PEREGRINE CONNECT

Saving Time and Money by Migrating Off of Microsoft BizTalk Server to Peregrine Connect's Neuron ESB

Peregrine Connect's Application Integration and Web Service Platform:
A Real-World Client Example of Valuable Time and Cost Savings

PEREGRINE CONNECT

Table of Contents

Why common application integration and web service composition approaches fail

Information technology (IT) departments in many organizations today are struggling with how to effectively and efficiently integrate applications for delivering transformational systems that drive business innovation. Even simple application integration changes that should take only a few hours or days to implement may take weeks or months due to overly complex, resource-intensive, outdated and custom-coded, point-to-point solutions or antiquated middleware solutions. An in-depth analysis of a real-world pilot performed for a client in the education industry conclusively demonstrates the overwhelming benefits of Peregrine Connect's Neuron Enterprise Service Bus (ESB)-based application integration solution for web service composition over an existing Microsoft BizTalk Server implementation.

The reliance on homegrown, point-to-point integration solutions engages an organization's IT in a perpetual cycle of redundant and excessive efforts expended to develop and sustain web service application interfaces. As an organization grows and evolves in striving to successfully compete in the rapidly changing and increasingly service-intensive marketplace, it will reach a breaking point in trying to operate based on such ineffective integration solutions. Trying to afford the time and cost, including the sophisticated and excessive developer resources, required to implement, maintain and support these connections will become highly impractical over time.

This white paper examines some of the most burning business and technical application integration challenges facing IT departments today and how Peregrine Connect middleware can effectively and efficiently help to address those challenges. Peregrine Connect's Neuron ESB is an application integration and web service platform for organizations of all sizes and industries committed to using the Microsoft .NET platform. Built entirely on Microsoft .NET Framework technology, Peregrine Connect facilitates an organization's ability to efficiently integrate, adapt, extend and scale .NET-based web service applications for overall improved operational performance, agility and cost savings. Providing both real-time and long-running business process orchestration and workflow capabilities, Peregrine Connect drives business innovation that is resilient to disruption.

A pilot performed for a client in the education industry serves as the basis for demonstrating the overall business value and key benefits of a Neuron ESB-based integration solution implementation. The pilot performed for the client specifically compares the Peregrine Connect Neuron ESB implementation to the organization's existing Microsoft BizTalk Server and custom .NET C# Windows Communication Foundation (WCF) implementations.

The conclusive benefits of the Neuron ESB Implementation of the client's service composition solution include:

- Reduced software licensing costs by approximately 12 times
- Reduced annual support costs by 20 times
- Improved overall performance of web service composition solution
- Streamlined solution implementation and maintenance
- Reduced solution implementation costs

Common integration challenges encountered today

An effective and efficient application integration solution strategy for web service composition is considered a cornerstone in successfully enabling technology innovation and business growth for organizations across many industries seeking to successfully compete into the future. Providing a 360-degree view of the various operational processes and supporting information systems that serve an organization and its external customers and business partners is therefore a primary business challenge for IT.

Another pressing business challenge concerns the need to enable automated, event-based operations with near real-time visibility of systems rather than relying on delayed batch processing between systems. Successfully enabling an automated, event-based business operation hinges on effective and efficient master data management in which a master system of record in conjunction with other related information systems are capable of reflecting any changes made in near real-time. An automated, event-based business operation effectively allows an organization to:

- Improve responsiveness for taking advantage of new business opportunities and addressing continually changing business needs and challenges in maintaining a competitive advantage.
- Increase operational efficiency and productivity to in turn realize valuable time and cost savings.

From a technical standpoint, the key challenges facing IT departments for adopting web services, application integration and business process automation platforms revolve around the need to leverage plentiful and more cost-effective, standard .NET developer resources (see figure 1). IT organizations struggle with trying to both find and afford the sophisticated and expensive, integration middleware product specialists with the skills required to write very complex integration solutions.

At the same time, organizations are seeking to minimize the number of developer resources required overall. Other developer resource-related challenges concern the:

- Acquisition of skilled resources for designing complex infrastructure capabilities, requiring plumbing and infrastructure; advanced coding; complex patterns; and monitoring, reporting and instrumentation
- Additional costs associated with acquiring skilled resources
- Development effort extended for months or years, increasing the time before a solution can be brought to market
- Higher costs of maintaining and extending infrastructure capabilities over the long term
- Inefficiencies in integrating, maintaining, adapting and scaling complex or custom-coded, point-to-point integration solutions
- Loss of technical knowledge capital and expertise with the developers who originally wrote the custom-coded integration solutions

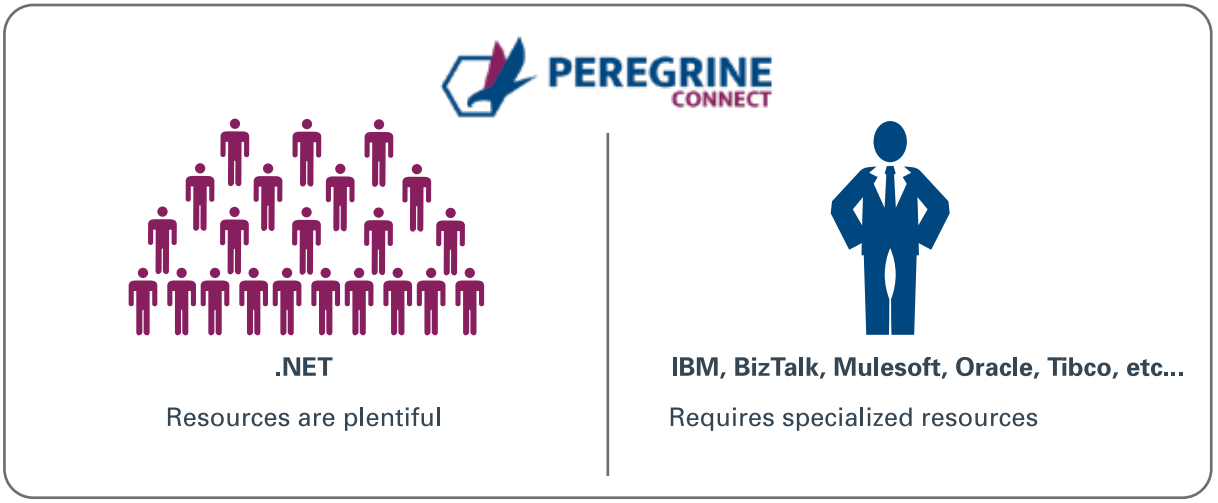

Figure 1: Peregrine Connect's implementations require minimal training

Gartner

"Planners and product managers that need an ESB to provide their deliverables should consider Neuron ESB because of its demonstrated ease of deployment and ease of use, as well as its ability to be used successfully by a broad set of resources, such as .NET developers, rather than more highly skilled (and more costly) Java developers."

—Jess Thompson, Research Vice President, Gartner

Beyond developer resource challenges, IT departments struggle with how to improve efficiency and productivity for handling the various application integration scenarios and types of programming interfaces. Developers also find it difficult to maintain their focus on addressing actual business needs and challenges due to valuable time and effort spent having to custom build integrated operating systems with point-to-point solutions. In other cases, IT may realize that it purchased the wrong product for efficiently scaling and supporting an organization's web service operations to accommodate growth and changing business needs.

Considering the complex mix of business and technical challenges associated with web service composition and application integration, IT departments seek a tool that can effectively align with business objectives while enabling an efficient and productive developer experience at the same time. Neuron ESB was therefore chosen for the pilot due to its focus on agility, ease of use and operational efficiency for use with the Microsoft platform.

Peregrine Connect's Neuron ESB provides businesses with real-time reliable messaging, business process orchestration, and application and web service integration options for truly distributed ESB solutions deployed in the cloud, on premise or in various hybrid topologies. Using Neuron ESB offers a secure, cost-effective and productive way of exposing and extending enterprise systems and services into new composable business capabilities.

A real-world client comparison of Peregrine Connect's Neuron ESB vs. Microsoft BizTalk Server

A real-world pilot and training performed for a client in the education industry overwhelmingly demonstrates the benefits of a Neuron ESB-based solution implementation for web service composition. The pilot objectives for comparing the Peregrine Connect implementation to the client organization's existing Microsoft BizTalk Server and custom .NET C# WCF implementations were to:

- Design a Neuron ESB-based solution that replicates the functionality of the existing Microsoft BizTalk Server-based solution
- Design a Neuron ESB-based solution that also replicates the custom .NET C# WCF-based solution the client previously built in an attempt to replace the existing Microsoft BizTalk Server-based solution
- Develop the Neuron ESB-based solution to include:
  - –A hierarchical, topic-based messaging topology for web service routing
  - –The organization's existing WCF-based web services
  - –Peregrine Connect-hosted web services
  - –Dynamic routing capabilities
  - –Scatter-Gather pattern for service composition
  - –Exception management framework
  - –Neuron ESB environmental variables
- Demonstrate XCOPY deployment in an ASP.NET web application framework from one isolated environment (quality assurance) to another (production)
- Conduct real-world performance testing, comparing the Neuron ESB-based solution against both the existing Microsoft BizTalk Server-based and custom .NET C# WCF-based solutions
- Provide advanced training for the organization's IT team regarding the use of Neuron ESB to develop complex business solutions

To accomplish these objectives, the organization's "retrieve college process" web service operation was migrated onto Peregrine Connect's hosting environment to effectively compare and contrast the solutions in terms of overall performance, maintainability and complexity. The retrieve college process web service was originally hosted in Microsoft BizTalk Server and then subsequently migrated to a custom .NET C# WCF-based solution due to performance issues. This particular web service operation was chosen for the pilot because it represented a typical web service built on the client organization's existing Microsoft BizTalk Server's infrastructure.

The retrieve college process web service operation essentially provides a consolidated view of an online or ground school through the consumption of six backend web services. When an information request is submitted containing a school identifier, the service returns basic school, degree, contact list, office hours and online community information. The existing version of the service implements the Scatter-Gather pattern in Microsoft BizTalk Server to dispatch information requests to the multiple backend services and then sends an aggregated response back to the client system users.

The primary objective of the pilot was to demonstrate improved performance of the web service solution using Neuron ESB in terms of request-response time versus the existing Microsoft BizTalk Server implementation while closely matching the performance of the custom .NET C# WCF implementation. The secondary objectives were to use Neuron ESB to provide a much simpler, easier to use, maintain, and more cost-effective application integration and web service solution. At the conclusion of the pilot, Neuron ESB successfully achieved both the primary and secondary objectives as evidenced by actual supporting metrics (see following section for metrics).

8.5 X Faster

The Peregrine Connect's Neuron ESB-based solution was approximately 8.5 times faster than the Microsoft BizTalk Server-based solution and nearly 3.5 times faster than the custom .NET C# WCF-based solution.

Improved overall performance of web service composition solution

Peregrine Connect's Neuron ESB-based solution implementation of the retrieve college process web service significantly outperformed both of the client's existing Microsoft BizTalk Server-based and custom .NET C# WCF-based solutions in terms of average response times (see figure 2).

Solution	Ground School
Peregrine Connect ESB-based solution	350 ms
Existing custom .NET C# WCF-based solution	1200 ms
Existing Microsoft BizTalk Server-based solution	3000 ms

Figure 2: Average response times

Neuron ESB's architecture is designed for more effective request-response, low-latency and high throughput scenarios. To summarize, the overall key reasons why the Neuron ESB-based integration solution performed better in comparison to the client organization's existing Microsoft BizTalk Server-based and custom .NET C# WCF-based solutions are:

Microsoft® BizTalk® Server

- Microsoft BizTalk Server routes all messages through its proprietary MessageBox database, creating undesirable latency in request-response messaging scenarios. The architecture also imposes severe limits on concurrency scale-out.
- Web service capability within Microsoft BizTalk Server is enabled through the use of its adapters. These adapters, however, do not submit the received requests directly to the MessageBox. The adapters instead use proxy messages, which are routed through Microsoft Internet Information Services (IIS) before they are submitted to the MessageBox database.
- Microsoft BizTalk Server does not provide a native hosting environment for web services. All web service calls must be proxied through IIS and COM interop.
- Microsoft BizTalk Server Orchestrations are stateful by design. Therefore at various points or shapes within the execution of a Microsoft BizTalk Server Orchestration, the current state of variables and messages are stored to the MessageBox database, causing more overhead.
- WCF-based web services that use data contracts generally require deserialization and serialization within the service, contributing to longer response times. However, note that additional analysis of WCF services is needed to confirm whether serialization alone is the reason for longer response times.

PEREGRINE CONNECT

- An optional stateless business processing environment, which reduces latency and increases concurrency through asynchronous configuration options.
- The elimination of an intermediate durable MessageBox-like database in the middle of all message interaction, which reduces latency even further while increasing concurrency.
- Direct interaction with existing web service and application integration adapters, which also helps reduce latency and increase concurrency.
- A native hosting environment for web services.
- Leveraging the asynchronous features within a Neuron ESB business process, coupled with the native Neuron ESB infrastructure services, the significant reduction in the request-response time was achieved over both the Microsoft BizTalk Server-based and custom .NET C# WCF-based solutions. Neuron ESB was effectively able to asynchronously call each of the six web services in parallel rather than in a sequential manner.

Streamlined solution implementation, adoption and maintenance

The time that it took to implement the Peregrine Connect's Neuron ESB-based integration solution for the client's retrieve college process web service solution took less than three days. The strategy applied for effectively streamlining the implementation time of the Neuron ESB-based solution leveraged the use of any existing artifacts from the client organization's Microsoft BizTalk Server-based solution while helping to minimize the need for custom coding.

To facilitate the streamlined Peregrine Connect implementation, Extensible Stylesheet Language Transformations (XSLTs) in the existing Microsoft BizTalk Server solution were used with some modifications. In specific, the modifications involved the consolidation of three XSLTs into one and required less than a few hours of effort to complete.

The existing Microsoft BizTalk Server-based integration solution implementation for the retrieve college process web service is an example of a Scatter-Gather pattern. A single web service call into Microsoft BizTalk Server invokes six subsequent calls to backend services, and the responses are then aggregated into a single response and returned to the client. This pattern was easily implemented as a Neuron ESB business process as shown earlier (see figure 4).

If a member from the client's IT development team who was more familiar with the XSLTs had completed the consolidation effort, even more implementation time savings may have potentially been realized. Additionally, to help minimize the need for any custom code development for the Neuron ESB implementation, the existing Scatter-Gather pattern sample provided within the Neuron ESB software development kit and documentation was used.

The client's existing Microsoft BizTalk Server-based integration solution implementation consisted of more than 100 orchestration shapes for choreographing the interaction with the six backend web services (see figure 3). The simplicity of the Peregrine Connect's Neuron ESB-based integration solution implementation was enabled through the creation of a Neuron ESB Business Process consisting of just 20 steps (see figure 4).

For the purpose of this comparison, the Neuron ESB Business Process Steps are virtually equivalent to Microsoft BizTalk Server Orchestration Shapes.

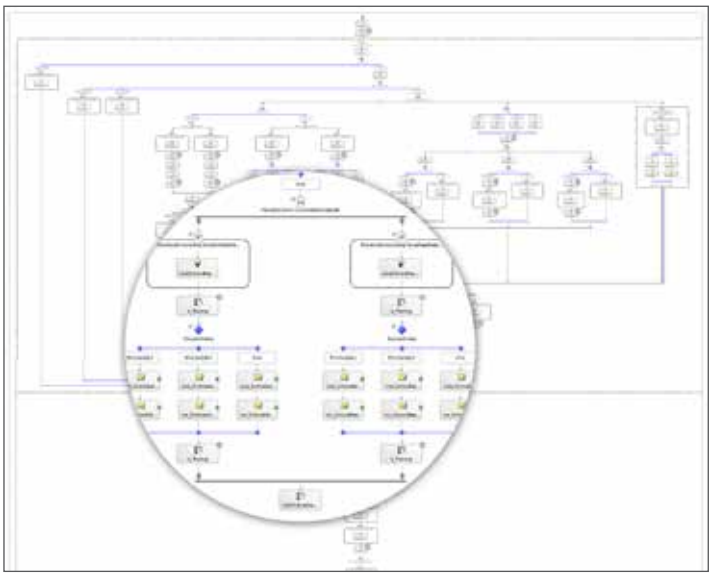

Figure 3: Existing Microsoft BizTalk Server-based integration solution (consisting of more than 100 BizTalk Orchestration Shapes)

Figure 4: Peregrine Connect's Streamlined Neuron ESB-based integration solution (consisting of approximately 20 Neuron ESB Process Steps)

In effect, Peregrine Connect's highly streamlined Neuron ESB-based integration solution demonstrated a dramatic, 80-plus percent reduction in the number of configuration steps required.

The Peregrine Connect pilot solution implementation demonstrated a dramatic reduction in the client's total costs—from more than $1.5 million to less than $100,000.

Peregrine Connect only requires the use of standard .NET resources and, combined with its ease of implementation and use, provides for dramatically reduced costs compared to other middleware products in the marketplace. For every dollar of Peregrine Connect purchased, only $1 or less is sold in implementation consulting services. The majority of Peregrine Connect customers do not outsource implementation services. Instead, they perform the implementation themselves with existing Microsoft .NET development resources (see Associated Press case study as an example). In contrast, for every dollar spent on most middleware products in the marketplace, an amount ranging from approximately $5 to $10 is commonly spent on implementation services.

Specific to the pilot performed for the education industry client, the Peregrine Connect's Neuron ESB-based integration solution offered dramatic savings in software and recurring maintenance costs in comparison to the client organization's existing Microsoft BizTalk Server-based and custom .NET C# WCF-based solutions (see figure 5).

Existing Microsoft BizTalk Server-based solution cost in production environment
(for software and recurring maintenance)

8 Microsoft BizTalk Servers + 2 SQL Servers		
Production	= 32 CPUs of Microsoft BizTalk Server	= $1.440MM
	= 8 CPUs of Microsoft SQL Server	= $50K
Annual Support		= $500K

Pilot Neuron ESB-based solution cost
(for software and recurring maintenance)

1 Neuron ESB Server + 1 SQL Server		
Pilot	= 2 CPUs of Neuron ESB	= $50K
	= 2 CPUs of Microsoft SQL Server	= $16K
Annual Support		= $15K

Estimated Neuron ESB-based solution cost in production environment
(for software and recurring maintenance)

1 Neuron ESB Server + 1 SQL Server		
Production	= 4 CPUs of Neuron ESB	= $100K
	= 2 CPUs of Microsoft SQL Server	= $16K
Annual Support		= $25K

Solution implementation costs
Microsoft BizTalk Server = 2 months of development and performance tuning = $64K
Neuron ESB = 3 days of development and performance tuning = $4.8K

Figure 5: Solution implementation and ownership cost savings

Reduce software licensing costs by approximately 12 times

Reduced annual support costs by 20 times

Overall in terms of product, services and operations, Peregrine Connect's Neuron ESB typically costs several factors less than Microsoft BizTalk Server, IBM, Oracle, Tibco and WebMethods (see figure 6). In the case of web service composition solutions, the cost savings obtained using Neuron ESB can be even greater. For this specific customer solution, the savings would be significantly greater:

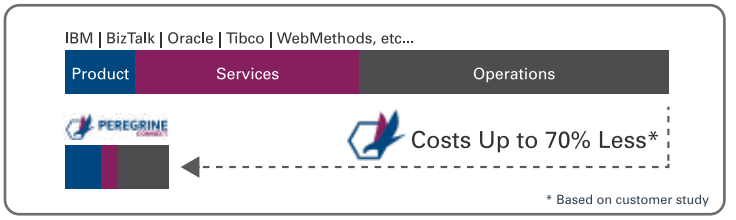

Figure 6: Peregrine Connect's Neuron ESB overall costs up to 7x less than the competition

Peregrine Connect's Neuron ESB-based solution implementation key features

In performing the pilot for the client organization, several Peregrine Connect's Neuron ESB features were utilized. A high-level description of the key features, including how they were utilized in the solution implementation, is provided in terms of:

- Topic hierarchy
- Business processes
- Exception management
- Deployment architecture

Topic hierarchy

Peregrine Connect and Neuron ESB provide a powerful hierarchical, topic-based publish and subscribe messaging system in which publishers can label each message with a topic name rather than addressing it to specific recipients. The Neuron ESB messaging system can alternatively route messages between parties (publishers and subscribers), adapters (technology "bridges") and service endpoints.

Neuron's topic model is composed of a subtopic hierarchy that can more intuitively reflect either an organization's structure or business requirements. However, Neuron ESB is unique in the industry in that it allows the business to determine the quality of service (QoS) attributes at the topic level (configurable within the Neuron ESB Explorer), and many topics of various QoS attributes can exist side by side. By providing this level of flexibility, organizations do not have to worry about changing their specific business requirements to meet the limitations imposed by other competing products.

Some of the critical QoS attributes include:

- Throttling
- Encryption
- Auditing
- Transport
- Transactions
- Durability
- Compression

Transport is a critical QoS selection, as it affects many aspects that businesses may require, including transaction and durability support. Transport selection also provides attributes like ordered messaging, guaranteed delivery, once-only delivery, scale out, latency and performance.

The lack of this level of flexibility within Microsoft BizTalk Server is one of the primary reasons that it falls short in web service-based scenarios. Every request processed and routed by Microsoft BizTalk Server MUST go through a database within a two-phased commit transaction, even if it is a web service request-response type of messaging pattern.

In contrast, Neuron ESB allows users to flexibly configure its messaging fabric specific to the business need.

In the case of a web service request-response scenario, Peregrine Connect's Neuron ESB can be configured to use in-memory routing. Neuron ESB provides several configurable transports for topics, including:

- TCP
- MSMQ
- Peer
- RabbitMQ
- Named pipes

The Neuron ESB messaging system manages the process of sending a message to all eligible recipients that have expressed interest in receiving messages on the assigned topic. This form of asynchronous messaging solution architecture is far more scalable than point-to-point messaging alternatives. Once the message sender or publisher creates the original message, they can then leave the task of servicing the recipients or subscribers to the messaging infrastructure.

Topic-based publish and subscribe messaging systems generally share several common attributes, which may include:

- Subscribers subscribe to one or more topics and only receive messages that are of interest.
- Publishers have no knowledge about the subscribers, including how many there are or where they live.
- Subscribers have no knowledge about the publishers, including how many there are or where they live.
- New systems, including publishers or subscribers, are either easily added or removed from the flow of information without the need for coding changes.

This type of messaging architecture sends messages only to those applications interested in receiving them without any knowledge of the identities of the receivers. What's unique about Neuron ESB's Topic implementation is that the underlying network transport and quality of service attributes can be individually configured at the topic level.

An initial step in performing the pilot was to define a topic hierarchy that resembled the client organization's existing web service composition solution. The client had approximately 50 web services hosted in either its Microsoft BizTalk Server-based solution or its custom .NET C# WCF-based solution.

Rather than reviewing all the different services, the following representational hierarchy was implemented based on discussions with the client:

- Actor
- Actor.School
- Actor.School.Ground
- Actor.School.Ground.GeneralInformation
- Actor.School.Ground.Newsfeeds
- Actor.School.Ground.Config
- Actor.School.Ground.GetListItems
- Actor.School.Online
- Actor.School.Online.GeneralInformation
- Actor.School.Online.Newsfeeds
- Actor.School.Online.Config
- Actor.School.Online.GetListItems
- Actor.Faculty
- Actor.Student
- Actor.Staff
- Partners
- Technical
- Technical.Authentication
- Technical.Authorization

The Actor.School.Ground.* and Actor.School.Online.* sets of topics were effectively used in implementing the retrieve college process web service with the Peregrine Connect's Neuron ESB-based integration solution.

Business processes

Peregrine Connect offers a flexible, easy-to-use, Business Process Designer with a drag-and-drop interface that ships with 44 configurable process steps that do virtually everything—from calling a service, supporting parallel processing, and updating a database or queue to detecting duplicate messages and parsing a Microsoft Excel file within Peregrine Connect's Neuron ESB Explorer (see figure 7). Developers can build custom, reusable process steps that can be registered in the Process Steps library and added to any custom business process.

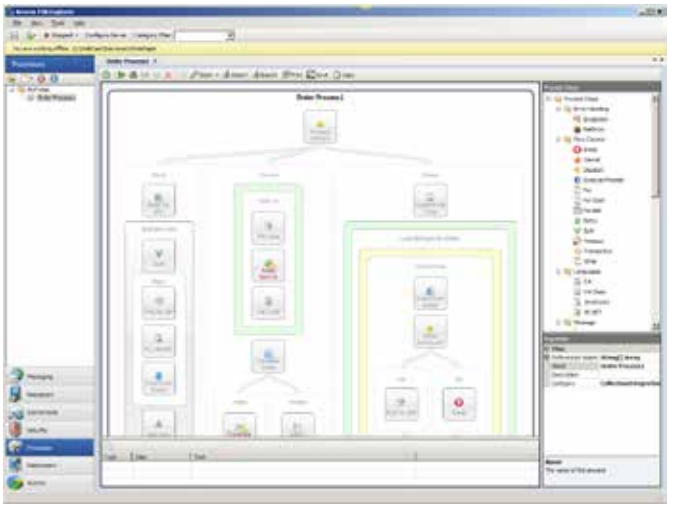

Figure 7: Peregrine Connect's Neuron ESB Business Process Designer

The Neuron ESB Business Process Designer ships with many .NET developer-centric user enhancements designed to improve developer productivity. Developers can perform real-time debugging at design time, so they can quickly test, diagnose and fix business processes during development. The business process debugging experience within Neuron ESB is similar to the experience provided by Microsoft Visual Studio for debugging .NET applications (see figure 8).

Additionally, the Neuron ESB Business Process Designer offers powerful extensibility features. Developers can write custom process steps, configurable through the Neuron ESB Explorer that allow other developers to reuse their capabilities. Neuron ESB also ships with programming language editors exposed through process steps, allowing users to write either VB.NET, C#, or Javascript directly within a process—without having to use Microsoft Visual Studio or compiling anything into assemblies (see figure 9).

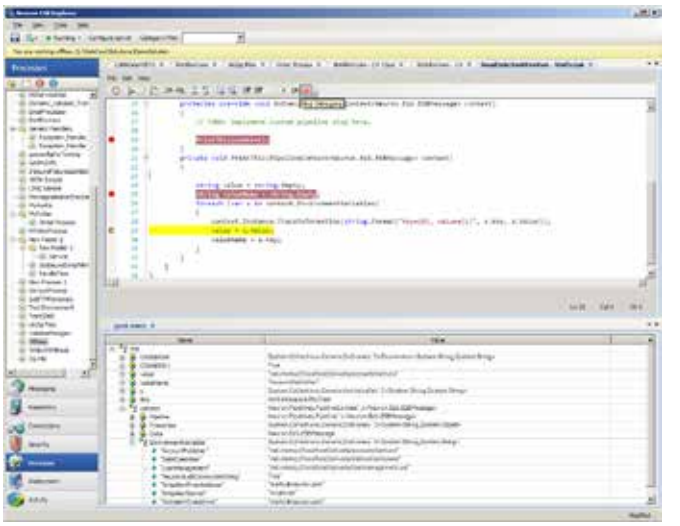

Figure 8: .NET debugging experience within Neuron ESB provides developers the ability to set breakpoints within processes and inspect messages, variables and state in real-time during development

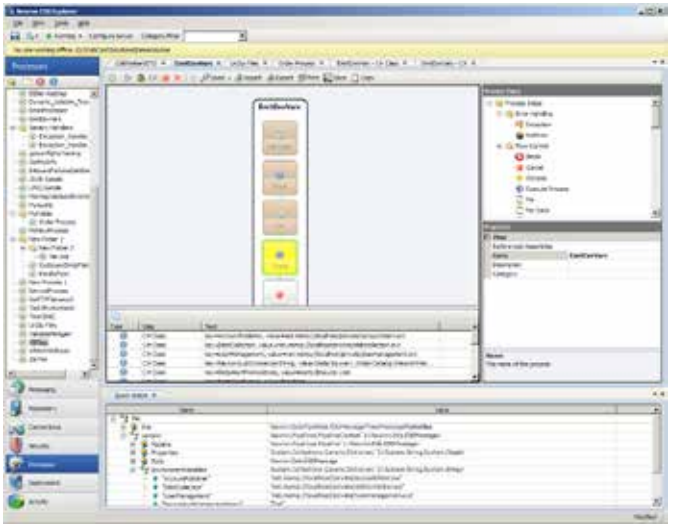

Figure 9: C# language editor with toolbar formatting options and full IntelliSense for the developer—nothing to ever compile

Using the Neuron ESB Business Process designer, developers can also:

- Set breakpoints on process steps and within language editors
- Use function keys F5 and F11
- Work using dock-able windows
- Disable process steps
- Create folders and categorize processes
- Perform cut/copy/paste process steps
- Zoom, print and save processes as images

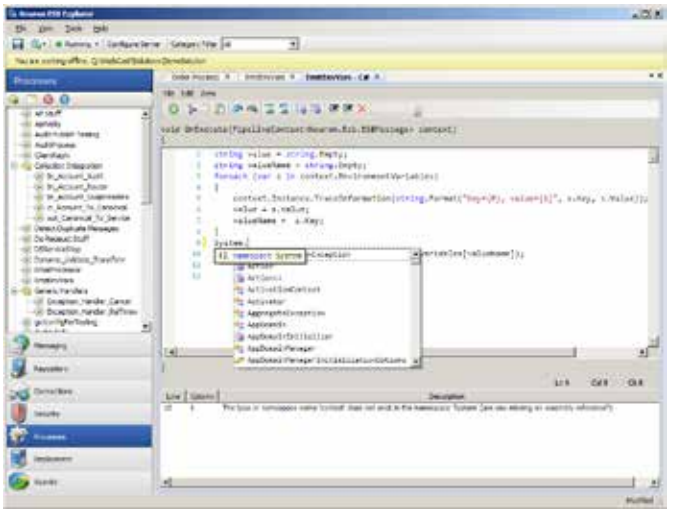

Figure 10: Peregrine Connect's Neuron ESB .NET language editor - Offering full Visual Studio experience for writing .NET code

Microsoft BizTalk Server does not ship with anything close to the capability, functionality, performance, extensibility and developer usability encapsulated within the Peregrine Connect's Neuron ESB Business Process engine and designer.

Neuron ESB-based solutions can be developed in a fraction of the time that it would take to build and scale out comparable solutions with Microsoft BizTalk Server while performing five times to 20 times faster.

Consequently, the Microsoft BizTalk Server Orchestration engine was designed for stateful, long-running processes, which focus on the execution of external .NET developer-provided assemblies. Because of this, it introduces significant overhead, usability and testing issues. Most capabilities have to be provided by the .NET developer since they are not provided by Microsoft BizTalk Server Orchestration.

Although not used as part of this pilot, Peregrine Connect and Neuron ESB also offer workflow and orchestration capabilities, supporting a user-friendly graphical design time environment, event tracking, long-running transactions, custom correlation, fault tolerance and distributed runtime environments (see figures 11 and 12).

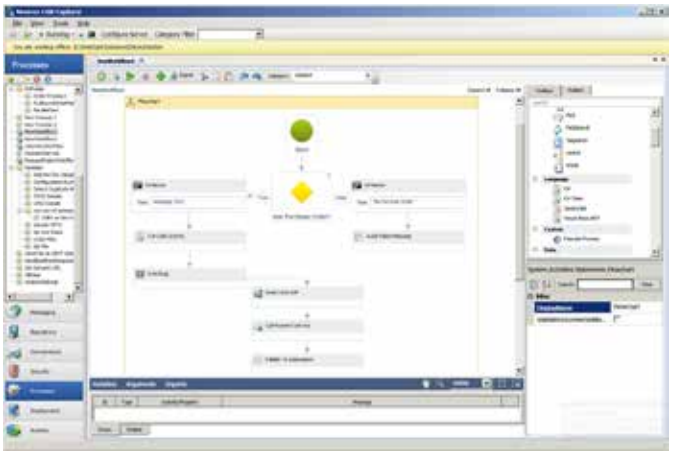

Figure 11: Neuron ESB Workflow Designer for creating long-running, persistent, fault-tolerant business processes

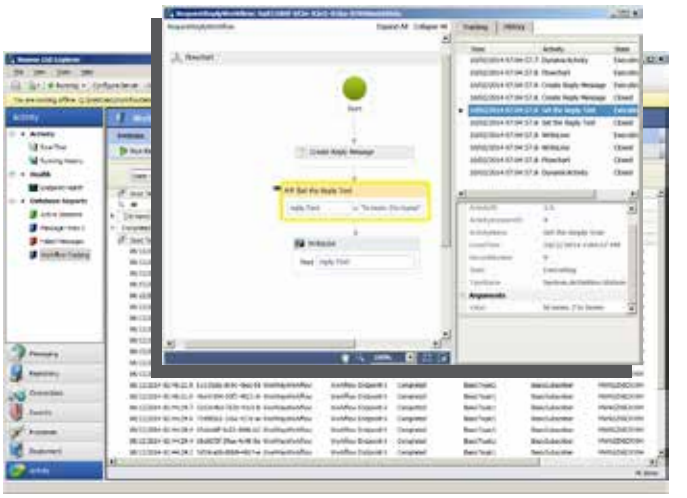

Figure 12: Graphical event tracking during Neuron ESB workflows allows users to see entire process flow state and health, including values assigned to all variables

Peregrine Connect's Neuron ESB business processes are ideal for basic ESB patterns, such as Validate, Enrich, Transform and Operate or VETO; Scatter-Gather or Simple/Dynamic Service Aggregation and Composition. The Neuron ESB Business Process engine capabilities are targeted for real-time web service environments, where performance, agility and time to market are driving factors. Whereas, Neuron ESB Workflow and Orchestration are targeted for those use cases where resiliency, tracking and fault tolerance are primary business drivers.

Exception management

Exceptions should always be managed within Peregrine Connect's Neuron ESB-based integration solution. An Exception Process Step that encapsulates all other steps is generally recommended for all processes. A catch block, an alternative block of code in C#, is therefore executed for exception management. Within the catch block of the Exception Process Step, one or more of several actions may be taken including, for instance:

- Use the Code Process Step to enrich the caught exception with business-specific and actionable information related to the cause of the error as well as the actual System.Exception object and then report the information to an external tracking system.
- Use the Audit Process Step, setting its action property to "failure," which will automatically write the failed message, its context and exception information to the Peregrine Connect database.

Peregrine Connect's Neuron ESB-based solution implementation of the retrieve college process web service composition follows this overall recommendation. The reusable exception management pattern can also be used to add features such as custom message management, custom error logging and email notifications.

A reusable and composable exception management pattern was not available within the existing Microsoft BizTalk Server-based solution.

Neuron ESB enables streamlined deployment by

Microsoft BizTalk Server does not support the concept of custom Orchestration extensions or activities.

Deployment architecture

encapsulating all elements of the integration solution into a simple configuration file. Deployment groups and environmental variables are used to assign the correct values for use at runtime per the deployment environment (see figure 13). There are no assemblies to manage or put into the global assembly cache. Complex database synchronization issues and the need for custom scripts and Microsoft Windows Installer or custom installer packages are therefore eliminated.

In effect, Peregrine Connect and Neuron ESB simplify what can otherwise become a very cumbersome deployment process for developers and administrators alike. Precious time may then be reallocated back to the development of the business solution, rather than to its deployment.

The education industry client organization utilizes four existing web service application environments for:

- Development
- Integration testing
- User acceptance testing
- Production

Facilitating the deployment of a Neuron ESB-based integration solution across these various environments would essentially require the creation of four deployment groups:

- Development
- Integration
- Regression
- Production

For each deployment group, a set of environment variables is defined to allow for environment-specific settings to be captured once and reused as the solution progresses from development through production. Examples of environment variables may include:

- Neuron ESB Audit Database Connection String
- URLs of service endpoints (or partial URLs, containing just the server name and port)

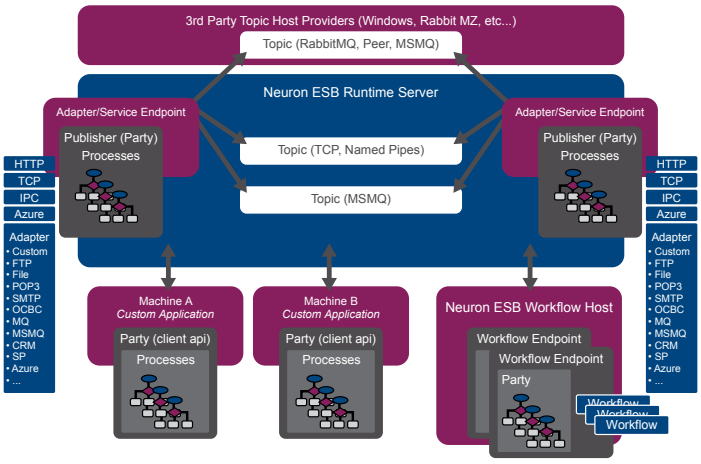

Figure 13: Peregrine Connect's Neuron ESB runtime architecture

Choosing an application integration and web service composition platform

The Neuron ESB application integration and web service platform effectively and efficiently extends the Microsoft .NET platform to provide real-time reliability, durable messaging, application integration, business processing and web service management. Expressly built with the .NET developer user experience in mind, Peregrine Connect and Neuron ESB offer a highly intuitive and easy-to-use user interface.

"We evaluated Peregrine Connect's Neuron ESB against other service buses, and Peregrine Connect came out as the clear leader. It wasn't a sledgehammer for a nail, as was the case with some of the other offerings we looked at. Peregrine Connect fit the bill perfectly and is now orchestrating all content processed by AP."

— Vince Tripodi, Vice President, Development, Associated Press

"We selected Peregrine Connect as the platform on which to build out our business transformation initiatives. Peregrine Connect had the capabilities and performance we needed to meet a number of our workflow and orchestration challenges. We can't get that from Microsoft BizTalk Server's orchestration engine."

— Mitch Bledsoe, Vice President, Information Technology, Assurant Specialty Property

"What requires hours with competitive alternatives, takes a matter of minutes with Peregrine Connect. We went from a months-long product release process to one that now takes mere weeks."

— Lester Henderson, Senior Architect, Compassion International

Key distinguishing features of Peregrine Connect

Peregrine Connect's Neuron ESB application integration and web service platform effectively and efficiently extends the Microsoft .NET platform to provide real-time reliability, durable messaging, application integration, business processing and web service management. Expressly built with the .NET developer user experience in mind, Peregrine Connect offers a highly intuitive and easy-to-use user interface.

- **Streamlined configuration and ease of use:** Drives events to and from other systems based on user-defined criteria and features an easy-to-use toolset and intuitive, hierarchical, topic-based publish and subscribe messaging system with configurable transport protocols, including MSMQ, RabbitMQ, TCP, named pipes and peer.
- **Codeless connections to Salesforce.com, Microsoft Dynamics CRM, NetSuite and more:** Enables more agile, effective integration for faster and simpler project deployment with codeless connections to ODBC, POP3, Microsoft Exchange, SQL, Microsoft SharePoint, SPTF/FTP/FTPS, IBM MQSeries, Apache ActiveMQ and more.
- **Distributed, hierarchical, topic-based message system:** Easily message-enables existing .NET applications using Peregrine Connect's Neuron ESB distributable, event-based client application programming interface (API). Peregrine Connect's client API allows existing .NET applications to scale and participate in an event-driven architecture.
- **Simplified service hosting and mediation:** Easily manages, extends, hosts and integrates all web service (SOAP/REST) endpoints in a single place while adding uniform security, policy, reporting, and management options to existing services.
- **Real-time business process design:** Quickly develops rules and decisions that execute in conjunction with business information in real-time for more than 40 out-of-the-box business activities, without the need for complex workflow tools, other technologies or specialized skills.
- **Step-by-step process debugging:** Allows process designers to quickly test, diagnose and fix business processes during development with similar design time debugging support as .NET apps using Microsoft Visual Studio.
- **Extendable business processes with language editors:** Expands business processes with C#, VB.NET or JavaScript within the Neuron ESB Business Process Designer without the complexity of managing or deploying .NET assemblies.
- **Workflow and orchestration:** Provides a user-friendly graphical design time environment, event tracking, long-running transactions, custom correlation, fault tolerance and distributed runtime environments targeted for those use cases where resiliency, tracking and fault tolerance are primary business drivers.

PEREGRINE CONNECT

About the Author

Marty Wasznicky, General Manager, Peregrine Connect

Marty has more than 24 years of experience in the software development industry. He joined Neudesic after serving for six years as a Regional Program Manager in the Connected Systems Division at Microsoft. His responsibilities there included building out the Microsoft BizTalk Server product integration business, managing a team of SOA/ESB/BPM field specialists, building strategic partner alliances, and collaborating on architecture and features for Microsoft distributed technologies. Marty also owned the development and architecture of the Microsoft Enterprise Service Bus Toolkit.

About Peregrine Connect's Neuron ESB

An enterprise service bus (ESB), Neuron ESB is an application integration and web service platform built entirely on Microsoft .NET. Neuron ESB-based integration solutions enable the delivery of real-time, event-driven communication with streamlined and cost-effective implementation, maintenance and support.
For more information and to download a free 30-day evaluation copy of Peregrine Connect,
www.peregrineconnect.com

If your organization wants to …

- Deploy application and web service integration solutions faster and more efficiently
- Increase productivity without increasing its developer base
- Realize considerable savings on deployment and associated software, hardware, maintenance, upgrade and support costs

… then it's time to explore how Peregrine Connect may benefit your organization.

To arrange a meeting with a Peregrine Connect representative, please contact Marty Wasznicky, at (310) 980-5495 or Marty.Wasznicky@peregrineconnect.com

PEREGRINE CONNECT

PEREGRINE CONNECT Management Suite

PEREGRINE CONNECT Design Studio

PEREGRINE CONNECT Runtime Platform

neuron esb

Peregrine Connect is an easy-to-use hybrid integration platform that simplifies integration design, deployment, and management

The Peregrine Advantage

- Adapt quickly - create and modify applications quickly and easily
- Connect with ease - reduce development efforts
- Save time and money - cut integration costs by up to 70%
- Scale for the future - quickly scale on-premises or in the cloud

peregrineconnect.com/biztalk-white-paper