



Developing Agility

*A Quarterly Newsletter for
Unisys EAE and AB Suite Customers*

Contents

- Page 2 **Make a Plan to Migrate**
Andy Wardle from Unisys UKMEA reveals the four essential steps to building a successful migration plan for AB Suite.
- Page 5 **Engineering Corner: Migration Best Practices**
Curt Leong discusses the six best practices ACUS has observed in helping organizations successfully migrate to AB Suite.
- Page 7 **CZ in Production with AB Suite**
This top Dutch insurer has completed a migration to AB Suite.
- Page 8 **Words of Wisdom**
Unisys field personnel share their best tips for successful AB Suite migration.
- Page 10 **A Better World for Developers: Microsoft Visual Studio + AB Suite**
Learn more about the benefits offered by the combination of these two technologies.
- Page 11 **Too Good to Wait: AB Suite ICs Deliver Significant New Features**
Find out more about the new capabilities offered by the latest AB Suite ICs.
- Page 11 **Calendar**
Check our calendar for information about upcoming events.

Make a Plan to Migrate

By Andy Wardle, Senior Architect Unisys, UKMEA

In the IT world, migrations are always approached with caution. But with proper planning, they don't need to be frenzied, hair-raising experiences. In fact, the migration experience can – and should – be quite the opposite during your move to Agile Business Suite (AB Suite) from Enterprise Application Environment (EAE).

Unisys has created a four-step process to simplify your migration. This article offers a framework from which to begin or enhance your AB Suite migration plan.

It is important to remember from the onset of the project that your migration to AB Suite is nothing different from *any* migration. The details will vary from previous efforts, but the basic premise and approach are the same. In that sense, this article is not teaching you anything you don't already know – it's simply a reminder of best practices.

Step 1: Review Your Current Environment

This step can be boiled down into one statement: *document what you have today*. The best way to ensure a successful migration is to know where you're starting from – such as your current hardware, software, utilities, procedures/processes, and so on. Be sure to assess the following areas in particular:

Features and capabilities

EAE provides a broad set of capabilities. Now is the time to take a step back to assess what features your developers are using. For example, are you using the local debug/test facilities? Do you use version control? Are you using some of the more unusual EAE functions, such as generate scripting, graphs, developer security, or changes to EAE skeleton files? Are you using the discontinued PowerClient or other products or interfaces that are no longer supported?

Everything but EAE

Step 1 is the ideal place to consider non-EAE applications and utilities that could be affected by your migration to AB Suite. You may need to upgrade or replace operating systems, utilities, and ancillary applications – and some may no longer be necessary. Some organizations focus their migration/testing efforts on the EAE application itself – disregarding integration with other applications or external systems. Don't let that happen to you!

Procedures vs. processes

Procedures are your formal, documented guidelines – and processes are how people actually execute. Be honest and thoroughly examine how your developers and operations staff members function on a day-to-day basis. Many times, this exercise will uncover differences you didn't even know existed. Once you document what's really taking place every day, you can better plan for the new world of AB Suite.

When Step 1 is complete, you should have a single document that is broken into various sections covering every aspect of your existing environment. The more formal you make this exercise, the more information you will extract, and the more accurate the final migration plan will be. >>

Step 2: Define Objectives and Plan for a Proof of Concept

After evaluating and documenting your current environment, Step 2 is about preparing to conduct a proof of concept (POC). A POC is critical to gaining confidence in AB Suite, Safe Passage, and your migration plan. It will help you work through potential problems and establish and confirm your requirements for a successful migration.

Be sure that your POC has a well-defined scope and objectives so that when it's done you are confident that you have all the information needed to move to AB Suite. It should be time-bounded and limited to a subset of the application migration workload.

During Step 2, you select an application or group of applications to run through the POC, define the POC activities, and develop a rough outline of what will become your formal migration plan. Throughout this pre-POC stage, consider the following:

- What skills – and associated training – do you need to complete the job?
- What additional or new software licenses are required, including Microsoft® Visual Studio® SQL Server®, source control, and other tools/utilities? Some of these may come with your AB Suite order, but this is the time to verify that everything you will need has been ordered and will be in place when you need it.
- What new hardware, particularly for development, will be necessary? (Many organizations we work with haven't updated hardware since their migration to EAE 3.3 – now is the time to consider a refresh.)
- What staff resources (internal/external) will be required? (This should go beyond developers to consider resource demands on operations, test, and end users.)
- How much will the migration project cost the organization?

The project migration plan you develop during this step will be confirmed during Step 3 when the POC is conducted. And remember – you have done migrations before so draw from that experience as you create the plan to move to AB Suite.

Step 3: Conduct a Proof of Concept

Step 3 simply enacts the major activities from the plan you created in Step 2 – testing everything you'll do when you finally go live. Organizations with many and/or large EAE applications, will run a subset of their systems through the POC. Those with smaller systems may use the POC as the first step to the actual migration to AB Suite. Key activities during the POC include:

- Training for personnel directly involved in the POC
- Migration of the POC application(s) using the procedures outlined in the *EAE 3.3 to Agile Business Suite Migration Guide*.
- Testing the new AB Suite Runtime environment, as well as any interfaces, to confirm safe passage from EAE (batch and online processing, of course!)
- Fully evaluating the AB Suite development environment, including System Modeler, Debugger, and Builder – and the supporting procedures/processes required for your particular environment.
- Testing the implementation of an application change to evaluate source control tools and test/refine associated processes.

The POC should help you investigate – and really experience – some of the new capabilities that AB Suite has to offer. The end result will be a well-thought-out migration plan that identifies the different areas that will be involved in the project and the resources required to do it. >>

Step 4: Create a Formal Migration Plan

You have completed the POC and are happy with the outcome. Now you need to finalize the formal migration plan, which will draw heavily from your experiences. Put simply, this step is “project planning 101” – set the objectives, establish your exit criteria, define the timeline and task sequence, and allow for corrective action.

But there are a couple of particularly relevant lessons we have learned while working with clients, including:

Establish a comprehensive set of functionality and performance tests

And, don't forget to capture data to drive these tests before you start the migration process so you have something with which to compare afterward. Also note the need to capture more than transaction logs. Be sure to assess and measure current performance metrics so you have a non-biased baseline for comparison.

Schedule training to match the migration timeline

The importance of training can't be emphasized enough – but that training must occur near the time your developers will be using AB Suite, not weeks or months in advance. If there is a delay in an earlier step that pushes the timeline back, the scheduled training should also be pushed back appropriately.

Summary

If you take nothing more away from this article than the following three points, we will have done our job:

- A *written* migration plan – with a quantified, formal business case – is essential. Without one, you have no chance of gaining that needed business buy-in.
- Key migration activities must be completed, including a review of your current environment and conducting a POC to test and enhance your formal migration plan.
- Training is essential. You could train yourself, but there is benefit in learning from people with more experience with the product. Unisys and some Unisys partners offer services that can be tailored to meet your needs.

Finally, remember that this project is not about AB Suite, it's about migration – and you have done this before. Rely on your past migration experiences – and take advantage of the expertise Unisys can offer, too. And, please keep your Unisys contacts updated regarding your plans so we can help make your migration to AB Suite a success!

Unisys is currently developing a service engagement that captures EAE transaction log files and re-plays them in the AB Suite environment – for a fraction of the cost of other test tools. You can expect to see this service being offered soon. Of course, if you are already using HP WinRunner, HP LoadRunner, or similar tools for other applications, they will serve the same purpose.

Engineering Corner: Migration Best Practices

By Curt Leong, Engineering Manager, Australian Centre for Unisys Software (ACUS)

ACUS has taken a keen interest in helping organizations with their migrations to AB Suite – and we plan to stay involved as more of you make the move. Along the way, we have worked closely with a number of customers – supplementing their own expertise and that of local Unisys field support personnel. As a result, there have been many lessons learned and this article covers six of the most important.

Looking back, the collaboration between Unisys engineers and field personnel and our customers has been a win-win situation all around – and resulted in a much stronger AB Suite. We've helped customers remove road-blocks and receive answers to critical questions. We've helped developers find more efficient ways of using the product – via on-site visits or through remote support sessions. And, most importantly, we've learned more about your needs and used that knowledge to fine-tune the product. For example, we've identified and made changes to AB Suite that have improved developer productivity, runtime report performance, and debugging – all based on working with our early adopters.

Six Best Practices for AB Suite Migration

1. *Develop a good understanding of the differences between EAE and AB Suite*

While AB Suite is the successor to EAE, it's grounded in a contemporary development philosophy and approach that are quite different from EAE. To get the most benefit from AB Suite, it's important to understand the differences, particularly in the area of development. There are several things you can do to build up that knowledge, including:

- Learn about the AB Suite development philosophy by viewing the [Object Orientation: Model Concepts](#) audio slideshow on the Unisys [Support](#) website.
- Get the right training *and reinforce that training with practice*. Plan a formal activity using AB Suite Developer to force yourself to go beyond what you learn in the classroom and apply your newfound knowledge to a real-world project.
- Carefully time training to align with your migration plans – don't train too early or too late.
- Pace the training – starting with core concepts, reinforcing those concepts with practical projects, running the next training class, doing more practice, and so on. You'll find that you retain a lot more information using this approach.

2. *Keep Unisys in the loop*

The sooner we know that you've started migration planning, the sooner we can help guide you to the available resources that will help make your project a success. Send an email to: absuite@unisys.com – we check this mailbox at least once a day.

3. *Share your clean, valid EAE 3.3 model file with Unisys*

When you send us your model, we'll take a quick look to make sure it will migrate without errors. We've found this gives organizations an extra measure of confidence as migration is starting.

4. *Conduct a proof of concept*

A proof of concept (POC) is an invaluable activity for testing your migration plan (see the [Make a Plan to Migrate](#) article in this issue of *Developing Agility* for more.) Customers who have done a POC are much better prepared to manage the actual migration and effectively function in the new AB Suite environment. Plus, POCs help reinforce the training you've received. >>

5. Consider how current processes will change with AB Suite

Because of AB Suite's enhanced capabilities in several areas, we recommend that you assess each of the following and adjust your current processes as needed:

- **Version Control:** Unlike EAE, where the Version Control optional product was a Unisys-developed solution, AB Suite comes with a third-party version control tool – Borland® StarTeam®. It's important to update your current processes to reflect this new capability and approach.
- **Ongoing development during migration:** With AB Suite, we offer a migration database, which is a special version of the AB Suite repository that helps you introduce changes from your EAE environment into your migrated AB Suite environment. As you undergo the migration, your EAE production system may continue to change. The migration database helps you keep the two environments in sync while you complete migration testing and make the final go-live determination.
- **Debugging:** As a plug-in to the Microsoft Visual Studio environment, AB Suite takes advantage of the integrated Debugger. The value of this close integration should not be underestimated and your current processes should be updated to reflect this greatly enhanced capability. For more about the integrated Debugger, refer to a previous Engineering Corner article, [Agile Business Suite Debugger](#), in the 1Q 2007 issue of *Developing Agility*.

6. Make sure your migration plan is solid

Migration planning is also discussed in more detail in the [Make a Plan to Migrate](#) article in this issue of *Developing Agility*. We'd like to note some additional best practices we have learned from our work with customers, such as:

- Clearly define your acceptance criteria and get buy-in from all parties so that there's common agreement around what must be accomplished to go live.
- Make sure your test plan is comprehensive – covering all system functionality and interfaces.
- Define your performance requirements from the new environment – and make sure you have the hardware to support those requirements. Check the Software Release Announcements (SRAs) for AB Suite, posted in the [eCommunity](#), for environmental requirements.
- Take note of the Interim Correction (IC) schedule, published on the Unisys [Support](#) website, and plan your go-live date to take advantage of the most current or planned release, so you have access to all the latest enhancements and corrections.

The entire ACUS team joins me in encouraging you to start the migration to AB Suite. We look forward to helping you be successful and continuing to learn more as we collaborate. >>

Who is Migrating to AB Suite?

The short answer: organizations of all sizes and types, including:

- *Financial services, government, manufacturing, and communications*
- *From one to more than 50 developers*
- *With applications having fewer than 10,000 lines of LDL to a few million*
- *From small databases to very large ones*

A number of organizations are well on their way to AB Suite – we've worked with customers that are:

- *Using the same model to deploy to different runtime platforms*
- *Continuing to develop with EAE but deploying with AB Suite*
- *Migrating green screens in EAE to graphical user interfaces in AB Suite*
- *Migrating 40+ developers to AB Suite*
- *Modernizing their EAE GUI client screens to newer .NET technology available with AB Suite*
- *Migrating from EAE on ClearPath MCP to AB Suite on Microsoft Windows®*
- *Migrating multi-language applications*
- *Moving from EAE Version Control to Borland StarTeam Version Control, which is included with AB Suite*

And this list is just a sample of the many different profiles, approaches, and goals that we have seen. Our point? Every migration is unique – and we're standing ready to help yours be a success.

CZ in Production with AB Suite



CZ is one of the top three healthcare insurance companies in the Netherlands. A long-time EAE customer, the company recently migrated its ARIS application to the AB Suite Runtime environment. ARIS has approximately 4,500 function points, 225 Specs, 130 Reports, and 100 end users.

After a period of intensive testing with good support from the ACUS team, the final migration step was a great achievement. The application went live on July 3, 2008 running under a Microsoft Windows operating environment. After several weeks in production, CZ reports that the performance and stability of the AB Suite Runtime environment looks good. CZ plans to continue maintenance and development of ARIS using EAE.

ClearPath Connection – Have You Read it Yet?

[Check the eCommunity](#) to view a recent issue and subscribe to this quarterly electronic newsletter for the latest news about Unisys ClearPath products, services, and events.

Words of Wisdom

Migration to Agile Business Suite is the focus of this issue of *Developing Agility* – and the discussion would not be complete without hearing from some other Unisys field personnel, who have been working diligently with customers like you on migration. We were lucky to catch a few of them in the office to gather the following tips.

Developing Agility: What is the single, best piece of advice you have for customers regarding migration planning?

Rob Henrichs (Unisys Netherlands): Do a trial migration and be sure to take enough time to test. This is the best way to build confidence in AB Suite and your ability to move forward.

Dave Jordan (Unisys North America): Don't forget to "think outside the box" and consider the impact of migration on other things, such as your network, WFLs, user accounts, and data.

Henk Hoogland (Unisys Netherlands): If your current EAE infrastructure and development is done through multiple departments, then get them involved from the start. The best way is to use a project room where they can sit together so that the lines of communication are short.

Gary Taylor (Unisys UKMEA): If you need to change the end-user client interface, then do so before migrating. This ensures that any issues are resolved before your migration – and they aren't wrongly attributed to the move to AB Suite.

Developing Agility: What are one or two best practices that you can share with our readers when it comes to migration planning?

Rob Henrichs: When you have multiple applications or runtime environments in EAE, consider using the Migration DB feature so you don't need to do a big bang migration. This way you can first migrate your production runtime environment and then your development environment at a later date. Also, schedule AB Suite developer and operations training right before migration so people can go from theory to practice with the shortest possible delay. It is a waste of time to take the courses months before going live.

Dave Jordan: Pick one or two client interfaces and focus on how they will work with your new environment. Also, keep things simple and don't make too many changes until the system is migrated. Embrace the challenge of coming up to speed in AB Suite – there are significant benefits once you migrate. Finally, don't migrate old things – clean up your EAE applications first.

Henk Hoogland: Get ACUS involved in the project from the start so the engineers have a good understanding of how your organization is currently using EAE. This will make it much easier to communicate during the migration project.

Gary Taylor: Take time to document/measure your current environment, particularly with regards performance. Then, set realistic expectations for post-migration performance. And don't short change your staff training. Yes, you can train yourself using a CBT. However, my experience is people come up to speed far faster with formal training (and, likely, for a cheaper price). As such, any migration plan should include formal training and, just as importantly, involve at least a minimal element of Unisys consultancy to ensure current best practices are followed. >>

Developing Agility: Is there any particular area where organizations under-prepare during the migration planning process? If so, what should be done during the migration planning stage to correct that oversight?

Rob Henrichs: Interfaces with other systems are often overlooked. Be sure to test interfaces during the trial migration and document all configuration issues.

Gary Taylor: There's a big difference between formal and informal testing. Quite often, sites will conduct testing by allowing existing users onto a test environment and asking them to test – without precisely defining what should be done. Users need more formal guidance, such as a tick sheet or checklist of areas to cover. This helps to ensure that all major areas are tested rather than just those that are easy for users to test. It also documents the requirements as to what level of data is required in the test environment (e.g. full production clone, just test data, etc.).

Developing Agility: What are one or two common concerns people have about migration and what is your advice to overcome them?

Rob Henrichs: Many EAE developers are concerned about learning to use Microsoft Visual Studio for AB Suite development. I recommend taking a generic Visual Studio class before taking the AB Suite training. Likewise, if object-oriented (OO) development is new to you, it's probably a good idea to take an OO concepts course.

Dave Jordan: I second what Rob said about building up your skills in Visual Studio. And the more you work with it, the more comfortable you will feel. Having two monitors can help, too.

Gary Taylor: I often hear concerns about how very different AB Suite is to EAE. It's important to remember that for end users there will be no difference at all. Yes, development is different. But even here, developers will be able to see their existing EAE code with very little change. So, for the majority of people they will see little-to-no difference. And for the small minority of people that do need new skills, Unisys can supply training to help them make the transition more easily.

Thanks to Rob, Dave, Henk, and Gary for taking the time to share their insights with our readers!



Rob Henrichs
consultant, Unisys
Netherlands,
20+ years working
with EAE and
AB Suite



Dave Jordan
consultant, Unisys
North America,
25+ years working
with EAE and
AB Suite



Henk Hoogland
consultant, Unisys
Netherlands,
16+ years working
with EAE and
AB Suite



Gary Taylor
senior architect,
Unisys UKMEA,
20+ years working
with EAE and
AB Suite

A Better World for Developers: Microsoft Visual Studio + AB Suite

The migration to AB Suite puts a vastly modernized toolset at the fingertips of EAE developers. This article provides a quick summary of how the combination of AB Suite and Microsoft Visual Studio offers many important advantages that are sure to make your future development efforts even more successful.

Use a Contemporary Development Tool

AB Suite is a very industry-current development toolset that puts the power of modern, model-based, object-oriented (OO) development techniques to work for your organization.

True to its shared heritage with EAE, the focus remains on the ability to quickly and incrementally evolve your systems to align with business needs – while providing the highest levels of productivity.

But as a Visual Studio package, AB Suite offers even more. It helps EAE developers enter a world where:

- Classes, methods, inheritance, and other OO concepts are part of the daily routine
- Integrated capabilities, such as debugging, enrich the development experience
- New levels of reusability are possible
- Graphical user interfaces are a natural extension of the development process
- Deployment to native .NET and Java Enterprise Edition environments becomes effortless

And, AB Suite does all this while helping you preserve and leverage all the “application intelligence” built through years of EAE development.

Build Composite Applications

AB Suite offers the ability to mix AB Suite-developed software components with other Visual Studio projects into one solution within one overarching development environment. And, that includes ASP.NET clients built with AB Suite, as well as with other tools. AB Suite is a true enabler of services-based solutions and supports your organization’s ability to take advantage of the value provided by a service-oriented architecture (SOA).

Promote Information Sharing Across Developer Teams

Because Visual Studio is a shared environment, AB Suite and non-AB Suite developers have much better visibility into what each other are doing. In fact, some AB Suite early adopters have reported that this shared environment is breaking down traditional walls between the various teams with varying skill sets and actually promoting a much more active level of engagement and collaboration. This unanticipated benefit means that organizations are getting even more value and, potentially, better applications thanks to the synergy between developer groups.

Access to More Developers

No discussion about the advantages of the AB Suite development environment is complete without a mention of the fact that there are vast numbers of Visual Studio developers in the market. This means your organization can more easily find resources to help with projects – without paying a premium.

A better world is waiting – what are you waiting for?

Here’s how you can get started:

- Do some reading, check out [Hood’s OO Reading List](#) in the 2Q 2006 issue of *Developing Agility* for suggestions
- Take a tutorial on Visual Studio and OO concepts
- Request an evaluation copy of AB Suite Developer (send an email request to ABSuite@unisys.com)
- Attend a Unisys AB Suite Developer training class or try the Getting Started with AB Suite Developer CBT
- Engage Unisys help to perform a Migration Readiness Assessment and/or Migration Planning Service

Too Good to Wait: AB Suite ICs Deliver Significant New Features

Three Interim Correction (IC) releases have been made available for AB Suite since May 2008: IC 1.2.1550 (May), IC 1.2.1600 (July), and IC 1.2.1650 (August).

What makes these ICs so special is the number of significant new features they contain – capabilities that were just too important to wait for the next major release.

AB Suite Developer new features include:

- EBCDIC Debugger Tool
- Debugger edit-and-continue improvements
- Debugger Report output format
- Advanced settings for import GUI

AB Suite Runtime for Microsoft Windows .NET new features include:

- OLTP
- User-defined station names
- Unicode data types

To learn more about the above features, read our [summary article](#) in the AB Suite section of the eCommunity.

And to get all the details regarding the contents of a particular IC, check the readme and documentation updates for each product component. You can find these files in the AB Suite section of the Unisys [Support](#) website.

Calendar

It's not too late to sign-up to attend one of our premier events for AB Suite and EAE customers, which are scheduled for September and October. See the details below.

What	Where	When
Agile Business Suite and EAE User Meetings (Choice of two dates)	Unisys International Management Centre , St. Paul de Vence, France	September 22-24, 2008 September 24-26, 2008
UNITE Annual Technology Conference	Caribe Royale Orlando , Orlando, FL	October 19-22, 2008

Specifications are subject to change without notice.

© 2008 Unisys Corporation.

All rights reserved.

Unisys is a registered trademark of Unisys Corporation. Microsoft, Visual Studio, and Windows are registered trademarks of Microsoft Corporation. Borland and StarTeam are service marks, trademarks, or registered trademarks of Borland Software Corporation or its subsidiaries in the United States and other countries. All other brands and products referenced herein are acknowledged to be trademarks or registered trademarks of their respective holders.