

VENDOR NEEDS AND STRATEGIES

Borland Delivers Domain-Agnostic Solutions with Janeva

Rikki Kirzner

Jagi Shahani, Consultant

IDC OPINION

Borland continues to stay on top of its game by providing developers with solutions that bridge the Java and .NET worlds. With its introduction of Janeva and C#Builder, Borland has provided an attractive value proposition for the development and deployment of "mixed environment" enterprise-class applications. Organizations that want to take advantage of .NET and yet preserve their investments in robust J2EE and CORBA back ends can do so with ease. Enterprises can do the following:

- Develop and deploy .NET applications that directly connect to high-performance J2EE or CORBA servers
- Allow .NET developers to continue to develop the front-end client and access J2EE or CORBA servers without becoming experts
- Accelerate life cycles with the model-driven approach to rapid development
- Not have to settle for bridging technology to produce applications for heterogeneous environments that support Java and .NET platforms

IN THIS STUDY

Borland Software continues to hold the coveted position as one of the top independent tools vendors for both Java and Windows application development. The primary reason is that Borland once again has transformed its understanding of the evolving requirements of the developer community to produce tools and environments that manage the convergence of the .NET and Java technologies. Recent acquisitions of TogetherSoft and Starbase have added to Borland's offering and have enabled Borland to provide its customers with best-of-breed full life-cycle development support, from analysis, design, and modeling to management, deployment, and maintenance.

Borland has begun delivering on its promises to provide seamless interoperability between Microsoft .NET, Windows, Java, and Linux platforms through proven tools and technology solutions that help developers effortlessly span these domains. Borland's success with its new tools will have a positive impact on the future growth of the developers tools market in new and profound ways that few thought possible.

This IDC document looks at how the company's acquisitions have given Borland a leading edge in providing extended development environments. It focuses on Borland's latest tools for .NET and C#, with particular emphasis on Janeva.

SITUATION OVERVIEW

Over the last decade, Borland Software has been through many changes while transforming and establishing itself as a leading provider of software development tools and technologies. As far back as the introduction of Delphi — which spawned a cult-like following — Borland has demonstrated that it is carving out an enviable position in the application development tools market. When the company introduced its JBuilder integrated development environment (IDE), it did not take long for Borland to become the preeminent leader in the Java application development tools space.

More recently, Borland has gone on an acquisition spree to obtain the crucial products and technology it needed to round out its offering and extend its development environment. The company's strategy has been reworked so it can heed its customers' calls for integrated application life-cycle management solutions. As a result, Borland is expanding beyond just software development to requirements, analysis, design, collaboration, and testing. Borland's acquisitions of TogetherSoft, Starbase, Caliber, and Bold allowed the company to acquire the technology it needed to complete the strategy it embarked upon. Rather than build life-cycle suites that provide breadth only, Borland is striving to go further to focus on "acceleration." The idea is to bring to market those technologies that speed up the application development life cycle and deliver orders-of-magnitude improvements in productivity.

Borland's strategy is to not only continue to be the leading independent vendor of application development tools and environments but also remain standards compliant and technology agnostic — the veritable Switzerland of the software development world. With its strategy in place, Borland has just released two key products that will no doubt make a significant impact on the enterprise: Janeva and C#Builder.

J A N E V A

Janeva is a compiler and a set of libraries that allow Microsoft .NET applications to connect to back-end J2EE or CORBA applications. Unlike Web services and other Java and .NET connectivity software (bridges), Janeva provides a tight, fine-grained integration to J2EE and CORBA by communicating to such back ends using the Inter-ORB Protocol (IIOP). What Borland has brought to market with Janeva is the ability for enterprises now to develop applications for mixed environments.

Organizations such as those in the financial services and telecom industry have made significant investments in the development of Java and C++ applications where the back-end servers are either J2EE or CORBA based. The reasons are obvious: Scalability, security, and transaction performance are key in these sectors, and J2EE and CORBA servers deliver these features reliably. However, there is no argument that Microsoft owns the desktop and that .NET is rapidly becoming the technology of choice for the front-end or presentation layer.

Until now, the .NET and Java/CORBA worlds have been separate and unconnected. Attempts have been made to span this chasm through the use of Web services and bridges. Unfortunately, the state of the technology today does not support the deployment of mission-critical applications. The Simple Object Access Protocol (SOAP) is still immature and is aimed at providing loose coupling between applications. Although SOAP may grow to support better security and performance, it will still not address the tight, fine-grained integration that is required for many applications. Recent introductions of bridges by a variety of vendors provide a means to span the gap between .NET and J2EE, but these bridges typically require additional hardware and software and can easily become the bottleneck and a single point of failure in a system.

Borland's Janeva is unique in that it allows the .NET developer to continue to develop the front-end client while making J2EE and CORBA transparent. The Janeva libraries reside alongside the .NET application and are called by it. No additional hardware or software infrastructure is required. Included with the development environment are a Java-to-C# compiler and an IDL-to-C# compiler that generate C# stubs and assemblies. The Janeva libraries are fully J2EE and CORBA aware and handle the task of translating .NET remoting calls to IIOP at runtime, thereby allowing a tight integration with the back ends. The .NET developer does not need to become an IIOP, J2EE, or CORBA expert to use Janeva. The net result is that .NET applications can be directly connected to high-performance servers, and the effort and training costs to achieve this are minimal.

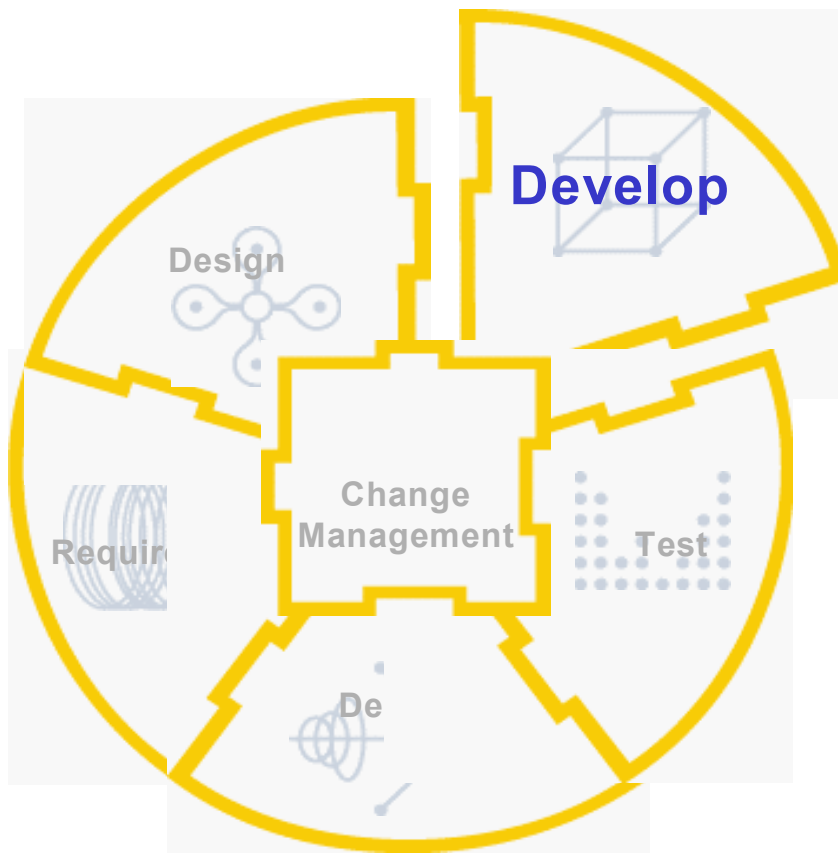
Janeva supports Microsoft's Common Language Runtime and, hence, supports C#, J#, Visual Basic.NET, and Visual C++.NET. Janeva is also tightly integrated with Microsoft Visual Studio and C#Builder.

C # B U I L D E R

C#Builder follows in the tradition of JBuilder by delivering an IDE for C# development. From a big-picture point of view, C#Builder is one piece of the larger integrated application life-cycle management suite that Borland is striving to develop and deliver. Current versions already provide a level of integration, and future versions will offer deeper integration (see Figure 1).

FIGURE 1

THE DEVELOPMENT PIECE



Source: Borland Software Corp., 2003

C#Builder is Borland's platform for .NET C# development for the enterprise. Other language support will be available soon. C#Builder already offers basic capability for VB.NET by allowing editing and compiling of VB source. Several key features distinguish C#Builder from other .NET development environments:

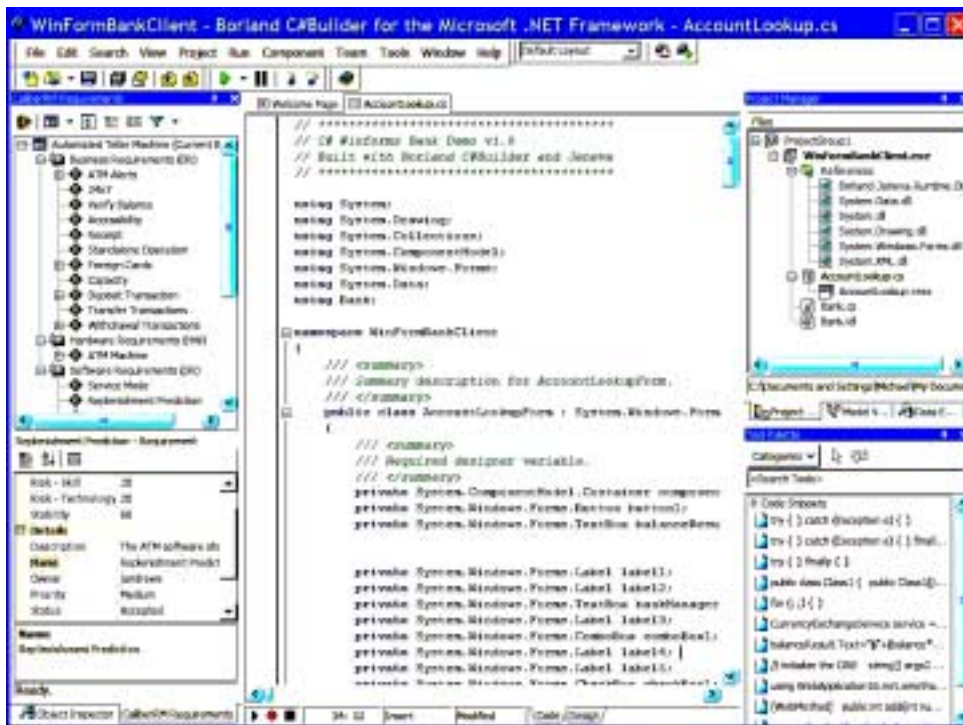
- ☒ **Adherence to proven standards.** Borland's approach in providing development suites is to embrace published and de facto standards and to find innovative ways to enhance capability. This is true of C#Builder, which is fully compatible with C# language specifications, the .NET Framework — including WinForms, WebForms, ASP.NET, and ADO.NET — and Microsoft Visual Studio for .NET. C# source files can be shared, and C#Builder can transparently import and export VS.NET project files. In addition, C#Builder includes the Janeva libraries to be integrated directly over IIOP with J2EE and CORBA servers.
- ☒ **Support for model-driven development.** C#Builder provides integration of the UML modeling environment development and runtime phases with Borland Enterprise Core Objects (ECO) for .NET to create a model-driven application design system. This .NET business-object runtime platform allows applications to import and export UML models between C#Builder and external modeling tools such as Borland Together into C#Builder. From within the ECO platform,

developers can also visually create and edit the UML model with the visual model designer, powered by Borland Together technology. Users of the Borland environment have reported orders-of-magnitude improvement in development times.

- ☒ **Support for mixed enterprise environments.** Although C#Builder supports standard database access via the Borland Data Provider for ADO.NET, developers can also access MSSQL Server, Oracle, DB2, and InterBase. Data Provider provides mapping between .NET data types and vendor-specific data types, thereby reducing complexity and also allowing the ability to change databases without recoding. A key feature called Live Data reduces coding required for ADO.NET development and displays data in visual controls at design time. And with Janeva, the C#Builder solution now allows the enterprise developer to deploy applications that have .NET clients with J2EE or CORBA servers.
- ☒ **Integration with key life-cycle components.** One of the key tools that has been integrated with C#Builder to deliver on the promise of acceleration of the life cycle and reduction in costs is Optimizeit Profiler. Optimizeit tracks performance and reliability issues throughout the development life cycle with CPU and memory use analysis. It can show developers how code is executed within the CLR and where performance improvements are needed. With real-time analysis of CPU and memory usage, garbage collection, and object allocations, developers can quickly pinpoint problem areas down to the method and line of code (see Figure 2).

FIGURE 2

OPTIMIZEIT INTEGRATION



Source: Borland Software Corp., 2003

FUTURE OUTLOOK

IDC views Borland's introduction of these new products as a means to further solidify its position in the application development tools market. Janeva and C#Builder are strategic releases that have put Borland in the position of being the "go-to guys" from the perspective of many professional developers for enterprise software development — whether it is for the desktop or the back-end server.

Rather than getting embroiled in the Java-versus-.NET debate, Borland has taken a tack to render this argument moot. Enterprises that prefer to go with .NET on the desktop but still have the need to deploy secure, high-performance, high-transaction, scalable servers can do so now with impunity. And those legions of Visual Basic programmers can now migrate to .NET and still be able to build enterprise-class applications that need Java application servers or CORBA compatibility.

The acquisition of TogetherSoft has given Borland an analysis, modeling design, and construction development suite that rounds out the development effort by adding more advanced design and modeling capability. With true round-trip engineering in place, the entire task of ensuring that documentation and designs are up to date and that development teams can track the progress of project completion is now greatly simplified. When coupled with Optimizeit, Borland has started to deliver on its strategy of acceleration of the development life cycle.

ESSENTIAL GUIDANCE

Although Microsoft and IBM have settled down in the rival camps of .NET and Java (notwithstanding IBM's acquisition of Rational), Borland Software has placed itself squarely in the neutral position. This could well play to Borland's advantage. The company could end up being viewed as the only vendor that is customer focused and bent on meeting customer needs rather than being technology biased. In fact, this is the only viable position that Borland can take so that it does not get lost in the clash between the titans.

More than anything else, enterprises that have heterogeneous environments and must leverage their resources across the somewhat disparate worlds of Java and Microsoft technologies can now develop and quickly deploy applications without waiting for the ultimate outcome of the raging battle. They can more effectively deploy their IT resources to take advantage of the technologies that each developer group is versant in and go to market quickly. In the end, they also avoid any costly mistake in having to choose one technology at the expense of the other.

Borland has come out as an early leader in helping developers avoid having to make an either/or choice when it comes to deciding if they want to create Windows or Java applications. This is a significant suite of tools for any company pondering the difficulty of creating future applications and Web services that must contend with these major heterogeneous environments.

LEARN MORE

RELATED RESEARCH

- ☒ *As Sun Sets Its Sights on the Future, It May Be Dropping the Developer Ball* (IDC #29203, April 2003)
- ☒ *Survival Strategies for Swimming with Big Fish in a Shrinking Tools Pond* (IDC #29111, March 2003)
- ☒ *Worldwide Application Design and Construction Tools Forecast, 2003–2007* (IDC #29077, March 2003)
- ☒ *Worldwide Analysis, Modeling, Design, and Construction Tools Forecast, 2003–2007* (IDC #29049, March 2003)
- ☒ *Worldwide Unified Development Environment Tools Forecast, 2003–2007* (IDC #29046, March 2003)
- ☒ *Worldwide Third-Generation Language Forecast, 2003–2007* (IDC #29038, March 2003)

COPYRIGHT NOTICE

This IDC research document was published as part of an IDC continuous intelligence service, providing written research, analyst interactions, telebriefings, and conferences. Visit www.idc.com to learn more about IDC subscription and consulting services. To view a list of IDC offices worldwide, visit www.idc.com/offices. Please contact the IDC Hotline at 800.343.4952, ext. 7988 (or +1.508.988.7988) or sales@idc.com for information on applying the price of this document toward the purchase of an IDC service or for information on additional copies or Web rights.

Copyright 2003 IDC. Reproduction is forbidden unless authorized. All rights reserved.

Published Under Services: Application Design and Construction Tools; Java and .NET