

The Future of Your DataEase for DOS Application

October 2007 PLM Consulting, Inc.

This paper attempts to consolidate information from a variety of prior publications and suggest possible strategies for companies heavily invested in DataEase for DOS applications.

Background: The DataEase for DOS (DFD) database management software was truly a remarkable and useful development tool. Over the years it has been used by thousands of companies worldwide to manage complex database systems. Many of our clients continue to use DFD as their primary database to run their businesses. Often, this database has been highly customized to integrate their core data operations into one database as opposed to earlier years when data management was done by several different software programs. Their databases range from hundreds of users to a single user.

The majority of them used Novell Netware as their network operating system. However, in recent years large numbers of companies have moved to Windows for their server. It seems that Novell has lost the server wars due to their lack of marketing and abandonment of earlier features that made them the server of choice for non-SQL databases. The main issue this presents is that Windows uses a completely different locking system than Novell and many DFD applications written to run on Novell are having a lot of problems switching to a Windows server.

The State of the Business: Because of the controlling influence of Microsoft, companies have fewer and fewer choices about the operating system and software installed on their systems. Microsoft seems to have gone out of its way to ensure that, eventually, everyone in America will be compelled to use its products. It has become increasingly difficult for other software companies to compete with Microsoft.

Pundits have, for many years, predicted that Windows would eliminate the ability to run DOS programs, but this has not yet proved true and DOS applications still run under the latest versions, including Vista. Still, as the operating systems get more and more complex it takes more time to ensure the operating environment is correctly established and maintained and Windows' tendency to change things without any notification only makes maintenance that much more difficult. It is entirely possible that Microsoft will eventually produce an operating system or a patch that keeps DFD from running at all.

Hardware components can wear out or break. Some of these components may be increasingly difficult or impossible to obtain simply because the market for them is dwindling as new advances in hardware render them obsolete. For example, there is the issue of Printers -- parallel port printers are becoming more difficult to find and more expensive, and DOS programs cannot easily print to USB printers.

What are the Choices?

Option A: Purchase off-the-shelf software and change your business rules to accommodate the built-in functionality. With this option you will be doing business just like everyone else using this software.

Advantages: Relatively inexpensive, widely supported.

Disadvantages: Your business rules aren't fully supported, if at all; unable to modify if your business changes; large disruption in normal operations for retraining, errors, etc. Not only that, but you will

likely be taking giant steps backwards in your automation. You may find yourself with islands of information in multiple programs and, isn't this what you corrected with your DFD database?

Option B: Rewrite your database in a new, Windows-based file-server type product.

Advantages: your business rules can be built in

Disadvantages: Expensive; potential disruption in normal operations for retraining; potentially long parallel testing period; potential new hardware requirements.

Option C: Migration to a SQL based database such as SQL Server or Oracle, using any number of products as a front end.

The major advantage of this approach is that SQL databases are widely used in business. As a result there are a very large variety of programs that can interface with them.

On the down side, it will likely be more expensive to build and maintain a custom SQL application than a file-server based app. For this reason, only customers with large data requirements or high user volumes will likely want to opt for this approach.

File Server Based Database Engines

Ffenics, (Database City)

DataEase for Windows, version 6.52 (Sapphire)

DataEase for Windows, version 7.x (Sapphire - not recommended)

Access (Microsoft)

FileMaker Pro (FileMaker, Inc.)

Alpha 5 (Alpha Software)

The differences between Ffenics and DataEase for Windows are discussed elsewhere but they are sufficient for us to recommend Ffenics over DataEase at this time.

So, what is involved in moving to a new database platform?

Well, consider for a moment that your existing database contains many of your business rules. These business rules exist because they are necessary for the proper operation of your business and they usually prevent bad data from being entered. Your existing database was built most likely over a number of years and consists of hundreds or thousands of hours of development time. The typical company has many thousands of dollars invested in their databases. How do you leverage that investment?

With the exception of Ffenics and DataEase for Windows (DFW), the other products have no method of incorporating your business rules without significant analysis and development time.

Unfortunately, because of basic differences in how Windows products handle data reporting, all of your reports will have to be rewritten in the new database software regardless of which one you choose.

The advantages of selecting Ffenics as your new platform are significant and may be the best choice. The most significant advantage is that you will be able to more directly migrate your existing DFD structure over since Ffenics supports the same general data structure and business rules as your DFD application. This means that you don't have to completely "reinvent the wheel". The process can also be seen as an opportunity to correct flaws, eliminate unused forms and reports as well as being able to take advantage of other features in Windows itself that are unavailable in the DOS product.

In other words, rather than looking at migrating to a new platform as a necessary evil, look at it as an opportunity to do the cleanup you have been needing to do for the past few years.

What is a Migration and How is it Done?

First, let's drop the term migration. When it comes to DFD, it's more aptly referred to as a conversion. The application structure and the data can be converted to Ffenics (or DFW) along with the scripts of your reports, but the rest of the process is pretty much a rewrite since the layout of DFD forms and reports has to be completely redone.

There is also the issue of whether a direct conversion is either needed or desired. Blindly doing a direct 1:1 conversion is probably the least effective use of time and money. There are many things that can be done differently in a windows application and Ffenics has a number of features that make both data-entry and reporting much more efficient. A 1:1 conversion will simply not take advantage of these new features.

Before any conversion is undertaken, there are several very important things that need to be done.

--Have a clear plan of action that is agreed upon and will be conformed to no matter what. This is very important because it is the middle of the project changes to the plan which cause the most confusion and end up wasting the most time and money.

--Do a thorough cleanup of the application before any conversion is started. Get rid of any unused, redundant, or worthless Forms and Reports (particularly reports). The less there is to convert, the faster the conversion will go.

--Identify any problems or changes that need to be addressed as part of the initial planning and determine if these changes should be done before, during, or after the conversion.

--Be prepared to provide timely assistance to the developer to answer any questions and do any necessary tests. Everybody's time is valuable but unnecessary delays have sunk many a project.

How Much is This Going to Cost?

Excluding the cost of the software itself, well, it depends... It depends on how complicated the application is, how big it is, and how much before-during-and-after cleanup is going to be required.

A basic database with 20 forms and 50 reports will require approximately 36 hours of work. Add in an additional 16 hrs for meetings, discussions, testing, and delivery and you have about 52 hours. This is assuming a completely clean database without a lot of very complicated reports and no new requirements.

The reality is that few databases are completely clean, there are always a few very difficult to convert reports, and there are always some issues to resolve. And the very fact that this is moving from a Character user interface to a Graphic user interface means that many user interface issues will have to be addressed.

So, What's the Solution?

The solution is to take things one step at a time. Start planning now and discuss what it's going to take to convert your application. Make sure that all the key players are on board with the decision.

Decide how much of the project you are going to take on yourself and how much you want to contract out. If you are the hands on type, then it will be to your best advantage to hire someone to just help you plan and get started. From that point on, you do what you can and get help with the tough stuff as you go. Whatever you do, don't try and go into this blind thinking that the fact that you are a wiz with DFD will make you a wiz with DFW or Ffenics. Experience has taught us the fallacy (and cost) of that.

Even if you plan on contracting the whole conversion out, you will want to be involved in the process. There is no substitute for the knowledge that you and your people have for how the job is done and how the app is used. Don't make the mistake of thinking that someone who knows nothing about what you do can somehow magically divine how to best set-up your new application.

Lastly, even if you know your business inside and out, don't be a slave to the way you always did things with your DFD app. There are always going to be other ways to do something. And it's entirely possible that the process of conversion can help you find ways to do things better and more efficiently. This is where the real value of a conversion can show itself. I've seen cases where people have saved literally days of work every month by means of a simple change made during a conversion.