



Developing Agility

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

Contents

- Page 2 **An Important Announcement for EAE Customers**
Get the details about Unisys plans for EAE 3.3 support.
- Page 3 **How Do You Measure Value?**
There's cost and then there's long-term business value. It's important to look at both when making an IT purchase decision. This article spells out the factors to consider when moving to Agile Business Suite.
- Page 6 **Q&A with Bob Supnik**
As Vice President of Engineering and Supply Chain for Unisys TCIS, Bob has executive responsibility for the EAE and Agile Business Suite programs. Meet Bob and get his perspective on the future of these programs in this interview.
- Page 7 **Engineering Corner: Smarter Debugging**
Paul Bourke, Debugger Technical Lead, shares tips and best practices that make using the Microsoft® Visual Studio® Debugger with Agile Business Suite easier and more effective.
- Page 11 **Mobile Device Integration with Agile Business Suite**
Learn more about how Agile Business Suite developers can design forms for mobile devices.

An Important Announcement for EAE Customers

Enterprise Application Environment (EAE) release 3.3 has been in the market since 2003. It enjoys a strong customer base that uses the product in many different ways. But one thing every one of our customers has in common is that EAE-built applications are essential to their organizations – supporting your most business-critical processes. As such, a migration of any sort, even to a new database or operating system level, is not to be taken lightly. And, neither is the migration to Agile Business Suite.

Agile Business Suite has been on the market for several years now – and its ranks are steadily increasing. It is the future for EAE customers. Now, we’re announcing the next step in our plans for EAE – so that your organization has time to make its own plans to migrate to Agile Business Suite.

The end of Phase 1 support for EAE 3.3, which will vary by platform, is as follows:

Operating Environment	End of Phase 1 Support Date	Revised End of Phase 1 Support (Updated December 2010)
Microsoft Windows®	31 December 2010	31 December 2011
UNIX® and Linux®	31 December 2011	31 December 2011
ClearPath MCP	31 December 2011	No end date*
ClearPath OS 2200	31 December 2015	No end date*

* Customers will be notified at least five years in advance if an end date is defined in the future

These dates mark the start of Phase 2 support for this product suite. 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 to errors they report that don't have a fix available.

Information about support and qualification of underlying software components for each platform is available on the Unisys EAE [Support](#) web site.

We have a wealth of no-charge migration aids, including white papers, How To videos, and more – all posted on the [Support](#) and [eCommunity](#) web sites. In addition, we offer a range of Agile Business Suite Transition services, which target discrete areas of your migration effort and provide the flexibility to tap into Unisys expert assistance when and where needed.

Questions Anyone?

If you have any questions about the plans for end of EAE 3.3 Phase 1 support, we want to hear from you. Please send a note to ABSuite@unisys.com and include your name, organization, country, email, and phone number.

How Do You Measure Value?

By Diane McGonigle, Agile Business Suite Portfolio Management & Marketing, Unisys TCIS

The word “value” has many meanings. At first glance, it’s about money or material worth – the currency needed to procure, or expected in return for, something. It’s easy to understand value in these concrete terms: look at the price tag, negotiate a bit if you can, settle on a figure, and you’re done. But when it comes to major purchases, there’s more to think about than the list price. And, that’s certainly true with IT investments. It’s not easy to put a price on innovation or on the ability to ensure better service, reduce risk, and respond quickly to business opportunities and market changes. However, these are all areas where the right IT investment can have a positive impact.

Consider the year that’s nearly complete. 2009 has presented some tough economic challenges, yet many organizations are continuing to move forward – investing in their future and preparing to seize new opportunities. In its annual *InformationWeek* 500 issue¹, the publication called 2009 a year of “relentless innovation” for the companies that rank in its top 500. These top-ranking organizations continue to view IT as a key enabler to long-term success.

That phrase – relentless innovation – is one that fits EAE customers well. You have built custom applications, refined them again and again to address changing business needs, integrated new technologies, and continued to return exceptional business value to your organization along the way. EAE shops do more with fewer resources and in less time than traditional development teams, and you are to be commended for that.

Now you have a decision to make about when to migrate to Agile Business Suite. And, that decision should be based on total value. It’s more than the software license fee (which, if you have Software Subscription Update licenses, is actually zero). It’s more than the time to plan and test your migrated application. It’s more than educating developers. And it’s more than upgrading or replacing hardware. Agile Business Suite offers all the capabilities and advantages of EAE – and more business benefits than ever.

When weighing the decision to move to Agile Business Suite, you must look at value from all angles – particularly in such key areas as cost, productivity, and innovation. In fact, *InformationWeek’s* survey² of the top 500 revealed the following as key areas where organizations plan to innovate with technology:

- Make business processes more efficient (60%)
- Lower IT or business costs (47%)
- Introduce new IT-led products or services for customers (37%)

Let’s examine each area of economic value in turn and recap what Agile Business Suite brings to the table.

Improved Efficiency and Productivity

Greater efficiency and productivity have been signature benefits of EAE for years – and Agile Business Suite has new features that deliver even more.

For developers, Agile Business Suite offers the opportunity to work in a truly integrated development environment, Microsoft Visual Studio, with modern modeling tools, highly integrated debugging, intuitive navigation, and more. While there are other programming options within Visual Studio, such as C++ and C#, these tools do not offer the tried-and-true capabilities upon which EAE developers rely, such as platform independence; one-button generate and deploy of the database, user interface, reports, and application from a single specification; >>

¹ Chris Murphy. “Relentless Innovation.” *InformationWeek*, September 14, 2009. Page 21.

² 2009 *InformationWeek* 500 survey.

and interoperability with other operating environments. No other development tool on the market can match Agile Business Suite's robust, platform-aware automation of a range of tasks, including development, test, deployment, administration, and maintenance.

In addition, Agile Business Suite enables a true model-driven development approach. It combines all the benefits of defining your application at a higher, business process model level with added features that enable generation of a complete, working system and keep the model and system in sync over time. See our white paper "[Moving Up the Model-Driven Development Curve](#)" for more on this topic.

Agile Business Suite also improves end-user efficiency and productivity. Like EAE, the toolset allows developers and business users to work side-by-side to model business process automation – at the highest level and before any code is developed. The result is faster delivery of new functionality that actually meets end-user requirements.



Because of safe passage, your stable, feature-rich, highly customized EAE applications can migrate to Agile Business Suite without change. This important capability reduces organizational risk and preserves both the end-user experience and all of your carefully developed business logic – an economic benefit that is often overlooked when considering conversion to an application package.

Agile Business Suite offers a wealth of graphical user interface (GUI) options – now including mobile devices – that streamline end-user access and boost usability. Whatever the technology device du jour, Agile Business Suite boasts an architecture designed to evolve and embrace it over time. Your organization can take advantage of new technologies without rewriting the user interface, offering greater end-user efficiency at a substantial economic advantage.

Finally, Agile Business Suite was born in an age of integration and interoperability that favors service-oriented architecture (SOA). It offers many powerful integration features to help your organization harness the power of this new development paradigm.

Manage Cost

From a pure cost perspective, Agile Business Suite has several advantages. First, the toolset costs less to license over the long term, and now includes the Automated Test Tool (ATT) as part of the Agile Business Suite 2.0 Developer. ATT is uniquely designed to capture and playback Agile Business Suite transactions, thus enabling you to automate this important development activity. The added benefit is higher quality applications with a shorter development cycle. The bottom line is reduced cost.

There are also economic benefits related to safe passage – especially when considering the alternative. Most estimates for replacing a mission-critical, custom application with a package solution range well over \$1 million, particularly when you add in any customization and integration work. With Agile Business Suite, you retain all the business process automation in your current applications without incurring any redevelopment costs. You will test the migrated applications, but that's required for a package and for a software conversion project, as well.

Agile Business Suite also allows you to evolve your applications and development tools as technology changes – without having to redevelop or migrate – while shielding developers from undue complexity. It builds a fully functional application from the model, shouldering much of the “heavy lifting” that is involved in generating systems that leverage new advances in technology. >>

Finally, because it's a package in Visual Studio – which is one of the most popular development environments on the market – Agile Business Suite is accessible to a broader pool of developers, who will come prepared with fundamental Visual Studio skills that reduce training costs and their time-to-competency.

Boost Innovation

There's no better way to gain a competitive advantage than by offering a new product or service that no one else has – that's what innovation is all about. Of course, innovation puts heavy demands on underlying IT systems. Fast turnaround on new custom applications or new functionality can make the difference between market leader and laggard. Agile Business Suite enables fast, effective innovation, offering a rare combination of IT capabilities – development, integration, and reuse – in one package. Agile Business Suite applications, like EAE before them, are built for change. It's been the hallmark of Unisys advanced development toolsets for decades – and Agile Business Suite continues that tradition.

Your Trusted Partner – Today and Tomorrow

Agile Business Suite continues Unisys long tradition of supporting organizations with mission-critical, high-volume, transaction-based applications. We understand what it takes to deliver bet-your-business security, reliability, availability, scalability, and performance – and that knowledge is at the very foundation of Agile Business Suite. When you consider everything that Agile Business Suite offers, there's no better economic value for your organization.

Q&A with Bob Supnik

Since joining Unisys TCIS in December 2008 as vice president of engineering and supply chain, Bob Supnik has quickly come up to speed on the broad TCIS hardware and software portfolio. Traveling the globe, he's participated in numerous "deep dive" meetings to immerse himself in Unisys technology — including discussions with Unisys Agile Business Suite development teams in Australia and India. Recently, *Developing Agility* had the opportunity to talk with Bob regarding what he's learned about Agile Business Suite and his vision for its future.

Developing Agility (DA): Thanks so much for speaking with us, Bob. Let's get started with your perspective on the near-term goals for Agile Business Suite.

Bob Supnik: Our most important goal is to grow the adoption of Agile Business Suite in the EAE customer base. It is important that our loyal EAE customers – who have built significant, innovative, and mission-critical applications using our technology – have the confidence to move forward to Agile Business Suite. Another goal is to streamline the product so that we can focus our resources on areas that are most used by – and most crucial to – our customers. We want to be responsive to their needs – and the best way to do that is to center our engineering efforts on areas that return the most value to them.

DA: What are you doing in engineering to help achieve these goals and what benefit will customers see from these activities?

Bob Supnik: Everyone on the engineering team is working hard to make migration to Agile Business Suite as efficient and risk-free as possible. We recognize that there are still some challenges with migration, but we have focused on eliminating of these challenges throughout 2009. We are looking at ways to further automate portions of the migration process. In addition, many of our most experienced staff have personal goals to produce new technical documents, such as Alan Hood's recent white paper "Agile Business Suite Development," that provide an additional level of guidance and expertise to help customers successfully prepare for and conduct their migration.

The engineering team is also committed to fostering a timely, effective, and bi-directional flow of information with Unisys technical services consultants and customer support analysts. Periodic conference calls with various engineering team leads allow technical staff from around the world to share lessons learned and make suggestions for ways to improve the product. "How to" documents also provide a mechanism to share knowledge on both sides with our customers. In the area of streamlining the product, we've taken the decision to focus on development for the ClearPath MCP and Microsoft Windows deployment platforms. This allows us to put more resources to work on areas that most benefit our customer base.

DA: What can you tell our readers about future plans for Agile Business Suite development?

Bob Supnik: We plan to enhance Agile Business Suite in several key focus areas. In the development area, we intend to further leverage the Microsoft Visual Studio technology by expanding our integration with other elements of that platform. Of course, we'll continue to focus on features that promote developer productivity and enhance support for a service-oriented architecture (SOA.).

DA: Any final words for our readers, Bob?

Bob Supnik: I've been very impressed by the talent and dedication of our engineering teams on the Agile Business Suite program – and equally impressed by what Unisys customers are doing with this tool, and its predecessor, EAE. Agile Business Suite represents a very viable path to the future for our EAE customers.

Engineering Corner: Smarter Debugging

By Paul Bourke, Debugger Technical Lead, GTC Australia, Unisys TCIS

Because Agile Business Suite uses Microsoft Visual Studio as the underlying integrated development environment, the Visual Studio Debugger is a developer's primary tool for performing quick testing. Unlike EAE, where Developer Test is a separately executable program, Debugger is a fundamental part of the development environment. While Debugger is similar to Developer Test in that there is no need to generate or compile your whole application before using the tool, with Debugger you never have to "leave" Developer to do some testing. This facilitates a "code-test, code-test, code-test" cadence to development that promotes quality and speeds delivery. This article reviews key examples of how Debugger differs from Developer Test and offers tips on how to best take advantage of these differences to speed your development efforts.

Edit-and-Continue, It's a Brand New World

With EAE, when you identify a bug in your code and want to make a correction, there's a certain process to be followed. You end your Developer Test session, use Developer to locate and correct the flaw in your specification, validate the change, and then start a new Developer Test session. While there is a window where you can see logic as it's executing in Developer Test, you cannot use that window to make the fix – you must find the error in Developer first and then make the change. And, if you've done a lot of prep work to set up your test – like running a complex series of 20-30 transactions or pre-processing 100s of records of data – well, you'll have to do it all over again when you start the new Developer Test session. Finally, raise your hand if you've ever made a fix and forgotten to perform the validation. Many of us know the frustration of returning to our testing and replicating all the setup, only to realize that our fix didn't take because we didn't validate – and we'll have to start all over again.

Welcome to the new world of edit-and-continue with Debugger. Now, when you run across a bug during testing, you simply click the development window tab (where the logic you are currently executing is visible), correct the code, and resume testing. Validation is performed on the fly, all your prep work is still intact, and you're working in a seamless environment where development and testing go hand-in-hand. And, if the on-the-fly validation finds an error, you just try again.

With edit-and-continue, you test as you develop – getting immediate feedback and confirming that logic is performing as desired. You really have to experience it to fully appreciate the boost in usability and productivity you get. What's more, frequent testing at the smallest unit of code is a best practice in any development shop – and Debugger helps you take it to a whole new level.

Debugger 101

To learn more about Debugger basics, take a look at a [previous Engineering Corner](#) that reviews features of the tool in more detail.

To Reorg or Not to Reorg, That is the Question

Whenever you start a Debugger session, Agile Business Suite goes through a "reorganization phase." At this point, any changes you have made to the model that impact the database are analyzed and tables will be reorganized – just as when you Build and Deploy a new version of the application. It is also at this point that changes to the configuration properties are picked up. In addition, as the Debug session is launched, the reorganization phase builds the Debug database for you, so it's important to let it run.

While waiting for a database reorganization to finish up is usually a good time to hit the snack machine in the break room, there are many more times when it's an unwanted delay. So, what about fixes you make during debugging that impact your database layout? Well, if your bug fixes result in the need for big changes to the underlying database, you'll probably want to let Debugger go ahead and reorganize before you resume testing. >>

However, there are many other situations where this isn't the case. Some types of database changes can be compensated for during debug – allowing you to quickly return to your code-test process. For example, if you've added a field, Debugger will still read and write records – pre-filling the field with blank or empty values. Many times, this is enough to keep your test session moving forward.

So here's my recommendation: when Debugger asks you "Should I do a reorg?" – and you know it's not necessary or at least not right away – just say "No" and get back to work.

Streamline Testing of Reports

Debugging reports is another area where a bit of practical advice can save a lot of time. First, I recommend that you set the "Deploy Reports" option on your top level folder to (false). Doing so prevents the automatic deployment of all reports below that level in the hierarchy. It's best to control this process yourself, selectively building and deploying reports when you deem it necessary.

Another tip for working smarter with Debugger is using folders to organize your reports. When you start a Debugger session, a C++ wrapper is generated and compiled for each of your reports. If you've created smaller groupings of reports using multiple folders or even placed the one report you're going to debug into its own folder, then the Debugger session will start much faster, as it will not create wrappers for the reports you're not going to use.

Now, let's assume that you've set Deploy Reports to (false) and organized your reports into folders as I've recommended. At this point, there are two ways the contents of a folder can be built: by right-clicking on it in Developer and selecting build, or by setting it as the root Deployment Folder in the Debugger configuration settings. If you are working on reports in the same folder all of the time (or for a long time), setting it as the Deployment Folder means you can just launch the Debugger and it will know where to go to find the Reports that need to be deployed. Of course, if you find you're changing folders a lot, then setting the Deployment Folder each time will grow cumbersome.

A Few Tips for the MCP Crowd

Debugger faithfully emulates the behavior of the MCP environment, including EBCDIC collating sequences – something that Developer Test didn't do. So, there's no longer a need for the workarounds many MCP shops adopted to compensate for the mismatch in an EAE Developer Test environment. In addition, a number of new tools for working with EBCDIC data have been available since Release 1.2 (via four ICs), including the ability to:

- Open and modify tables from a Microsoft SQL Server® database containing EBCDIC data
- Perform SQL queries against a SQL Server database containing EBCDIC data
- Open and modify EBCDIC files
- Convert files from EBCDIC to ASCII/Unicode or vice-versa
- Perform command line file conversions as per the previous point

These tools come in handy when you need to get direct access to data in your test database.

With EBCDIC handled, Debugger provides an even better platform for testing systems that will eventually deploy to MCP environments. This means more testing can be performed off your ClearPath server – freeing up processor cycles for other tasks and eliminating the wait time for builds. It certainly doesn't dismiss the need to test on MCP, but it does allow you to do more and better testing on your workstation before you move to the next step. And don't forget, we now have the Automated Test Tool (ATT), which was made available with Agile Business Suite 2.0. ATT allows you to capture and replay transactions and compare the results. (Read more about ATT in the [4Q2008 issue of Developing Agility](#).) >>

Tips for Debugging in a Windows Terminal Services Environment

Unlike other Visual Studio languages, such as C# and Visual Basic, developers working in the same project who run Agile Business Suite via Terminal Services are able to use the Visual Studio Debugger to test their applications. However, there are some special configuration changes required to allow multiple developers/testers to use Debugger at the same time.

To use Debugger in a Terminal Services environment, you must ensure that different users' debug sessions do not clash with each other – which can happen when the applications share the same name, are registered as COM objects on the same machine and have the same name in the Agile Business Suite Administration Tool.

The best way to eliminate the conflict is to create a configuration for each user – giving it a name that uniquely identifies it as a Debug configuration for that person, such as DebugFF for Fred Flinstone's configuration. In addition, you must ensure that three Segment configuration properties are made unique for each user:

- **COM ProgId:** the ID used to register the wrapper, which must be unique at the machine level
- **Alternate Name:** the system name in Admin Tool, which must be unique within Admin Tool
- **DB Schema Name:** the name of the schema that will be created in the Debugger database, as well as the name of a SQL user created to log into the Debugger database

To ease administration and management, it's best to use names that are easily identified on sight, such as ClientFF for Fred's configuration of the Client system. All three Segment configuration properties can be set to the same value, so long as they are unique to that user – and do not clash with Windows Runtime systems or other COM objects on the machine.

Test versus Debug: What to Use When

To wrap up this article, I'd like to include an excerpt from an upcoming technical paper written by Alan Hood. The following table summarizes the range of test and debug features offered within Agile Business Suite – and provides guidance about which one should be used when.

Capability	Advantages	Best used for...
Debugger (Using Windows Configuration)	<ul style="list-style-type: none"> • Instant startup from within Developer. • Edit-and-continue features. 	<ul style="list-style-type: none"> • Interpretive debugging and unit testing where relatively small databases are required. • Systems targeting a Windows runtime platform.
Debugger (Using MCP Configuration)	<ul style="list-style-type: none"> • Instant startup from within Developer. • Edit-and-continue features. • Faithfully emulates MCP environment. • EBCDIC tools provided to enable external viewing and editing of database and flat files. 	<ul style="list-style-type: none"> • Interpretive debugging and unit testing where relatively small databases are required. • Situations where platform-specific execution for MCP is required.

continued >>

Capability	Advantages	Best used for...
Debugger with Host Database Access (HDBA) (Using MCP Configuration)	<ul style="list-style-type: none"> • Instant startup from within Developer. • Edit-and-continue features. • Faithfully emulates MCP environment. • Retrieves data directly from DMS II database. • Supports very large databases. <ul style="list-style-type: none"> – Eliminates the need to transfer a large database from one platform to another. • Enables debugging of data-dependent problems using “live” host-based data. 	<ul style="list-style-type: none"> • Interpretive debugging and unit testing with small or large databases. • Environments where platform-specific execution for MCP is required. • Situations where debugging data-dependent problems with large amounts of data is required.
Automated Test Tool (ATT)	<ul style="list-style-type: none"> • Easy to use. • Automates capture and playback of test cases. • Works with generated ASP .NET and Visual Basic .NET clients. • Test case capture can be done by non-development personnel. • Test case playback can be done by people who are not experienced with the application. • Easy to edit test cases • Scripted execution automates the inclusion of unit or regression tests into the build/release process. • No additional charge; it is included with AB Suite Developer 2.0. 	<ul style="list-style-type: none"> • Unit testing and regression testing of existing application components. • Automated regression testing after a system Build or Rebuild. • Leveraging less experienced, lower skilled personnel for routine testing.

With Agile Business Suite, the Visual Studio Debugger, and ATT, developers have very powerful, intuitive tools that not only speed development but improve overall quality – two of the many significant benefits you’ll gain from migration to Agile Business Suite.

Mobile Device Integration with Agile Business Suite

Anytime, anywhere information access is expected by today's tech savvy customers and employees. So, the ability to access your business applications via browsers running on mobile devices is becoming more important.

Recent product enhancements included in the latest set of Client Tools ICs for Agile Business Suite improve integration with mobile device browsers. For example, we now support the Safari web browser, which runs on the iPhone.

To provide more information about the ins and outs of mobile device integration, our engineering team has developed a white paper: [Developing Web-based Applications for Deployment to Mobile Devices Using AB Suite Client Tools ASP.NET Generator](#)

The paper is downloadable from the Unisys Support web site. If you don't have access to the site, [send us a note](#) and we'll forward a copy. But remember, organizations are permitted to have multiple sign-ons to access Support – and there's a ton of great information posted there. So, request your sign-on today!

Specifications are subject to change without notice.

© 2009 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, SQL Server, Visual Studio, and Windows are registered trademarks of Microsoft Corporation. UNIX is a registered trademark in the United States and other countries, licensed exclusively through X/Open Company Limited. All other brands and products referenced herein are acknowledged to be trademarks or registered trademarks of their respective holders.