

USING MICROSOFT ACCESS FOR GREATER EFFICIENCY

What's Inside

INTRODUCTION	2
CHAPTER 1 - COMMON USES FOR DATABASE APPLICATIONS	3
CHAPTER 2 - WAYS MICROSOFT ACCESS CAN IMPROVE YOUR PRODUCTIVITY	5
CHAPTER 3 - DECIDING IF ACCESS IS RIGHT FOR YOU	7
CHAPTER 4 - WORKING WITH MICROSOFT ACCESS	9
CHAPTER 5 - COMMON SURPRISES, MISUNDERSTANDINGS, AND FRUSTRATIONS	16
CHAPTER 5 - MICROSOFT ACCESS LIMITATIONS AND MISCONCEPTIONS	17
CHAPTER 6 - HOW TO DEPLOY YOUR ACCESS SOLUTION TO END USERS	18
CHAPTER 7 - HOW TO EVOLVE YOUR DESKTOP ACCESS APPLICATION	23
CHAPTER 8 - RESOURCES TO HELP YOU	24

Introduction

For nearly 20 years Microsoft Access has been a simple, relatively easy solution for tracking and maintaining information. I emphasize the word “relatively” because it isn’t without frustration, and requires time to learn. But compared to other, more complicated business application tools, Access is fast to work with, nearly every business Windows machine has it installed, and with little effort can save tens to hundreds of hours of mundane labor. Access is not a utility knife you will want to apply to every problem, but it can be incredibly useful where rapid, intuitive solutions are needed, when developers are committed to larger projects, and you have the interest to go deeper into creating software tools without necessarily becoming a programmer. This book will help you understand when (and when not) to use Access, what types of Access applications you can create, and how to get the most out of Access to make you much more productive.

Chapter 1 - Common Uses for Database Applications

Database applications are pervasive today, we use them or rely on them in nearly every business or commercial interaction whether we know it or not. They store, organize, and supply information very efficiently and minimize the possibility of duplication and erroneous data (that presumes, of course, that the database is well designed). Despite being widespread, they are not widely understood.

There are numerous ways to store, organize, and later retrieve information – from a paper sticky note to an Excel spreadsheet to a full database application. In this eBook the aim will be to help you understand when a database like Microsoft Access will give you more value (even with some extra effort up front) when compared to other methods. To begin with, let's look at some common scenarios and whether a database can provide you with more value than another method like an Excel spreadsheet:

Scenario	Most Effective Method
Maintaining a list of phone numbers	Spreadsheet
Maintaining a list of person contact information	Spreadsheet
Maintaining customers and associated contacts	Database
Maintaining a list of products and prices	Spreadsheet
Maintaining lists of products and prices that vary by location, customer, or some other second element	Database
Maintaining orders	Database
Maintaining a list of projects	Spreadsheet
Maintaining projects, tasks, invoices, etc.	Database
Maintaining sales forecasts for multiple territories or reps	Database

What you've likely noticed from the few examples above is that a spreadsheet is a solid way to track simple lists that are not likely to contain duplicate information. Lists that you can use to lookup information quickly, but are not closely associated with other things. A database is most useful in cases where the information you need to manage, organize, and report on is actually a collection of many things (projects, tasks, invoices). Databases are also indispensable where you need to consolidate and calculate data from many sources, such as a sales forecast that comes from multiple people in the organization.

Commonly, organizations begin with a spreadsheet tool like Excel to handle a job best suited for a database. And the reasons are clear, Excel is relatively easy to understand, lets you calculate, summarize, and visualize your data. Excel is installed on nearly everyone's Windows machine, can be opened on most mobile devices, and is essentially a free-form data management tool. If you are reading

this, however, it's likely you have started to see that Excel's capabilities are stretched and not always 100% reliable for your needs. In this next section, we'll look at how a database application built using Microsoft Access can make you and your organization more productive, and allow you to focus on more important matters.

Chapter 2 - Ways Microsoft Access Can Improve Your Productivity

Maintaining information in Excel (or shudder, on paper) may be familiar and fast to create or change, but the drawbacks can have a significant impact to your operations. Moving to Access as a solution to manage information centrally and even automate repetitive tasks will result in a solid time and financial benefit to your organization. Here are some of the key ways Access can make you more productive:

Save time fixing errors and recovering from inconsistent information

With one source of information, you minimize duplication, errors, and inconsistent values. In spreadsheets values are repeated many times, so a misspelling, empty cell, or inconsistent ways of referring to the same thing (customer, product, service, item, etc.) can mean trouble when you are running a search, creating graphs, or trying to calculate your data. Microsoft Access can help you control user input not only by making many fields a dropdown selection rather than free form text, but also by limiting the types of information a user can enter (e.g., only dates or numbers), as well as making some fields required before information is saved. Keeping your information consistent and error free takes minutes if not hours that you could be doing other more important work.

Eliminate the need to consolidate and rationalize different information

If you store your information on paper or spreadsheets, it's likely you have multiple people providing you with the information. Copies of paper forms or separate spreadsheets requires work to consolidate the information into a single source. If the incoming information is in different formats, it compounds the problem making you need to spend time rationalizing the different formats into one consistent format. Without it, you can't create reports and analyze the information.

Faster access to the latest information

You and others working in your database need only change a single record and all other related things in your database will automatically "see" the change. In Excel, you would need to change every occurrence of a duplicate values. In a database, there is only one customer record to change. Every other record (orders, contacts, etcetera) that relates to that customer record will never need to change because they are joined together by way of automatically assigned numeric "keys" that represent the relationship between them, rather than something that may change over time, such as a name or address. Because your database is the single source of truth, you and your users also have the advantage of fast access to information. You don't need to hunt for the latest information, it's available in the shared application.

Reporting and analysis

While basic information tracking is generally the goal of a process, it's important not to forget the immense value that comes from being able to analyze and learn from the information available to you. If your data is in spreadsheets or on paper, it will be incredibly difficult to find out how long it takes to complete a specific task, how many orders you received this year, or answer other important questions. Here, the productivity gain comes from the insight and potential to improve your processes, products, or other activities to be more effective overall.

Automate repetitive processes

Microsoft Access has the potential, with some additional effort, to automate routine tasks that consume time and require little human intervention. For example, mailing a report every day to key people, export data to a place that other programs can use the information, or help you [collect information via email](#).

Chapter 3 - Deciding if Access is Right For You

There are two major considerations when deciding whether Access is right for your situation. The first is, assuming you will be the primary person to do the work, your own appetite for learning to work with Access. The second consideration is whether Access' capabilities align well with your project requirements.

Assessing Your Appetite to Learn

While Microsoft Access has been available worldwide since 1995 (nearly 20 years at the time of this writing), the concept of databases and designing screens and reports is quite foreign to most. So despite years of refinement and improvement, Access still requires a level of patience and willingness to learn that not all prospective users are prepared to commit. After working with Access for nearly 20 years, and working with many new users, it is fair to say that you should expect no less than 2 to 4 hours of solid, dedicated learning by trial and error, frustration, and ultimately - reward. Besides the gratification of mastering a new challenge, the productivity gains are often even greater.

Matching Your Project Requirements to a Solution

Requirement	Spreadsheets	Microsoft Access	Microsoft SQL, MySQL Server, or other	Applicable to your Project?
User interface	Grid	Forms	Requires PHP, .NET, Java, or other programming to create we UI	
Validate basic user input (e.g., for a single field/cell)	Yes	Yes	Yes	
Complex user input validation or events (if field X = Y, and field Z is empty, perform some action)	Challenging to accomplish	Yes	Yes	
Data arranged in a few columns without much repetition	No	Yes	Yes	
Rows/records supported	Excel 2000 and Excel 2003: 65,535 rows Excel 2007 and Excel 2010: 1,048,576 rows	Unlimited (up to 2GB/table)	Unlimited	

Requirement	Spreadsheets	Microsoft Access	Microsoft SQL, MySQL Server, or other	Applicable to your Project?
Simple cross-references	Yes (vlookup and hlookup)	Yes	Yes	
Complex data analysis/queries	Limited	Yes	Yes	
Events based on user actions (when user exits a field/cell do X, when they mouse over a field/cell do Y)	Limited	Yes	Yes	
Multiple users at the same time	No	Yes	Yes	
Frequent mass-update operations	Challenging to accomplish, time intensive	Yes	Yes	
Multiple data entry forms	No	Yes	Yes	
Mail Merge with Word	Yes	Yes	Yes	
Reporting	Challenging to accomplish	Yes	Yes	
Automated backups	Challenging to accomplish	Possible with VB coding, backup software, or UI Builder for Microsoft Access	Yes	
Record-level auditing	No	Possible with VB coding or UI Builder for Microsoft Access	Yes	
Ability to hide the inner workings of your project from the user	Challenging to accomplish	Yes	Yes	
Expertise Required	Low	For simple databases: Low/Moderate For complex requirements: Moderate-to-High	High	
Attaching files/links to fields	Limited	Yes (MS Access 2007 and Access 2010)	Yes	
Sharing information on the Web	Yes (Excel 2007 and Excel 2010)	No ¹	Yes (with a web front-end like PHP or ASP.NET)	

¹ Microsoft discontinued Access Web app support as of April 2018.

Chapter 4 - Working with Microsoft Access

What to Expect

As mentioned earlier, there is an unavoidable learning curve when staring out with Microsoft Access. The same can be said of course of using a spreadsheet application as well, and if you have a level of mastery with Excel, you will find many concepts translate into Access. But entering into your project with the expectation of some small level of healthy frustration as you learn how things work will serve you well in the end. Rather than abandoning your project, you can simply plan to spend a few hours watching YouTube videos, or if you prefer, playing around in Access using trial and error.

For those of you wondering how learning and working with Access compares to Excel, or more complex solutions, this continuum should help:

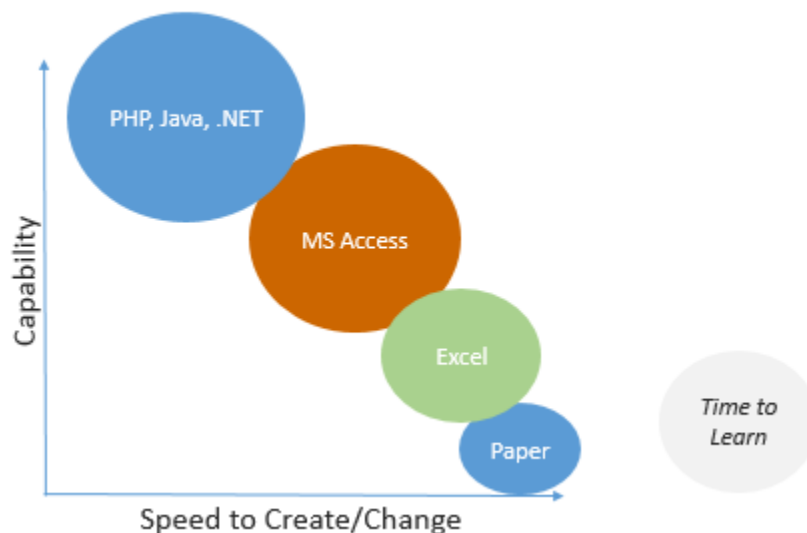


Figure 1: Solution Continuum

The key message is patience will be rewarded with a tangible amount of time savings in the end. Investing a few hours to learn and create your application can yield days and weeks of time saved.

What to Expect – for Excel power-users

If you don't consider yourself an Excel power-user, feel free to skip to the next section. But if you are an Excel wizard, there are a few suggestions here to help make your life easier with Access.

1. In Excel everything is presented in a “flat” layout with information organized in columns. In Access, this is possible too. But it isn't the way information is stored. Expect to have your information separated into multiple tables (think of them as separate tabs in Excel, almost). You

can bring everything into on single grid view using a Query. But as you design your database, think about what things are related closely (e.g., Customer name, phone number, headquarters address) and what things are related but really separate (e.g., orders, contacts). If the information is directly about the thing you are working with, it probably belongs in the same table. If there could be multiple things related to each other, it likely belongs in a separate table.

2. Fields in a table are relatively similar to a column in Excel. At the top of most Excel columns you put the name of the information that will be displayed below. In Access you have a field name, and then everything in that field will store that type of information.
3. Be careful renaming things. In Excel renaming a column header rarely has a big impact. Renaming a tab even has little impact. In Access, renaming a field or table can have big downstream impacts. That doesn't mean avoid renaming, but be aware that if you had a field "Customer Name" and decide to change it to "CustomerName" then you'll need to locate all the places (forms, queries, reports) that were accustomed to the old name, and update them to the new name.

When to Use Templates, a Design Tool, or Start from Scratch

Over the past two decades, hundreds of templates have been created and made available for Access. Microsoft maintains a small library, and you can locate many other MS Access templates online (both free and for sale). There are few design tools available for Access, OpenGate Software's Designer for Access being the primary one that facilitates the design process. The remaining option is to begin a database from nothing.

Following the common theme of this book, the right decision for you will largely depend on your desire to spend time learning how to design a database and entry screens. Spending the time to learn how databases are designed can be very beneficial if you are in a technical industry, or have interest in deepening your understanding of computer science. The need to build skills in software applications is only growing more important, so the time invested is rarely wasted in the long run. So starting a new database using only the Access design interface can be an excellent way to learn using a real-world problem to solve for your organization. But if you lack the time or patience, using a design tool like Designer for Access or downloading a template that most closely matches your project needs will help you get started much faster, and reduce the learning curve required.

Access Navigation and Design Concepts

Once you learn your way around Access, you will find that has many similarities to other Office tools.

Below are a few basic navigation actions to know:

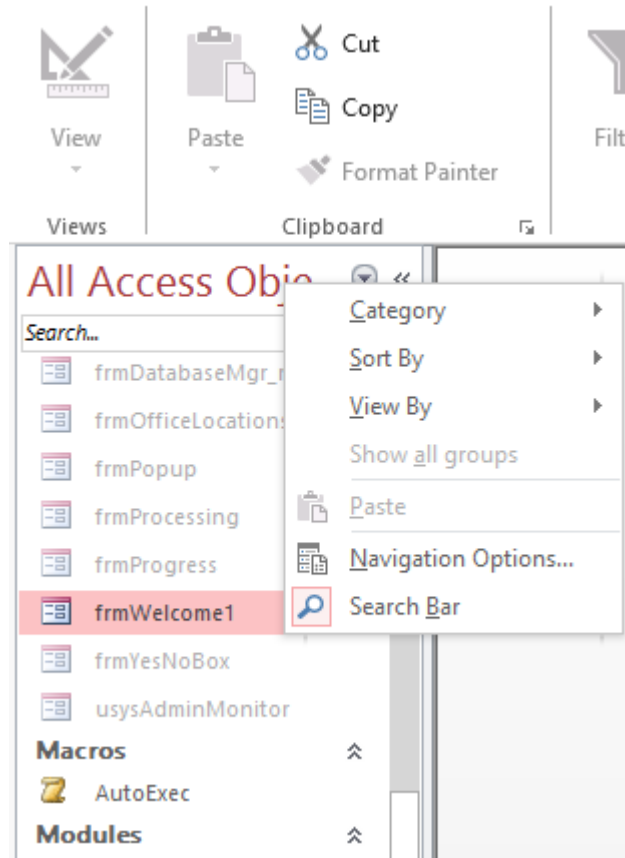
Navigation Pane

The navigation pane in Access 2007 and above is the way to get to all of your forms, reports, tables, and other objects. You can sort, search, and manage all of the different items in your database using the navigation pane.

Tip: Right-click the top of the navigation pane to see more advanced options.

Caution: Be careful renaming things unless you understand what other items are looking for something with that name. If you rename a form from “My Form” to “MyForm”, and a macro or other form is looking for the “My Form” you will get error messages.

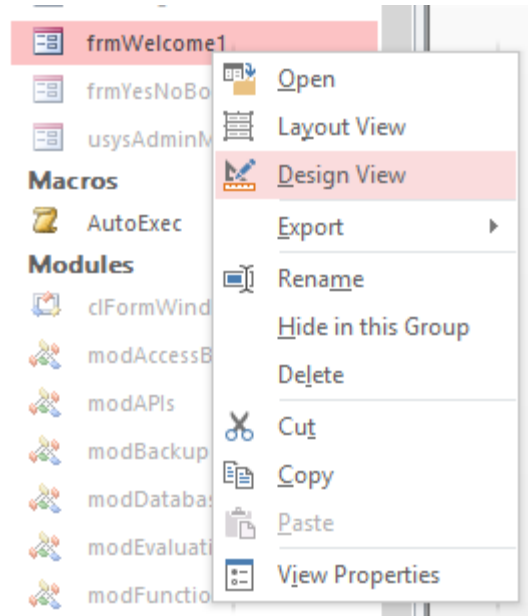
Tip: Items that are grayed out can be hidden using the “Navigation Options...”. And anything starting with “Msys” is a part of Access that every database contains. You won’t be able to delete them or change them. But they won’t hurt anything.



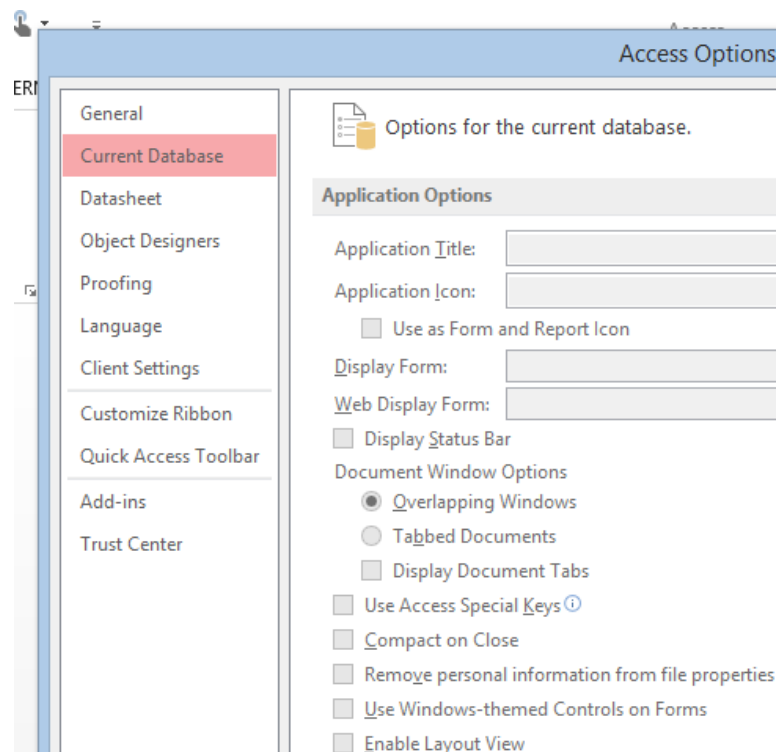
Design View

Right-click any item in the navigation pane and you'll see an option for Design View. Use the Design View for making changes to anything in Access.

Tip: Layout View is similar to Design View, but it allows you to view your data and make simple layout changes at the same time. Perfect for instances where you want to make sure a field is sized correctly.



Options View



Types of Objects

There are several major concepts related to the things you work with in Access. At the highest level, you have Tables, Queries, Forms, Reports, Macros, and Modules. Tables store the actual data. Queries let you join Tables, filter data from one or more tables, and do advanced calculations on the data. Forms

let you see data from Tables or Queries in a friendly interface, and edit records. Reports do nearly the same, but they are formatted more for printing rather than on screen viewing. Macros let you automate tasks and make Access perform actions like opening a form or printing a report when clicked. Lastly, Modules let you add Visual Basic programming to your applications.

Tip: In Excel, a Macro is really Visual Basic automatically generated by Excel after recording steps you perform. In Access, there is no similar recording feature, and Visual Basic is done “by hand” in Modules.

The next level of objects are those that you work with on a Form or Report, which are called “controls.” These are things like a text box (place where you can show/enter data), combo box (a dropdown box), label (something that displays static text), or tab control (where you can place lots of different controls on a form but use less screen real estate because users need to click on each tab to see that information).

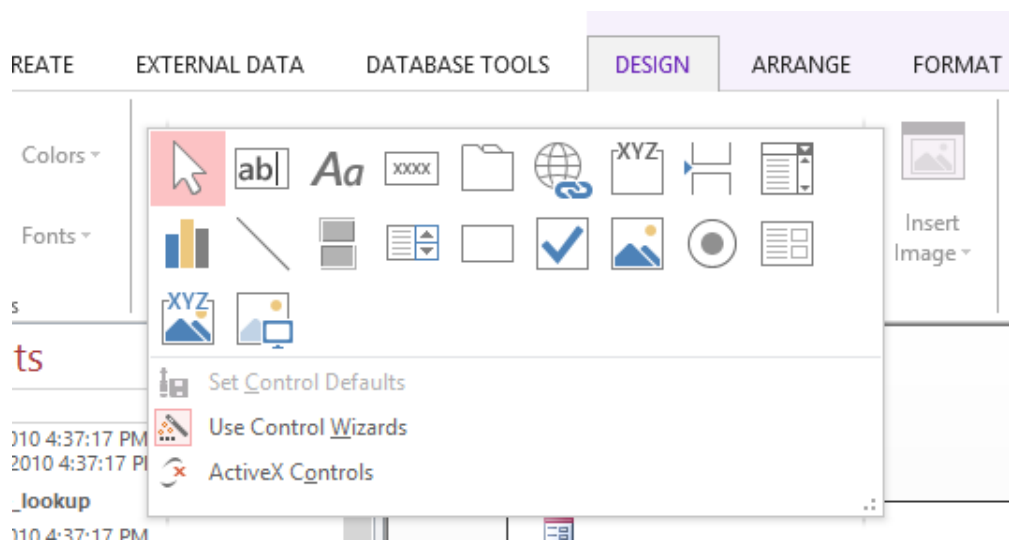


Figure 2: Controls available for Access desktop apps

Drag & Drop

If you have experience with PowerPoint, the Access desktop databases will have a familiar user experience for you. Everything on screen acts like a movable object. You can select a text box, label, or anything else in a form or report, and drag it where you would like it. Using the mouse you can draw a box around several different controls and drag them as a group where you want.

Properties

When you open a Form or Report in the Design view, they seem incredibly simple. Just boxes with names inside them, or “Unbound.” The depth of what you can do with each control comes when you

double-click it. Access will open the Property Sheet, where you will see the wide array of options available for each control: what happens when it is clicked, whether users can even edit the data, whether it is visible, how the data should be formatted, etcetera.

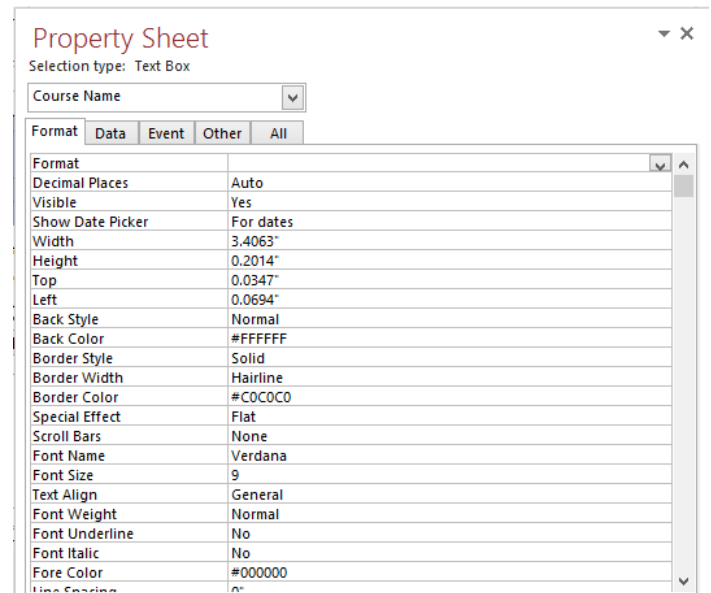
Format Options that control the look and feel of the control you are working with.

Data Settings that dictate whether what field in a table the data should come from, whether it can be edited, and any special validation that should take place.

Event Actions that should happen when a user clicks on, moves away from, or interacts with the control. The actions can be a Visual Basic routine, or a Macro.

Other Miscellaneous options like whether the field should be included in the controls a user can tab between, and what text should display when the user hovers over the control.

All All of the options from each tab shown in one long list.



Relationships

Relationships are used by any database to define how different things should be joined together. In Access desktop apps, you can visually explore relationships between tables by selecting “Database Tools” from the Ribbon, then “Relationships.” When you are working with a Table and select the “Lookup” option for the type of data a field will store, Access will automatically create relationships behind the scenes.

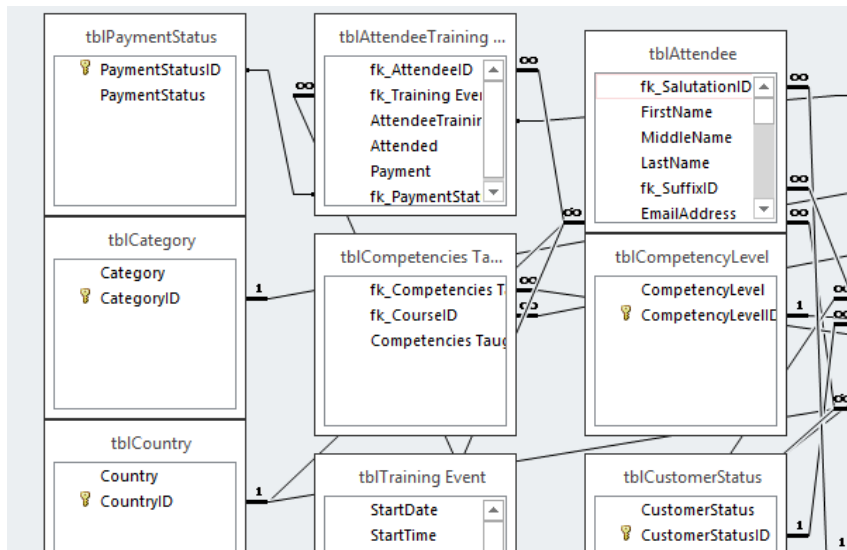


Figure 3: Relationships in a complex database

Designing relationships consumes entire books, but there are some very simple things that will help you avoid common mistakes:

1. Don't repeat information in more than one table. If you are tracking Orders and Customers, don't store Customer Name in the Orders table.
2. Make sure each table has an Autonumber field that has a name similar to the table name. For example, “CustomerID” as opposed to “ID.”
3. When joining two different tables, join them using the Autonumber field, not a text field from the other table. You can always show the user the text value even though in the background the number is being stored. If you used a text field to relate two tables, say Customer Name, what happens when you change the customer name in the customer table? It breaks the relationship. Joining with an AutoNumber field makes sure the one thing that can't be edited for a record always remains the same. George Washington's name might change in the Customer table, but the automatically assigned record number 1 will not change.

Chapter 5 - Common Surprises, Misunderstandings, and Frustrations

- There is no true “save” button or need in an Access desktop database. When you enter information in a record and click away from it, the information is automatically saved.
- Be careful renaming things, whether it is a form, table, or even a field in a table. A common mistake is for someone to rename a field in a table, only to find out it causes many errors downstream in queries, forms, and other things that are expecting a field with a specific name. Better option: re-label the field on forms/reports so the user sees a friendly or understandable name, even if the table might store a name that isn’t perfect.
- Be wary of the “Available Fields” helper for Forms and Reports. The tool lets you drag field into your Form/Report, and is smart enough to understand if things might be related to each other. But in the background it creates a query that may not actually return the records you want, or worse, you receive messages that you cannot edit the data you are viewing in the forms! Better option: Create your forms such that they focus on one table. Then add Subforms to your form to display related information. The same holds true for reports, although since reports are “read-only” you are less likely to encounter difficulty since editing data in reports is not needed.
- If you haven’t distributed your database to other users, you can keep track of different versions of your database just by creating copies and giving the files names like “MyDatabase 7.15.14.accdb” so you know when it was created, and can tell if there is a newer version you are working on just by the data. The only real need for this is if you are copying the file between computers, sharing with a consultant, or you want to be sure you can restore a previous version if you happen to make a big mistake with your work.
- You may be alarmed to find that you've created a query that returns no records, or many fewer than you expect. Either your criteria is too strict, you've joined tables incorrectly, or, you need to define the relationships between the fields more precisely. Take an example query, where you join the field CustomerTypeID and fk_CustomerTypeID in the Customer Type and Customer tables, respectively. What if you have records in the Customer table that haven't been assigned a Customer Type value? They will be missing from your query! Access, by default, assumes you want to see only records from two joined tables where the values match. An empty value on a matching field won't return a row if the query join Double-click the lines joining the two tables in your query to see options to make sure you see all the data you need.

Chapter 5 - Microsoft Access Limitations and Misconceptions

For being such a niche tool, there are many misconceptions about what Access can and can't do, and what limitations exist.

Database Size

Microsoft, wanting to ensure they make a clear delineation between Access and their more expensive and robust database, SQL Server, has capped the size of any one Access database to 2GB. That is actually a massive amount of data, and rarely will you hit that size with just plain tables. If you store attachments you can easily approach 2GB. But if you are simply storing several million rows (yes, million) with tables that might have 10-30 fields, you won't approach the 2GB limit. And if you are a stalwart Access fan who doesn't ever want to move to SQL Server, you can even link to multiple different databases that might each be 2GB in size by just distributing tables into separate Access files.

Fields, Tables, Queries, etc

Access limits you to 255 fields per table or query, and 255 tables, 255 queries, you get the picture. It is possible, by creating queries that might join all fields in several tables (using a Select * from each table), to hit the 255 field limit in a query. If you find that a table has 255 fields, there is a strong chance you want to revisit your design to see if in fact your trying to use Access like an Excel spreadsheet.

Security

IT magazines and organizations at times use Access (and Excel) as the poster child for proliferating, unsecured sensitive data. Access can be secured to a greater extent than even Excel, but by no means is it 100% secure in the same was as you can achieve by storing your data in a server database. You can, with appropriate measures, password protect and encrypt your database – making it inaccessible to all but the most experienced and determined hackers.

Corruption

There was a time that Access databases could become corrupted, and loss of data would occur. This is very unlikely today, despite the fact there are several online services for helping to recover corrupt Access databases. While data corruption is not a concern, certainly make backup of your database file(s) if you have any critical data that you cannot afford to lose. More commonly, the Visual Basic in Access will encounter issues. You may receive a message from Access such as "File Not Found:" and have no idea where the error is coming from. In these cases, try decompiling your database. To do so, you open your database using a special command. Here are instructions to do so on UtterAccess.com:

<http://www.utteraccess.com/wiki/index.php/Decompile>.

Chapter 6 - How To Deploy Your Access Solution to End Users

A common question raised by many Access users creating a multi-user database is how best to deliver your solution so that it is secure, fast, and upgradeable. Access database deployment within your organization, or to your end customers, should be well thought out in advance to avoid performance issues, reliability problems, or potentially significant security or data loss issues. This article discusses key best practices to deploy your Access database.

Splitting Your Database

This item is a must to make upgrades to your end users easier. A split database in MS Access simply means you will have one file that contains your tables only. This is sometimes called a "back end" or "data file." For simplicity, we'll refer to it as your Access Data File. The second file is what is usually called your "front end" database. There is really no data stored in the file, it only contains your forms, queries, reports, macros, VB modules, and links to your Access Data File. The advantages are:

Store your data file anywhere

With your data separated from the user interface (forms/queries/reports) and business logic (VB/macros), you can keep the data anywhere that your users are able to gain access. Most commonly, on a shared network drive or even a shared folder on your own machine (as long as your machine is on when the file is needed by other users!). You can back up your Access Data File whenever you need to, and relocate the file if needed. You will need to make sure the front-end database file(s) always know where the Access Data File resides. If it moves, you can manually restore the links to the new location, do so programmatically, or use a tool like UI Builder for Access to help prompt the user to find the new location.

Upgrade Your Front-End Databases without Data Loss

Storing your forms/reports/macros/VB in the same file with your tables also presents a problem when you need to make updates. Particularly if you plan to deliver your database to users at a different location where you don't have direct access to the database file. Sending users a new version won't work, as the data would be overwritten, or you would have to ask the users to import their existing data tables. With a split Access database configuration, you simply send them a new version of your front-end file and their data remains intact.

Avoid Multi-User Collisions

If you store all your objects in one file, including your tables, Access will usually have difficulty when multiple users attempt to open the same database file. While Access does have record-level locking, you may still receive errors that another user is currently in the database and you cannot make changes.

To avoid this, you provide each end user with their own front-end database file, each linked to the same Access Data File.

Splitting your database is quite easy with the built-in Access tool. Be sure to make a copy of your database before you begin just to be safe. In Access 2000/2003, select "Tools>>Database Tools..." from the menu, then "Database Splitter..." In Access 2007/2010, select the Database Tools ribbon tab, then "Move Data" and then "Access." The wizard will take you through the process of selecting the tables to move, and what the your data file will be named. It may seem trivial, but we recommend adding "_datafile" to the end of the name so you are sure you know this is the data file, not a front-end your users can delete.

When preparing your front-end database file for delivery to users, you will want to consider what you want to allow your users to be able to do with your front-end file. If you want to allow them to make design changes, open tables directly, edit or view VB source code, then you can deliver the file "as is." More commonly, you will expect users to have much more limited capability to simply view the forms and reports as you have prepare them, without being able to view or change the form/report/code design. The easiest way to do so is to compile Access into an MDE or ACCDE file format. A standard Access database file is in a "semi-compiled" state which means Access does some level of compilation of any VB code you have created on the fly. Compiling your Access database simply means that a new file is created with the extension ending in "e" instead of "b," and the objects are fully compiled such that a user cannot open a form or report's design view, nor can they inspect your VB code. This protects your intellectual property, prevents many types of tampering, and also makes your database load (only slightly) faster. A few other points to help with front-end preparation:

Navigation and Forms

Make sure your database includes clear navigation, and we recommend you also use our 10 Tips for a More Dynamic Access Form to add polish and make sure your users are more productive.

Remove Unnecessary Menus and Options

In most cases, you'll want to hide the Access database window (2000/2003) or Access Navigation Pane (2007/2010) and provide your own Access menus. To hide the database window in 2000 and 2003, select "Tools>>Startup..." from the menu, and uncheck the "Display Database Window" option. In Access 2007 and 2010, select the Office icon, then the Access Options button. In the dialog, select "Current Database" and uncheck the "Display Navigation Pane" option. Lastly, all versions allow you to disable Full Menus (the ribbon in Access 2007/2010) and right-click Shortcut Menus if you want to further limit user's options.

Security Measures

Compiling your front-end into an ACCDE or MDE format helps protect some of your intellectual property, but if you are planning to store sensitive information in your Access project, we recommend you implement the necessary security measures to be sure your data is protected to the extent needed. In some cases, that may simply be hiding the navigation pane to minimize the possibility that a user can find your tables without Access know-how. In more extreme cases, it may mean password protecting and/or encrypting the front-end and data file.

Access Workgroup Security

Because the Access 2007 file format does not support Workgroup Security, we recommend staying away from this approach since it will limit your ability to move to the 2007 format when it becomes advantageous or even required by future versions of Access.

Table Security

Ultimately, a determined user will likely be able to get to the data if they have enough Access know-how. A linked table in your front-end is technically a "trusted connection" where if the user can open the database window or navigation pane, they will be able to view data in the linked tables even if the data file is password protected. If you hide the database window/navigation pane, disable full menus, and also disable the "Access Special Keys" option, you have made a large step to preventing a user from being able to get to the tables.

Navigation Security

Even after you secure your database tables, you may decide you want to limit what information some users view in your forms and reports, and what forms and reports they can even launch. The Enterprise Edition of UI Builder offers the ability to create user-level menus in Access using the user's windows login or a custom menu prompt, or you can create your own custom menus and routines to log users in and present different options to them. At a form level, you may want to consider whether you will obscure certain information based on the user, such as National ID or Tax ID.

Data File Locations and Performance

Your Access Data file (back end) should be located on a network folder where all users can open files in the directory. You can also use this mechanism to limit who can access your database. For example, you may set up your file server folder permissions to only allow the 10 users you want to be able to open your database. If you are delivering a solution to your client, you may not have control over where the data file resides, but you can provide guidance on setting up network folder security, and making sure the folder is included in nightly backups.

When deciding where to store the Data File, it is also important to consider the file server age, hardware, and network latency. The last thing you want is a client or user complaining that your application is painfully slow. In some cases, it may just be a slow server. We also recommend creating a persistent connection to a table in the Data File for each front-end when it is open. Simply put, Access will attempt to lock and unlock the Data File each time you close a form and open another. By creating a persistent connection, Access keeps the Data File open for the entire time the front-end is open. To do this, you can either launch a hidden form at startup that uses a table with a few records in the Data File, or open a recordset to a table in the Data File, and keep it open until the front-end is closed. If you are using a tool like OpenGate's UI Builder, this is handled for you when UI Builder starts up. UI Builder creates a connection to the event log table (tblEventLog) that remains open until UI Builder closes.

If your solution will be used by a single end user, and you store their Data File on their machine, keep in mind that it will be easiest to have a consistent directory to store the Data File on your machine (for development) and the end user's (for production). If you both use the same path on your local PCs, you won't need to remap the linked tables when your end user installs your front end database file.

Front-End Installation

The front-end database files should be installed on each individual's PC. It will load faster, and you minimize the risk that two users open the same front-end on a shared network folder (causing the Access multi-user error messages we covered previously). Importantly, your end users will not need the full version of Access unless you are allowing them to make design changes to your forms/reports/code. You can save a great deal in MS Access licenses by leveraging the free Access Runtime edition. When you install the front-end, you can use the free Access 2007 developer extensions from Microsoft to create an installation file, or use a free installer like Advanced Installer (what we use and love). Make sure your main front-end form or an easy-to-access form has the version clearly labeled. That way if you need to troubleshoot, you or the user can quickly tell what version they are running.

It's also important to note that the Program Files directory in Windows Vista and Windows 7 is no longer considered a "read/write" directory for most purposes. Consequently, you'll want to install the front-end file in one of the new and accepted places to installed read/write programs, such as:

The user's Documents folder

C:\Users\<<USERNAME>>\AppData\Roaming\<<YOURFOLDER>>\

Upgrading Your Front-End

When the time comes to upgrade your front-end file with new changes, all you need to do is release an update to the end users with the front-end file only. They simply need to run your provided installation file, or replace their current copy of the ACCDE/MDE front end file you initially provided.

Changing Your Data File

If you have direct access to the Data File, making changes to the tables is usually easy. Your biggest task will be to make sure all users have exited their front-end databases so that you have full read/write access to the table design.

Chapter 7 - How to Evolve Your Desktop Access Application

When you begin to have a large number of users, your database is growing very large, or you notice that performance over the network is slow, you may want to consider a few options to evolve your solution. As written about earlier, you can “upsized” your database to Microsoft SQL Server or MySQL Server. There are quite a few advantages to move your important data from Access to a Microsoft SQL Server or MySQL Server.

Better Access Performance

If you presently have your database split into a front-end and back-end, with the back-end data file stored on a shared file server, then you know Access can slow to a crawl when the network is busy and more than one user is trying to work in your database. Why the slowdown? Imagine you and two other people are trying to read the same book. In all likelihood, you are each trying to read a different page, but because you are all holding a single book, it is a slow process. You may try taking turns reading a page at a time, or holding the book “just-so” to let each person see part of their page. Either way, it's uncomfortable, slow, and inefficient. Microsoft Access, built mostly for the desktop, has similar challenges.

A true server, like Microsoft SQL Server or MySQL Server, is far more effective at letting multiple users work in the same database. Taking our book example, now imagine you had the ability to split the book up into sections and give each person their page and a few before and after it. You would only need to worry about sharing when two people needed the same page. Microsoft SQL Server or MySQL Server are far faster because they are designed to handle this sort of situation.

Lastly, in the case of a server-based database, the Microsoft SQL or MySQL server's actual hardware does far more of the heavy lifting and computing. When you store your Access back-end on a file-server, the file server simply stores the data and your local machine does all the heavy lifting, including pulling all the data down from the network to your machine, whether you need it all or not.

How much faster can a Microsoft SQL or MySQL server-based database run compared to a file server-based database? The answer is, of course, “it depends.” It will likely run far closer to what you would expect if the database was entirely on your local PC than when located on your file server.

Chapter 8 - Resources to Help You

There is a vast community of Access developers, consultants, and enthusiasts to help you with your project. Below is a small list of resources that have been particularly useful over the years:

Tutorials

- [YouTube](#) For visual learners, there are many quality tutorials on YouTube.
- [DatabaseDev.co.uk](#) A collection of excellent tutorials and examples for many common requirements.

Templates

- [Microsoft](#) Available when Access opens (2007 or higher), search for many templates.
- [OpenGate](#) Templates for the most commonly needed Access databases.

Examples

- [AccessWeb](#)
- [AccessFreak](#)
- [Access Blog](#)

Experts

- [UtterAccess](#) Hundreds of very dedicated Access experts providing advice free of charge
- [Access World Forums](#)
- [DBForums](#)