

The Road from Access to SQL Server

Paul Litwin
Lead Programmer, Collaborative Data Services
plitwin@fhcrc.org

CDS Seminars & Training Classes

- ◆ CDS Brownbag seminars
 - June 14, Upsizing Access to SQL Server, Paul Litwin
 - July 12, Introduction to ASP.NET, Paul Litwin
 - Aug 9, Life Made Easier w/ Regular Expressions, John Syre
 - Sep 13, Hacked: Understanding and Preventing Malicious Input Attacks, Paul Litwin
- ◆ Training classes
 - July 20-Aug 2, 8:30-12:30, ASP.NET with C# Class, Paul Litwin
 - SQL Server for Access Developers, sometime in Sept/Oct timeframe
- ◆ Send me email (plitwin) to get on announcement list for seminars/classes

2

Me.About

- ◆ **Developer**
 - Focus: ASP.NET, ASP, VB, C#, SQL Server, Access, ...
 - Microsoft ASP.NET MVP, MCSD
 - Based in Seattle, WA
 - After 11 years on my own, I recently took a position as Lead Programmer at Fred Hutchinson Cancer Research Center in Collaborative Data Services (CDS)
- ◆ **Co-Founder and Owner**
 - Deep Training
 - www.deeptraining.com
- ◆ **Conference Chair/Speaker/User Group Leader**
 - Chair, Microsoft ASP.NET Connections
 - Speaker at Tech*Ed, VSLive, DevDays
 - Member INETA Speakers Bureau
 - Web Developer Meeting Leader, .NET Developer's Assn
- ◆ **Author**
 - Author/co-author of a dozen books, including...
 - Five editions of Access Developer Handbooks
 - ASP.NET for Developers
 - Access Cookbook

3

ASP.NET with C# Master Class, Version 1.9d (01/04)

©2002-2004 Deep Training/Litwin Consulting, Inc. All Rights Reserved Worldwide. Unauthorized duplication is strictly prohibited.

Genesis of this Talk

- ◆ Cancer Epidemiology Research Cooperative (CERC) needed to upsize a number of Microsoft Access applications to SQL Server
- ◆ Collaborative Data Services (CDS) was asked to perform the upsizing

4

Agenda

- ◆ Why Upsize?
- ◆ Upsizing the Data
- ◆ Upsizing the Application
- ◆ Tools & Recommendations

5

Should You Upsize?

- ◆ Reasons to Upsize
 - Scalability
 - Security
 - Robustness
 - Interoperability
 - ◆ Why Not Upsize?
 - Access is very accessible
 - Package deal (data, forms, reports, code, etc. all in one convenient place)
 - Cost of programming/training
 - Cost of SQL Server
 - Personnel (need a DBA)
- } CDS may be able to help here

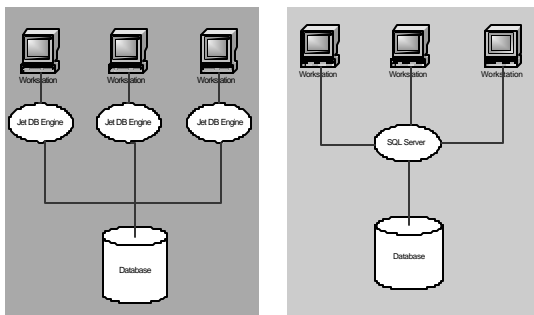
6

Scalability

- ◆ Access by default employs a file-server design
 - Each user instantiates a copy of Jet database engine
 - Performance adversely affected when going beyond a dozen or so users
- ◆ Access can also function as a front-end to SQL Server
 - Client/server design
 - Most database work is offloaded to SQL Server database engine
 - Scales much better

7

File-Server vs. Client/Server



File-Server

Client/Server ₈

Security

- ◆ Medical records need to be kept private (HIPPA)
- ◆ Access has ok security
 - Difficult to apply correctly
 - Many users leave unsecure backup copies around, etc.
 - Can still be cracked as long as you can get to the database (e.g., www.lostpassword.com)
- ◆ SQL Server has much better security
 - Users can be prevented from having direct access to database file
 - Supports Windows integrated security

9

Robustness

- ◆ File-server design of Access makes it easy to corrupt an Access database
 - Every user runs his/her own personal copy of Jet engine that is reading/writing data
- ◆ SQL Server much more robust
 - Only one instance of the SQL Server database engine can read/write data
 - Supports clustering, failover, hot backups
- ◆ Because of Access file-server design it should never be used as the database behind a Web site

10

Interoperability

- ◆ SQL Server data is more accessible; can be accessed from
 - Microsoft Access
 - VB, V C++ 6.0 apps
 - .NET apps
 - Web apps
 - ASP, ASP.NET, PHP, JSP, Java, etc.

11

Upsizing the Data

- ◆ Options
 - Use Access Upsizing Wizard
 - Use Data Transformation Services (DTS)
 - Rebuild Schema from Scratch and then transfer data using DTS

12

Access Upsizing Wizard

- ◆ Pros
 - One stop friendly upsizing
 - Upsizes:
 - Tables
 - Primary keys
 - Foreign keys (relationships)
 - Validation rules
- ◆ Cons
 - If any problems are encountered in upsizing schema or data it just skips the table

13

Data Transformation Services

- ◆ Pros
 - Import Wizard plus more sophisticated DTS package designer
 - Upsizes tables nicely
 - Can transform data as it's upsized
 - Can log errors
- ◆ Cons
 - Does not upsize:
 - Primary & foreign keys
 - Validation rules

14

Rebuilding Schema from Scratch

- ◆ Pros
 - Upsizing effort represents a good opportunity to redesign database
 - Rethink design
 - Normalize design
 - Rename fields/tables to more SQL Server friendly names
- ◆ Cons
 - Requires a lot more work
 - Chance that you'll miss something

15

**Recommendations
Upsizing Data (1 of 2)**

- ◆ If you have the time/money and plan on using the app for some time
 1. Redesign database from scratch
 2. Use DTS to import data

16

**Recommendations
Upsizing Data (2 of 2)**

- ◆ Don't quite have the time/money for redesign
 1. Create Empty SQL Server DB
 2. Use Access Import Wizard to upsize schema only to DB
 3. Use DTS Import/Export Wizard to upsize data but save package & don't schedule to run immediately
 4. Go to DTS node in Enterprise Mng, open package, select Package|Properties and enter an error file name under Logging tab
 5. Execute package
 6. Examine error log for errors. Fix if necessary and redo

17

DTS Upsizing Tip

- ◆ When using DTS Import/Export Wizard you may encounter errors because of wrong sequence of tables containing foreign keys
- ◆ This is easy to fix
- ◆ Just re-run wizard including only tables with foreign key errors
- ◆ Should re-execute with no errors

18

Application Upsizing Options

- ◆ Access MDB linked to SQL Server
- ◆ Access ADP linked to SQL Server
- ◆ .NET desktop application
- ◆ ASP.NET web application
- ◆ Other desktop/web application

19

Access MDB Linked to SQL Server

- ◆ Least impact
- ◆ Upsize data, redirect links to SQL Server and fix up potential problems
- ◆ More flexibility than ADP solution
- ◆ Potential issues
 - Code that uses OpenDatabase
 - Complex queries or queries with VBA functions
 - Queries/forms/reports returning large number of records (> several thousand)
 - Separate date & time fields
 - Unique indexes with nulls

20

Access Data Project (ADP) Linked to SQL Server

- ◆ Can improve performance over MDB solution because Jet Engine is no longer involved in data access
 - Uses ADO/OLEDB instead of DAO/ODC/Jet
- ◆ Potential issues (> than MDB solution)
 - All stuff from last slide plus...
 - All DAO code will have to be recoded as ADO code
 - Can't be used when you need local Access tables
 - no local Access data store
 - Can't be used when you need to place tables in multiple SQL Server databases
 - only points to one SQL Server database

21

MSDE

- ◆ Free version of SQL Server
 - No enterprise manager, query analyzer
 - Limited replication/analysis services, etc.
 - 2GB max db size
 - Workload governor actually slows down SQL Server if 8 concurrent connections exceeded
 - Still requires a DBA in all but most trivial cases
 - Usually not recommended

22

.NET Desktop/Web Application

- ◆ These options will require most work
- ◆ If you are considering recoding to move to ADP solution you should really consider complete rewrite in .NET instead
 - Better suited for future
 - Potentially significantly better performance
 - Can create rich desktop app or thin browser-based app
 - Does require .NET expertise, however

23

Comparison of App Upsizing Options

Option	Pros	Cons
Upsized MDB	Easy; more flexible than ADP	Old technology; may be slower
ADP	Can perform better than MDB	More recoding; not as flexible
.NET Desktop	New technology; rich desktop UI	Big effort; needs to be installed
.NET Web App	Fast modern; thin Web UI; no install issues	Big effort; big shift in UI for some users

24

Recommendations Upsizing App (1 of 2)

- ◆ Have the time/effort for rewrite of application
 - ASP.NET (VB .NET or C#) for thin, browser-based app with no install issues
 - Windows Forms app (VB .NET or C#) for richer desktop app

25

Recommendations Upsizing App (2 of 2)

- ◆ Only have time for minimal upsize
 - MDB application with linked SQL Server tables

26

CDS Upsizing Tools

- ◆ I've created a few tools you may find useful if going the MDB route
 - ItemizeLinks.mdb
 - useful if you have apps with links to more than one back-end Access database
 - DSNStripper
 - Converts DSN-dependent links to DSN-less links
 - Makes deployment easier for linked upsized MDB apps
 - AccessCheck
 - VB 6 app that checks if Access database is most up-to-date, and if not, copies newer version before opening database

27

Recommended Tool

- ◆ SPEED Ferret
 - Find and replace utility that works with Access 97-2002, VB 5/6, SQL Server
 - Helps you to safely make changes to object (table, field, etc.) names
 - <http://www.moshannon.com>
 - Costs \$199

28

Conclusions

- ◆ Upsizing Access-only apps to SQL Server is a good idea for improving scalability, security, robustness, & interoperability
- ◆ Use a combination of Upsizing Wizard & DTS to upsize data
- ◆ Linked MDB usually preferable to ADP
- ◆ If you can afford it, consider rewriting app as ASP.NET or Windows Desktop app in VB .NET or C#

29

Thank You

- ◆ Paul Litwin, Collaborative Data Services
 - plitwin@fhcrc.org
 - x3426
- ◆ Slides and utilities
 - download from www.deeptraining.com/fhrcr

30
