

## **Upgrading to SQLBase 9.0**

By Suren Behari, Product Manager – Team Developer,  
Charles McLouth, Product Manager - SQLBase and  
Martin Teetz, WW Product Manager

**September 2004**

**GUPTA™**

## Table of Contents

<b>UPGRADING TO SQLBASE 9.0 .....</b>	<b>1</b>
<b>ABSTRACT .....</b>	<b>4</b>
<b>INTRODUCTION .....</b>	<b>4</b>
<b>WHAT'S NEW IN SQLBASE 8.0 .....</b>	<b>5</b>
COM+ Transaction Support .....	5
What's New? .....	5
What does this really mean? .....	5
SQLBase Management Console .....	6
What's new? .....	6
What does this really mean? .....	7
New ODBC driver.....	7
What's new? .....	7
What does this really mean? .....	8
Updated OLE DB Provider .....	8
What's new? .....	8
What does this really mean? .....	8
Long Database Identifiers .....	8
What's new? .....	8
What does this really mean? .....	9
Finding similar sounding strings.....	9
What's new? .....	9
What does this really mean? .....	9
New Security Features .....	9
What's new? .....	9
What does this really mean? .....	9
Easy Upgrading .....	10
What's new? .....	10
What does this really mean? .....	10
<b>WHAT'S NEW IN SQLBASE 8.1 .....</b>	<b>10</b>
Full support and integration into Visual Studio .....	10
Full support and integration into Visual Studio.NET .....	10
Full integration into DTS and SQL Server Enterprise Manager.....	10

Full support of Delphi and PowerBuilder .....	10
All new .NET data provider NDP .....	11
Enhanced ODBC driver and OLE DB provider .....	11
<b>WHAT'S NEW IN SQLBASE 8.5 .....</b>	<b>11</b>
Multiple Outer Joins and ANSI SQL99 join syntax .....	11
What's new? .....	11
What does this really mean? .....	12
How does it work? .....	12
Multiple Isolated SQLBase Installations .....	14
What's new? .....	14
What does this really mean? .....	15
How does it work? A step-by-step instruction. ....	15
Overall enhanced database performance .....	21
What's new? .....	22
What does this really mean? .....	22
Updated database drivers .....	22
What's new? .....	22
How does it work? .....	22
Very smooth migration to SQLBase 8.5 .....	22
What's new? .....	23
<b>WHAT'S NEW IN SQLBASE 9.0 .....</b>	<b>23</b>
Freedom of Choice .....	23
What's New? .....	23
What does this really mean? .....	23
Linux Performance .....	24
What does this really mean? .....	24
Linux Application Support .....	24
What's New? .....	24
What does this really mean? .....	24
Smooth migration to SQLBase 9.0 and to Linux .....	24
What does this really mean? .....	24
<b>CONCLUSION .....</b>	<b>25</b>

## Abstract

This paper outlines the advantages of upgrading to SQLBase 9.0 by introducing the new features of SQLBase 9.0 as well as the features introduced in the 8.x versions of SQLBase, while providing information on what the new features really mean for existing SQLBase developers and end-users.

## Introduction

SQLBase is GUPTA's core flagship product that delivers the technology required for today's enterprise applications, be they distributed over the Internet or an Intranet, or expected to function in a highly integrated, distributed environment.

With SQLBase, companies can manage data far from the corporate data center with administration-free operation and low resource requirements. Best of all, SQLBase can significantly reduce the overall cost of developing and deploying a remote application employing an embedded database.

Through new features, critical enhancements to tried-and-true technology, adoption of emerging industry standards, and support for both the Windows and Linux Operating Systems, SQLBase 9.0 supports the evolution of your business applications.

SQLBase 9.0 will increase the potential of business applications because it:

- further **reduces maintenance** associated **cost**
- is **faster**, and offers greatly improved performance
- is **flexible**, by allowing deployment on Windows or Linux
- offers **unmatched embedding support**, to run multiple ISV applications that use SQLBase on one computer and cross-platform capabilities
- offers **painless upgrading** to SQLBase 9.0

**SQLBase 9.0 is  
40% faster  
than SQLBase  
7.5**

## What's New in SQLBase 8.0

### **COM+ Transaction Support**

#### **What's New?**

- Support for COM+ Transactions
- SQLBase Resource Manager

#### **What does this really mean?**

- By supporting COM+ Transactions, one can now integrate SQLBase into enterprise transactions. The boundaries of a transaction are not limited by a database anymore. Transactions cover complex business processes that include updating databases from different vendors and at different sites.

SQLBase's COM+ Transaction (Microsoft Transaction Server - MTS) support allows one to fully integrate SQLBase with COM+ Transactions including full commit and rollback support for overall transactions. Many programming languages including GUPTA's Team Developer can be used to create COM+ transactions. For example one combined transaction can move money from a local bank account in SQLBase to a central account in Microsoft SQL Server. The entire transaction either succeeds or if fails, is rolled back in all participating databases.

Microsoft Transaction Server (MTS) is a distributed runtime environment for COM objects that provides an infrastructure for running objects across a network. MTS is a combined object request broker (ORB), resource manager, and transaction monitor.

MTS provides automatic transaction management, database connection pooling, process isolation, automatic thread pooling, automatic object instance management, resource sharing, rolebased security, transaction monitoring within distributed applications, and more.

These services are necessary for scaling server-side components and supporting a substantial number of concurrent client requests. MTS performs all of these services automatically, and without the need for application developers to write special code. A developer can therefore develop server-side components that behave as if only a single client is connected at a time.

- SQLBase Resource Manager (SQLBrm), as the name suggests, is responsible for managing the resource - GUPTA SQLBase database. A resource manager is responsible for managing a resource that can be reversed or recovered.

SQLBrm is responsible for handling the actual commit or abort transaction with the database. The Distributed Transaction Coordinator (DTC) will get the status from the components of a transaction and if the status is to commit or abort as described above, it is the DTC will coordinate with SQLBrm and SQLBrm will do the actual commit or abort against the SQLBase database.

**Distributed  
Enterprise  
Transactions  
with SQLBase  
and COM+**

**SQLBrm is  
responsible for  
handling the  
actual commit  
or abort  
transaction  
with the  
database**

SQLBrm also handles client and server communication threads by enlisting them when they are required and de-enlisting them when they are no longer required.

If there is a shutdown in SQLBase, SQLBrm will handle the shutdown of the communication threads, it will shut down the listener, halt any active communication threads and stop all sessions.

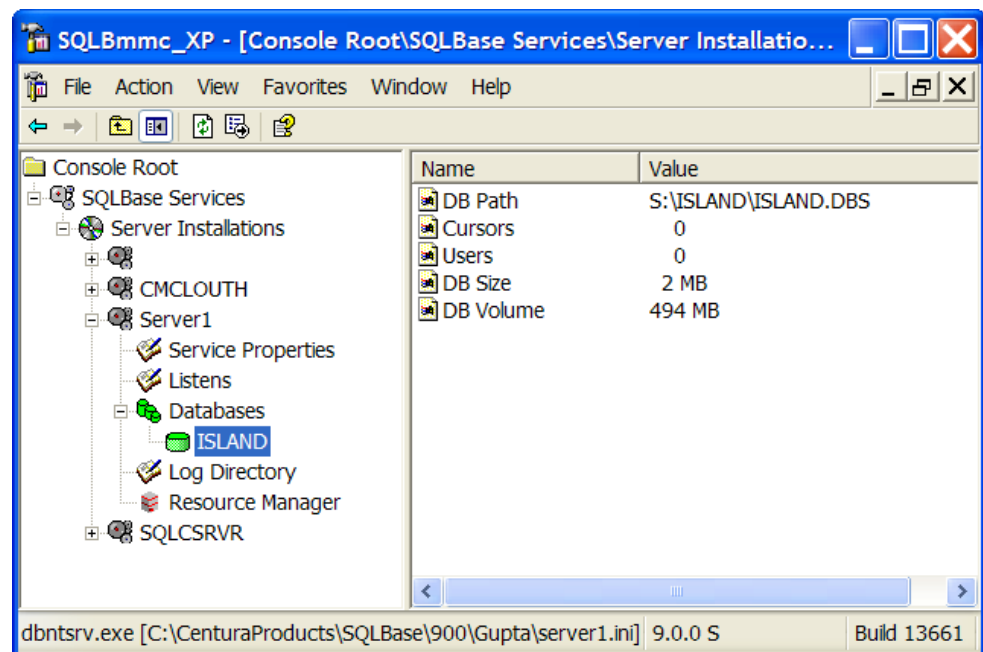
### **SQLBase Management Console**

The main purpose of the SQLBase Management Console is to replace the SQLBase Server Monitor (SSM). The new SQLBase Management Console allows one to view statistics about databases and transactions. SQLBase Management Console is a Microsoft Management Console (MMC) Snap-In and looks and behaves just like your Windows management tools.

## **The new SQLBase Management Console manages Enterprise Transactions**

The Microsoft Management Console (MMC) is a tool used to create, save, and open collections of administrative tools called consoles. Consoles contain items such as snap-ins, extension snap-ins, monitor controls, tasks, wizards, and documentation required to manage many of the hardware, software, and networking components of the Windows 2000 system. You can add items to an existing MMC console and configure them to administer a specific system component.

In SQLBase Management Console SQLBase and the new transaction service required for MTS transactions can be started and stopped. SQLBase Management Console displays detailed views of database statistics like size of the database and log files and statistics about MTS transactions that have been executed or rolled back.



### **What's new?**

- Start and stop the SQLBase Server if run as an application.
- Start and stop the SQLBase Server if run as a service (previously called an NT Service)
- Displays all installations of SQLBase Server

- Displays state, (Active or Inactive), of all SQLBase Server Installations
- Allow the user to switch between the two different installations
- Change the startup mode of SQLBase to automatic, manual, or disabled
- Change the event log level to 1, 2 or 3
- Turn the Server Console on and off
- Show the current communications protocols being used for the current installation of the SQLBase Server
- Display the SQLBase Server version number
- Display the operating system version and service pack
- Displays all database names and sizes for any installation of SQLBase Server
- Under each database node the SMC displays:
  - The path or location of this database
  - The number of cursors connected to this database
  - The number of users connected to this database
  - A statistics node that displays this database size, and the free space on the volume where this database is located
- Displays log file location and space available for any installation of SQLBase Server
- Start and stop the SQLBase Resource Manager if run as an application
- Start and stop the SQLBase Resource Manager if run as a service (previously called an NT Service)
- Change the startup mode of SQLBase Resource Manager to automatic, manual, or disabled
- Display state and statistical information about the SQLBase Resource Manager

### ***What does this really mean?***

The purpose of the SQLBase Management Console is to provide an alternative interface for starting and stopping SQLBase Server whether it is running as a service or as an application. When running as a service, SQLBase can always be started or stopped from the Services applet of the Control Panel. In a secure environment, however, the Services applet may have usage restrictions. The SQLBase Management Console allows a logged-on user to perform the same functions through a separate interface. To use the Snap-in, a logged on user must be a member of the Administrator group.

### ***New ODBC driver***

#### ***What's new?***

- Developed in-house by GUPTA Technologies
- Fully multi-threaded
- Support for most of the advanced 3.x ODBC features (bookmarks, descriptors, diagnostics, etc)
- Support for COM+ (MTS) transactions
- Ability to enable connection pooling via ODBC driver manager
- Complete backward compatibility with the 2.x driver
- SQLBase specific extensions to the driver

**The new ODBC Driver for SQLBase supports COM+ Transactions**

### ***What does this really mean?***

- Having the ODBC Driver developed in-house guarantees faster maintenance response directly from GUPTA support and engineering
- Being a multi-threaded Driver allows for greater scalability of ODBC applications
- By supporting most of the advanced 3.x ODBC features like bookmarks, descriptors and diagnostic APIs, the new ODBC Driver for SQLBase allows one to better integrate SQLBase with Visual Studio and other tools
- The support for MTS transactions allows one to access SQLBase in a distributed environment using COM+ via ODBC
- Connection pooling enables an ODBC application to reuse an existing connection from a pool, so the ODBC application does not have to go through the complete connection process for any subsequent connection
- The ODBC Driver provides support for the deprecated 2.x Driver functions, which is transparent to the application, thus allowing existing applications to work with the new driver against any version of SQLBase

### ***Updated OLE DB Provider***

#### ***What's new?***

- New Interfaces and new methods for existing Interfaces
- Fully free threaded implementation
- Support for COM+ (MTS) transactions
- Ability to take advantage of connection pooling through OLE DB Services
- Supports automatic enlistment into COM+ transactions

#### ***What does this really mean?***

- The new Interfaces were added for MTS support while the new methods provide the ability to query schema information, for example
- Being free-threaded allows for greater scalability of today's demanding applications
- The support for MTS transactions allows one to access SQLBase in a distributed environment using COM+ via OLE DB
- Connection Pooling enables an application to use a connection from a pool of connections that do not need to be reestablished for each use. Once a connection has been created and placed in a pool, an application can reuse that connection without performing the complete connection process. Any application that makes frequent calls to the data store for connections can benefit from pooling.

### ***Long Database Identifiers***

#### ***What's new?***

- SQLBase 8.0 database object names (tables, columns, etc.) can be 36 characters long
- SQLBase 8.0 database names can be 16 characters long

**SQLBase 8.0  
Database  
names can be  
up to 16  
characters  
long**

**SQLBase 8.0 helps by providing a list of similar sounding values using the new @soundex function**

**What does this really mean?**

- Descriptive database object names make it easy for end users to create ad-hoc queries and reports since it is easier to understand what the content of an object is. The maintenance of the data model is easier since new DBAs can understand the structure faster when descriptive table and column names are being used

**Finding similar sounding strings**

**What's new?**

SQLBase 8.0 provides two new functions:

- @Soundex – used for phonetic searches
- @Difference - Compares two Soundex codes and indicates how similar the two codes are to one another

**What does this really mean?**

The SOUNDEX algorithm has its basis in history aiding searches for previous generations. The algorithm was standardized in *The Art of Computer Programming, Volume 3: Sorting and Searching* by Donald E. Knuth. The algorithm allows you to compare words that are spelled differently, but sound the same in English. Note that this algorithm works less effectively in other languages, but still has value in Latin based languages.

Searching for names can be difficult when the exact writing of a name is unknown. Multinational corporations for example often have a problem finding foreign names in databases. SQLBase 8.0 helps by providing a list of similar sounding values using the new @soundex and @difference functions.

@Soundex may also be used in an index. Creating a @Soundex index may speed up selects on @Soundex. Consider this table and its index:

```
CREATE TABLE DICT (WORD VARCHAR(22), DEF VARCHAR(60));  
CREATE INDEX DICTXSOUND ON DICT (@SOUNDEX(WORD));
```

For queries like: `SELECT WORD FROM DICT WHERE @SOUNDEX(WORD) = @SOUNDEX('SMYTHE');`

**New Security Features**

**What's new?**

SQLBase 8.0 now provides a means of encrypting unload files that can be in turn loaded into an encrypted database.

**What does this really mean?**

Text files of unload data can be a security leak. Administrators are able to read sensitive corporate information like payroll information. The new secure Load/Unload feature encrypts unloaded text files using the same encryption algorithm as SQLBase TE 56/128. SQLBase 8.0 further limits unauthorized data access making it a first choice for secure systems.

**The new secure Load/Unload feature encrypts unloaded text files**

**SQLBase 8.0  
makes  
upgrading many  
installations  
very easy**

### ***Easy Upgrading***

#### ***What's new?***

The upgrade process to SQLBase 8.0 is transparent to the application and end-user that accesses it.

#### ***What does this really mean?***

Upgrading many deployed SQLBase installations by unload/load can be a complex task.

SQLBase 8.0 makes upgrading many installations very easy by avoiding any administrative action necessary to upgrade a SQLBase installation.

ISVs need to only include the new SQLBase 8.0 database files into their installation process. When an application first connects to the new SQLBase Database Engine, an automatic conversion of the database file from an older version of SQLBase to the new SQLBase 8.0 format takes place.

## **What's New in SQLBase 8.1**

### ***Full support and integration into Visual Studio***

Developers using Visual Basic 6 or other programming languages of Visual Studio 6 can fully leverage SQLBase for their database applications. Updated SQLBase ODBC and OLEDB interfaces allow for easy integration into ADO, COM+ and ASP web applications.

### ***Full support and integration into Visual Studio.NET***

SQLBase 8.1 allows you to leverage all .NET programming languages to develop business database applications. Applications can be desktop Windows or ASP.NET web applications. All Visual Studio.NET tools like data browsers and query builders can be used to access SQLBase. The SQLBase .NET data provider fully integrates into the ADO.NET class framework for database access, XML data exchange and Web Services integration.

### ***Full integration into DTS and SQL Server Enterprise Manager***

DTS stands for Data Transformation Services and is a Microsoft tool to exchange data between SQL Server and other databases like SQLBase. With SQLBase 8.1 DTS can for example be used to move data from decentral SQLBase databases to a central SQL Server database and vice versa. The updated OLEDB provider of SQLBase 8.1 also integrates SQLBase into the SQL Server Enterprise Manager. This integration allows viewing and manipulating SQLBase data from within the SQL Server Enterprise Manager a tool similar to SQLBase SQLConsole.

### ***Full support of Delphi and PowerBuilder***

Delphi integrates third party databases using OLEDB and ADO. Delphi offers several tools and classes to access databases using ADO. The Delphi TADO objects can be used to access SQLBase using the SQLBase OLEDB provider or ODBC driver. The Delphi Command Text Editor can be used to generate SQL statements that will be executed against SQLBase. The Delphi Object Tree View can display SQLBase databases,

tables and other database objects. Delphi developers now can create desktop and web applications based on SQLBase.

### **All new .NET data provider NDP**

The SQLBase NDP (.NET data provider) is the first available for embedded databases. The SQLBase NDP allows to access SQLBase from any .NET programming language like for example C#, VB.NET and others. The SQLBase NDP also integrates SQLBase into Visual Studio.NET, allowing you to leverage all VS.NET database tools, wizards and classes for SQLBase programming.

### **Enhanced ODBC driver and OLE DB provider**

The SQLBase ODBC driver now supports Type 2 driver functionality. Both the ODBC driver and the OLEDB provider have been enhanced to integrate tighter with Visual Basic 6, C++, Delphi and other programming environments. They also integrate better with third party reporting solutions like Crystal Reports.

## **What's New in SQLBase 8.5**

### **Multiple Outer Joins and ANSI SQL99 join syntax**

You have more important things to do than look after a database – that's why you choose SQLBase in the first place. Now, SQLBase 8.5 can cut your development time by supporting multiple outer-joins. Why is this important? It means the queries you write that conform to the ANSI/SQL99 standard will work for all database providers that support the standard. SQLBase 8.5 includes the full ANSI/SQL99 join syntax, including inner join, natural join, right outer join and left outer join. SQLBase 8.5 provides these new features in best SQLBase manner, lean, smart and fast. SQLBase retains its small memory requirements and still executes state-of-the-art complex queries involving the full ANSI join syntax including Multiple Outer Joins.

### **What's new?**

- Right Outer Join, Left Outer Join: Data analysis using the new outer join syntax including multiple outer joins now is very easy in SQLBase 8.5. Querying all customers, invoices and invoice items including those customers that do not have an invoice and all invoices or invoice items that do not have a customer assigned is a snap in SQLBase 8.5.
- Natural Join: The new ANSI SQL99 join syntax greatly reduces typing effort for SQL queries. "Natural Join" automatically detects the key columns that will be used for the join. No need to type the key columns in a select statement. Very important for application developers is the possibility to leverage the same SQL statements to run it against SQLBase, SQL Server or Oracle. So there is less special coding effort for applications that access different types of databases.
- Join Using: The difference to a natural join is that the join column is specified in the using clause. Again important for application developers is the possibility to leverage the same SQL statements to run it against several backends supporting SQL99 join syntax.

- Join On: "Join on" is similar to the old join syntax but lets you run the SQL against any database supporting SQL99 join syntax.

**What does this really mean?**

- Using SQL99 join syntax you can create application code that is similar for different backends like SQLBase, Oracle and SQL Server. This means less coding effort for your multi database applications.
- SQL99 join syntax includes support for multiple outer joins. Complex data analysis queries are a snap with SQLBase 8.5.
- New SQL99 join syntax can greatly reduce SQL typing effort. Join Using for example automatically detects the right join columns in joined tables.
- Some query and reporting tools like Crystal Reports 9 require SQL99 join syntax. Crystal Reports 9 generates SQL99 join syntax and sends it to SQLBase using OLEDB or ODBC database drivers, these queries fail with pre SQLBase 8.5 database versions.

**How does it work?**

Every client that interacts with SQLBase 8.5 can use the new ANSI SQL99 join syntax. To demonstrate the use, GUPTA has created several samples for use with the SQLTalk tool and the GUPTA Island sample database.

1. ANSI SQL99 Join Syntax can be enabled during installation of SQLBase 8.5 or later using the new config file keyword `ansijoinssyntax=0/1`. `ansijoinssyntax` set to 0 means SQLBase 8.5 operates in backward compatibility mode and does not allow ANSI SQL99 join syntax. You can use this mode if you do not want to modify your database objects in case they include some of the new reserved keywords of SQLBase 8.5. When `ansijoinssyntax=1` is set, you can take advantage of the new join syntax like natural join, join using and join on. The table below lists the four options for valid join syntax in SQLBase 8.5 for a simple 2 table join. The first syntax is the old SQL89 join syntax. All four select statements produce the same result, you can try them in SQLTalk running against the island sample database.

Join Type	SQL Select Statement
SQL89 Equal Join	<code>select * from company, invoice where company.company_id = invoice.company_id;</code>
SQL99 Natural Join	<code>select * from company natural join invoice;</code>
SQL99 Join Using	<code>Select * from company join invoice using (company_id);</code>
SQL99 Join On	<code>Select * from company join invoice on company.company_id = invoice.company_id;</code>

**Multiple Outer Joins give extended query and data analysis power**

2. Outer Joins are active when ANSI SQL99 join syntax is enabled in SQLBase 8.5. Your queries can contain any number and any combination of Left Outer Join and Right Outer Joins. To test the new syntax simply run the queries in SQLTalk connecting to the Island sample database. The sample use natural joins to reduce SQL typing effort.

<b>Outer Join Type</b>	<b>SQL Select Statement</b>
Left Outer Join	select company_name, cont_first_name, cont_last_name from company natural <b>left outer join</b> contact;
Right Outer Join	select company_name, cont_first_name, cont_last_name from company natural <b>right outer join</b> contact;

3. Multiple Outer Joins are any number of combinations of right outer joins and left outer joins. Below are samples based on the Island sample database. Use the queries in the table below to test the syntax in SQLTalk. Outer joins retrieve all rows from joined tables even when in the joined table no associated rows are present. For example give me all customer data and their orders including all customers that do not have orders. The order data of the customers that do not have orders is all blank (NULL).

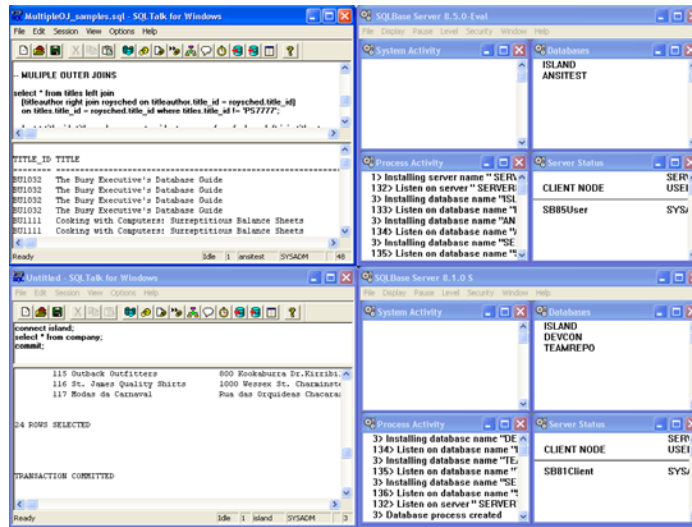
<b>Multiple Outer Join Type</b>	<b>SQL Select Statement</b>
Multiple Left Outer Join	select company.company_name, invoice_date, amount_paid, item_no, style from company natural <b>left outer join</b> invoice natural <b>left outer join</b> invoice_item;
Multiple Right Outer Join	select company.company_name, invoice_date, amount_paid, item_no, style from company natural <b>right outer join</b> invoice natural <b>right outer join</b> invoice_item;
Mixed Multiple Outer Joins	select company.company_name, invoice_date, amount_paid, item_no, style from company natural <b>left outer join</b> invoice natural <b>right outer join</b> invoice_item;

4. The advanced usage of Multiple Outer Joins is Nested Multiple Outer Joins. You can use brackets to group joined tables in the way to obtain your desired query results. The outer joins in brackets are executed first and can be seen as a temporary table, which then will be joined with the tables outside the brackets for the final query results.

<b>Nested Multiple Outer Join</b>	<b>SQL Select Statement</b>

	<pre>select company.company_name, invoice_date, amount_paid, item_no, style from company natural left outer join ( invoice natural right outer join invoice_item );</pre>
	<pre>select company.company_name, invoice_date, amount_paid, item_no, style from ( company natural right outer join invoice ) natural left outer join invoice_item;</pre>

### Multiple Isolated SQLBase Installations



**SQLBase 8.5 introduces easy and low support application deployment**

Cut your support time with easier installation: You already know that SQLBase is the world leader in embedded databases and because of its popularity, you may have run into some installation problems where your customer has another application using a different version of

SQLBase. For example users of some of the most popular payroll applications, business and personal finance or parcel shipping software (all have SQLBase embedded in them) may have created an installation and support problem for you. But not anymore! Now you can install your application that embeds SQLBase 8.5 without any configuration issues. Your application embedding SQLBase 8.5 can run in complete isolation from an existing older SQLBase installation, without any SQL.INI or SQLBase client/server run-time file conflicts. Your application embedding SQLBase can have a configuration file named anything like "yourinfile.ini". There is no need for a SQL.INI file anymore. SQLBase 8.5 client and server can be forced to use a specified configuration file. All SQLBase drivers like JDBC, OLEDB, ODBC, .NET Data Provider and the SQLBase API support generic configuration files. SQLBase 8.5 makes sure your application can be deployed and used successfully even when the target system has other SQLBase installations (yes, even previous versions!). This unique feature further reduces SQLBase Total Cost of Ownership since application installation support decreases quite a bit.

#### What's new?

- Multiple SQLBase 8.5 running at the same time on the same machine
- One SQLBase pre 8.5 and SQLBase 8.5 running at the same time on the same machine

- Complete SQLBase client isolation
- Complete SQLBase server isolation
- Generic configuration file names (My.Config.File, etc.)

### ***What does this really mean?***

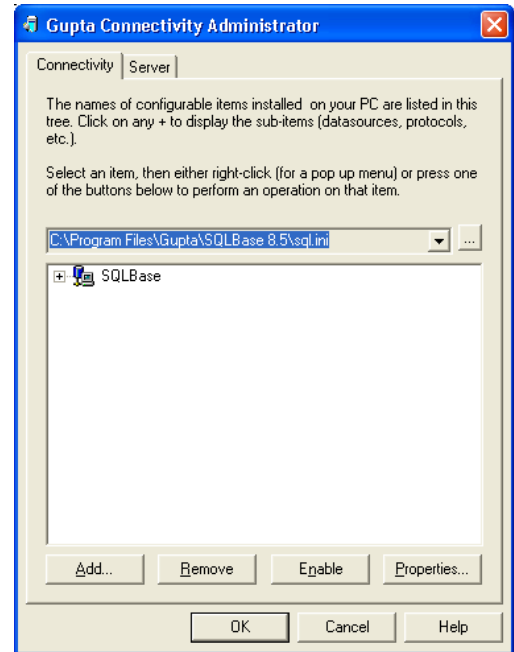
- Painless configuration of multiple ISV applications that use SQLBase. One application using an older release of SQLBase and one or more new applications using SQLBase 8.5 can run at the same time on one machine. For customers of your vertical applications that means less installation and configuration problems and more sales opportunities for your applications.
- Create custom installers for your vertical applications including SQLBase that allows you to run your application in complete isolation from other SQLBase installations on one computer at a customer site.

### ***How does it work? A step-by-step instruction.***

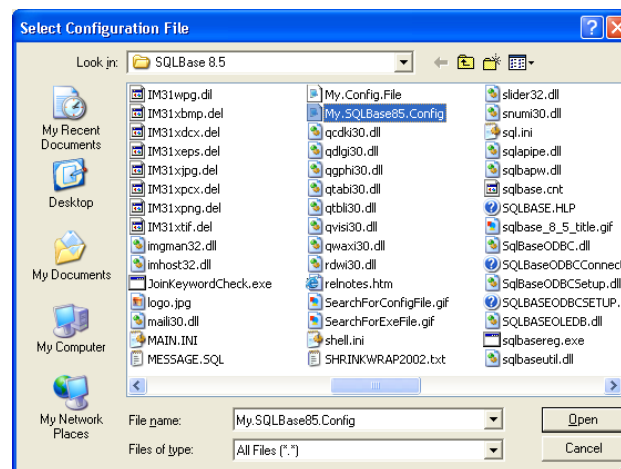
Given the scenario that you already have a SQLBase 8.1 or older SQLBase version on your computer that has been installed by a business software package, you now want to add another software application that should use SQLBase 8.5. And both software programs, the one using SQLBase 8.1 or older and the one using SQLBase 8.5 should run at the same time on this computer without any conflicting configuration settings. In the following scenario the new software package for SQLBase 8.5 is SQLTalk. We use SQLTalk just as an example how your application could work in this new flexible multiple installations scenario.

1. Install SQLBase 8.5 on a machine that already has an older SQLBase product installed. After the installation of SQLBase 8.5 you have two directories containing a SQLBase installation. The installer of SQLBase 8.5 does not allow you to install over an existing older SQLBase installation. During installation, the installer checks whether the server name you would like to use is already used by another server. If the server name is already in use you will be asked to enter a different server name. When the installer asks whether it should start SQLBase after the installation check no, because you need to change several settings before you can get going.
2. Use the control panel to remove the path from the new SQLBase 8.5 installation. By default the path of the SQLBase 8.5 server is "c:\Program Files\Gupta\SQLBase 8.5". This way only applications that are located in the SQLBase 8.5 directory will use this configuration; all other SQLBase applications will use the old existing installation. In this sample we will use SQLTalk as our client application.
3. After the installation of SQLBase 8.5 has finished rename "sql.ini" of the SQLBase 8.5 installation to "My.SQLBase85.Config".

4. Start the SQLBase Connectivity Administrator from the start menu. Select the button to the right of the sql.ini path to change the configuration file name.

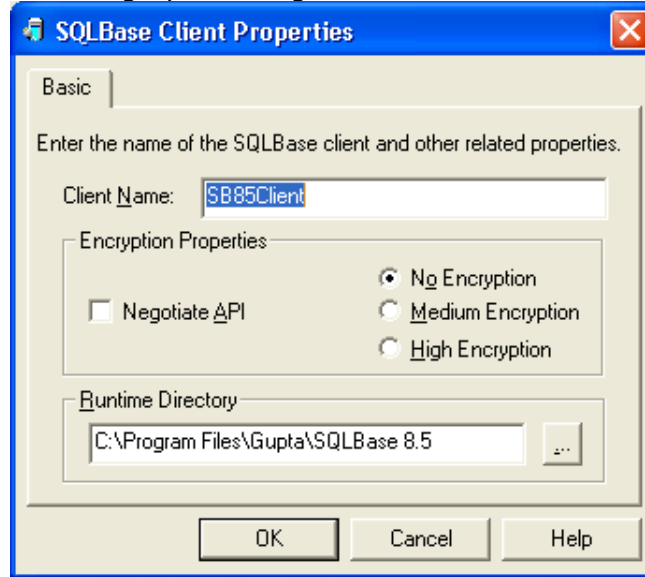


5. Select the new configuration file name My.SQLBase85.Config in the Select Configuration File Dialog and press Open.

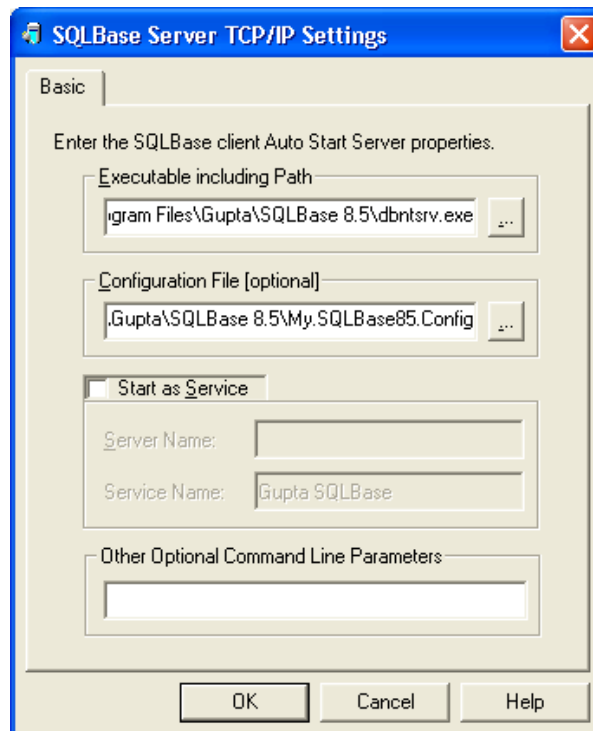


**Configuring multiple installations can be done with a few mouse clicks**

6. Select SQLBase in the Connectivity Administrator tree list box and press the properties button. This dialog let's you define a client name that has already been set by the SQLBase 8.5 installer, you do not need to change anything in this dialog. You can use the data field for the Runtime Directory to specify the location of the SQLBase 8.5 runtime DLL's. You can close this dialog by selecting cancel.

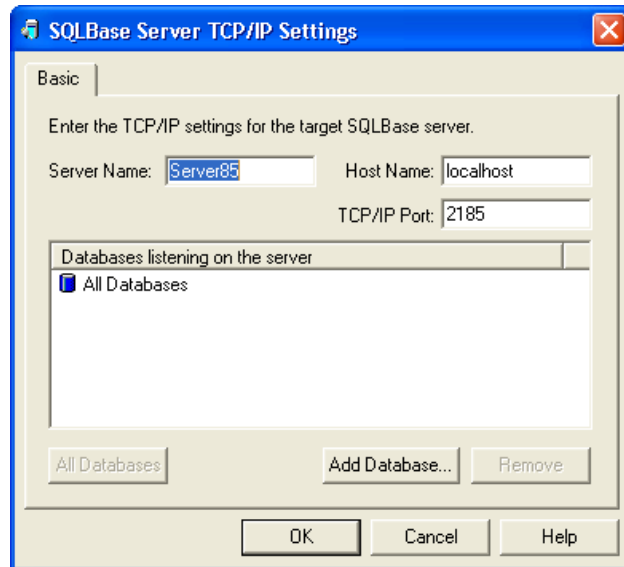


7. Now select TCP/IP in the tree list box and then press the properties button again. The TCP/IP Server Settings dialog opens. This dialog is used to set the SQLBase autostart file name and optional configuration file name that should be used when the server is autostarted. Please note that autostart now also works for TCP/IP SQLWS32 protocol when used for local connections. The first data field in this dialog should already be filled in by installer provided settings. By pressing on the button



to the right of the configuration file name field a new dialog opens that let's you select a new configuration file. Select the new configuration file My.SQLBase85.Config and press the open button. Now you are almost done with your new configuration.

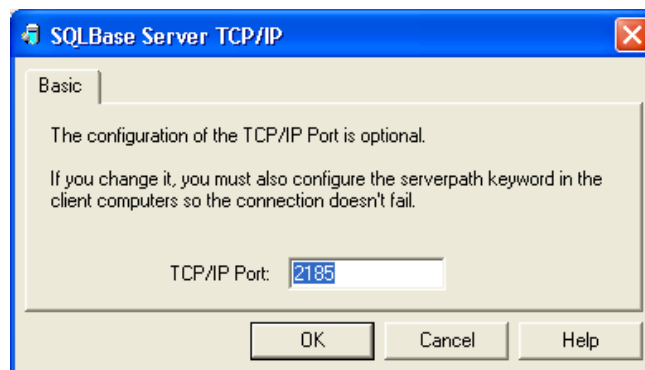
- The next step in the Connectivity Administrator is to specify a different TCP/IP port for the new SQLBase 8.5 installation. The existing SQLBase installation uses 2155, the new installation



needs to use a different value for client and server. To specify a new IP port select the line of your server name that shows when you expand the TCP/IP protocol in the tree list box and press the properties button. The

dialog to your right opens and you can specify a new port. In this case you can use 2185 for the SQLBase client. Be sure to use the same port number for the server as well. The next step shows how you can set the server listen port.

- Now for the last step in the Connectivity Administrator change to the server tab of the Connectivity Administrator. The server tab shows a list of available SQLBase servers including your older SQLBase installation. Expand the new SQLBase 8.5 server

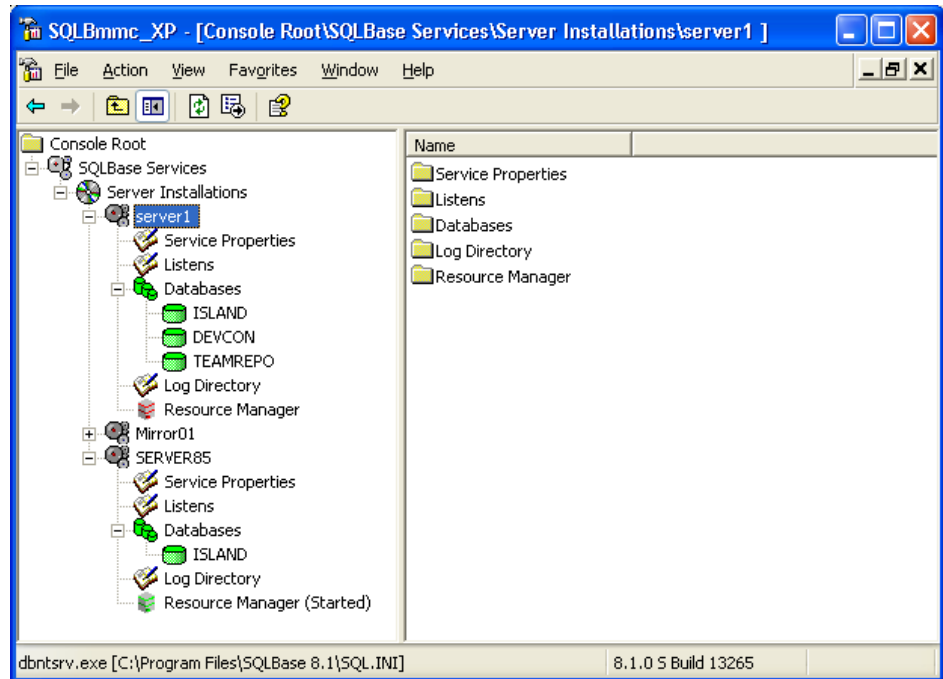


installation and then expand the Listening Protocols. Now select the TCP/IP protocol line and press the properties button. Fill in the new

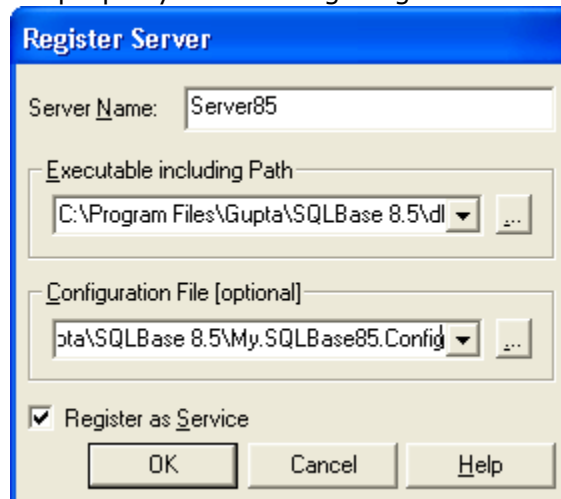
TCP/IP port number 2185 to make client and server port equal. Now you are done with Connectivity Administrator. Press OK and then again OK in the Connectivity Administrator server tab.

- The final steps are to configure the new configuration file name in the SQLBase Management Console. This is necessary so that SQLBase will use the new configuration file name even when started as a service. To do so, start the SQLBase Management Console from the Windows start menu. You will see a Window similar to the one displayed below.

- Expand the SQLBase Services entry and then also expand the Server Installations entry. You should see the server names of both of your SQLBase server installations. Move your mouse over the SQLBase 8.5 installation and do a right click with your mouse. In the menu select Unregister the server.



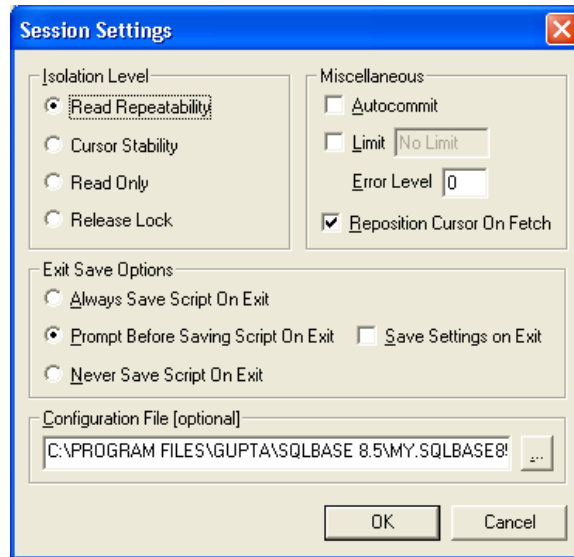
- Select the Server Installations entry in the tree list on the left pane of the SQLBase Management Console and open the property menu doing a right click on it. From the menu select



Register to create a new entry for the SQLBase 8.5 installation including a reference to the new configuration file. The buttons to the right of the executable path and the configuration file let you easily set the server executable name and the new

configuration file that should be used. Select your server executable file. The file name depends on the user level you have installed. Now you are ready to complete your installation by specifying your new configuration file name My.SQLBase85.Config. Press OK to save the changes. Now you are ready for prime time. You can now close the SQLBase Management Console.

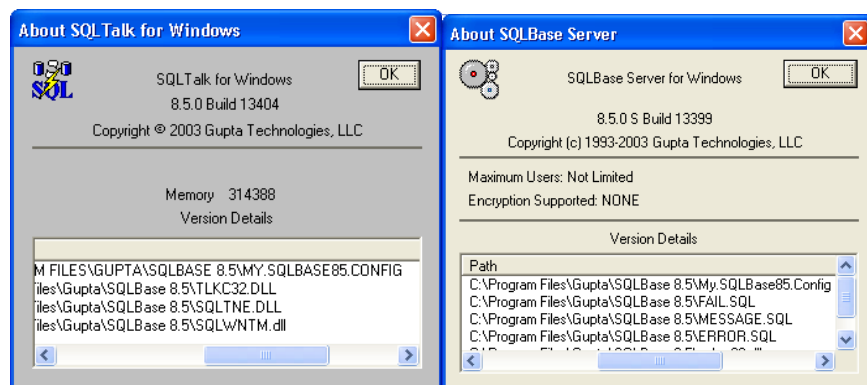
13. To test your new configuration, just start SQLTalk from the start menu. Before connecting to a database you need to tell SQLTalk



to use a different configuration file. In the Options menu select session settings. The session settings dialog let's you specify the newly created configuration file. To do so just click on the button to the right of the configuration file data field. In the dialog that

opens select My.SQLBase85.Config as your configuration file and press ok. In the Session Settings dialog you can now check the Save Settings on Exit checkbox to make sure SQLTalk remembers to use your new configuration file. Now you can press OK and you are back in the SQLTalk main window. In the upper pane just type "connect island;" and execute by ctrl-enter. SQLBase 8.5 will start automatically using your new configuration settings. Congratulations! You are successfully using the new multiple installations feature of SQLBase 8.5. Now you could also start your older SQLBase installation and use your application to connect to it. Both SQLBase installations will run and operate well in complete isolation.

14. GUPTA has provided tools to verify what configuration file a SQLBase 8.5 server or SQLTalk are using. When you select "About" in the Help menu of SQLBase 8.5 or SQLTalk 8.5 the programs display the exact configuration file including path that is being used. The About window also displays path and file names of the DLL's that have been loaded by the program. This makes it easy for developer's and database administrators to manage and test multiple installations.



15. All the steps above can also be done manually by editing your configuration file using a text editor like notepad. Following is the complete listing of My.SQLBase85.Config that has been created by the above configuration process. The important SQLBase 8.5 related lines are marked in bold font. Please refer to the SQLBase 8.5 manuals for detailed descriptions of the new configuration file entries.

```
[win32client.comments]
[win32client]
clientname=SB85Config
clientruntimeidir="C:\Program Files\Gupta\SQLBase 8.5"
```

```
[win32client.dll]
comdll=sqlws32
;comdll=sqlapipe
;comdll=sqlwsspx
;comdll=sqlspx32
```

```
[win32client.apipe]
autostartserverpath="C:\Program Files\Gupta\SQLBase 8.5\dbntsrv.exe"
serverpath=Server85
```

```
[win32client.ws32]
autostartserverpath="C:\Program Files\Gupta\SQLBase 8.5\dbntsrv.exe" "INI=C:\Program Files\Gupta\SQLBase 8.5\My.SQLBase85.Config"
serverpath=Server85,localhost,2185/*
```

```
[dbntsrv]
```

```
ANSIJoinSyntax=1
```

```
dbdir=C:\Program Files\Gupta\SQLBase 8.5
dbname=Island,sqlws32
servername=Server85,sqlws32
[dbntsrv.dll]
comdll=sqlws32
```

```
[dbntsrv.ws32]
listenport=2185
```

### ***Overall enhanced database performance***

Have you been praised by your users lately? By installing SQLBase 8.5 you will greatly increase the user satisfaction level. Complex long running queries will be accelerated by the much improved sort cache allocation algorithm and new memory allocation in SQLBase 8.5. So get ready for some very good user feedback after you have installed SQLBase 8.5.

We've also improved the SQLBase Lock Manager to minimize conflicts like timeouts and deadlocks between reading and writing processes accessing one database. This improves the overall application performance and behavior for multi-user installations.

**SQL.INI has been enhanced with several new keywords like clientruntimeidir and autostart-serverpath**

### **What's new?**

- Overall Performance up by 10%, 40% compared to SQLBase 7.5.1
- Sort Performance dramatically increased
- Timeout and Deadlock Improvements further increase multi-user performance

### **What does this really mean?**

If you are migrating from SQLBase 7.5 you will immediately notice a 40% increase in overall performance and if migrating from 8.1 you will notice a 10% increase in overall performance. Queries that perform Sorts (ORDER BY, GROUP BY), in most cases, will perform more than 25 times faster (+2500%) and in some circumstances up to 800 times faster (+80000%.) Additionally, transactions failing as the result of a lock contention are reduced by more than 30%.

Thus, by upgrading to SQLBase 8.5, you can be "praised by your users" for dramatically increasing performance of individual tasks, increase throughput, and enhance the user experience by reducing lock contentions.

### **Updated database drivers**

All SQLBase database drivers have been updated to support the new ANSI SQL99 join syntax. You can continue to use your favorite development environment like Visual Studio, Visual Studio.NET, Delphi or Java IDE's to create applications that use SQLBase and take full advantage of the new join syntax including multiple outer joins.

### **What's new?**

- The SQLBase 8.5 .NET Data Provider now includes support for scrollable result sets and ODBC escape sequences.
- The updated OLE DB provider includes support for ODBC escape sequences.
- Creating Java apps for SQLBase is getting a lot easier. The SQLBase 8.5 JDBC driver has been updated to contain many more methods of the JDBC 1.2 API and the JDBC 2 API to enhance support for J2EE.

### **How does it work?**

The updated SQLBase 8.5 drivers will be installed during SQLBase 8.5 installation. The drivers are available for download from the GUPTA website. Look for the SQLBase Driver Pack in product downloads.

### **Very smooth migration to SQLBase 8.5**

Most customers choose SQLBase because they have more important things to do than look after a database. GUPTA is very committed to steadily improve Total Cost of Ownership of SQLBase. Many things that are usually done by a database administrator are completely automated in SQLBase. In SQLBase 8.5 GUPTA is going one step further in keeping the customer effort for database version migration as low as possible. This is one important factor of the ultra low Total Cost of Ownership SQLBase offers.

- For a very smooth migration to SQLBase 8.5 you can switch off using the new ANSI SQL99 join syntax and run in backward compatibility mode so you don't need to make any changes to your existing databases.
- Upgrading large SQLBase installations using SQLBase 8.5 is very easy. SQLBase 8.5 automatically converts existing older SQLBase databases to the latest database format. Existing installations can be as old as SQLBase 6.1 or later for SQLBase 8.5 to auto convert them. The automatic database conversion saves huge amounts of maintenance work usually necessary when upgrading to a newer database release. SQLBase continues to make your life easier.

**What's new?**

- Automatic migration of existing SQLBase databases to SQLBase 8.5 format.
- Backward compatibility mode allowing you to switch off the new ANSI SQL99 join syntax and avoid new keyword name conflicts.

**What's New in SQLBase 9.0**

GUPTA is committed to providing our customers the tools and technologies necessary to compete successfully in delivering cost effective applications quickly and easily without platform limitations. SQLBase 9.0 fulfills this commitment by providing a database solution that can be deployed on Linux as well as Windows.

**Freedom of Choice**

SQLBase 9.0 extends its features of "Zero admin, low maintenance database", "Easy embedding into vertical solutions", "Scalable", and "High Performance" by providing "Freedom of Choice by" with flexible cross platform deployment on Linux or Windows

**What's New?**

- SQLBase 9.0 for Linux supports most distributions of the Linux Operating System including:
  - Red Hat Desktop 3.0
  - Red Hat Enterprise Linux WS 3.0
  - Red Hat Enterprise Linux ES 3.0
  - SUSE Linux 9.0
  - SUSE Linux 9.1 Professional
  - SUSE Linux Enterprise Server 8
  - SUSE Linux Enterprise Server 9
  - SUSE Linux Desktop 1.0

**What does this really mean?**

- SQLBase 9.0 for Linux opens new markets for your vertical applications
- SQLBase 9.0 for Linux creates new opportunities for software houses that develop individual software applications
- SQLBase 9.0 for Linux creates new opportunities for enterprise business solution deployment
- Lower Total Cost of Ownership by combining the low TCO of Linux with the low TCO of SQLBase to increase your bottom lines.

**Flexible cross platform deployment on Linux or Windows**

## **SQLBase 9.0 for Linux outperforms Windows by 15%**

### **Linux Performance**

GUPTA's testing of SQLBase 9.0 for Linux shows that it outperforms SQLBase 9.0 for Windows (and all prior versions of SQLBase for Windows) by 15%.

#### **What does this really mean?**

If you are migrating from any prior version of SQLBase to SQLBase 9.0 for Linux, you will immediately notice a 15% performance increase. Primarily due to the efficiency of the file systems supported by Linux, you will notice a greater impact during disk intensive activities such as LOAD, UNLOAD, UPDATE STATISTICS, & CHECK DATABASE operations

## **CAPI, ODBC, & JDBC available natively for Linux**

### **Linux Application Support**

SQLBase 9.0 extends SQLBase's exhaustive connectivity options (CAPI, ODBC, .Net, OLEDB, & JDBC Driver) for the windows platform by providing native connectivity options for Linux.

#### **What's New?**

- CAPI for Linux
- ODBC for Linux (supporting iODBC & unixODBC Driver Managers)
- Type 4 JDBC Driver
- SQLTalk for Linux

#### **What does this really mean?**

- High performing SQLBase applications can be created to run in Linux shells utilizing SQLBase's CAPI.
- Web developers can now create web applications on Linux via PHP connecting to a local SQLBase database on Linux or a remote SQLBase database (Linux or Windows.)
- Routine tasks can be written and performed in Perl (or other scripts supporting ODBC.)
- SQLTalk for Linux allows system administrators and developers to perform ad-hoc queries and scripts against SQLBase without leaving the comfort of a Linux shell.

## **SQLBase automatically converts existing databases to SQLBase 9.0 format.**

### **Smooth migration to SQLBase 9.0 and to Linux**

Most customers choose SQLBase because they have more important things to do than watch after a database. SQLBase 9.0 offers auto-migration from later database releases to make deployment of SQLBase as easy as possible. SQLBase automatically converts existing databases to SQLBase 9.0 format. Additionally, the SQLBase file format between Windows and Linux is 100% compatible yielding complete flexibility.

#### **What does this really mean?**

- Existing SQLBase databases can be conveniently migrated to 9.0, merely by installing 9.0, copying the existing database, and connecting to it.
- Since SQLBase's file format is compatible between Windows and Linux, databases can be moved from one platform to another merely by copying the database file.

## Conclusion

SQLBase 9.0 is the fulfillment of GUPTA's commitment to providing customers the tools and technologies necessary to compete successfully in delivering cost effective applications quickly and easily without platform limitations.

SQLBase 9.0 brings large benefits to customers deploying vertical applications by allowing them to target new consumer markets or gain a competitive edge with cross-platform deployment.

SQLBase 9.0 also provides IT departments the "Freedom of Choice" to choose the platform that is the most cost effective for their organization.

Copyright © 2004 Gupta Technologies LLC. GUPTA, the GUPTA logo, and all GUPTA products are licensed or registered trademarks of Gupta Technologies, LLC. All other products are trademarks or registered trademarks of their respective owners. All rights reserved.