta entrevista para saber como a Movir conduziu esse processo de migração e descobrir quais são os principais benefícios que Thakoer espera obter como novo usuário do Agile Business Suite.

Developing Agility (DA): Como é o ambiente de TI da Movir? Quantos aplicativos vocês utilizam? Eles oferecem suporte a quais funções empresariais? Qual é o tamanho desses aplicativos? E quantos profissionais trabalham na sua equipe de desenvolvimento?

Ron Thakoer (RT): A Movir opera três aplicativos EAE: dois oferecem suporte direto às operações comerciais e o terceiro é uma ferramenta de controle de versão personalizada relativamente pequena. O MOSYS, nosso principal aplicativo, possui 120 ISPECs e 200 REPORTS, mantém um banco de dados de produção que tem em torno de 10 GB e é utilizado por cerca de 110 usuários finais.

O segundo aplicativo, que é muito menor, se assemelha muito ao sistema MOSYS, mas foi especificamente desenvolvido para oferecer suporte a serviços de seguro de proprietários de pequenas empresas.

Contamos com cinco desenvolvedores do Agile Business Suite.

DA: Que benefícios comerciais o senhor espera obter com o Agile Business Suite?

RT: O sistema entrou no ar em junho de 2009, portanto a primeira etapa após a migração tem sido dar tempo para que nossa equipe de TI possa se familiarizar com o novo ambiente. Após esse período inicial, esperamos aproveitar o Agile Business Suite para otimizar determinados processos, por exemplo, aumentar a integração entre o Agile Business Suite e nossos aplicativos de front-office.

Além disso, pretendemos usar o Agile Business Suite para ampliar o escopo de um recurso de automatização da criação de correspondências que desenvolvemos no Microsoft® Word. Nós também utilizamos o Infolmage, um sistema de gerenciamento de documentos da Unisys, e esperamos simplificar o processo de integração dessa ferramenta com os aplicativos Agile Business Suite.

DA: E quais são os benefícios tecnológicos?

RT: É claro que esperamos que o Agile Business Suite ajude a aumentar a produtividade de nossos desenvolvedores, mas nosso principal objetivo é trabalhar em um ambiente orientado a objetos (OO). Com um OO, poderemos otimizar a estrutura de nossos aplicativos, o que melhorará nossa capacidade de mantê-los e aprimorá-los de forma rápida e eficiente.

DA: Antes de fazer a migração para o Agile Business Suite, o senhor migrou o front-end gráfico no aplicativo EAE do PowerClient (Graphical Interface Workbench) para o Microsoft Visual Basic® .NET. Por que escolheu o Visual Basic .NET? Os usuários finais notaram alguma diferença após a implementação do Agile Business Suite?

RT: Nós usávamos uma interface de usuário final que integrava o PowerClient e o Visual Basic 6 em nosso aplicativo EAE. No entanto, como o suporte a esses dois produtos estava com os dias contados, sabíamos que era preciso adotar tecnologias mais modernas. Como a empresa já usava o Visual Basic 6, foi uma escolha natural, do ponto de vista operacional, fazer a migração para o Visual Basic .NET usando o gerador fornecido nas ferramentas de cliente do EAE e do Agile Business Suite. Também utilizamos o Visual Basic .NET para adicionar códigos personalizados à interface de usuário final. >>

Conseguimos aumentar significativamente a eficiência do usuário final. Por exemplo, quando uma função do Visual Basic .NET é invocada, nossos usuários podem permanecer na mesma janela. Como essa interface foi modernizada primeiro, nossos usuários puderam usá-la com os aplicativos EAE por alguns meses antes de fazermos a migração para o Agile Business Suite. Além disso, conseguimos migrar as interfaces gráficas do usuário atualizadas do Visual Basic .NET para o Agile Business Suite sem qualquer problema e sem que os usuários percebessem a diferença.

DA: *Que recursos do Agile Business Suite foram fundamentais para influenciar a sua escolha de migração?*

RT: O fator determinante foi a robusta funcionalidade de integração do Agile Business Suite. Esses recursos já são importantes no momento e passarão a ser cada vez mais importantes à medida que a empresa começar a enfrentar futuros desafios.

DA: *A Movir passou a usar o Agile Business Suite 1.2 em junho de 2009. Como tem sido a experiência de usá-lo em um ambiente de produção? O que os seus desenvolvedores acham do Agile Business Suite Developer?*

RT: A migração para o Agile Business Suite nos possibilitou manter a qualidade de nossas operações sem comprometer a qualidade. Todos os funcionários das diversas equipes da Unisys com as quais trabalhamos nos ofereceram um suporte de valor inestimável ao longo do projeto. Ficamos especialmente satisfeitos com a prontidão dos engenheiros do Agile Business Suite, que fizeram reparos rápidos que ajudaram a evitar maiores atrasos.

Do ponto de vista do desenvolvimento, alguns de nossos desenvolvedores já usam o Agile Business Suite Developer regularmente e esperamos que outros desenvolvedores comecem a aproveitar todos os benefícios da ferramenta em breve.

DA: *O que o senhor aprendeu com o projeto de migração que vale a pena compartilhar com outras organizações de TI? Quais foram as atividades mais importantes durante o projeto de migração?*

RT: Nós conduzimos a migração da mesma forma que conduzimos outros projetos e acho que esta é uma abordagem importante. Pensar dessa forma nos ajudou a selecionar os recursos internos, compor o case empresarial e contratar a equipe de suporte da Unisys necessária para conduzir o processo.

No geral, a atividade mais importante foi permitir que nossos usuários finais testassem os aplicativos migrados. Também usamos os serviços e o conjunto de ferramentas do Unisys Business Application Test Manager (BATMan) para reproduzir diversas transações de aplicativos do EAE no aplicativo Agile Business Suite migrado, o que nos deu a tranquilidade de poder testar várias áreas do aplicativo antes de colocá-lo no ar. É óbvio que os testes tomam tempo, mas esta é uma etapa crucial para garantir uma migração bem-sucedida.

DA: *Quais são os seus planos para novos desenvolvimentos com o Agile Business Suite?*

RT: Por ser o nosso aplicativo de back-office mais importante, o MOSYS está sempre sendo atualizado e ampliado para atender aos crescentes requisitos dos usuários finais. Portanto, nossa primeira atividade com o Agile Business Suite é usar o Agile Business Suite Developer para criar novos recursos inovadores para o MOSYS.

BATMan garante uma transição tranquila para o Agile Business Suite

Fornecido como um serviço, o BATMan automatiza o desenvolvimento de scripts de teste que capturam transações reais do EAE ou do Agile Business Suite para ajudá-lo a testar aplicativos online antes de eles entrarem no ar. O BATMan reproduz essas transações em um ambiente de teste e permite que você confirme uma reprodução bem-sucedida validando se os resultados correspondem às transações originais. Com isso, o BATMan permite que você tenha mais segurança para mover aplicativos de um ambiente de teste para um ambiente de produção.

Aprimorando o uso de aplicativos migrados por meio da modelagem para reutilização

Por Grant McCauley, Líder Técnico de Modelo do Agile Business Suite no GTC da Austrália da Unidade de Sistemas e Tecnologia da Unisys

É fácil perceber os benefícios dos recursos de modelagem UML, como os recursos oferecidos pelo Agile Business Suite, quando você está desenvolvendo um novo aplicativo. Mas ainda há benefícios a serem obtidos com os aplicativos existentes, principalmente se você estiver fazendo a migração do EAE para o Agile Business Suite. Este artigo analisa maneiras simples e direitas de utilizar os recursos de modelagem do Agile Business Suite para facilitar o suporte aos aplicativos EAE migrados e à reutilização do código do EAE. Aproveitando essas dicas, você poderá aumentar a produtividade a longo prazo e ter uma experiência de modelagem prática enquanto trabalha com os sistemas EAE que conhece (antigos).

Um passo de cada vez. Quando você faz a migração para o Agile Business Suite, o modelo do EAE passa a ser um modelo do Agile Business Suite e vários elementos básicos são criados automaticamente. Objetos, classes, métodos, atributos — todos os bons elementos orientados a objetos — estarão disponíveis quando você fizer a migração. Já falamos sobre etapas básicas dos elementos orientados a objetos na seção Canto da engenharia de edições anteriores, portanto não é necessário entrar em detalhes. Mas recomendo que você consulte essas edições quando tiver tempo.

Após criar o modelo do Agile Business Suite do seu aplicativo, há muitas outras maneiras de fazer com que seja mais fácil entender, manter, utilizar e reutilizar o novo modelo. Eu vejo as oportunidades divididas nas seguintes áreas gerais:

1. Analisar
2. Organizar
3. Documentar
4. Desenvolver

Vamos analisar essas áreas com mais detalhes.

Analisar

Quando o seu modelo for migrado para o Agile Business Suite, um bom primeiro passo é na verdade dar um passo para trás e analisar o que você tem. Não é necessário fazer mudanças só para dizer que está fazendo alguma mudança, portanto reserve um tempo para analisar o seu modelo de “todos os ângulos”. Use as guias (propriedades, dependência, etc.) para explorar as diferentes visões de modelo disponíveis no Agile Business Suite Developer. Confira as dependências e como as propriedades, como tipo, proprietário, heranças e multiplicidade, foram definidas para os conjuntos de objetos. Você também pode experimentar os recursos da caixa de diálogo Search (Alt+S) e o Quick Navigator (Alt+G) para analisar ainda melhor o sistema e o modelo.

Organizar

Agora que você já tem uma boa noção da aparência do modelo do aplicativo no Agile Business Suite, é hora de começar a organização. Quando falo em organização, estou me referindo à adoção de etapas para fazer com que seja mais fácil entender, reutilizar e reestruturar o modelo. Mais uma vez, você não estará começando do zero, pois as áreas e atividades funcionais do EAE foram transportadas para o Agile Business Suite durante a migração.

Nesta etapa, recomendo o uso de três ferramentas: pastas, dicionários e diagramas. >>

Atualizador orientado a objetos

Para obter mais informações sobre conceitos orientados a objetos, consulte estes artigos do Canto da engenharia:

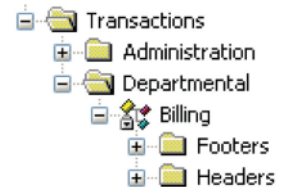
- [Quando é bom aplicar um estereótipo](#)
- [Encapsulamento](#)
- [Polimorfismo](#)



Pastas

A metáfora da pasta é uma excelente maneira de organizar melhor o seu modelo e deve ser a primeira etapa da organização. Use as pastas do Agile Business Suite da mesma forma que você usaria uma pasta física: para agrupar itens relacionados. Por exemplo, você pode:

- Colocar todas as transações relacionadas a uma área comercial específica em uma pasta
- Criar uma pasta que contenha todas as classes externas ou todos os relatórios ou todas as transações administrativas
- Usar pastas dentro de um relatório para organizar as estruturas do cabeçalho e do rodapé



As pastas são tão flexíveis que a mesma entidade pode ser usada em várias pastas e é possível criar pastas dentro de outras pastas.

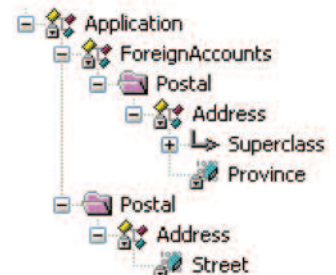


Dicionários

No Agile Business Suite, o dicionário é um tipo especial de pasta usado especificamente para organizar objetos que funcionam como definições de outros objetos, como classes que você deseja reutilizar com facilidade em diversas áreas do aplicativo. Diferentemente do EAE, no Agile Business Suite você pode criar quantos dicionários precisar em vários níveis do modelo (no qual eles se aplicarão a todos os objetos definidos nos subníveis de um determinado nível).

Por que ter mais de um dicionário? Bem, pelos mesmos motivos que você teria uma pasta, mas com o benefício adicional de os dicionários facilitarem a reutilização, definindo automaticamente a relação “herdada” quando o nome de um atributo ou variável corresponder a um objeto que faça parte de um dicionário.

Por exemplo, vários dicionários podem ser úteis se você estiver combinando a funcionalidade de dois ou três aplicativos ou desejar redefinir determinados termos ou definições em uma área local do modelo. Em uma parte do modelo você pode criar um dicionário que defina um endereço com rua, apartamento, cidade, estado e CEP. Em outra parte, que esteja relacionada a outros países, você pode redefinir o endereço para que ele inclua um código postal longo, um país e uma província. Em cada área será possível se referir a um endereço como um objeto único e selecionar o objeto apropriado automaticamente.



No EAE, as definições em um dicionário eram limitadas a tipos primitivos, como uma cadeia de caracteres e um número. No Agile Business Suite, as definições também podem ser classes que incluem atributos, lógicas e até apresentações. Ao criar entradas em um dicionário, você cria um padrão para itens usados com frequência no seu modelo — o que pode ser o primeiro passo para uma reutilização efetiva de itens.

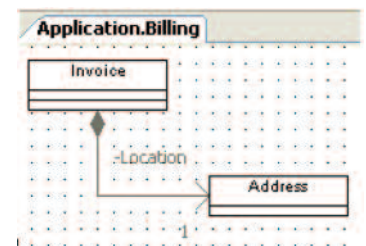


Diagramas

Os diagramas são outra ferramenta de organização. Como uma pasta, você pode arrastar itens relacionados para um diagrama — criando uma imagem UML dos conteúdos. Mesmo que não haja linhas (dependências/relações) conectando esses elementos em um diagrama, os diagramas continuam sendo úteis, pois permitem que você tenha um agrupamento lógico de itens relacionados. Como dizem por aí, “uma imagem vale mais que mil palavras”.



O uso de qualquer uma dessas dicas de organização não afetará o funcionamento do seu aplicativo. Elas são apenas ferramentas que facilitam a organização dos desenvolvedores. >>



Documentar

Documentar se refere ao conjunto de ações que ajudam a esclarecer para os usuários do seu sistema a finalidade de objetos, lógicas, etc. Os diagramas UML são apenas uma maneira de descrever um sistema. Esses diagramas criam visualizações de um aplicativo e mostram relações estruturais entre classes.

O Agile Business Suite também possui os excelentes recursos “document as you go” que facilita a inserção de comentários e descrições à medida que você projeta e desenvolve. Esses mesmos recursos podem ser usados no sistema migrado — pense nessa possibilidade como um recurso “document as you discover” para formalizar seu conhecimento institucional sobre as várias funções de um sistema que você já conheça bem. Você ainda tem a possibilidade de adicionar documentos que serão consultados por públicos diferentes. Informações sobre as funções comerciais que são compatíveis com um objeto e planos futuros para ampliar uma função podem ser úteis para desenvolvedores. Já os usuários finais podem consultar documentos que expliquem como utilizar um Ispec ou a origem de dados em uma lista.

Comece inserindo documentos em uma pasta e crie subpastas com informações mais específicas de acordo com o seu tempo e interesse. Se estiver trabalhando em uma área do modelo e descobrir algo interessante, abra a janela de documentação e registre sua descoberta. Uma função parecida com o WordPad faz com que seja fácil registrar descobertas.

O Agile Business Suite também possui um novo recurso que permite a extração da documentação do aplicativo. As páginas de comentários, que são guias de referência navegáveis baseados em HTML, oferecem uma solução mais robusta e fácil de usar, em comparação com as especificações de impressão padrão do EAE. A documentação online é fácil de manter quando é automaticamente gerada a partir do sistema sempre que necessário.

Desenvolver

Até este ponto, todas as ações recomendadas não alteram o modelo do aplicativo. Analisar, organizar e documentar apenas fazem com que seja mais fácil entender como o aplicativo funciona. Desenvolver um sistema significa aprimorar e ampliar o aplicativo, o que é a principal característica da modelagem para reutilização. O desenvolvimento de um aplicativo engloba quatro etapas: eliminar, encapsular, decompor e generalizar.

Eliminar

Eliminar significa refinar as funções das entidades em um aplicativo, removendo recursos desnecessários ou redundantes que não são mais úteis. Como o EAE oferece apenas entidades de alto nível, sempre haverá oportunidade para aprimoramentos. Por exemplo, grupos podem ter sido usados para definir estruturas, sem utilizar o recurso de serialização de estruturas ou modelos insertables podem ter sido usados em formulários sem a necessidade de utilizar códigos de macros especiais. Eliminar recursos não utilizados simplifica a definição de uma entidade e também aumenta a possibilidade de ela ser reutilizada.

Portanto, monitore os objetos do Agile Business Suite que não estão utilizando seu comportamento estereotipado e insira-os novamente em uma classe (sem estereótipo), para que o objeto possa aproveitar a orientação de objeto da melhor maneira possível.

Encapsular

O encapsulamento facilita a reutilização, pois agrupa funcionalidades comuns e controla o acesso a essas funcionalidades, eliminando efeitos colaterais indesejados. Quando um objeto é bem encapsulado, ele se torna uma “caixa preta”, fornecendo uma interface na qual pode ser usado, mas mantendo os detalhes da implementação ocultos no seu interior. Com essa abordagem, você pode reutilizar o objeto em qualquer parte do modelo e saber exatamente como ele se comportará. Para ser mais específico, todas as alterações internas no objeto não terão impacto externo. >>

Você pode começar a encapsular objetos alterando a visibilidade dos membros do objeto. Por padrão, a maioria dos itens de dados são migrados do EAE como atributos “públicos” e podem ser acessados a partir de qualquer parte do aplicativo. Caso deseje impedir que os dados de um Ispec sejam modificados fora dessa classe, configure a propriedade de visibilidade dos dados como “privada”. A alteração da visibilidade permite identificar rapidamente a lógica de fora da classe que está tentando acessar os dados internos, porque você receberá um erro quando tentar validar o modelo. O próximo passo no processo de encapsulamento é mover essa lógica para um método dentro do mesmo proprietário como dados.

Experimente fazer isso pela primeira vez com um ou dois atributos para avaliar o impacto. À medida que você trabalhar no modelo, poderá observar o surgimento de padrões que resultam em um número menor de métodos, na eliminação da duplicação de códigos e no crescimento de classes com mais recursos.

Veja este exemplo: Se o modelo tiver uma classe Ispec chamada ACCT e você fizer com que o atributo BALANCE seja privado, quando validar o modelo, poderá descobrir que há vários locais fora do ACCT nos quais há lógicas com o seguinte formato:

Subtract TRANTOTAL ACCT.BALANCE : Subtrair o total do saldo da conta

Você poderá observar que próximo ao ACCT há lógicas que verificam o saldo e reportam um erro se o saldo estiver abaixo de um determinado limite. As condições que determinam se o saldo de uma conta pode ser alterado devem ser determinadas pela classe da conta, que é o “proprietário” da conta. Ao colocar a lógica dentro de uma classe e disponibilizá-la por meio de um método único, você impede que o saldo seja atualizado incorretamente, pode simplificar o código em outras partes do modelo e facilitar a alteração da lógica de atualização se os requisitos comerciais forem alterados no futuro.

Decompor

Decompor não está relacionado a decomposição. Em computação, decomposição é o processo de criação de objetos menores e mais gerenciáveis a partir de objetos maiores. No EAE, era fácil copiar lógicas de um Ispec para outro — e de um aplicativo para outro. Com o Agile Business Suite e com os conceitos orientados a objetos, a abordagem é diferente. Funções comuns devem ser definidas como classes e métodos, que podem ser compartilhados com um ou mais aplicativos. Isso reduz a complexidade e possibilita a reutilização. A decomposição combina lógicas repetitivas em um pequeno número de métodos e agrupa esses métodos em classes a partir das quais eles podem ser reutilizados.

Considere um exemplo de gerenciamento de estoque. Quando o sistema estiver processando uma venda no débito, no crédito ou uma devolução, o estoque do item será atualizado (com um acréscimo ou uma dedução). Sendo assim, faz sentido definir uma nova classe chamada de StockLevel que possui um método UpdateStock para fazer acréscimos e deduções sempre que for preciso alterar os níveis do estoque. Com essa abordagem, caso ocorra uma grande alteração na forma de gerenciamento dos níveis de estoque, a lógica só será alterada em um local, mas o impacto na funcionalidade será sentido em todo o aplicativo.



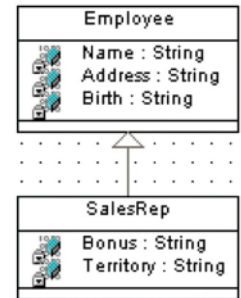
Você poderá ver dicas de padrões usados com frequência se reservar um tempo para organizar seu sistema em um diagrama UML. As relações no diagrama mostrarão os elementos de dados e lógicas comuns e muito mais — todas as oportunidades de decomposição. >>

Generalizar

A última técnica a ser utilizada é a generalização, que significa encontrar maneiras de utilizar elementos herdados para simplificar o modelo. Pesquise características e comportamentos comuns que sejam compartilhados por mais de uma classe e defina-os em uma nova superclasse. Em seguida, você pode permitir que objetos se tornem subclasses que herdam e ampliam a classe dos objetos pai. Quando os requisitos forem alterados, basta alterar a superclasse e a herança permitirá que as alterações sejam refletidas nas subclasses como um passe de mágica.

O exemplo clássico de generalização é um funcionário. Os funcionários possuem um conjunto definido de características e comportamentos: nome, endereço, data de nascimento, etc. A partir desses dados, é possível definir determinados tipos de funcionários, como representantes de vendas. Esse tipo de funcionário herda todas as características dos funcionários, mas também possui atributos especiais específicos, como estrutura de bonificação e território.

Procurar oportunidades de generalização permite agrupar itens de dados, lógicas e até apresentações comuns em um grupo generalizado e, em seguida, definir especializações para criar subclasses.



Aprimorando o uso do modelo do EAE

A migração para um ambiente de desenvolvimento orientado a objetos ajuda a gerar ganhos de produtividade para a sua organização. Seguir as etapas descritas neste artigo é uma excelente forma de se familiarizar com o Agile Business Suite — à medida que você migrar seus modelos para a estrutura UML.

Silverlight oferece nova opção de interface de usuário

Em julho de 2008, dedicamos [grande parte do Developing Agility](#) ao tema das interfaces de usuário, apresentando as várias opções que permitem que os desenvolvedores simplifiquem a utilização dos aplicativos EAE ou Agile Business Suite. Essas opções incluíam os GUI painters no EAE e no Agile Business Suite, ferramentas clientes, como o ASP.NET Generator, e controles personalizados, como o Web Form Renderer. Também destacamos a possibilidade de criar um gerador personalizado usando o Client Tools Customization Kit (Kit de personalização de ferramentas clientes) ou usar um gerador montado por outro fabricante experiente.

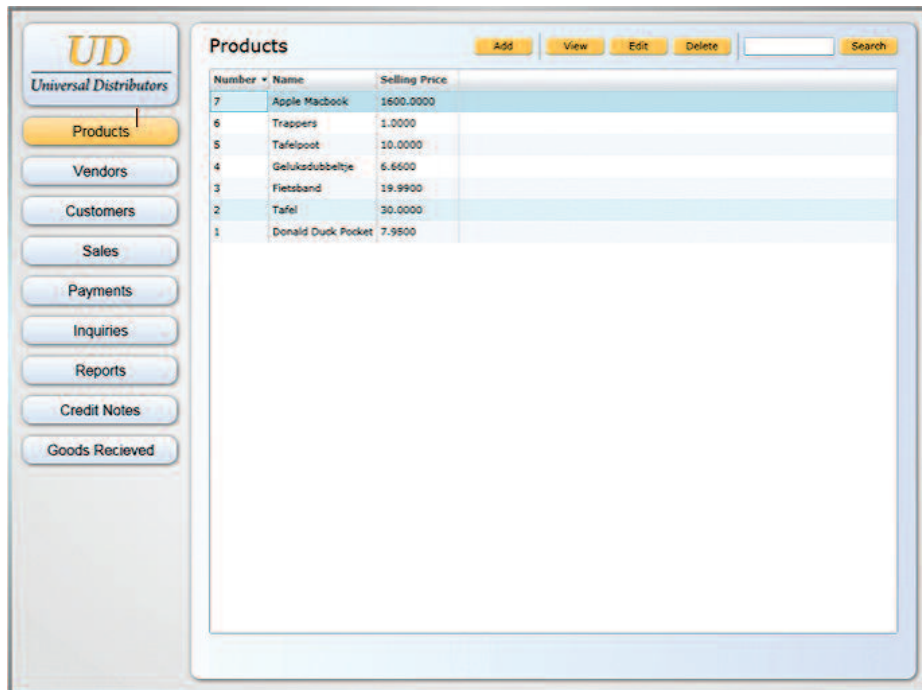
Agora, estamos lançando mais um produto, a ferramenta de desenvolvimento de plug-ins de navegador do Microsoft® Silverlight™ que é outra opção para que você possa desenvolver uma interface de usuário gráfica ainda melhor.

Você pode obter mais informações sobre o [Silverlight](#) no site da Microsoft e até baixar uma cópia do kit de desenvolvimento. O Silverlight oferece vários recursos úteis para ajudar a aprimorar a sua experiência de usuário final. É importante que os desenvolvedores do EAE e do Agile Business Suite saibam que disponibilizar Specs para aprimoramento via Silverlight é tão fácil quanto utilizar outras opções de interface gráfica do usuário. Basta gerar as transações que deseja aprimorar como Serviços da Web e começar a trabalhar. O Silverlight é um plug-in do Visual Studio. Portanto, se você estiver trabalhando com o Agile Business Suite, perceberá que os dois ambientes são muito parecidos.

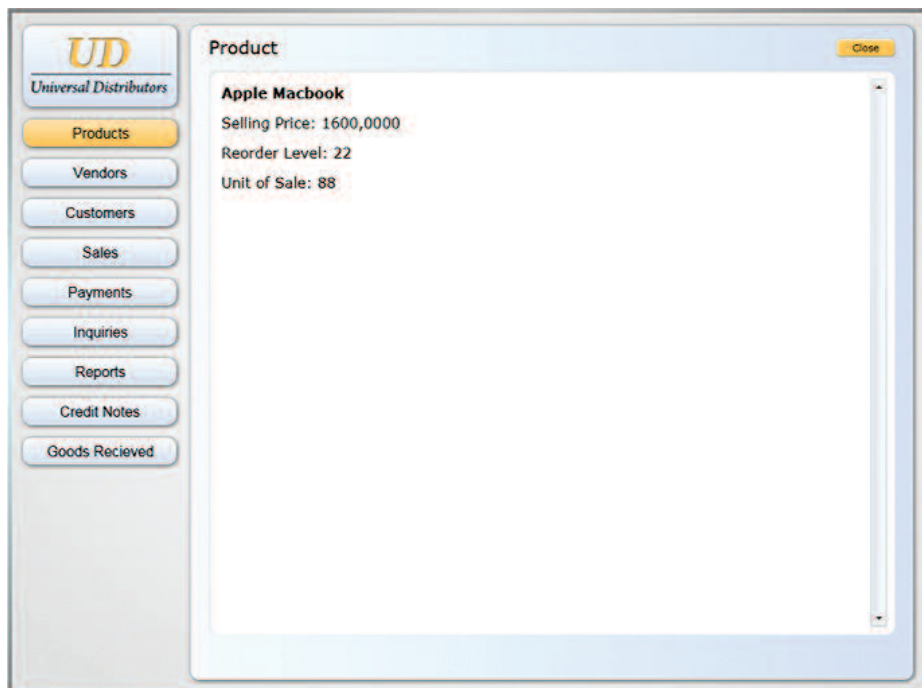
Por coincidência, um grupo de estudantes universitários holandeses acaba de concluir um projeto que estudou o uso do Silverlight com nosso velho amigo aplicativo de “Amostra”. Usando o Agile Business Suite e o Silverlight 2.0, os estudantes conseguiram transformar uma apresentação com tela básica em um excelente front-end em apenas algumas semanas e sem conhecer profundamente as ferramentas (mas com algum conhecimento do Visual Studio).

O trabalho da equipe começou da mesma forma que recomendamos: avaliando a interface antiga e discutindo ideias para atualizar a apresentação. É importante destacar como eles usaram o Silverlight para combinar as telas Lista de Produtos, Detalhes e Manutenção em um formulário. >>

Na sequência de capturas de telas mostrada abaixo, o usuário clica no botão Products para obter uma lista de todos os produtos usando um excelente controle de grande.



Outro clique em um determinado produto fornece mais detalhes, como mostrado na próxima captura de tela. >>



Por fim, os usuários autorizados podem manter essas informações de produto usando o mesmo formulário.

The screenshot shows a web application interface for 'Product Maintenance'. On the left is a vertical navigation menu with buttons for 'Products', 'Vendors', 'Customers', 'Sales', 'Payments', 'Inquiries', 'Reports', 'Credit Notes', and 'Goods Recieved'. 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.

Em cada uma das etapas acima, apenas a parte da tela que é diretamente afetada será atualizada. Além disso, os estudantes usaram a mesma metáfora para a manutenção de fornecedores e clientes, o que possibilita uma experiência muito consistente para os usuários.

No final do projeto, os estudantes concluíram que o Silverlight é mais rápido e simples de usar e oferece uma boa maneira de atualizar a aparência de aplicativos antigos.

Independentemente da ferramenta escolhida pela sua organização — o Silverlight ou uma das várias outras opções — recomendamos que você aprimore a usabilidade dos aplicativos EAE e Agile Business Suite com novas interfaces de usuário. Esse é um investimento que trará benefícios a longo prazo.

In Memoriam

É com pesar que comunicamos o trágico falecimento de Andre Paridaens, vítima de um acidente de moto no dia 1º de maio, na Bélgica. Andre levou seu entusiasmo incansável e sua experiência sem precedentes a muitas áreas tecnológicas, como o EAE e o Agile Business Suite, para o benefício dos clientes e de seus colegas na Unisys. Ele se destacava por seu estilo e sua conduta e sua perda está sendo muito sentida por todos que o conheciam. Andre tinha 55 anos de idade e deixou sua esposa e dois filhos.

Não tivemos o cuidado de comunicar à comunidade do EAE o falecimento de Colin Zealley, no dia 3 de outubro de 2007, após uma breve enfermidade. Colin trabalhou na Unisys por 36 anos, tendo iniciado sua carreira na Sperry. Colin foi um arquiteto técnico altamente respeitado, grande defensor do EAE e colaborou por diversas vezes com grupos de discussão de uma variedade de tópicos. Colin imprimiu sinceridade a tudo que fez, talvez mais notadamente quando representou seus colegas como membro do Unisys UK Works Council. Sua personalidade marcante e seu senso de humor vão fazer falta.

Calendário

Existem muitas oportunidades de aprendizagem de valor inestimável disponíveis para você. Consulte a seção Webcasts e Eventos da [eCommunity](#) para obter informações atualizadas.

O que	Onde	Quando
Workshops de práticas recomendados do OS 2200	Bruxelas, Bélgica	24 de setembro de 2009
Workshops de práticas recomendadas do MCP	Houten, Holanda Bruxelas, Bélgica Milton Keynes, Reino Unido Paris, França	De 29 a 30 de setembro de 2009, De 7 a 13 de outubro de 2009
Reunião de usuários do EAE/Agile Business Suite	Paris, França	13 de outubro de 2009
Visão geral do ClearPath e Workshop de melhores práticas do OS 2200	Estocolmo, Suécia	15 de outubro de 2009
Evento para usuários do DACH e Workshop de práticas recomendadas do OS 2200	Basel, Suíça	22 e 23 de outubro de 2009
UNITE Conferência de tecnologia anual	Hyatt Regency Minneapolis, Minneapolis, MN	8 a 11 de novembro de 2009
Simpósio sobre o BIS (Business Information Server)	Unisys Roseville, MN	12 a 13 de novembro de 2009

As especificações estão sujeitas a alterações sem aviso prévio.

© 2009 Unisys Corporation.
Todos os direitos reservados.

Unisys e o logotipo Unisys são marcas comerciais registradas da Unisys Corporation. Microsoft e Visual Basic são marcas comerciais registradas e Silverlight é uma marca comercial da Microsoft Corporation. Todas as outras marcas e produtos aqui referenciados são reconhecidamente marcas comerciais ou registradas de seus respectivos detentores.