# Case Study: Access Desktop Application Linked to Web form with MySQL Database

Tony Bayliss FCCA, Senator System Consultants Limited

[www.senatorsys.co.uk](http://www.senatorsys.co.uk)

[tbayliss@senatorsys.co.uk](mailto:tbayliss@senatorsys.co.uk)

Presentation to UK Access User Group, May 2017.

Thanks to all who attended and especially those who pointed out improvements.

Some valid security points were raised which I will action, particularly giving each remote user a separate ACCDE on the terminal server.

Users have very restricted access to the terminal server and cannot access the control panel (including the ODBC administrator), install any programs or add-ins, create databases or run Excel queries but more can be done to tighten it up.

All suggestions were and are very welcome.

Slide 1

The idea behind this talk was not to provide a comprehensive view of MySQL or how to use it with Access. Most Access developers, including me, use Microsoft SQL Server as a default back-end and I am not qualified to compare the two.

Instead, it is a case study showing what started as a simple Access application on a local network and ended up linked to a MySQL database populated via a web form. My experience is if you do need to create an application connecting to a MySQL database, Access desktop using ODBC seems a good and valid option if you love Access desktop applications (don’t we all).

Slide 2

I am a qualified accountant and a Sage reseller and became a database developer via Dataease for DOS back in the 1980s - a very fast programmable RDBMS and RAD tool. The Windows version was a very slow RDBMS and a good reason to switch to Access! Amazingly, in August 2016, Dataease released a new update to the DOS version that runs in Windows 10! So there must be users and live applications out there, some still used by large plc companies but they’re well due an update.

Slide 3: My previous AUG talks. More details on the list server and my web site.

Slide 4

This system handles applications and admissions for various grammar schools which are in the news at the moment as the government wants to open new ones. I went to a grammar school myself back in the 1970s, but I’m not convinced it’s the way forward for secondary education - unless they want me to setup their admissions in which case I can see some of the positives.

Slide 5

The initial specification was just an Access application on a local network to

record and process the annual applications to sit the entrance test:

* Applicants parents/guardians completed forms by hand and school admissions staff keyed in these details to the database.
* Admissions forms for the child to bring to the test were printed and sent out by post along with details of the test.
* Passport photos of the child were scanned and stored in a folder as JPGs file names just referring to candidate numbers not names and printed as part of the Admissions Form.
* An email button for those supplying email addresses was later added to acknowledge the acceptance of the application.
* A spreadsheet and printed lists of candidates was generated to go to the test examiners.
* Labels were produced to stick to the exam papers.

Slide 6

The trickiest part was the school allocation routine. There were 5 state-funded grammar schools each with a limited number of places. Each parent could nominate a first, second or third choice. When the exam results arrived, they were imported from a spreadsheet and then the allocation routine started with the highest scoring child, allocating their first choice, working down the list until it came to a child whose first choice school was full, then it allocated their second choice if that was not full and so on. For each of their three school choices, it also records how many places the child is below the cut off number of places. Anyone wanting the full VBA, let me know.

The school could adjust these automatic allocations based on reasons such as proximity to chosen schools, siblings, etc, via an appeals system for parents. Now it’s all handled by the local authority based on rules laid down by the DfE so my code is no longer needed and lies unloved on my hard drive.

Meanwhile the schools started up a web site using a web design and hosting company called XHost. Initially this was just a brochure until parents started asking to apply online and XHost were asked to create an online admission form.

Slide 7

Initially, XHost emailed the details of online applications to the schools’ office staff and they keyed them in which was obviously not ideal, so we all got together and came up with a plan to try out with test data: XHost would adapt the forms so that it saved online applications in a MySQL database stored on the web server. The schools wanted to keep my Access application as much as possible so I would try to link it to the MySQL database. Then the schools could manage and process the online records in the Admissions office as they had always done via Access and add postal applications.

MySQL itself is used by massive corporates and it’s everywhere on the web so I wasn’t concerned about MySQL as a database, just my lack of knowledge and it was another database to learn. In fact I use MySQL a lot without knowing it as it is the database behind WordPress which I used to create my own simple web site [www.senatorsys.co.uk](http://www.senatorsys.co.uk) among others. Almost everyone using a third party web host will see MySQL on their CPanel. The tests went very smoothly and we have been live since 2008.

XHost created the schema (equivalent to the SQL Server DB) and the basic table on their web server and I then used MySQL Workbench to connect to it. MySQL Workbench is the standard Windows tool to manage MySQL databases, downloadable free (as is the database) from the MySQL web site. (There are other third party tools – HeidiSQL is very popular).

Slide 8

(In the demo, I opened MySQL Workbench and connected to the live application data, showing several records added in the last hour. This slide shows the table of primary schools downloaded from the Dept for Education Edubase which we store in another MySQL table for the parent and staff to look up the correct address and DfE code for the child’s primary school).

To connect, you need of course the server name, username, password and port number which is all straightforward and it then works very like SQL Server Management Studio. Between myself and XHost, we created the fields such as the candidate number which is the Primary Key (Type INT for integer and Auto Incremented. Alan commented that BIGINT caused problems until Microsoft fixed it recently).

There are migration tools for MySQL, nothing like as comprehensive as SQL Server migration tools like Andy Couch’s MUST but we were effectively starting a new system for a new year’s intake so I would adapt my Access front-end manually to best work with the new database.

We imported 20,000 records to test and linked it to Access which I will show shortly. Probably only 6 users at any one time and not a huge record set but performance was and is excellent.

(I then gave a very quick demo of MySQL Workbench, showing how you can add and edit field names, types, etc., add tables, use stored procedures, import and export, etc. You can also run SQL queries and edit the data of course).

(Next I showed the live Access front-end used by school staff hosted on a terminal server):

The terminal server could only be accessed by me and the schools through a controlled gateway which caused a few issues but they trusted the security of this.

Next I showed the MySQL ODBC settings using ODBC Administrator. This of course requires filling in the server name, password, database name and port number. There are lots of advanced settings but I didn’t have to change any - for me it just worked. I can do all the things you can do with any external table in Access using forms, SELECT/INSERT/UPDATE/DELETE queries, modules and reports but I need the MySQL Workbench for database changes such as adding a table, indexes or field changes, etc.

Slide 9

This shows the my Access application which the users open via an ACCDE. In the background is the Main Menu which requires a login and password before the user can get further with user access controlled by VBA code as per my system demonstrated previously to the User Group and available on my website.

From the main menu, the user opens a continuous form which lets them browse and select applicant’s details as completed on the online form. The data source for the browse screen is a simple Access query on the live MySQL admissions table with criteria set according to the Year on the Main Menu and the user’s school and login.

Just like SQL Server, you don’t use Ctrl-F but from the continuous form use the right-click options for searches, sorts and filters. These respond in a fraction of a second despite reading live from the remote web server.

At this point I was out of time, but there were other features I could have shown:

1. The Notes History button lets the schools add notes for each candidate (automatically date-stamped and tagged with the user name) stored in a local table in the back-end ACCDB.
2. If the parent has got it wrong, the school can search and select the child’s primary school details, linking to the downloaded Department For Education database stored in MySQL.
3. The email acknowledgement to the parent confirming acceptance to sit the test goes straight out via an SMTP server – we considered sending these via the local users’ Outlook which is possible but the customer is happy this way. We used XHost's mail server for this, and I have the code (redacted to hide the mail server’s password). I can send the VBA to anyone interested.
4. Finally a feature that the school like is my Proper Case Name & Address fixer which is not perfect but cleverer than the basic VBA functions, fixing double/triple spaces and coping with hyphenated and apostrophe’d names like N’Dour, O’Reilly, Al-Khalili, Fink-Nottle - even Ashby-de-la-Zouch. I’ve used this elsewhere and there is a demo on my web site which is good fun, if I say so myself. As you can imagine, many of the parents are awful typists.
5. There is a duplicate entry check created using the Access query wizard, which finds any applications with the same Post Code, Surname and Date of Birth. This could obviously include twins. We discussed possible ways of preventing parents from making duplicate applications on the PHP form, e.g. by asking for something unique to each child like NHS No but the annoyance factor to the parent and Data Protection Act issues decided on just this check.)

Final summing up:

If I’d been approached to do this job, I may well have handed it over the whole job to someone else who knows PHP and web forms. Or worse, I might have tried doing it with AWA. As it happens, the PHP part of it is fairly straightforward (Slide 10 shows very basic simple code). The parent only adds records – they obviously can’t see any others or update their own once submitted so I most of the clever stuff is in the Access front-end and the clients are very happy with it.

Also I feel pretty confident about handling MySQL. It's free, powerful and if you have a friendly collaborator like XHost, you can put together good solutions with elements of web forms and good old Access desktop.

XHost is run by my excellent collaborator on this project, Russell Clarke.

Contact Details at [www.xhost.co.uk](http://www.xhost.co.uk)

OUTLINE OF SCHOOLS ADMISSION SYSTEM

School staff & System Admin connect to Terminal Server

School staff & System Admin connect to Terminal Server

MS Access Application

Hosted on terminal server connects to MySQL database via ODBC. Insert, Select, Update, Delete etc via Forms, Queries, Reports, VBA Procedures.

Parent enters application via PHP form hosted on web server.

INSERT INTO command populates MySQL database

MySQL Database stores Candidate Details. Hosted on Web Server