

QUICK TAKE



January 28, 2005

NEON Systems Scores Big By Acquiring ClientSoft

NEON Has The Technology To Beat In The Services From Mainframe Market

This is the second document in the "Web-Service-To-Host Consolidation In 2005" series.

by **Phil Murphy**

with John Rymer and Mike Gilpin

EXECUTIVE SUMMARY

NEON Systems' acquisition of ClientSoft is the most significant announcement of the year in the mainframe connectivity market; its union of complementary products will reset the competitive landscape in 2005. The new product — ShadowRTE — combines features from three vendors. NEON's traditional mainframe speed combines with the deep CA-IDMS and dynamic introspection features in InnerAccess Technologies' z/Services. ClientSoft's Service Builder gateway provides an off-mainframe runtime option, an Eclipse-based Studio for Java, and a Microsoft Visual Studio plug-in for .NET. The resulting product — already in an early beta release — is unparalleled in the industry today. Make room at the top of your shortlist for NEON.

NEON'S FUTURE IS PROMISING

NEON, InnerAccess, and ClientSoft were head-to-head competitors on many deals in the Web-to-host and Web-service-to-host space. As such, each studied its competitors closely, looking for chinks in the armor of its foes, and each found undeniable promise in its competitors' technologies and made the most of whatever faults it could find. Highly competitive deals eventually came down to one tactic — beat the other companies on prices, leaving a smaller and smaller profit margin for all of them. It comes as little surprise, then, that the companies started to entertain merger and acquisition thoughts.

Web-to-host software is a valuable tool that can quickly extend legacy transactions to the Web, but it had often been dismissed as proprietary screen-scraping and as not fast enough to handle the demands of very high volume. Taking those criticisms in order:

- **The screen scraper critics were once right, but the technology has changed.** Web-to-host technology today is far more capable than early screen scrapers that offer IT shops the opportunity to extend legacy transactions to remote users via the Internet in a variety of ways (emulation, interface re-engineering, and screen components) to suit a variety of needs.¹
- **Web-service-to-host adopts WSDL and SOAP standards, sheds the proprietary shackles.** One tenet of service orientation is that a service can run on any platform; WSDL and SOAP are the standards that make services reachable, no matter where they reside. Several of the Web-to-host vendors have changed their products to wrap transactions via WSDL and expose them via SOAP. ClientSoft — by virtue of its acquisition of Tanit and InnerAccess Technologies — had premium products in this space. Web-service-to-host takes the proprietary argument away.²

- **NEON makes it all faster — how fast remains to be seen.** NEON has moved the speed boundary to an as-yet-untested place; only time and reference customers can say for sure where the new boundary is. However, customers that need both speed and services built from legacy transaction have a new and excellent choice.

With better and faster available technology, NEON competes favorably in the Web-service-to-host arena based on speed, and it competes favorably in the EAI arena based on utility and alignment to service-oriented architecture (SOA). The key remaining questions are: What is the gap between this and a highly architected solution? Do you — or when do you — need a highly architected solution? And how many of your applications need the higher performance?

WHERE DO SPEED AND RELIABILITY COME FROM?

NEON's ShadowRTE was written in such a fashion as to run within a single address space using cross-memory services instead of CICS for the Web/CICS Transaction Server, the 3270 Bridge, and all other CICS-based code paths. This approach eschews supervisor calls (SVCs) in favor of directly invoking low-level services — all in the interest of speed. Tasks that don't leave the address space, that don't travel the longer paths, and that don't wait for supervisor interrupts take significantly less time to complete their work.

As for reliability, NEON's development paradigm requires that instrumentation and testing for every function be designed and built before any functional code is written. This self-discovering, ready-for-testing approach makes the end product much leaner, tighter, faster, more reliable, and simpler to fix.

WHAT COULD GO WRONG?

Acquisitions can be extremely difficult matters, taking several months to perhaps years to fully resolve themselves. Company culture, product mismatch, and other well-known issues can cause problems. Other potential issues include:

- **Culture is king, but there can't be three kings.** Management pays far too little attention to cultural differences until they have already done their damage. Seemingly insignificant issues can be blown out of proportion. For example, mainframe-centric technical staff, especially the deep, deep Assembler talent that are the masterminds of NEON's mainframe technology, are a green-screen-centric breed. Some may fight the technology that is widely accepted in the other firms, such as desktops, email, instant messaging, and calendaring.
- **Managing development across three geographic locations.** NEON will have staff in three very different locations: Toronto (InnerAccess), Miami (ClientSoft), and Houston (NEON). Whether and how they can they collaborate across the geography that divides them must be addressed.

Centralizing staff isn't likely, but some moves will be inevitable. Forced relocations may cost NEON some employees; then again, a move from Toronto to a warmer climate may retain an employee. The point is this: Expect some turnover — it is only natural.

- **NEON's existing base of clients expect speed.** Will Web-facing transactions be fast enough? To some degree, this argument is a nonstarter. ShadowRTE will likely be faster than all but the highly architected solutions. However, NEON's traditional customer base may not like the decreased transaction speed.
- **ShadowRTE must be priced competitively.** NEON's traditional products were expensive. Can NEON craft a price point and pricing model that suits its new customer base? MIPS-based pricing doesn't make any sense in the competitive Web-service-to-host market, concurrent user pricing doesn't make sense in NEON's traditional space, and metering by the number of transactions has never been well-accepted.
- **NEON must mitigate concerns of company size.** Is NEON big enough to compete with IBM? In a word, yes. NEON has been successfully competing for years. How and whether that will change remains to be seen. NEON's comparatively small size — a net loss of \$98,000 for fiscal 2004 on \$15.4 in revenues, which can be attributed in part to restructuring — may lose it some deals in which company size matters more than technology and the promise of future success.

WHAT IS THE LIKELY OUTCOME?

NEON stands poised to be the de facto choice for companies looking to extend the life of their legacy applications, especially if they want to avoid an all-IBM situation. ShadowRTE lets clients use their choice of IDE to extend legacy transactions as services — delivering business value today, while leaving a migrate-while-you-operate option open for tomorrow. NEON's approach holds the promise of significant breakthroughs in speed and ease of use that neither Web-service-to-host competitors nor pure enterprise application integration (EAI) vendors can match. If all of this works out, customers can avoid waiting for more architecturally elegant solutions and can still get the speed improvements over traditional Web-service-to-host.

Poor management execution is just about the only worry that NEON has left, but, given the speed at which these deals have come together, the choice of acquisition targets, and the availability of a beta version of Shadow RTE so soon after the acquisition, it is apparent that NEON's management is doing many things right.

RECOMMENDATIONS

TAKE STOCK OF YOUR APPLICATIONS

NEON's combined product offers a standards-based method to wrap legacy transactions as services with unprecedented speed and reliability. The question for customers is which applications need it and which should be done first.

- **Understand NEON's offering and how migrate-while-you-operate may change your plans.** NEON's offering changes the competitive landscape. Also, the layer of abstraction afforded by a service-oriented interface will permit a more staged and orderly back-end migration after the business needs are met. Fully understanding these two points sets the stage for analytical action.
- **Which projects in the queue would benefit?** Take a look at your new project queue. Does the availability of this type of capability change any existing project plans? Does the increase in speed and reliability mean that an existing plan for a highly architected solution can happen soon?
- **What about existing systems — what opportunities exist?** Many companies decline to tackle certain projects that could provide high value to customers, but a highly architected solution is cost-prohibitive. This offering brings higher speed at lower cost — reviving the potential of those opportunities.
- **Consider a gradual migration toward service orientation using tools like this.** With the tool already purchased to extend legacy transactions to the Web, consider the opportunities to simplify and consolidate legacy applications, eliminating redundant code in favor of a single service to perform functions — this promotes a gradual migration to service orientation and, in so doing, should lessen the legacy application maintenance burden.

ENDNOTES

- ¹ Web-to-host technology continues to succeed because it fills the need for “right now” Web-enabling of what are otherwise perfectly suitable applications, extending them to new groups of users at compelling speed-to-market and price/performance points. See the March 7, 2002, Planning Assumption “Market Overview: Web-To-Host Tools — A Crowded Market, But Compelling Technology.”
- ² Web-service-to-host inserts a layer of abstraction by wrapping transactions as services, permitting a migrate-while-you-operate approach to application modernization and migration. See the March 21, 2003, Planning Assumption “Web-Service-To-Host Modernizes Legacy Application Portfolios With SOAP, WSDL, And New Migration Options.”