WHITEPAPER

First Look at Advantage Database Server 8.0

Cary Jensen, Ph.D., Jensen Data Systems

SYBASE iAnywhere

TABLE OF CONTENTS

1 Overview

2 What's New in Advantage 8.0?

- 2 Updated Advantage Data Architect
- 3 Online Backup
- 3 Replication Support
- 4 SQL Updates
- 4 SQL Persistent Stored Modules
- 5 General Iimprovements to Advantage SQL
- 5 Security Enhancements
- 6 Improved Performance
- 6 Enhanced Triggers

6 Additional Improvements

- 6 Transaction Savepoints
- 7 New Field Types
- 7 Larger DBFs
- 7 Updates to the Advantage .NET Data Provider
- 7 Application Identification

8 Summary

8 About The Author

Advantage 8.0 is the latest version of iAnywhere's Advantage Database. This article begins with a brief overview of Advantage. It continues with a look at many of the updates and enhancements that extend the Advantage Database Server's position as the leading low maintenance, high performance server for small to medium sized applications.

OVERVIEW

The Advantage Database Server (ADS) from iAnywhere is a low maintenance, high performance remote database server (RDBMS). What distinguishes ADS from other databases is that it supports a wide range of high-end database features with an ease of use that is virtually unheard of in this type of software.

Advantage Local Server (ALS) is a free, file-server based database that is virtually 100% API (application programming interface) compatible with ADS, providing developers with a no-cost solution where the benefits of a remote database are not yet needed. But the real value of ALS is that it provides an effortless migration path for applications as they mature to the point where the features of a true RDBMS are required.

Most of the topics in this article apply to both ADS and ALS. When that is the case these products are referred to generically as Advantage.

The high-end features supported by Advantage rival those found in the most popular database servers. These features include a wide range of security options, triggers, stored procedures, views, referential integrity constraints, extensive SQL support, online backups, full text search, table and field-level constraints, and sophisticated indexes. Furthermore, ADS supports replication, database clustering, and over the Internet connectivity.

Advantage is notable for its extensive connectivity options. These include Windows and Linux ODBC (open database connectivity) drivers, an OLE DB provider, a Java class 4 JDBC (Java database connectivity) driver (ADS only), a .NET data provider, Delphi TDataSet descendants for Win32 and VCL for .NET, Borland Data Providers (BDP) for .NET, a Kylix TDataSet descendant for Linux, Windows and Linux PHP drivers, Windows and Linux Perl DBD drivers, an RDD (replaceable database driver) for CA-Visual Objects, a Clipper RDD, a Crystal Reports driver, as well as Windows and Linux client APIs for C++, Visual Basic (Windows only), and Delphi/Kylix.

While sophistication in a database server is often accompanied by complexity of configuration and management, Advantage serves as a refreshing exception. In most cases, Advantage installations require only minimal configuration, such as the selection of a collation language.

Even more important, most ADS installations require no maintenance and no DBA (Database Administrator). It is common for an ADS installation to run for years without requiring attention. (All file-server based databases are inherently susceptible to potential corruption, which is one of the reasons why most developers deploy ADS over ALS.) This is one of the reasons why Advantage is so popular with vertical market software developers.

Another benefit of Advantage is its support for both a navigational model and a SQL set-base model of data access. Advantage is based on ISAM, the indexed sequential access method. ISAM databases make extensive use of indexes, providing for high-speed searches, ranges, joins, sorts, and record-level navigation.

Because of its ISAM architecture, Advantage can perform tasks that are unthinkable in set-based SQL databases. For example, you can use Advantage to open a 20 million record table, set a sort order, and then navigate to the last record (or the first record, or any record, for that matter). In most cases such an operation takes a fraction of a second. This same task in a set-based database could easily take hours.

Importantly, while Advantage supports high-speed navigational operations, it also provides extensive support for ANSI 92 standard SQL operations. In most cases, these operations are highly optimized, providing SQL-based performance on par with most set-based SQL databases.

Advantage's support for navigational operations make it the preferred database for migrating file-server based applications; like those originally designed around Clipper, dBase, Paradox, MS Access, and other similar file-server based database formats. (In fact, Advantage directly supports both Clipper and FoxPro DBF formats.) Since these types of applications often rely on navigational operations, migrating them to a set-based SQL database generally requires an extensive re-write.

But Advantage is also the smart choice for new development. By supporting both navigational and set-based operations, you choose which data access methods are appropriate for your application's needs. For GUI (graphical user interface) applications, this flexibility gives you a wider range of options for designing your user interfaces.

WHAT'S NEW IN ADVANTAGE 8.0?

In keeping with the last two major point upgrades to Advantage, iAnywhere has included a large number of feature and performance enhancements in Advantage 8.0. But to begin with, it is important to note that Advantage 8.0 is mostly forward compatible with Advantage 7.x client applications.

Specifically, in most cases, currently deployed applications can connect to, and use Advantage 8.0. While those applications based on 7.x cannot access all of the new features available in 8.0, this forward compatibility means that you do not necessarily have to recompile and redeploy all of your existing applications before you upgrade to Advantage 8.0.

On the other hand, there are a number of features of Advantage 8.0 that you can use without redeploying existing Advantage 7.x client applications. For example, as soon as you deploy Advantage 8.0 you can immediately benefit from the new live backup capabilities. Then, as time permits, you can update your client applications to take advantage of Advantage 8.0's additional features, such as the updated .NET data provider.

However, due to the nature of some of the enhancements added to Advantage 8.0, there are some specific instances where upgrading from Advantage 7.x also requires modifications to your client applications. For example, Advantage 8.0 adds a number of new SQL reserved words, including CURSOR, RIGHT, and SAVEPOINT, to name a few. Any existing SQL statement that uses one of these new words as an un-quoted field reference must be modified.

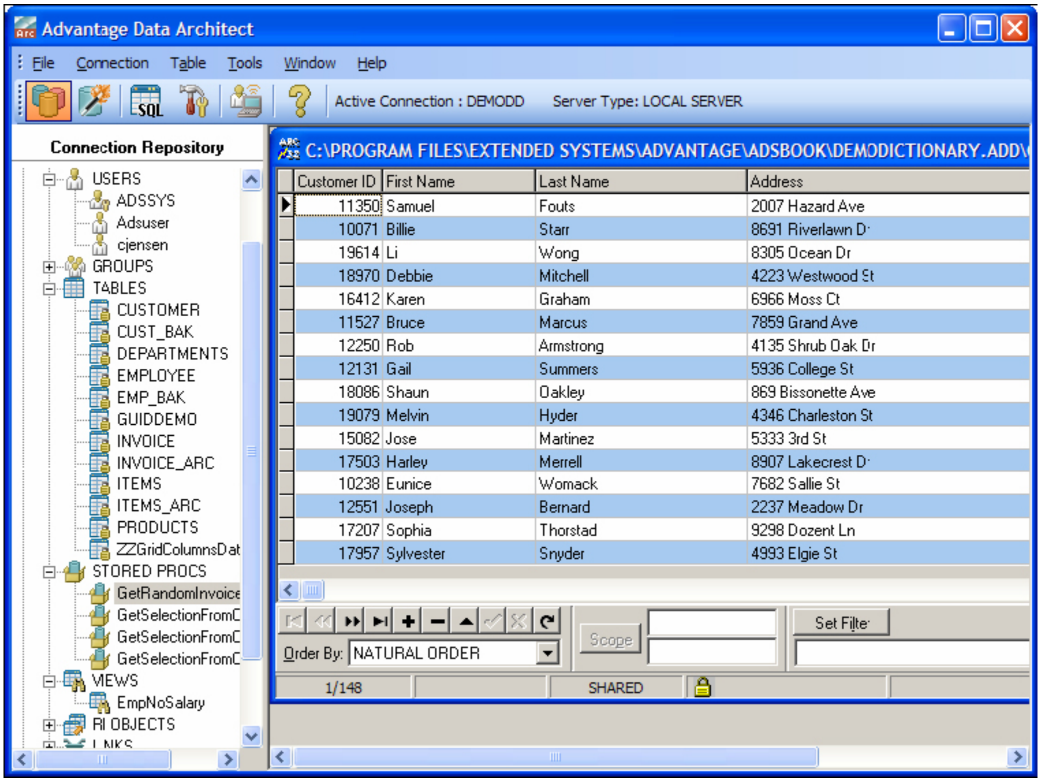
Fortunately, most existing Advantage client applications will run using Advantage 8.0 without modification. For a list of changes that might affect your existing application, refer to the readme.txt file located in the directory where you install Advantage.

The remaining sections of this article take a look at the most interesting additions and enhancements found in Advantage 8.0. For a comprehensive list of updates and improvements, please refer to the Advantage v8.0 help file, which can be accessed by selecting Help | Contents from the new Advantage Data Architect main menu.

Updated Advantage Data Architect

While database servers themselves are non-visual service applications, many databases provide you with some type of GUI utility for creating, configuring, and managing your databases. With Advantage, this utility is called the Advantage Data Architect.

With the release of Advantage 8.0, the Advantage Data Architect has received its first complete re-write since it was originally released. The updated look can be seen in Figure 1, which shows an open database with an open table from the main screen of the updated Advantage Data Architect.



Customer ID	First Name	Last Name	Address
11350	Samuel	Fouts	2007 Hazard Ave
10071	Billie	Starr	8691 Riverlawn D·
19614	Li	Wong	8305 Ocean Dr
18970	Debbie	Mitchell	4223 Westwood St
16412	Karen	Graham	6966 Moss Ct
11527	Bruce	Marcus	7859 Grand Ave
12250	Rob	Armstrong	4135 Shrub Oak Dr
12131	Gail	Summers	5936 College St
18086	Shaun	Oakley	869 Bissonette Ave
19079	Melvin	Hyder	4346 Charleston St
15082	Jose	Martinez	5333 3rd St
17503	Harley	Merrell	8907 Lakecrest D·
10238	Eunice	Womack	7682 Sallie St
12551	Joseph	Bernard	2237 Meadow Dr
17207	Sophia	Thorstad	9298 Dozent Ln
17957	Sylvester	Snyder	4993 Elgie St

Overall, the look and behavior of the Advantage Data Architect conforms much more closely than previous versions to standard 32-bit Windows applications. In addition, great thought was given to your access to features. The result is a consistent and user-friendly interface that simplifies your work.

Online Backup

Probably one of the most anticipated additions found in Advantage 8.0 is online backup. With online backup, you can create backups of your database, even while it is in use.

Advantage's online backup is implemented through a command line utility, adsbackup.exe. This utility can be invoked manually, initiated by a feature implemented in a client application, or executed on a regularly scheduled basis through a scheduling utility, such as the Scheduled Tasks applet included in the Windows operating system. By scheduling regular backups, you ensure that a current backup is available for restoring your database in case something happens to your live data.

Advantage's online backup utility supports both complete as well as differential backups. Differential backups take less time to execute than complete backups by duplicating only those records that where changed, deleted, or added since the last backup.

Replication Support

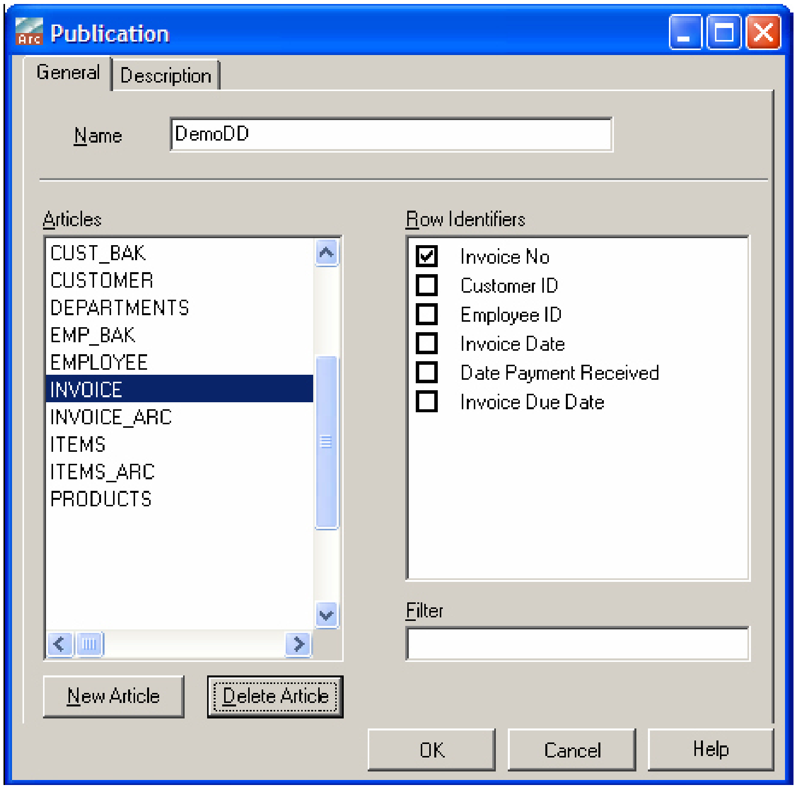
Replication, while similar to online backup in that a copy of your database can be made while it is in use, provides two additional capabilities. These are the ability to make a backup that stores data off-site, as well as the ability to update two or more databases. Replication support requires ADS and a separate license.

Off-site replication permits you to make live backups to a remote location. These backups can then be used to restore your database in the case of a catastrophic event, such as a fire, flood, or any other event that could damage or destroy the server or the facility in which the server is located.

Replication also permits you to ensure that two or more separate database servers share their data. For example, the server at your company's headquarters can download recently added data to each of your company's regional offices during a replication event.

Replication can either be unidirectional or bidirectional. In addition, you can configure replication to replicate all of your tables, or only those tables for which replication is needed.

Replication is configured through a publisher/subscriber model. Figure 2 shows a publication being configured, including which tables will be available for replication, and which fields will be used to uniquely identify records.



Name: DemoDD

Articles:
- CUST_BAK
- CUSTOMER
- DEPARTMENTS
- EMP_BAK
- EMPLOYEE
- INVOICE
- INVOICE_ARC
- ITEMS
- ITEMS_ARC
- PRODUCTS

Row Identifiers:
- ☑ Invoice No
- ☐ Customer ID
- ☐ Employee ID
- ☐ Invoice Date
- ☐ Date Payment Received
- ☐ Invoice Due Date

SQL Updates

There are many updates to Advantage's support for SQL in 8.0. Many of these are related to performance, and are mentioned in a later section of this paper. This section will specifically discuss two areas of Advantage's SQL improvements. These are the addition of SQL persistent stored modules (PSM) and the introduction of new SQL keywords and capabilities.

SQL Persistent Stored Modules

SQL scripts in Advantage 8.0 now support most of ANSI SQL PSM 2003. ANSI (American National Standards Institute) SQL PSM 2003 specifies the syntax and semantics of statements that add a procedural capability to the SQL language in the form of functions and procedures. It includes statements to direct the flow of control, define variables, make assignments, and handle exceptions.

SQL PSMs are particularly useful for declaring stored procedures and triggers, particularly if your client application development environment does not support the creation of DLLs (32-bit Windows), share object libraries (Linux), or .NET managed assemblies. For example, if your development language is Java, Perl, or PHP, SQL PSMs provide you with a powerful tool for creating stored procedures and triggers without resorting to another compiler, such as Delphi or C# for .NET.

Even if you are using a language that can compile external stored procedures and triggers, SQL PSMs offer a solution for creating triggers and stored procedures that are operating system independent and require no additional files to be deployed to your servers.

The following are some of the keywords supported by Advantage's SQL PSM: CACHE, CATCH, CLOSE, CONTINUE, DECLARE, EXECUTE IMMEDIATE, END, FETCH, FINALLY, IF, LEAVE, OPEN, RAISE, RETURN, TRY, and WHILE. In addition, SQL PSM permits you to declare variables, assign values to variables, include regular SQL statements, and use variables as query parameters.

General Improvements to Advantage SQL

Advantage 7.x support for SQL is excellent, and in this release it gets even better. For starters, Advantage SQL now supports both FULL and RIGHT OUTER joins. Additional improvements include permitting the use of subqueries anywhere an expression is permitted, permitting alias column names in HAVING clauses, and the introduction of an escape character in LIKE conditions.

Advantage SQL scripts can now also return a cursor. Specifically, if SELECT is the last statement in a SQL script (a SQL statement containing two or more SQL statements separated by semicolons), that script will return a cursor to the records fetched by the SELECT statement.

Another new feature in Advantage SQL is the support for query logging. Through the system administrator account (ADSSYS), the sp_EnableQueryLogging and sp_DisableQueryLogging system stored procedures turn logging on and off.

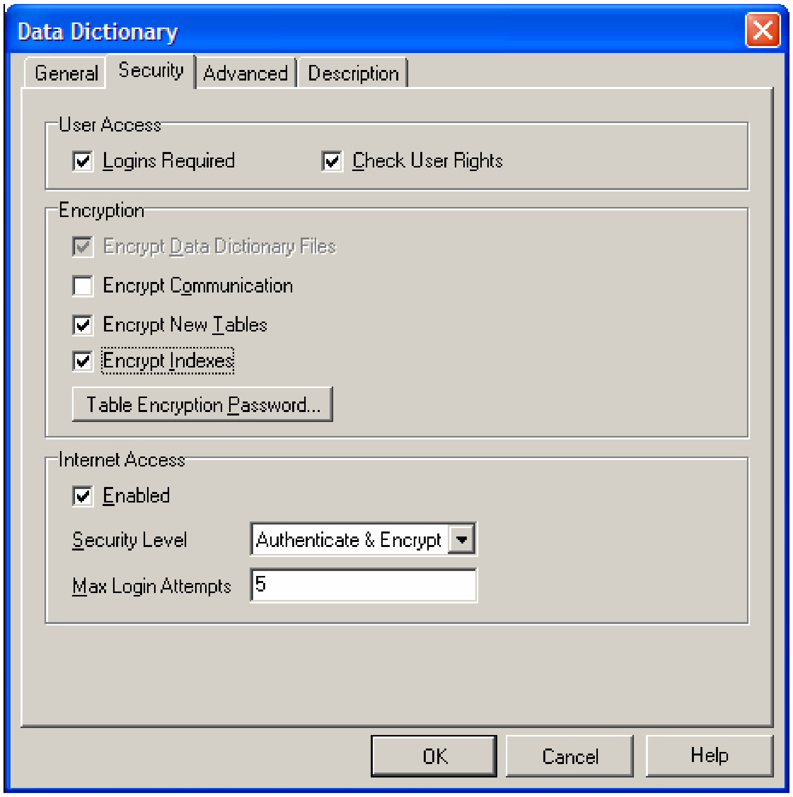
When query logging is enabled, queries and their optimization levels are written to a table of your choosing. In addition, the sp_ViewQueryLogging system stored procedure can be used to report which connections are currently logging queries.

Finally, support for query parameters has been improved. For SQL scripts, each SQL statement in a SQL script can now have a different number of parameters, as well as different parameter types. Previously, if you passed parameters to a SQL script it was necessary for each SQL statement in the script to use those same parameters.

Security Enhancements

Most database developers recognize the need for security in their applications. To address these needs, Advantage 8.0 includes three valuable enhancements that improve your data's security.

The first of these is the option to encrypt indexes. Normally, database tables and indexes are invisible to the user, being stored on secure directories on the server. Now, those stand-alone applications where the data resides on the user's machine can use encrypted indexes to prevent sensitive fields participating in the index from being viewed. Figure 3 shows the Security page of the Data Dictionary Properties dialog box where you use the Encrypt Indexes check box to enable index encryption.



User Access

☑ Logins Required ☑ Check User Rights

Encryption

☑ Encrypt Data Dictionary Files

☐ Encrypt Communication

☑ Encrypt New Tables

☑ Encrypt Indexes

Internet Access

☑ Enabled

Security Level: Authenticate & Encrypt

Max Login Attempts: 5

Another security enhancement is the option to encrypt client-server communication on local networks. Prior to this release, only those clients connecting to ADS over the Internet had an option to employ encrypted communication. You use the Encrypt Communication check box shown in Figure 3 to enable this feature.

Finally, ADS now supports server-side aliases. Server side aliases are symbols that define for a server where the data is located on the network. A client application passes this symbol to the server, which then resolves the alias to a directory path based on a server-side INI file. This feature eliminates the need for shares, and keeps the location of the data hidden from the client applications.

There is one additional benefit to server-side aliases that does not relate directly to security. A server-side alias can point to a network attached storage (NAS) device. This permits an Advantage database to be located on a NAS, improving your data storage and backup options.

Improved Performance

In addition to security, another issue high on the list of priorities for database developers is performance. And while Advantage has always been a top performer, iAnywhere has managed to squeeze even better performance out of this release.

Some of the improvements that affect performance include a new caching scheme that more aggressively caches indexes and static cursors. Other operations benefiting from improved performance include transaction processing, SQL parsing, filter and record count operations, and table restructuring.

As mentioned in passing earlier in this article, many performance improvements have also been included in the SQL engine. These include improved subquery and correlated subquery performance, improved view performance, and improved GROUP BY performance.

Enhanced Triggers

Triggers are routines that execute in response to a record-level event, such as a record being modified, inserted, or deleted. Advantage 8.0 introduces a number of trigger improvements, making them more valuable than ever.

One of the more welcome updates to triggers is that you can now write to the record that has been inserted or modified from within an AFTER trigger. Previously, you had to write an INSTEAD OF trigger if you wanted to change data being inserted or edited. In general, AFTER triggers are much easier to write than INSTEAD OF triggers.

Another enhancement can be found in the caching of triggers. In previous versions, it was necessary to restart ADS after updating a DLL or .NET managed assembly trigger. Now, in most cases simply replacing the DLL or .NET assembly is sufficient.

There are also two new features related to triggers. First, you can now place an INSTEAD OF trigger on a static view. This trigger can then be used to update one or more of the tables involved in the view. Through this mechanism you can now have editable views, including those that rely on two or more joined tables.

The last new feature related to triggers is the ability for the administrative account to disable triggers temporarily. While in most cases you will not want to disable triggers in a deployed application, there are some operations, especially those that occur only during the development process, that benefit from the ability to disable triggers.

ADDITIONAL IMPROVEMENTS

This last section includes a look at additions and improvements that fall into the miscellaneous category. These include transaction savepoints, new field types, larger DBFs, Advantage Data Provider enhancements, and application identification. Each of these is discussed in the following sections.

Transaction Savepoints

Transactions in Advantage now support savepoints. A savepoint identifies a state during a transaction to which you may want to rollback to, without having to rollback the entire transaction.

You can have multiple savepoints for a single transaction. You specify which savepoint you want to rollback to when you want to perform a partial rollback. When you rollback to a given savepoint, those operations within the transaction that were executed prior to the savepoint remain intact.

New Field Types

Advantage 8.0 introduces two new field types: RowVersion and ModTime. Both of these fields are automatically assigned their value by Advantage at runtime.

When you include a RowVersion field in a table, that field can hold an unsigned 64-bit integer that is automatically assigned each time a record is inserted or changed. This is an auto-incrementing value, with the last record inserted or modified containing the highest value. RowVersion is particularly useful for detecting record changes when applying changes from a cached dataset, such as a .NET DataTable.

ModTime is a time stamp field that is automatically assigned a value each time a record is inserted or changed. ModTime eliminates the need to write an insert and update trigger to record the time of the most recent change to a record.

Larger DBFs

Previously, DBFs could be no larger than 4 GB in size (2 GB in Linux prior to glibc 2.1.2 and kernel 2.4). This is still the case with 32-bit Windows operating systems using FAT32 and NetWare 5 or greater with the NSS file system.

However, with 32-bit Windows using NTFS, NetWare 5 and greater with traditional file systems, and Linux glibc 2.1.2 and kernel 2.4 and higher, the maximum size of a DBF is substantially greater. Specifically, with these operating systems a DBF can be the maximum number of records supported by DBF (2,147,483,648) multiplied times the size of each record. For example, if each record in a DBF is 1,000 bytes in size, the maximum size of that DBF would be just under two terabytes (2,147,483,648,000 bytes).

Updates to the Advantage .NET Data Provider

Advantage 8.0 includes updates in two areas of its Advantage Data Provider for .NET. The first is the introduction of AdsHelper, an implementation of the Data Access Application Block for .NET.

AdsHelper simplifies the process of creating and managing AdsDataSets, AdsDataReaders, and AdsExtendedReaders. Like the Data Access Application Block supplied by Microsoft in the Microsoft Enterprise Library for the .NET framework, AdsHelper permits you to write more portable code and do so more quickly.

The second major area of enhancement is found in the AdsExtendedReader. The AdsExtendedReader is an implementation of IDataReader, with a twist. Unlike most IDataReader implementations, AdsExtendedReader offers both forward and backward navigation, as well as table editing capabilities.

In Advantage 8.0 the AdsExtendedReader can do even more. The following are some of the new methods you will find in the AdsExtendedReader: IsRecordLocked, IsRecordDeleted, IsTableLocked, and DeleteIndex. In addition, the read-only property LogicalRecordNumber has been added.

Application Identification

Client applications can now be programmatically identified using Advantage's new application identification feature. By default, application identification resolves to the name of the executable that is connected to Advantage. If you want, you can also programmatically set the application identification for a particular client application through the sp_SetApplicationID system stored procedure.

You read an application's identification by executing the sp_GetApplicationID system stored procedure. This stored procedure is particularly useful from within triggers and stored procedures, permitting you to note which of your client applications is triggering the routine.

SUMMARY

Advantage 8.0 represents yet another significant step forward in the evolution of the Advantage line, with additions and improvements in almost every aspect of the product.

More importantly, these updates are made without sacrificing the ease of use, ease of deployment, and ease of management that are the hallmark of this powerful database. In fact, Advantage 8.0 makes these tasks even easier.

In short, the new and improved features found in 8.0, combined with the power and flexibility that this product is known for, solidifies the Advantage Database Server's position as the best remote database server for small to medium sized applications.

For more information about the Advantage Database Server, visit http://www.AdvantageDatabase.com, and follow the links for Advantage Database Server.

ABOUT THE AUTHOR

Cary Jensen is President of Jensen Data Systems, Inc., a consulting, training, and software development company. He is the award-winning author of 19 books, hundreds of magazine articles, and more than 75 software training courses. Cary has a Ph.D. in Human Factors Psychology from Rice University. For more information about Cary, Jensen Data Systems, or Cary's training offerings, visit http://www.JensenDataSystems.com.

Contact Us

iAnywhere Solutions, Inc.
Worldwide Headquarters
United States
One Sybase Drive
Dublin, CA 9
4568-7902 U.S.A.

For General Information:
AdvantageInfo@iAnywhere.com
North America T 1-800-801-2069 1-519-883-6898

For Specific Regional Product Information:
(800)-235-7576 ext. 5030 (208)-322-7575 ext. 5030
Benelux +31(0)73 6235359
Germany +49 (0)7032/ 798-200
United Kingdom +44 (0)117-333 9000