

Developing Something for Nothing with SQL Server

**A Closer Look at SQL Server Express and How it
Can Work for You**

*Written by
Jeremiah Peschka
SQL Server Developer
HMB Information System Developers*



White Paper

**© 2009 Quest Software, Inc.
ALL RIGHTS RESERVED.**

This document contains proprietary information, protected by copyright. No part of this document may be reproduced or transmitted for any purpose other than the reader's personal use without the written permission of Quest Software, Inc.

WARRANTY

The information contained in this document is subject to change without notice. Quest Software makes no warranty of any kind with respect to this information. QUEST SOFTWARE SPECIFICALLY DISCLAIMS THE IMPLIED WARRANTY OF THE MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Quest Software shall not be liable for any direct, indirect, incidental, consequential, or other damage alleged in connection with the furnishing or use of this information.

TRADEMARKS

All trademarks and registered trademarks used in this guide are property of their respective owners.

World Headquarters
5 Polaris Way
Aliso Viejo, CA 92656
www.quest.com
e-mail: info@quest.com

Please refer to our Web site (www.quest.com) for regional and international office information.

Updated—July 2009

CONTENTS

- INTRODUCTION 1**
- GETTING STARTED WITH FREE OR BUNDLED SOFTWARE..... 2**
 - FREE SOFTWARE 2
 - Freemium Software* 2
 - THE RISKS OF FREE SOFTWARE 3
 - Lack of Support* 3
 - No Continuals Updates* 3
 - Legal Issues* 3
 - THE LIMITATIONS OF BUNDLED DATABASE SOFTWARE 4
 - Scalability* 4
 - Transactional Integrity* 4
 - Concurrency* 4
 - Backups* 4
- SQL SERVER: EXCEPTIONAL FREEMIUM SOFTWARE 5**
 - SQL SERVER EXPRESS EDITION 5
 - ADDRESSING THE LIMITATIONS OF BUNDLED DATABASES 5
 - Scalability* 5
 - Transactional Integrity* 5
 - Concurrency* 6
 - Backups* 6
 - LIMITATIONS OF SQL SERVER EXPRESS EDITION 6
 - CPU Limitations* 6
 - Memory Limitations* 6
 - Size Limitations* 7
 - SQL Server Agent* 7
- WORKING WITHIN THE LIMITATIONS OF SQL SERVER EXPRESS EDITION 8**
 - DEALING WITH DATATYPES 8
 - Example* 8
 - HISTORICAL DATA 9
 - SCALE OUT, NOT UP 9
 - SQL SERVER AGENT 9
- MOVING UP 10**
 - CHOOSING THE RIGHT EDITION 10
 - THE UPGRADE PROCEDURE 10
- CONCLUSION 11**
- ABOUT THE AUTHOR 11**
- ABOUT QUEST SOFTWARE, INC. 12**
 - CONTACTING QUEST SOFTWARE 12
 - CONTACTING QUEST SUPPORT 12

INTRODUCTION

Software development is an expensive and time-consuming process, not just in terms of time and salaries but also in terms of software and hardware costs. It can be tempting to pick a widely available free solution with the intent of migrating to an enterprise solution in the future. However, it is important to consider potential issues before they arise. Will the free solution scale? Does the free solution have a support agreement? Will it be as easy as possible to continue to grow your systems beyond the original development server?

This paper explains how SQL Server resolves these questions: it is available in a free edition that can easily scale to enterprise levels.

GETTING STARTED WITH FREE OR BUNDLED SOFTWARE

Free Software

It is very easy to download free software, set it up, and begin developing a project. The most tempting aspect of free software, of course, is that there's no cost to deploy it in both development and production. In fact, developers frequently install the same version of the software on their machines that is used in production.

Freemium Software

Some software is described as "freemium." Wikipedia defines freemium as "a business model which works by offering basic services for free, while charging a premium for advanced or special features." This business model has proven to be a very successful way of making money while still delivering business value and good will to customers who only require basic features.

Some examples of the freemium business model include:

- Basecamp from 37 Signals—This project management software is offered in many different monthly plans to meet a variety of business needs.
- Remember the Milk—Basic list-making services are free, while premium services include mobile device software and premium support.
- Last.fm—This music recommendation service is free, but paying subscribers can create unlimited personal radio stations, enjoy an ad-free browsing experience, and access beta features before they reach the public.

The freemium business model extends beyond the web to the desktop, with software like Trillian and MusicMatch Jukebox (which has been a freemium offering since 1999). And freemium software can even extend to the server and database.

The Risks of Free Software

For all of their benefits, free software products do carry various risks for those that download and use them.

Lack of Support

Many free software offerings lack any form of support from the vendor or developer unless you pay for the freemium offering, if one is available. Support is often available through third-party vendors, but the quality and availability of that support is difficult to verify.

No Continuals Updates

Software purchased from major vendors typically has a definite lifecycle. As a product nears the end of its life, the vendor will typically inform customers and present a plan for an alternative product offering. Free software, on the other hand, comes with no guarantee of how long the software will remain in active development and no promise that a future alternative will be available as a free offering.

Legal Issues

Some types of free software incur a potential for legal difficulties. Discussion of legal issues is beyond the scope of this document; however, it is important to consider that there may be legal risks, depending on how closely the software and your development project are coupled.

There are many different licensing models for free software. Some models require that:

- A copy of the license be redistributed with the compiled software
- The original source be distributed
- Any changes to the source code be distributed—or made available—with the finished product

In this last case, there is potential that any proprietary changes to the free software would have to be distributed.

The Limitations of Bundled Database Software

Free software is frequently intended to replace a large, commercial software product—in the case of database software, free software would be replacing SQL Server. Because free software meets different business needs than freemium or commercial software, its developers often make fundamentally different design decisions that result in the lack of important features: bundled database software is frequently intended to serve the needs of business users, rather than replace a large commercial database server.

Scalability

Some free software offerings are included in a larger suite of software and serve a niche purpose: to make it easy to implement single-user desktop solutions. These applications are capable of scaling in a limited sense, but they tend to break down when attempting to scale beyond a handful of users.

Transactional Integrity

Transactional integrity (the ability of a database to consistently isolate changes) is vital to the proper functioning of a relational database management system (RDBMS). This functionality is built into all commercial offerings and many open source offerings. Bundled database solutions, however, frequently forgo transactional integrity in the database and move the management to the application layer. This means that your developers have to manage transactions in data access code or potentially even implement and maintain transactional support.

Concurrency

Concurrency refers to a database's ability to handle multiple users accessing the same data. While modern database systems have a large number of options for concurrency control built in, desktop-based offerings do not have as rich of a concurrency control mechanism built in, frequently leaving you to develop your own solutions.

Backups

Database backups are critical for maintaining data integrity and recovering from disasters. Most commercial and free database systems offer a variety of ways to create database backups. However, desktop databases frequently require direct interaction through menus to create a database backup and do not support any automation of database backups. Users often turn to a third-party backup solution, but this approach presents one problem: with a desktop database, a single user can perform an operation that locks a database file, preventing the backup solutions from accessing the file and creating a backup. Commercial database backups avoid this problem through their support for concurrency.

SQL SERVER: EXCEPTIONAL FREEMIUM SOFTWARE

SQL Server Express Edition

SQL Server Express Edition is Microsoft's free version of SQL Server. It comes bundled with SQL Server Management Studio Express, a free version of Microsoft's tool for managing and developing SQL Server. SQL Server Express Edition is a full-featured version of the SQL Server storage engine (also referred to as the database engine or the OLTP engine) that can be readily deployed on desktops and servers alike.

Once a database has outgrown the limitations of SQL Server Express Edition, there is a clear cut and easy upgrade path to move to SQL Server Standard or Enterprise Edition.

Addressing the Limitations of Bundled Databases

SQL Server Express Edition overcomes the limitations of bundled database software discussed above.

Scalability

Scalability is important when moving beyond a single user desktop application and expanding into a large, multi-user application. SQL Server Express Edition provides support for scalability by easily allowing multiple concurrent connections. In addition, it is possible to run multiple instances of SQL Server Express Edition on the same computer—making it possible to run multiple applications across many instances of SQL Server Express Edition, as well as spreading the application load across multiple instances of SQL Server Express Edition.

Transactional Integrity

SQL Server Express Edition uses the same core database engine as SQL Server Standard Edition and SQL Server Enterprise Edition. It provides full support for a variety of levels of transactional integrity needed for different data access scenarios. Transactional support is not only built in at the server level; it is also a feature of the SQL language itself. SQL Server provides advanced support for nested transactions and can even take advantage of distributed transactions spanning multiple servers, if the Distributed Transaction Coordinator has been properly configured.

Concurrency

SQL Server was designed from the ground up to give multiple users the ability to access the same data. By default, SQL Server works to eliminate read/write blocking that can degrade application response times. SQL Server also supports *pessimistic concurrency control*: data being read can be blocked from being modified until the read has completed. In addition, multiple transaction isolation levels are available to specify the exact type of concurrency control desired.

Backups

SQL Server Express Edition fully supports the variety of backup options available in the commercial versions of SQL Server. Administrators can back up the entire database at once, as well as create differential backups and log backups.

Limitations of SQL Server Express Edition

SQL Server Express Edition does have limitations that affect development and deployment. If any of these limitations become a problem, you can easily upgrade to a premium edition of SQL Server, as explained later in this document.

CPU Limitations

SQL Server Express Edition is limited to one CPU per instance of SQL Server Express Edition. Microsoft's licensing for SQL Server defines a CPU as a single physical CPU, regardless of the number of logical processors or cores contained in that CPU. For example, a server with one quad core CPU would be considered to have one CPU for the purpose of licensing SQL Server.

Memory Limitations

SQL Server Express Edition is limited to one gigabyte of memory per instance. That is, regardless of the amount of memory on the server, SQL Server Express Edition will use only up to one gigabyte of memory. Multiple copies of SQL Server Express Edition can be installed on the same server using the *named instances* feature, and each will only use one gigabyte of memory. This approach, called "sharding", is discussed later in the section detailing upgrade paths from SQL Server Express Edition.

The memory cap on SQL Server Express Edition can rapidly become a performance bottleneck, so it is wise to monitor memory use when using SQL Server Express Edition in a production environment.

Size Limitations

SQL Server Express Edition also limits the physical size of the database to four gigabytes. This limitation is per database, not per server. Therefore, when designing tables and indexes, you need to consider what kind of data will be stored in each column and what the precision of that data needs to be. Can you get away with a SMALLDATETIME instead of a DATETIME? Will a TINYINT work instead of an INT? These design considerations can dramatically impact the size of the database.

SQL Server Agent

Finally, SQL Server Express Edition does not ship with SQL Server Agent, a Windows service for scheduling and executing regular tasks in SQL Server. SQL Server Agent is important for maintaining a healthy SQL Server infrastructure throughout the enterprise datacenter. Of course, there are ways to overcome this limitation.

WORKING WITHIN THE LIMITATIONS OF SQL SERVER EXPRESS EDITION

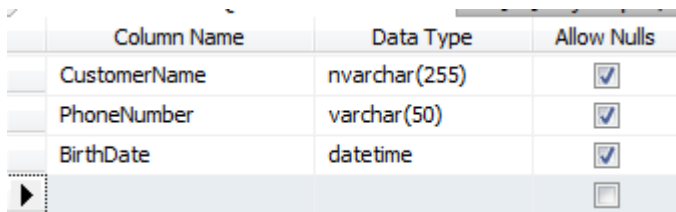
The limitations of SQL Server Express Edition should not prevent anyone from deploying an application to production. If anything, the limitations of SQL Server Express Edition should encourage database architects and developers to use querying, storage, and physical database design best practices.

Dealing with Datatypes

Regardless of which database system you're using, it is important to carefully consider the datatypes used in a column, but the four gigabyte physical database size restriction and one gigabyte memory restriction of SQL Server Express Edition make these design decisions even more important. Choosing the wrong datatype can have drastic effects on storage within the database.

Example

Suppose an application needs to store customer information: customer name, telephone number, and date of birth. It might be tempting to create these columns like this:



Column Name	Data Type	Allow Nulls
CustomerName	nvarchar(255)	<input checked="" type="checkbox"/>
PhoneNumber	varchar(50)	<input checked="" type="checkbox"/>
BirthDate	datetime	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

However, if the application will not be used for data that requires Unicode character sets, up to 255 bytes of storage will be wasted for each row in this table. In addition, the PhoneNumber column might need to be only 10 characters wide, or could be stored defined as an integer to save additional space. Likewise, rather than using a *datetime* to store BirthDate, *smalldatetime* or even *date* could be used to save on the number of bytes taken up by each row.

Being careful about your use of data types will not only enable an application to remain on SQL Server Express Edition longer before reaching the four gigabyte storage limit; it also will have a positive effect on application performance because indexes will be smaller and will have faster response times.

Historical Data

Many applications do not need to keep historical data ready for immediate retrieval; even if users need to occasionally review historical data, the response time does not need to be instantaneous. In this situation, you can reduce the storage pressure on your production server by moving historical information to a separate storage mechanism, such as another database on the same server, a separate server, or some other storage medium altogether.

You can also save space by taking into account the required granularity of the historical data. In some cases, you can store summarized historical data in a separate database and maintain detailed historical data offline for record-keeping or auditing purposes. Careful planning can result in tremendous storage savings.

Scale Out, Not Up

It is also possible to split database load across multiple servers using a technique known as sharding. In effect, routing software in the application layer determines where to store data in different databases based on some pre-determined criteria (e.g., the last two digits of a Social Security number). Sharding balances the load of an application across multiple servers and splits the CPU, memory and disk requirements across multiple servers.

However, sharding requires that a separate piece of software be developed and maintained by staff developers. Sharding is a fairly well known and established technique; however it is a specialized area of knowledge and implementation details are not widely available.

SQL Server Agent

SQL Server Express Edition does not include SQL Server Agent, and SQL Server Management Studio Express Edition is unable to create or manage SQL Agent jobs. It is certainly possible to create a reasonable facsimile of SQL Server Agent using a combination of scheduled tasks, batch files and PowerShell, and command-line SQL tools. However, these solutions tend to be brittle and are potentially troublesome for other DBAs and system administrators to use.

Denny Cherry, a SQL Server MVP, has created a standalone SQL agent that can be downloaded and installed from his website at <http://itknowledgeexchange.techtarget.com/sql-server/tag/standalone-sql-agent/>. While it is not integrated with SQL Server, it is a well-documented solution that can be understood by DBAs and Windows system administrators alike.

MOVING UP

At some point, SQL Server Express Edition may no longer be sufficient to meet the needs of your application: your storage needs may increase beyond four gigabytes, sharding the data may not be a viable option, or developing custom solutions to overcome the solution's limitations may be overly expensive. Luckily multiple upgrade paths are available for moving from SQL Server Express Edition to either the Standard Edition or the Enterprise Edition.

Choosing the Right Edition

Several editions of SQL Server are available: Web, Workgroup, Standard, and Enterprise. Each has different features and different limitations. A full discussion of each edition is beyond the scope of this document, but Microsoft provides a comparison chart (available at <http://www.microsoft.com/sqlserver/2008/en/us/editions-compare.aspx>) to help you choose the right edition for your needs.

The Upgrade Procedure

Moving from SQL Server Express Edition to a different version of SQL Server can be simple. You need to:

1. Back up the databases.
2. Uninstall SQL Server Express Edition.
3. Install the new edition of SQL Server.
4. Restore the original databases.
5. Continue working as if nothing happened.

CONCLUSION

SQL Server is a powerful enterprise-level database engine that can meet a variety of business needs, from small departmental applications to enterprise-level applications. Despite its limitations, the free SQL Server Express Edition can often meet initial development and deployment needs if database best practices are followed.

ABOUT THE AUTHOR

Jeremiah Peschka is a SQL Server developer with HMB Information System Developers in Columbus, OH. Jeremiah has broad IT experience in systems administration, web design, application development, and database development, spread across diverse platforms. Currently, he specializes in developing solutions based on the SQL Server platform. Jeremiah is currently the Group Leader for CBusPASS, the Columbus chapter of PASS, and the Group Leader for the PASS Application Development Virtual Chapter.

Learn more about Jeremiah at the following:

- Blog: <http://facility9.com>
- Twitter: <http://twitter.com/peschkaj>

ABOUT QUEST SOFTWARE, INC.

Quest Software, Inc., a leading enterprise systems management vendor, delivers innovative products that help organizations get more performance and productivity from their applications, databases, Windows infrastructure and virtual environments. Through a deep expertise in IT operations and a continued focus on what works best, Quest helps more than 100,000 customers worldwide meet higher expectations for enterprise IT. Quest Software helps organizations deliver, manage and control complex database environments through award-winning products for Oracle, SQL Server, IBM DB2, Sybase and MySQL. Quest Software can be found in offices around the globe and at www.quest.com.

Contacting Quest Software

Phone: 949.754.8000 (United States and Canada)
Email: info@quest.com
Mail: Quest Software, Inc.
World Headquarters
5 Polaris Way
Aliso Viejo, CA 92656
USA
Web site: www.quest.com

Please refer to our Web site for regional and international office information.

Contacting Quest Support

Quest Support is available to customers who have a trial version of a Quest product or who have purchased a commercial version and have a valid maintenance contract. Quest Support provides around the clock coverage with SupportLink, our web self-service. Visit SupportLink at <http://support.quest.com>

From SupportLink, you can do the following:

- Quickly find thousands of solutions (Knowledgebase articles/documents).
- Download patches and upgrades.
- Seek help from a Support engineer.
- Log and update your case, and check its status.

View the ***Global Support Guide*** for a detailed explanation of support programs, online services, contact information, and policy and procedures. The guide is available at: [http://support.quest.com/pdfs/Global Support Guide.pdf](http://support.quest.com/pdfs/Global%20Support%20Guide.pdf)