Attachmate

WHITE PAPER

Safe and Economical Reuse of Legacy Applications

When web enablement is the best fit

The uncertain economy has resulted in tighter budgets, and tighter budgets typically call for using what you already have. It's not surprising, then, that IT organizations worldwide have adopted this driving principle: make sure that existing assets are providing the biggest value to the broadest audience.

That's why more and more IT focus is lasered on the extension and reuse of core enterprise applications. And as organizations progress to the modernization of legacy applications, they quickly confront issues of risk and cost. The good news is that technology advances can help you meet your goals without breaking your established business processes ***or*** your budget.

In the case of mainframe applications, the target audience and skill sets have often been specifically preordained. Because they were never built with broader use in mind, these monolithic systems have their logic tightly intertwined with their data. But they might not be as "locked up" as many IT professionals think. Two good options for legacy modernization are service enablement and rejuvenation. Let's look at them individually.

Service Enablement

Service enablement is the process of deconstructing an application into individual building blocks. It's done by encapsulating functionality in the form of discrete, interoperable components that become independent of their source application. You can use the resulting services in any combination needed to build new business processes.

The more applications you service enable, the closer you move to a service-oriented architecture. SOA is well-known for its ability to make IT infrastructures more open and agile; reusable services present a new world of opportunities for combining older and newer technologies. This approach is especially notable when it comes to mainframe modernization, which is considered by many as the "last mile" of SOA because it's often delayed until the urgency becomes impossible to ignore.

In a real-world example, a state government agency recently used service enablement to streamline its traffic-ticket processes, which had been requiring too much hands-on attention. The state IT team extracted specific services from their core mainframe application and used them to create a composite application that automates the transactions. The new system eliminates manual data entry by reading an XML stream and automatically entering the data into the master green-screen application.

Extension of legacy applications via service enablement is becoming more prevalent as IT managers discover better ways of accomplishing the task. In fact, an abundance of published information on service enablement and SOA already exists. For that reason, this paper focuses on another, related modernization option: rejuvenation.

Rejuvenation

Although service enablement is an effective way to make applications more flexible, not all enterprises need to pursue a services strategy for each and every application. There are some cases where service enablement would not add enough business value to justify this approach. That's why it's critical to take a cold, hard look at your actual needs. And part of that assessment calls for a realistic evaluation of the skill set required to use the application.

An IT manager might want to retain the present skill set and majority of the application workflow, but make it easier for end users to handle. This goal is most commonly (and quickly) achieved by providing a new web interface and, as needed, simple controls over application workflow. It's a process that is becoming popularly known as application rejuvenation.

The decision to provide a new UI is typically driven by users' needs to have a more modern front end or more streamlined interaction with the application. Simplified data entry is another driving factor. In some cases, improved efficiency calls for the ability to do error handling directly in the UI. Rejuvenation can almost always address issues like these.

Depending on the needs of your enterprise, applying a modern UI can have substantial advantages over service enablement. For starters, it's a faster and simpler approach. Instead of creating services and

using them to develop new or composite applications that perform a different function, you just leave the application as-is and give users an uncluttered, dynamic front end. The application, processes, and users stay the same but the user has a different experience. (This approach works well for new users, too, because extensive training is not necessary.)

The real advantage of rejuvenation is that you are reusing an existing application, already proven in the enterprise. This reuse speeds time to market and minimizes the risks associated with a new application rollout. These attributes – time savings and risk reduction – are critical in reliably attaining your return on investment.

Rejuvenation: The User Perspective

Take a green-screen order-entry application for example. Simple rejuvenation means that end users no longer have to deal with obscure character-based commands, or rely on "sticky notes" to know the next action needed for completing an order. With rejuvenation, users can step through a basic workflow that's spelled out on a friendly web screen right in front of them. They get the kind of interface they need to be most productive, without being bogged down in behind-the-scenes complexity.

In a real-world example of rejuvenation, a large medical center recently simplified access to patient data residing on their AS/400 application. The hospital provided a web interface that lets doctors access data in a new way, but with all of the functionality they've become accustomed to. The browser-based front end provides an intuitive layout with comprehensive, real-time information that medical staff can readily put into action. The result is better patient care.

Rejuvenation: The IT Perspective

From an IT perspective, rejuvenation is advantageous because you can leverage so much of the application infrastructure as it sits, with all its inherent benefits. The method is fast and cost-effective because no new host code or modifications to the application are required. Let's take a closer look at some IT capabilities that come with application rejuvenation:

- **Leverage an application, in whole or in part, to accommodate changing business strategies.** With a rejuvenation approach, you can tailor an application for new audiences, beyond the original intended use. Rejuvenation even allows the creation of multiple different workflows for different audiences from a single application.
- **Avoid the costs and risks of a platform migration or rewrite.** By leveraging the existing enterprise or legacy assets where they are, your chances of downtime and diminished customer service are mitigated. You retain all the abilities and uptime protections of the mainframe environment.
- **Inherit and extend security.** You directly leverage the full level of security, authentication, and authorization from the enterprise platform. Rejuvenation makes the application safely extensible without undue burden as the existing protections remain in place. You also maintain control over the provided workflow, and that means you can offer specific, safe uses for intended audiences.
- **Use skills and resources you already have.** With rejuvenation, you can leverage existing in-house middle-tier technologies to facilitate reuse of applications. Extending the legacy aplication is a simple matter of understanding its workflow and attaching a modern UI. The usual low set of legacy needs will allow a savvy mid-tier expert to easily web enable a legacy application. No new legacy expertise is necessary.

So rejuvenation, applied to the right circumstances, clearly has strong IT benefits. These benefits become even more significant when dealing with mainframe applications, which are typically closed, complex, and difficult to leverage.

Getting Rejuvenation Right

Benefits of rejuvenation notwithstanding, the simple concept of tying mid-tier technology to a legacy application is not a concise, actionable plan. It's critical to understand first how the legacy application is married to the mid-tier, and which set of mid-tier technologies to use.

The best way to move forward is to design a system that capitalizes on the inherent benefits of the fast-paced technology advancements occurring in the mid-tier. The goal: with the least possible change to legacy systems, create a hand-off from the legacy application that's not tied to a specific mid-tier implementation or desktop platform.

One way to do it: create a new web UI

An example of this approach might be for a legacy application to feed its embedded UI to a mid-tier tool. The tool should then be able to route the legacy UI in whole or part to a web server. (From the web server perspective the routed UI appears simply as blocks of content.) There, modern web-design and web-application capabilities can be applied to mix, match, and overlay the legacy UI.

If the mid-tier UI is completely auto-generated from the legacy feed, then the project scope for rejuvenation

is small and the project duration is short. A nice outcome perhaps, but one that is bound to become a problem. The mid-tier-generated UI is basically fixed to that of the legacy application and will not easily accommodate changes – like the modifications that inevitably arise after the project is up and running. In this case, the finished product is a static solution directly tied to the legacy UI. What you get is yet another legacy asset.

A better way: create *building blocks* for the UI

The difference between a tool that creates a web UI and one that creates the building blocks for a web UI (where control over div tags and use of custom cascading style sheets [CSS] can be applied) is substantial. Note that, if the legacy UI is simply used to generate web-application building blocks, then another step is needed to generate an actual mid-tier UI. But this additional step allows the mid-tier look, feel, and flow to be independent of the legacy application.

With a building-block approach, standard web infrastructure and web-design tools can be used to generate the actual UI. This process provides the freedom to use the legacy UI and flow as-is, or to deviate from it in any way necessary. Also, as web infrastructure and tools advance, these advancements get inherited, so the new abilities can be applied to the legacy-derived building blocks. What you get is an open-ended method for easy extension of a legacy application.

The Value of Web Technologies

If you've decided that rejuvenation is the right approach for your enterprise, you might be wondering about the best way to get it accomplished. Real-life implementations have shown that web technologies consistently deliver the most effective, reliable results. Tools such as XML, HTML, and CSS seem to be made for projects like rejuvenation.

[Note: The web technologies discussed here are not to be confused with web services, which are typically associated with SOA. While it's true that web services technologies are a subset of web technologies, we are concerned here only with presentation web technologies, especially as they relate to application rejuvenation.]

When doing a rejuvenation project, you can take advantage of the expansive and fast-moving world of web technologies to allow the use of web browsers as ubiquitous clients. In fact, web browsers are becoming more and more like the desktop itself. Now, thanks to modern web technologies, browser applications are not just "re-skinned" HTML representations of legacy applications. Instead, the robust new browser applications can provide the rich application experience users need to be as efficient and productive as possible.

Modern web technologies can help you meet evolving expectations about application performance across the spectrum of internal and external audiences. In addition, you can make use of the common and current web skills already existing on your IT staff. And there's another reason to consider web technologies: Standard web infrastructures can be easily leveraged. So, as new technologies emerge, you'll be on the right platform and ready to meet your next IT challenge.

Attachmate Verastream Host Integrator: Built for short- and long-term modernization strategies

Of all the application-reuse solutions on the market, only Attachmate® Verastream® Host Integrator uses web technologies to deliver both service enablement and web enalement with equal ease. Using Verastream, you can encapsulate mainframe data and logic via the application interface for participation in today's SOAs, as easily as you can provide a quick-turnaround UI for that application. And you can transform the full range of enterprise host applications into SOA assets

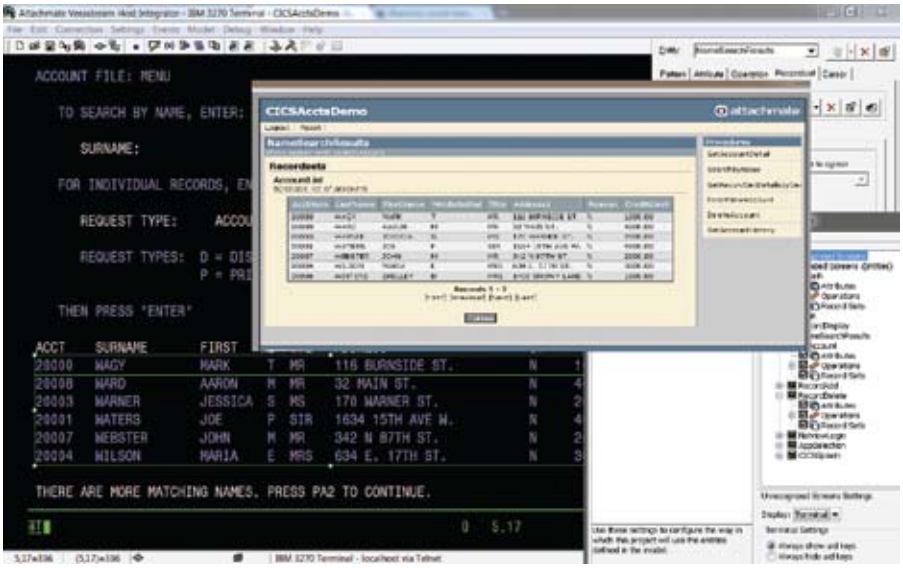
Using Verastream for Rejuvenation

Verastram Host Integrator is both a tool and platform that can safely extend the reach of your enterprise assets. For web enablement needs, a Verastream rejuvenation process follows two simple conceptual steps:

1. Target the legacy appliation and decide what level of its UI to leverage.
   - For the desired auto-generated content, pick a default theme for the look and feel.
   - Use a GUI to examine, identifiy, and model the parts of the legacy UI where additional control is desired.
2. Create or extend Verastream-generated cascading style sheet (CSS) to define the mid-tier UI.
   - Leverage any mid-tier web development environment of your choice to pull in Verastream content as services or divs.
   - Create custom forms to govern presentation of any specific aspect of the legacy application that should not be processed through the standard or enhanced CSS.
   - Define customer CSS for control over display attributes and layout.

by exposing business processes as web services, XML, Java, or .NET components, as easily as you can show someone how to use a browser.

With Verastream, you can take an incremental approach and build toward SOA, or quickly complete that one-off rejuvenation project that you have been postponing. Either way, you get rapid results because you can use existing development skills and familiar IT tools. Whether your environment is IBM System z, IBM System i, Unix, OpenVMS, or HP e3000, Verastream Host Integrator can help modernize your legacy assets—without disturbing mainframe-application code or daily business operations.



With the Verastream design tool, you can maintain complete control of the selection process when converting green screens to web GUIs.

The Verastream design tool

Verastream Host Integrator uses an advanced design tool to tie a legacy UI to the mid-tier interface. The design tool was specially built to analyze and control legacy applications. A typical Verastream modeling process involves these steps:

1. Open the tool.
2. Create new model and name it.
3. Add legacy screens you want to tailor in the mid-tier interface.
4. Optionally, use the design tool to create server-side procedures to embed into the mid-tier workflow.

Using this modeling process, any unidentified screens can be identified for auto-generation or specifically excluded from the mid-tier UI.

The Verastream web builder

You can use the Verastream web builder to generate the components (building blocks) for the mid-tier UI. Using a Verastream Host Integrator model, you can build either ready-to-use web application projects or component interfaces, which provide developers with the building blocks necessary to create their own custom applications. Here are the typical steps:

1. Choose mid-tier environment, Microsoft .NET, Java, or ASP / JSP procedure-based web applications.
2. Expose both procedures and screen funtionality in one web-based presentation.
3. Rejuvenate both modeled and unmodeled screens.

With the Verastream web builder, modeled screens can be displayed as HTML-based forms or as a simulated host screen, while unmodeled screens can be displayed as simulated host screens. The web builder lets you tie any modeled screen to specific HTML-based forms as desired.

The power of this solution lies in the leveraging of auto-generated content from the host application with the ability to use pinpoint directions to overrule auto-generation. (For example, you might want to create a custom form or specifically invoke a custom drop-down box in a single screen, without having to manually tailor the rest of the application.) This ability, combined with the use of standard CSS, divs, and HTML forms as the technology basis for layout, imparts the freedom and power of the web to build and extend your legacy applications.

You end up with a continuously extensible solution that allows macro and micro control over presentation. In addition, you have the ability to build a legacy-application UI consistent with current web 2.0 workflow and behavior expectations.

Fitting the Solution to the Need

Any IT department considering web enablement should become familiar with the available technology approaches, to be sure you are not over- or under-correcting your actual problem. When you realistically

assess your present and future business demands, the need for SOA might become obvious to you. Or, that standalone modernization project might emerge as the only one you really have to handle for the foreseeable future. As many businesses are discovering, the need usually lies somewhere between those two scenarios. What is less ambiguous is the very real need for flexibility in any chosen solution.

With the rapid advance of web application technologies, it's easier than ever to build a full-featured modern UI that can help you web enable your legacy applications. So don't rule out rejuvenation if that approach solves your problem most effectively. It's a jump-start modernization option offering compelling benefits – notably, less risk and less cost – that are not found with other methods.

If your needs are less well-defined, remember that Verastream Host Integrator is designed to take the guesswork out of your decision making; it can handle the spectrum of demands, from on-the-fly rejuvenation to full-blown SOA. And with Verastream, the incremental steps you take now to solve specific problems can constantly and reliably be reused to build additional solutions down the road.

About Attachmate

Attachmate delivers advanced software for terminal emulation, legacy modernization, and managed file transfer. Our NetIQ business provides solutions for automating IT processes and managing performance, security, and compliance of distributed IT. With our technologies, more than 65,000 businesses worldwide are putting their IT assets to work in new and meaningful ways.
www.attachmate.com

Attachmate

Corporate Headquarters
1500 Dexter Avenue North
Seattle, Washington 98109
TEL 206 217 7500
800 872 2829
FAX 206 217 7515

EMEA Headquarters
The Netherlands
TEL +31 172 50 55 55
FAX +31 172 50 55 51

Asia Pacific Headquarters
Australia
TEL +61 3 9825 2300
FAX +61 3 9825 2399

Latin America Headquarters
Mexico
TEL +52 55 9178 4970
FAX +52 55 5540 4886

WEB attachmate.com
E-MAIL info@attachmate.com

For regional office information, visit **www.attachmate.com**.