SYBASE®

Advantage™ Database Server Delphi Components

FEATURES

- Provides native client/server access to the high performance Advantage Database Server

- Eliminates the need for the BDE and dbExpress for easy development and deployment

- Components are compatible with any third-party database components

- Easy conversion from Paradox, InterBase, Access, Pervasive, and SOL Server files using Advantage Data Architect

- TDataSet Descendant source code included

- Supports data access via SQL as well as direct table access and server-side control unique to Advantage

- Features native VCL components parallel to TTable, TQuery, TDatabase, and TStoredProc

- Provides powerful RDBMS functionality including: transaction processing, triggers, stored procedures, referential integrity, and security

- Fully scalable from local to peer-to-peer to client/server environments with no code changes necessary

For developers using Delphi, C++Builder, and RAD Studio, the Advantage Delphi Components are a complete set of drivers that provide native data access to the Advantage Database Server in WIN32 applications. Advantage Delphi Components provide Visual Component Library (VCL) components for Win32. Thus, providing seamless integration directly into the Delphi IDE.

For developers using Delphi Prism, the Advantage .NET Data Provider is the perfect ADO.NET solution. Please reference the Advantage .NET Data Provider data sheet for specific data provider details. The Advantage .NET Data Provider is not included in the Advantage Delphi Components download, but can be found in a separate stand-alone download.

POWERFUL DATA ACCESS FOR ALL DEVELOPERS

Whether you're an experienced .NET developer, are just converting to the framework, or prefer to continue using Win32 forms, the Advantage Delphi Components provide the specific components for your needs. The Advantage Delphi Components include TDataSet components for Win32. Advantage also provides a native ADO.NET Data Provider, for developers who do not want to use the VCL, for Delphi Prism developers, or for developers who require interoperability with other ADO.NET classes and controls.

Advantage TDataSet Descendant for Win32

Using the Advantage TDataSet Descendant, Win 32 developers can continue to program as they always have using standard TTable, TQuery, TDatabase and TStoredProc methods and properties. The Advantage TDataSet Descendant consists of a set of native components that provide developers with easy access to the Advantage Database Server and Advantage Local Server. The Advantage TDataSet Descendant does not require the BDE, nor dbExpress, providing a direct, high-performance connection to Advantage.

Advantage .NET Data Provider

For developers using Delphi Prism, the Advantage .NET Data Provider can be used for seamless and full featured ADO.NET support. In addition to traditional ADO.NET functionality, the Advantage .NET solution provides easier conversion to the .NET Framework than other database engines, because Advantage supports both the expected disconnected recordset functionality in ADO.NET as well as direct navigational/ISAM access to the database via the AdsExtendedReader class.

FULL SCALABILITY—WRITE ONCE, DEPLOY ANYWHERE

Advantage applications can be deployed in stand-alone, peer-to-peer and client/server environments with one set of source code. Advantage does not require an alternate data provider or different set of source code for different network environments. The Advantage Delphi Components will automatically determine if the Advantage Database Server is available directly or whether the Advantage Local Server should be used. Therefore, you only need to develop one application with one version of code for local, peer-to-peer or client/server file access. In addition to one-source scalability, the Advantage Database Server supports Windows, Linux and NetWare for cross-platform distribution. One set of source code can allow you to deploy to multiple environments easily and seamlessly.

SUPPORTED PRODUCTS

Advantage provides two installations depending on which Borland or CodeGear development environment you are using.

Advantage TDataSet Descendant
- Delphi 6, 7
- C++Builder 5

NOTE: The following environments are not officially supported, but are provided as a courtesy: Delphi 3/4/5 and C++Builder 3/4

Advantage Delphi Components
- Borland Developer Studio 2006
- Delphi 2007
- C++Builder 2007
- CodeGear RAD Studio 2009
- Embarcadero RAD Studio 2010

NOTE: The following environments are not officially supported, but are provided as a courtesy: CodeGear Turbo Pro products
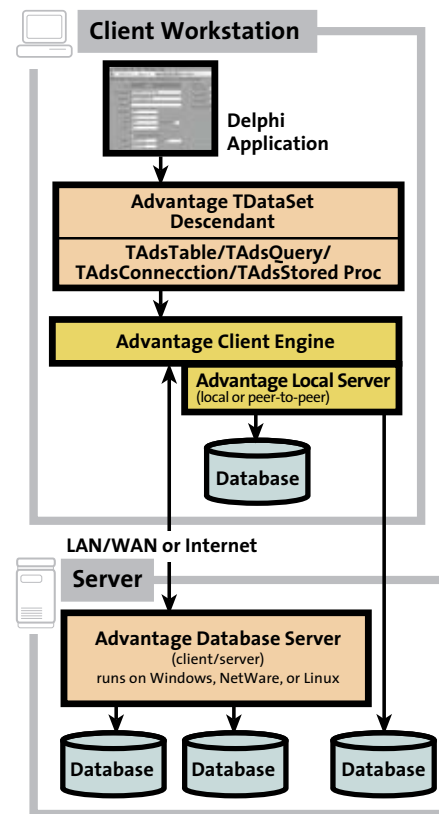
Advantage .NET Data Provider
- Delphi Prism

ADVANTAGE SPECIFICATIONS
Server operating systems (via the Advantage Database Server)
- Novell NetWare 5.x or greater (IP, IPX)
- Microsoft Windows x86 (IP, IPX)
- Microsoft Windows x86_64 (IP)
- Linux x86, x86_64 (IP)

Client operating systems
- Microsoft Windows 2000/XP/Vista/2003/2008

Client Workstation

Delphi Application

Advantage TDataSet Descendant

TAdsTable/TAdsQuery/ TAdsConnecction/TAdsStored Proc

Advantage Client Engine

Advantage Local Server
(local or peer-to-peer)

Database

LAN/WAN or Internet

Server

Advantage Database Server
(client/server)
runs on Windows, NetWare, or Linux

Database   Database   Database

SUPPORTED DATA TYPES

Advantage ADT table data types

AutoIncrement	4-byte read-only integer value from 1 to 4,294,967,296 unique for each record in the table.
Binary	Variable length BLOB containing binary data. The size of each field is limited to 4 GB. The binary image data is actually stored in a separate file, called a memo file, to reduce table bloat.
Character	Fixed length character field from 1 to 65,530 bytes that is stored entirely in the table.
Cicharacter	Case insensitive fixed length character field from 1 to 65,530 bytes that is stored entirely in the table.
Date	4-byte date field.
Double	8-byte IEEE floating point value in the range 1.7E +/-308 (15 digits of precision).
Image	Variable length BLOB containing image data. The size of each field is limited to 4 GB. The binary image data is actually stored in a separate file, called a memo file, to reduce table bloat.
Integer	4-byte long integer values from -2,147,483,647 to 2,147,483,647
Logical	1-byte logical (boolean) field.
Memo	Variable length memo field containing character data. The size of each field is limited to 4 GB. The memo data is actually stored in a separate file, called a memo file, to reduce table bloat.
ModTime	Timestamp field automatically updated when a record is updated
Money	Currency data stored internally as a 64-bit integer, with 4 implied decimal digits from -922,337,203,685,477.5807 to +922,337,203,685,477.5807. The Money data type will not lose precision.
NChar	Fixed length Unicode character field stored entirely in the table.
NMemo	Variable length Unicode character data stored in the separate memo file.
Numeric	Fixed length (exact representation) numeric up to 32 bytes.
NVarChar	Variable length Unicode character data stored entirely in the table.
Raw	Fixed length data-typeless raw data field from 1 to 65,530 bytes.
RowVersion	64-bit auto-incrementing integer value
ShortInteger	2-byte short integer value from −32,767 to 32,767.
Time	4-byte value representing time of day.
TimeStamp	8-byte value representing date and time of day.
VarBinary	Variable length binary data stored entirely in the table.
VarChar	Variable length character data stored entirely in the table.

DBF table data types

AutoIncrement	4-byte read-only integer value from 1 to 4,294,967,296 unique for each record in the table.
Binary	Variable length BLOB containing binary data. The size of each field is limited to 4 GB. The binary image data is actually stored in a separate file, called a memo file, to reduce table bloat.
Character	Fixed length character field from 1 to 65,530 bytes that is stored entirely in the table.
Date	8-byte date field.
Double	8-byte IEEE floating point value in the range 1.7E+/-308 (15 digits of precision).
Image	Variable length BLOB containing image data. The size of each field is limited to 4 GB. The binary image data is actually stored in a separate file, called a memo file, to reduce table bloat.
Integer	4 byte-Integer values from -2,147,483,648 to 2,147,483,647.
Logical	1-byte logical (boolean) field.
Memo	Variable length memo field of up to 65,530 bytes. The size of each field is limited to 4 GB. The memo data is actually stored in a separate file, called a memo file, to reduce table bloat.
Money	Currency data stored internally as a 64-bit integer, with 4 implied decimal digits from -922,337,203,685,477.5807 to +922,337,203,685,477.5807. The Money data type will not lose precision.
NChar	Fixed length Unicode character field stored entirely in the table.
NMemo	Variable length Unicode character data stored in the separate memo file.
Numeric	Fixed length (exact representation) numeric up to 32 bytes.
NVarChar	Variable length Unicode character data stored entirely in the table.

TimeStamp	8-byte value representing date and time of day.
VarBinary	Variable length binary data stored entirely in the table.
VarChar	Variable length character data stored entirely in the table.

Advantage Database Server (client/server)

- Maximum number of transactions — limited by memory
- Maximum number of connections — limited by memory
- Maximum number of files opened simultaneously — limited by memory
- Maximum number of tables — limited by memory

Advantage Local Server (local and peer-to-peer)

- Maximum number of transactions — unsupported
- Maximum number of connections — limited by memory
- Maximum number of files opened simultaneously — limited by memory
- Maximum number of tables — limited by memory

Database maximums

- Maximum ADT table size
  - Windows NT/2000/XP/2003 with NTFS — 16 exabytes (18,446,744,073,709,551,616 bytes)
  - Windows NT/2000/XP/2003 with FAT32 — 4 gigabytes (4,294,967,296 bytes)
  - NetWare 5 or greater with NSS file systems — 16 exabytes (18,446,744,073,709,551,616 bytes)
  - NetWare 5 or greater with traditional file systems — 4 gigabytes (4,294,967,296 bytes)
  - Linux pre-2.1.2 — 11 glibc and pre-2.4 kernel — 2 gigabytes (2,147,483,648 bytes)
  - Linux glibc 2.1.2 — 11+ with kernel 2.4+ — 8 exabytes (9,223,372,036,854,775,807 bytes)
- Maximum DBF table size — Maximum record count (2,147,483,648) multiplied by record length (depending upon operating system and file system)
- Maximum number of records — 2.1 billion
- Maximum record length — 65,530 bytes
- Maximum field name length — 128 characters for ADT tables, 10 characters for DBF tables (dictionary-bound Visual FoxPro tables can have names up to 128 characters)
- Maximum number of columns per table — ~3,500 for ADT tables, 2,035 for DBF tables

www.AdvantageDatabase.com

SYBASE®