



Developing Agility

*A Newsletter for Unisys EAE
and Agile Business Suite Customers*

Questions?
ABSuite@unisys.com can help.

Contents

- Page 2 | **Agile Business Suite 2.0 IC1400: Enhancing Developer Productivity** With the release of Interim Correction (IC) 1400, we've updated the underlying functionality of Agile Business Suite 2.0 to streamline and simplify development efforts – and make it easier for EAE users to make the move to AB Suite.
- Page 3 | **Important Migration Reminder for EAE Customers Running Microsoft Windows, Linux, or UNIX** Phase 1 support for EAE on Microsoft® Windows®, UNIX®, and Linux® platforms is set to expire on December 31, 2011. With a short six months left until we hit this date, it's important to get your migration plans underway today.
- Page 4 | **Engineering Corner: State and Stateless Transactions** “Stateful” and “stateless” transactions have different uses and benefits in the EAE and AB Suite worlds – but blending them together can help you modernize your applications in a variety of ways. Read on to learn how to do just that.
- Page 7 | **Partner Corner: Information Exchange Group** Well versed in all things EAE and AB Suite, Information Exchange Group provides a host of helpful services for these development tools. Check out this article to learn about what they can offer you.

Want your own copy of *Developing Agility*?
[Sign up](#) in the eCommunity today!

Agile Business Suite 2.0 IC1400: Enhancing Developer Productivity

By Maarten Schneider, Worldwide Marketing Manager, Enterprise Application Environment and Agile Business Suite

With many thanks to the early adopters of Agile Business Suite (AB Suite) across the globe – who provided numerous useful and thoughtful suggestions for improving the usability of the System Modeler functionality in AB Suite 2.0 – we are pleased to announce that we have significantly enhanced AB Suite 2.0 with Interim Correction (IC) 1400. With IC1400, AB Suite 2.0 better utilizes the underlying infrastructure of Microsoft Visual Studio® to increase the consistency of the behavior between System Modeler and Visual Studio project types.



For example, AB Suite Developer will now provide greater detail about your existing application by separating the class view pane and the related members into two panes. This way, anytime you select an object in Developer, the bottom pane automatically fills with pertinent information about that object.

In addition, we've added a new search function in class view, which provides a filtered view that quickly takes you to where you maintain your application – and provides easy access to information about Owner, Inherits, or Type. What's more, IC1400 introduces a "Synchronize Class View" function that brings you from any element to the right spot in the class view.

We've also provided a simplified view of the frames and content in Reports, which streamlines maintenance.

Together, these features – as well as many of the smaller, yet helpful updates we've packed into Agile Business Suite 2.0 IC1400 – will make it much easier for Enterprise Application Environment (EAE) users to make the decision that now is the right time to migrate to AB Suite.

Because many of the updates contained within IC1400 were designed to incorporate familiar functions into AB Suite, EAE users will find it quite easy to manage their applications following a migration to AB Suite. The new search functions, for example, will help to shorten the learning curve and keep productivity at a high level.

What's more, because this IC contains many changes that maximize the use of familiar Visual Studio features, AB Suite is even easier to learn and use for C# developers who are used to similar functionality.

EAE Developers: Now is the Time to Migrate

As you can see, this IC holds tremendous value for EAE shops that make the move to AB Suite. As such, I encourage you to begin your migration plan, if you haven't done so already. Please check the Documentation section of the [Unisys Support Web Site](#) for detailed information about AB Suite IC1400.

If you need assistance incorporating the migration into your 2012 budget, please ask your Unisys technical consultant to guide you in setting up a draft migration plan. Or, send Diane McGonigle, AB Suite migration manager, an [email](#) and she will guide you to the right person.

Important Migration Reminder for EAE Customers Running Microsoft Windows, Linux, or UNIX

By Diane McGonigle, Agile Business Suite Migration Manager, Unisys TCIS



Phase 1 support for EAE 3.3 on Microsoft Windows, UNIX, and Linux platforms will be ending on December 31, 2011. With this date roughly six months away, I wanted to take this opportunity to remind you to plan your migration to Agile Business Suite and review some of the benefits of moving to the new environment.

Migrating to AB Suite running on Windows gives you a new .NET based runtime environment that eliminates the expenses associated with renewing the COBOL licenses you had in EAE. And, this change in environments is actually transparent – even though AB Suite generates C#, you can continue to develop using LDL (now called LDL+ in AB Suite). You'll be able to make the move to a modern, Windows based development environment with little or no change to the application itself.

There are already several organizations running AB Suite on Windows in high-volume production environments. For example, [Thomasville Utilities](#) – the subject of a recent *Developing Agility* feature – recently completed a successful migration and is now running a mission-critical billing, collection, and work order and inventory management application on AB Suite.

In addition, [TIES](#), an education technology collaborative, runs hundreds of thousands of transactions per day on a migrated Windows application – and is bringing new school districts on board each week. Likewise, [Harris School Solutions](#), another provider of education software, has migrated about 20 percent of its participating districts to the AB Suite version of their SchoolMax application.

If you currently use EAE on either UNIX or Linux, please contact [me](#) or your Unisys sales representative to discuss migration options to a supported AB Suite runtime platform – and to learn about the potential savings you can realize in the process.

Make a Plan to Migrate

January 1, 2012, marks the start of Phase 2 support for EAE 3.3 Runtime for Windows, UNIX, and Linux operating environments. During Phase 2, customers with valid maintenance contracts will continue to receive telephone and online assistance from Unisys Customer Support, obtain fixes to known problems, and if possible, receive workarounds when they report errors that don't have a fix available. Please note that if you are not able to complete your migration by the end of the year, you can obtain extended support (where you can receive fixes for new problems through hot patches) for up to one year at an additional charge.

If you have questions about making the move to AB Suite, please contact Diane McGonigle, AB Suite Migration Manager, at diane.mcgonigle@unisys.com for a copy of our migration checklist and advice on the best way to get started.

Engineering Corner: State and Stateless Transactions

By Alan Hood, Consulting Engineer, Agile Business Suite, Unisys TCIS

In computing terms, the concept of “state” refers to a transaction environment where a user logs into a server from a specific known location, terminal or otherwise, enters unique credentials, and is able to access information tied to what was entered during his or her session.

Most traditional online transaction applications take advantage of state – and depend on the fact that user-specific information is retained from one transaction to the next. Doing so eliminates the burden of reentering credentials or data every time a new transaction is transmitted back to the application. In EAE and Agile Business Suite environments, this kind of information is typically stored in GLB.WORK, so any application using this function is likely performing “stateful” transactions.

Although state is clearly useful in a number of ways, many modern transaction-processing environments and protocols – such as service-oriented architecture (SOA) and Hypertext Transfer Protocol (HTTP) – require “stateless” interactions. In this situation, each unique request is treated as a separate transaction that is unconnected to any previous request. As such, stateless transactions function as independent pairs of requests and responses, meaning that the host server never retains session or status information for the duration of multiple requests as it would in a state environment.

As fundamental components of many modern computing approaches, stateless transactions are independent of any previous transaction and are characterized by the term “ACID,” or:

- **Atomicity** – an all or nothing approach that requires the entire transaction to be completed, or no update will be made.
- **Consistency** – stateless transactions ensure the integrity of persistent (database) data.
- **Isolation** – data that has not been updated cannot affect any transaction.
- **Durability** – the integrity of a transaction is ensured from the point a system receives it.

However useful they may be, stateless transactions are not ideal for all forms of transaction processing. For example, many transactions require additional identity, authentication, and security information, much of which is not available in a stateless environment – and it would be cumbersome for the end user to enter it over and over. Likewise, large messages that carry a lot of information between user and server benefit from a state environment, which eliminates the need to repeatedly exchange this data.

So, while both transaction models have unique advantages, the key from a development perspective is to unify the two in ways that maximize the benefits of each while enabling applications to be consumed using modern protocols. EAE and AB Suite feature tools that allow you to do just that. >>



Stateless Protocols in EAE and AB Suite

Though many EAE and AB Suite developers may think that Component Enabler mandates the use of transaction state in an application, it is actually determined by the way the application has been written. While the presence of GLB.WORK often designates the use of state, making specific changes to other areas of the application code can help you bring stateless standards into an otherwise stateful application. In addition, going stateless is not an all-or-nothing proposition – more about this in the next section.

For example, EAE and AB Suite support Open DTP (formerly XOpen DTP or Open OLTP), and can be used to expose/consume in- and out-bound services. To do so, you can define the EAE or AB Suite application as a server and the individual Ispecs as services. Then, when you generate the application, the View Buffer that is used by the calling “client” program is automatically generated. Similarly, you can call out to an external Open DTP service by importing its View Buffer into a special type of external Ispec, called an OLTP View Ispec (OVI). Any outbound transactions are initiated using the familiar AUTO External command. This setup supports Global Transactions with 2 Phase Commit (2PC) and is able to operate alongside standard transactions and GUI clients using the same transaction libraries.

For this approach to work, the Ispecs must be stateless (i.e., information in GLB.WORK will not be saved). Also, COPY.FROM and “big buffer” Ispecs are not supported. And, data types must conform to XCOMMON buffer definitions – which means they must be simple, fixed length character or number types.

In addition to Open DTP, EAE and AB Suite can leverage Web Services. The Component Enabler Client Generator for ASP .NET Web Services enables you to define Ispecs in a bundle and generate the client interface using a drop-in generator. The ASP .NET Web Services client generator will produce XML and SOAP definitions that allow users to call the Ispecs using standard Web Service protocols. Also, like Open DTP, the Ispecs exposed using this interface must be stateless, and the data types must be simple character and number types. Some key differences between this and the Open DTP implementation include support for COPY.FROM Ispecs, and the ability to define data fields as an input, output, or I/O.

You Can Create the Perfect Blend of State and Stateless

Using the techniques discussed above allows you to write applications that accommodate both state and stateless protocols. Any Ispec (outside of the restrictions noted above) can be defined as an OLTP service or Web Service – without affecting other state and stateless interfaces.

Blending state and stateless transactions in the same application will significantly aid your modernization efforts and make it easier to prepare your applications for consumption in SOA or expose them for use on mobile devices.

Let’s look at a practical example of how this can be done. Say you want to take a traditional green screen banking application and optimize it for use on the Web, without removing the identity and authentication characteristics of the stateful transactions. Remember as you start the process that GLB.WORK is where this type of information is typically saved between transactions. >>

To maintain the necessary state properties of the application as you prepare it to perform stateless transactions, your first step should be to create a new output Ispec (which we'll refer to as "SWORK" throughout this example) to store log-in information each time a new user accesses the application, along with anything else you would normally keep in GLB.WORK. When building SWORK, you should create a special key that acts as the index into SWORK for the user's session. This key will be used to retrieve any needed state transaction data from SWORK, which carries this information between transactions in a single session. The SWORK key is the only additional piece of information that would need to be passed back and forth to the client application during each transaction. Once the user logs off, SWORK should clear the data until the next logon.

If you'd like to build in additional security controls – which may be required for any application transmitting sensitive personal or financial information – you can change the value of the SWORK key with each user session, or even on a transaction-by-transaction basis, if appropriate. To simplify any changes you make, think about creating two Global Logics, which we'll call "GET-SWORK" and "PUT-SWORK." Insert GET-SWORK at the beginning of the transaction to retrieve the state data from the previous transaction, and use PUT-SWORK at the end of the transaction to store state data that will be required in any transactions made until the point where the user logs off or a timeout closes the session.

What's Your State?

Have you done any work to make your EAE or AB Suite application work in a stateless environment? How did you approach this project? [I would love to hear from you](#) – and maybe we'll feature your solution in an upcoming issue of *Developing Agility*.

Have questions or comments about state and stateless transactions in EAE or AB Suite? Send a message to ABSuite@Unisys.com and we'll be happy to help you out.

Partner Corner: Information Exchange Group



Founded in 1995, Information Exchange Group, Inc. (IEG) was formed for a clear purpose – to help businesses maximize the value of the systems, applications, and data powering their day-to-day operations.

Now, some 16 years later, IEG brings substantial experience and expertise to EAE and Agile Business Suite shops looking to design and update their user interfaces, change the appearance and behavior of form elements, and simplify and streamline migrations.

Building Your UI

To help your user interface (UI) remain modern and in line with your business and IT needs, IEG's Interface Builder (IB) provides flexible, automated form-building functionality that enables you to update your UI with ease – and extend and enhance existing applications without significant modification.

Initially released in 1999, the present form of IB delivers true bi-directional coordination with AB Suite – enabling the tool to be initiated as part of an AB Suite Build or Rebuild process or configured to initiate AB Suite Builds itself.

IB automates the process of adding graphical functionality and customizing client or server processing. The tool helps you address a host of UI modernization needs – such as optimizing screen size for different browsers and ensuring fonts are up to date with recommended settings – many of which would require considerable time and IT resources if performed manually on a screen-by-screen basis.

Customizing Forms to Your Needs

With Dynamic Web Alteration (DWA), another powerful tool IEG provides for EAE and AB Suite environments, developers gain a whole new set of possibilities to further enhance web forms.

For instance, you can use this tool to determine under what conditions to invoke such actions as changing a field's background color or disabling a button. And because DWA changes the ASPx form as part of the ASP .NET form Life Cycle, it enables nearly all HTML attributes or JavaScript styles to be dynamically applied to any addressable form element.

When you complement DWA with IEG's Dynamic Web Module (DWM), you can integrate changes based on data dependencies like user logon values. This means that you can control the end user display down to a specific user, if needed. And since DWM is part of your application, all DWA requests become part of your application data repository. >>

Making Migrations Easy

And to ensure that any changes in environment occur as smoothly as possible, IEG's Information Migration process addresses the movement of application data and source code. Ideal for migrations between platforms, across platforms, or from EAE to AB Suite, the process creates report programs, tailored specifically to each of your applications, which automate the transfer of data from existing repositories into a target repository – regardless of whether it's EAE or AB Suite. And because this is done in a platform-independent format, you'll be able to migrate easily from a Windows based platform to an MCP server, for example.

In addition, IEG can help you to bring your EAE applications from a UNIX or Linux system over to AB Suite on MCP and/or Windows servers. When migrating, the process evaluates EAE specifications and automatically replaces reserved words, removes control sequence highlighting, and proposes/changes LDL verb inefficiencies in DETERMINE and LOOK.UP constructs. For web deployment, the Information Migration process assists with processes like resizing the grid used for painting to spread out the form in both directions.

IEG prides itself on being able to address customer requests in a short amount of time. Interested to learn more about what IEG can do for your business? Visit their [web site](#), or contact them today about performing a complementary proof-of-concept project.

Specifications are subject to change without notice.

© 2011 Unisys Corporation.

All rights reserved.

Unisys and the Unisys logo are registered trademarks of Unisys Corporation. Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both. Microsoft, Visual Studio, and Windows are registered trademarks of Microsoft Corporation. UNIX is a registered trademark of The Open Group. All other brands and products referenced herein are acknowledged to be trademarks or registered trademarks of their respective holders.