

# DATABASE

## TRENDS AND APPLICATIONS

Solutions for the Information Project Team • [www.dbta.com](http://www.dbta.com)

Volume 19, Number 10 • October 2005

## BlueFinity Provides the Key to Microsoft .NET for Multivalued Developers

Founded in 2002 by a group of developers with long experience using MultiValue database technology, BlueFinity is dedicated to creating developer tools for the Microsoft .NET environment. Its flagship offering, mv.NET, is a fully managed .NET data provider which allows the full benefits of the .NET architecture to be realized by users of all MultiValue applications.

The dominant product of its type, mv.NET can be used with all major MultiValue databases. Moreover, it is database release independent, so even legacy MultiValue applications can be easily integrated into the most contemporary information infrastructures. In short, MultiValue applications will no longer be islands within a company's overall architecture. MultiValue application development is now accessible to all .NET programmers.

### Key Features

Microsoft's .NET technology is universally used to develop Rich Client, Web and Web Services applications. mv.NET offers several unique features to facilitate that process when linked to a MultiValue database as the backend. It excels by providing the complete development environment in the form of a series of plug-ins to Visual Studio .NET (a beta for VS2005 is also available now, with a full release to follow VS2005's official release from Microsoft). New MultiValue applications can now be developed without ever needing to leave Visual Studio.

A key aspect of all .NET implementations is managing the connections between the user interface and the database. mv.NET's Session Manager makes certain that those connections are handled in the most efficient and cost-effective way, ensuring that users get outstanding response from their MultiValue applications. Users want fast response times and Data Fetch on Demand technology enables outstanding application response to be achieved, for instance, if large volumes of data must be returned to the application. It parses the data in smaller packets, with no need for developer intervention. Of course, developers who wish to customize the way the data is returned for specific applications have the ability to do so.

Another significant feature is mv.NET's ability to dynamically manage the normalization of data from MultiValue databases to ADO.NET. Consequently, for developers working with Microsoft's ADO.NET technology, MultiValue databases now appear just like another SQL database.

### Product Structure

mv.NET has three major components: Core Objects, Binding Objects and Adapter Objects. Core Objects, as its name suggests, is the backbone which consists of the server, client and connectivity components together with a full .NET data class library and Data Manager utility. In many cases, mv.NET uses the native high performance interfaces provided by the MultiValue database suppliers.

mv.NET is a 'must have tool' for the serious .NET programmer in our MultiValue community

Binding Objects complements standard .NET binding to provide the complete environment for advanced databound programming. By making Visual Studio MultiValue-aware, the binding process is significantly simplified for both out-of-the-box and third-party .NET controls. This makes mv.NET a true RAD tool for Rich Client development.

Adapter Objects is a full ADO.NET managed data provider implementation and provides a full read/write interface for MultiValue databases. MultiValue data is dynamically normalized into ADO.NET recordsets making this the way forward for true SQL-like application development and reporting, for example, using products like Crystal Reports.

### Case Study

Drexel Management Services has been a value-added reseller in the MultiValue marketplace since the early 1980s. The company has developed its

own vertical point of sale application for restaurants as well as a credit card processing application that it markets through other VARs. It also provides consulting and support operations for other MultiValue applications and VARs. "We work with a wide variety of applications and platforms," said Drew Conboy, the company President.

In the mid-1990s, Drexel became heavily involved with Microsoft Windows. Adding a graphical user interface proved to be a significant challenge faced by vendors of MultiValue applications. Over time, Drexel explored many products but all fell short.

Last spring, Drexel began to work with mv.NET, which has the feature set for which they had long been looking. It is very flexible and works with a wide range of backend databases. A state-of-the-art GUI can be added to their applications without needing substantial alterations to the time-tested application itself. The mv.NET learning curve has been short and has been greatly aided by the strong support BlueFinity provides from its United Kingdom headquarters and support office in Cleveland, OH.

"In the long run mv.NET will help the MultiValue community remain competitive," said Conboy. "You can maintain the integrity of your existing MultiValue business logic and add the feature-rich GUI that comes with the .NET environment." Conboy concludes, "mv.NET is a 'must have tool' for the serious .NET programmer in our MultiValue community."



### CONTACT INFORMATION

**BlueFinity International, Inc.**  
[www.bluefinity.com](http://www.bluefinity.com)  
[sales@bluefinity.com](mailto:sales@bluefinity.com)

**Drexel Management, Inc.**  
[www.drexelmgmt.com](http://www.drexelmgmt.com)  
[sales@drexelmgmt.com](mailto:sales@drexelmgmt.com)