

contents

- 193** Microsoft Access 2000
- 195** Lotus Approach
- 196** FileMaker Pro 5
- 198** Corel Paradox 9
- 200** Sun StarBase 5.1
- 202** Beginner's guide
- 203** Web-enabled databases
- 206** High-end server databases
- 209** Table of features
Editor's Choice

• Databases reviewed by Tim Anderson.

Databases

Traditional databases are facing a new challenge to survive in a web-enabled world. So how are established and new databases coping with life in cyberspace?

The market for Windows database managers is anything but lively, being dominated by the leading office suites. Microsoft Access is the most widely used, followed by Lotus Approach and Corel Paradox. Then comes FileMaker Pro, a Mac migrant with a loyal following. All these have been around for years and many people use whatever database is bundled

with their PC, but it is important to make an informed choice if you are about to embark on a new database project, so we have rounded up the latest versions for in-depth testing. We also take a look at the new kid on the block, StarBase, part of Star Office which is available free from its new owners Sun Microsystems.

Despite the apparent inactivity on the product front, things are changing in the

database world. The importance of the desktop database, where your data is stored and managed on a single machine, is diminishing. Whether this is a simple address book, or a large business database, or the outcome of years of research into the flight paths of the bumblebee, the chances are you will want to access the data in different ways from different kinds of devices.

break into cyberspace



Businesses want their data web-enabled, both for internal use and for customers to browse products and place orders. Hobbyists want to share their data with other enthusiasts. Everyone wants a quick way to look up phone numbers and email addresses. What this means technically is the future of database management lies with server databases that can accommodate many different

kinds of client, from traditional desktop applications to websites and PDAs.

Alongside this new desire for flexibility, the typical PC has become powerful enough to handle heavyweight server databases like those from Oracle and IBM. These run across a variety of platforms and are well placed to grab new markets from those unwilling to commit to Windows for all their future

database requirements, and conscious that Unix-like operating systems still dominate the web. The web has also brought new opportunities for products like mSQL, popular among those setting up small web-based database systems.

The challenge for the likes of Access and Paradox is how well they can survive in a web-enabled world, so we have paid special attention to that issue.

Microsoft Access 2000

Part of the ubiquitous Microsoft Office, at least in its Professional guise, Access is the one to beat in the Windows desktop database market. It arrived in 1992 and was an instant success. At that time many database applications were still DOS-based, using dBase or Paradox tables. Access was not only a well-behaved Windows application, but was also modelled more on high-end SQL database server than on the DOS heritage.

Access stores each database in a single file with an MDB extension, including not only data but the forms, reports, queries and program code that goes into a typical database application. The main interface shows the contents of a database in a container window with a navigator and detail panes, showing groups of objects including Tables, Queries, Forms, Reports, Pages, Macros and Modules. Some

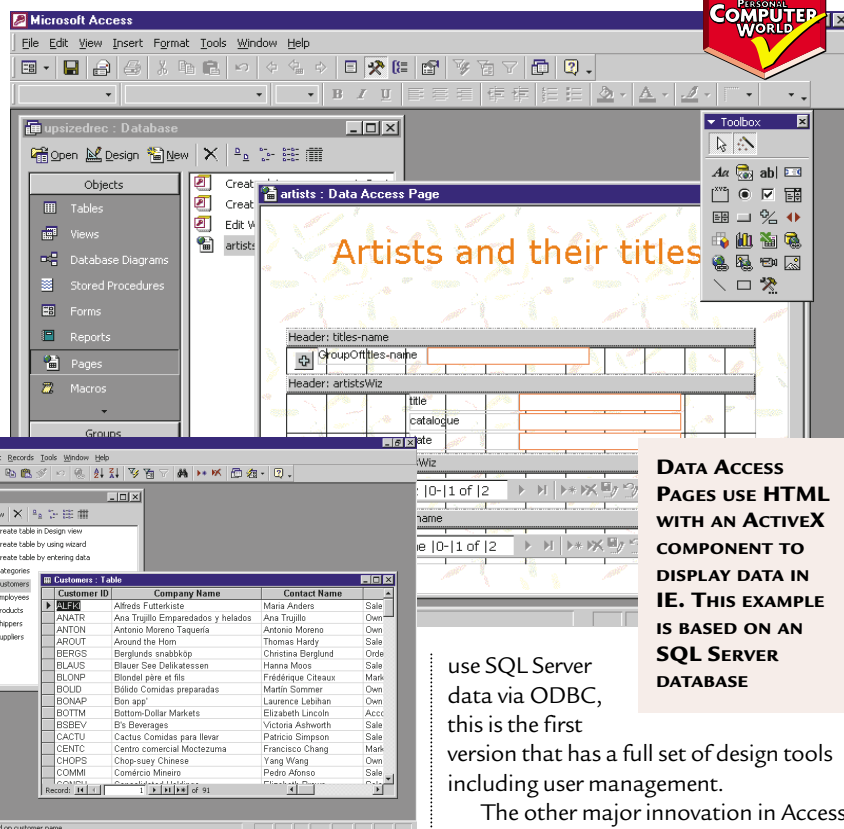
of these require explanation. Data Access Pages are HTML pages which use ActiveX components to display Access data forms in a web browser. Macros are the simple way to automate things, with predefined actions you can arrange in a sequence with simple point-and-click actions. Modules contain Visual Basic (VB) code and are far more powerful, although using them means real programming.

There is a lot to like in Access. The table designer is easy to use and lets you set default values, validation rules and required status as well as the usual data type and field

size. Tables open in an Excel-like grid, with instant sorting achieved by right-clicking a column header.

Intuitive searching and filtering is available from toolbar icons. The Query builder has an excellent two-way SQL designer, which lets you switch between editable SQL code and a graphical query designer. Reports are another strong feature, although challenging to novices. However, a wizard can create a simple report for later modification.

The table designer is easy to use and lets you set default values



A CLEAN, UNCLUTTERED INTERFACE IS KEY TO SUCCESS

DATA ACCESS PAGES USE HTML WITH AN ACTIVE X COMPONENT TO DISPLAY DATA IN IE. THIS EXAMPLE IS BASED ON AN SQL SERVER DATABASE

One flaw is that arguably Access has too broad a reach, from novices to professional database developers. This is particularly true in the form designer where the developer features make it over-complicated for beginners. The form designer is not properly integrated with the VB code editor, because Access has its own native forms engine, but uses the standard VBA development environment for editing program code.

Another limitation of Access is that its native database engine and file format is not suitable for networked applications

beyond a certain number of users. How many depends on the detail of the system, but it could be as few as

five or as many as 100. If performance is slowing down a client/server system may be more suitable. Access 2000 makes this easy by supporting SQL Server as a second native database engine. A cut-down version, the Microsoft Database Engine, is free to deploy, or Access can work with the full edition of SQL Server 7.0. While Access has always been able to

use SQL Server data via ODBC, this is the first

version that has a full set of design tools including user management.

The other major innovation in Access 2000 is Data Access Pages with which you can create browser-based Access forms, although they are designed for intranets rather than for the wider web and require Internet Explorer. An Office licence is required for each user who accesses a Data Access Page. There are more ways than this to publish Access data on an intranet or on the web. Access has ODBC and OLEDB drivers that work with Active Server Pages, VB Web Classes, and other web development systems that work with ODBC. As a file-based database system, an Access MDB is not suitable for busy websites, and you will need a Windows web server. There is no JDBC driver so it is not ideal for Java development.

PCW DETAILS

★★★★★

Price £326.43 (£277.81 ex VAT)

Contact Microsoft 0345 002000

www.microsoft.com

Good points Well designed and powerful interface, native support for SQL Server

Bad points Complex advanced features can make it tough going for novices

Conclusion An excellent choice if not pushed beyond its limits

Lotus Approach

Lotus acquired Approach as a desktop database for the SmartSuite office suite in the days before Lotus became part of IBM. One of its original selling points was it did not inflict yet another database format on the world, but works with a number of different database types including ODBC. The default format is the trusty .DBF, in its dBase IV variant.

Approach does not pretend to be the most powerful of databases, but instead aims for ease of use. The starting point when working with Approach is defining

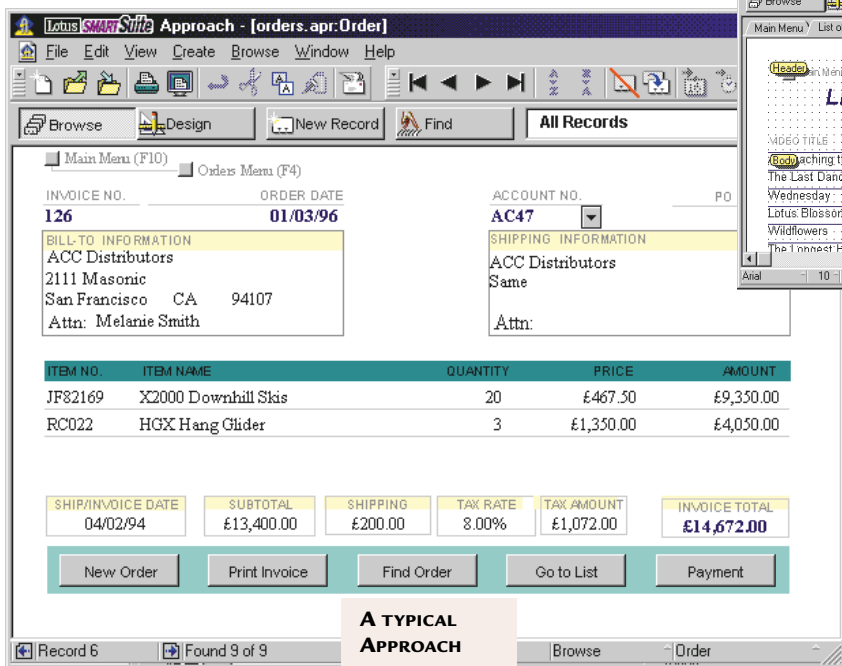
button activates the form so you can begin working with data. A single form is not usually sufficient, so you can add new pages that become tabs in the main view. Reports and queries are simply additional tabbed pages. Any form can serve as a query – clicking the Find button lets you enter example values into fields and search the database on those values.

Custom applications can be built in Approach. Like Microsoft it has a simple macro builder and a complete programming environment called

integrity of the data. Lotus does not provide any special links to WordPro, although you can embed an Approach view into a 1-2-3 spreadsheet using OLE. Also, Approach has its own tools for creating form letters and labels.

There are a few nods toward web integration, but nothing of much substance. You can publish an Approach

IN DESIGN VIEW, APPROACH FORMS USE THIS DISTINCTIVE FLOATING PROPERTY INSPECTOR



A TYPICAL APPROACH APPLICATION

a database table or opening a connection to one of the supported server databases. If you are defining your own table, you get a conventional dialog where you enter field names, data types, field size, and special features like default values and validation rules, all presented in something close to plain English. A clever aspect is that even when using standard dBase IV or FoxPro data files, which do not support engine-level validation or referential integrity, Approach supplies these features. It does so by using a database container file, with a .APR extension, through which the underlying data is managed.

Once you have defined your data, Approach uses a form designer as the centre of operations. In Approach, the form designer is part of the main application window. Clicking a Browse

button activates the form so you can begin working with data. The interface is restrictive for developers, so you would not want to create advanced applications. It is also prone to flashing, jerky effects when used under program control, since the code is automating user actions.

On the face of it, Approach is highly suitable as the database manager in an office suite. It is also a handy tool for creating simple clients for server data, and has a driver for Lotus Notes. Unfortunately, Approach depends on its own database container to enforce rules and add features to database files, which is fine when working from within the application, but problematic when you want to work with the data from outside. You can work with the underlying tables, which are most likely to be in dBase format, but doing so risks breaking the

view as a jDoc document, a Java-based utility that creates an accurate but static reproduction of the data for viewing on the web. You can also save as HTML, which is preferable unless you need to preserve the exact layout of a report. There is a facility to bundle the Approach database container with its associated database files into a compressed .APT file, making it easier to work with if you need to access the data from an FTP (File Transfer Protocol) server.

The winning feature of Approach is that it is bundled as standard with SmartSuite, or can be bought standalone. It is adequate for everyday office tasks and makes an effort to be easy to use.

PCW DETAILS

★★★

Price SmartSuite Millennium £439.45 (£374 exVAT)

Upgrade and Competitive Upgrade price £162.15 (£138 ex VAT)

Approach 97 £49.35 (£42 ex VAT)

Contact Lotus 0870 6006123

www.lotus.com

Good points Ease of use, uses standard underlying format

Bad points Poor suite integration, less powerful than Access

Conclusion Great value but not outstanding in its own right

FileMaker Pro 5

FileMaker was born on the Mac and brings that platform's ease-of-use philosophy to Windows. In version 5 some effort has been made to make it more Windows-like, but fortunately the interface changes are superficial. The Mac version keeps in step, making this the obvious choice if you want to share data on a mixed Windows-Mac network. Like Access, FileMaker is aimed at both novices and developers, although the primary focus is on providing a database manager that allows non-experts to manage data easily.

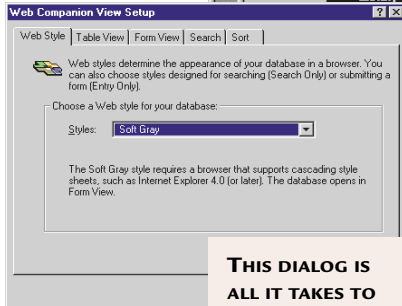
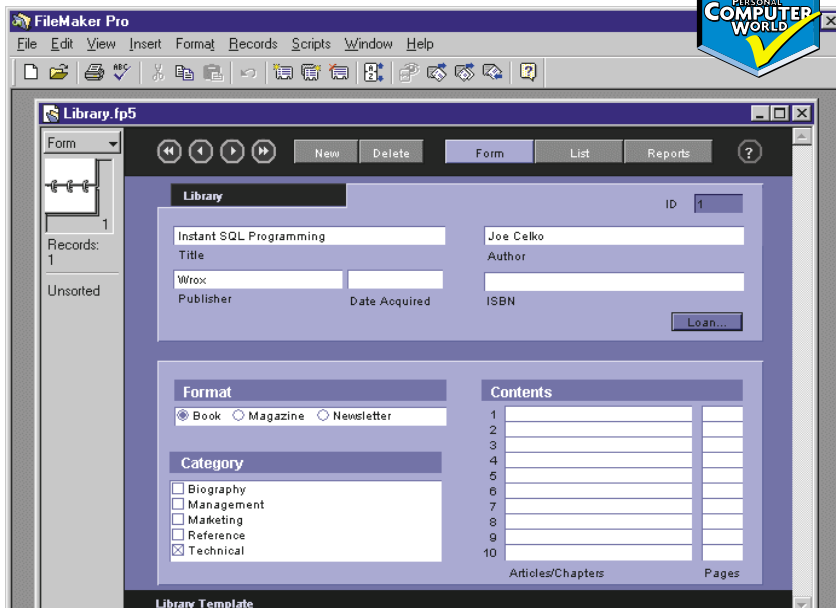
FileMaker's interface has a lot in common with Approach. A new database begins with a field definition dialog, then a form is automatically created. The interface is centred on FileMaker's form designer, which can be switched between Browse, Layout and Find modes.

FileMaker was there before Approach, and of the two its attempt to be accessible without compromising database management capabilities is more carefully thought-out.

FileMaker Pro has a native database format and is smart about how data is organised. A tricky decision in most database managers is how to choose the field size – too small and you end up with truncated names or obscure abbreviations; too large and there is a lot of wasted space. FileMaker has a variable field size, so if you define a field as Text, it can take up to 64,000 characters. There is no need for a separate Memo type. Another plus is that you do not need to define indexes. FileMaker does this on-the-fly, according to the queries created. These great features make designing a database more intuitive.

To design a form in FileMaker you select items from a tool panel, not floating but docked at the side of the form, and lay them out on a form with the mouse. Reports use the same layout tools. You can add many different layouts to a database, and switch between them via a drop-down menu.

FileMaker supports relational databases, but it pushes the flat-file type



THIS DIALOG IS ALL IT TAKES TO PUBLISH FILEMAKER DATA ON AN INTRANET. MOST, BUT NOT ALL THE STYLES REQUIRE IE

as far as it will go. Features like lookups enable a field in one table to be filled by looking up data from another, and repeating fields, where a single field can hold more than one value. It is best to avoid these features,

particularly repeating fields, and use related tables instead. FileMaker layouts can contain portals, or subforms, which show related data from another table.

Despite its ease-of-use, FileMaker is powerful. The basic file-sharing edition supports up to 10 users, but you can upgrade to a server version for 250 users. The maximum database size is 2GB. There is also a developer version that lets you extend the script library with your own functions, or deploy FileMaker applications using a runtime edition.

Version 5 has two significant features for Windows users. The first is an ODBC driver for FileMaker data. In previous versions it was all but impossible to read FileMaker databases other than with FileMaker. Now any application that supports ODBC can read data in FileMaker's format.

Next comes support for ActiveX

automation, which lets you control FileMaker from programming tools like Visual Basic. Unfortunately FileMaker's implementation of this is poor – all you can do is run existing FileMaker scripts. However, it is an improvement on earlier versions.

Finally, the software has an integrated web server that makes it genuinely easy to publish data on an intranet or the web, if you have a permanent connection or an ISP that supports this feature. FileMaker uses XML and scripts to recreate layouts as web pages, with dynamic read-write access to the data. Think of this as a black box, since FileMaker is no easier than other web development tools if you want to customise this feature extensively. For immediate results though, it is superb.

FILEMAKER HAS A CONSERVATIVE DESIGN BUT HAS BEEN CAREFULLY THOUGHT OUT FOR NON-EXPERTS

PCW DETAILS



Price £233.82 (£199 exVAT)

Contact Filemaker 0845 6039100

www.filemaker.com

Good points Easy to use and powerful, with new ODBC driver

Bad points Mac origins and individualistic approach can get in the way

Conclusion The best blend of usability and performance

Corel Paradox 9

Paradox is another long-established product that began as a successful DOS database from Borland. It was a casualty of the migration to Windows, particularly since Borland changed it so drastically that the Windows version had little in common with its DOS predecessor. A couple of versions later Borland pulled out of the end-user application market and licensed Paradox to Corel, but the database still uses the Borland Database Engine, developed and maintained by Borland. This gives Paradox the same relationship to Delphi, Borland's premier development tool, that Microsoft Access has with Visual Basic. Corel does not bundle Paradox with all versions of the WordPerfect Office suite, but keeps it for the Professional and Enterprise versions or as a standalone purchase.

Of all the desktop products in this test, Paradox is the most perplexing for newcomers. Corel has worked hard to

supply improved Experts, but has not succeeded in making it even remotely user-friendly. This is partly a jargon problem. In the table structure design dialog, the Text field type is called A, which stands for Alpha, while S means Short, for an integer up to 32,767. Paradox has two kinds of index, the primary key and secondary indexes. So to index a field you have to select a tab called Secondary Index, choose Add and type in a name, then select one or more fields to define the index – a rigmarole for a simple operation. The fact that you have to press Shift-F2 to edit a memo field is hardly intuitive either. Paradox does not belong in an office suite where end users will encounter it, but should be positioned like Visual dBase and Visual FoxPro as a developer's tool.

For those who have the patience to learn Paradox, it is an outstanding desktop database manager. It is characterised by an advanced database format, with good support for referential integrity, and a comprehensive range of tools including two query builders, form and report designers, and a data model designer that lets you view and modify

When it comes to the web, Paradox has some interesting features

PARADOX HAS SUPERB TECHNOLOGY, BUT DOES NOT EASILY FIT ANY MARKET CATEGORY

the links in a relational database.

Paradox has its own development language called

ObjectPal. Despite its name it is object-based rather than object-oriented, so you cannot define new classes and create custom objects. Even so, it is powerful and well-structured. Its main disadvantage is that because it is unique to Paradox, there is not much incentive for developers to learn it. By contrast, VB is so widely used that Windows developers are almost compelled to learn it. Ironically, Corel has

licensed VBA but has not yet put it into Paradox.

When it comes to the web, Paradox has an interesting set

of features. In true Corel style there is a bewildering range of options. There is a bundled web server to which you can start saving HTML files exported from Paradox. You can save as static HTML, or as an HTTP template. In this case, the file contains the layout, but not the actual data. Custom tags in the template enable Corel web server to retrieve the data dynamically when the page is requested. You can also use the Corel Web Server Control, a web server ActiveX control which overrides the standalone Corel Web Server. Pop it on a form and

ALL DONE WITH PARADOX: THE COREL WEB SERVER CONTROL AT WORK

when the form is run so too is the web server.

Paradox 9 has yet another set of web tools, called JParadox. This includes a set of Java servlets and requires a web server that supports them, such as Sun's Java Web Server or any web server with JRun installed. A Paradox JDBC driver and a Web Form Designer utility let you design a form using data-aware controls connected to Paradox data, and publish it as a Java applet. This is great technology, but nobody can pretend it is easy to use.

PCW DETAILS

★★★★

Price £146.88 (£125 ex VAT)

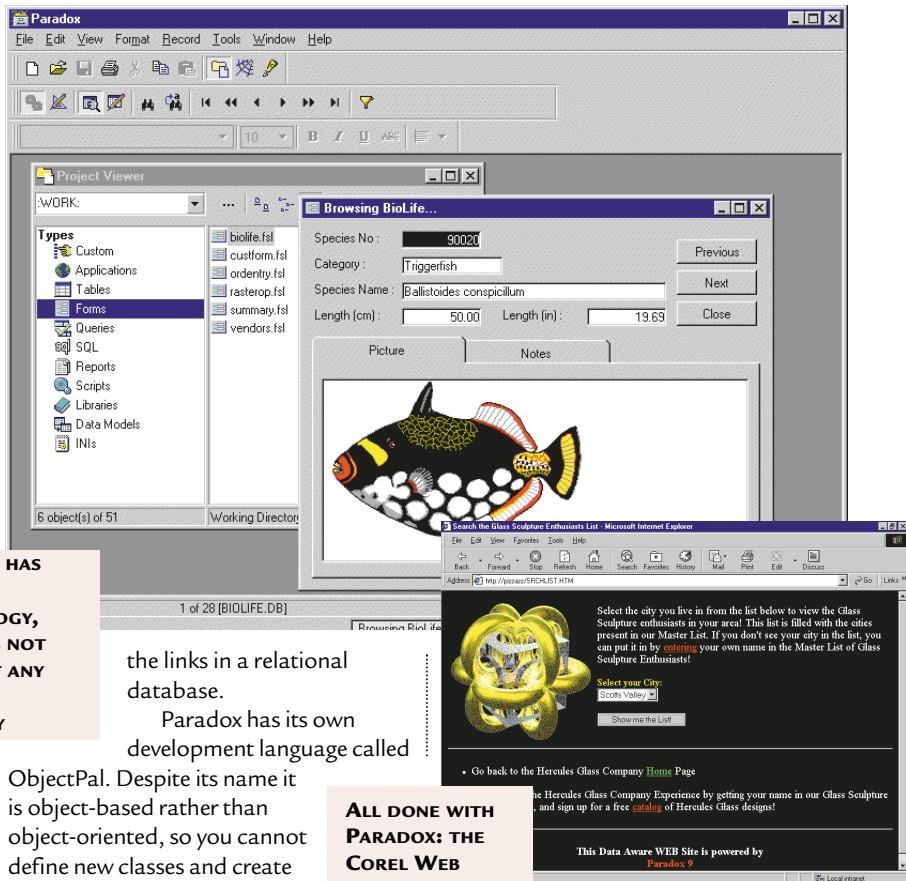
Contact Corel 0800 973189

www.corel.com

Good points An advanced database manager including great cross-platform web technology

Bad points Too complex to bundle with an Office suite. Proprietary ObjectPal language

Conclusion The problem for Paradox is that it is too hard for end users, while developers have plenty of other tools to choose from



Sun StarBase 5.1

Working within StarOffice, the office suite acquired by Sun as an alternative to Microsoft Office, is the StarBase database manager. It can connect to a variety of database formats including Access, Oracle, DB2, or any JDBC or ODBC data source, along with a table designer, form designer with a sub-form facility, SQL-based Query builder, and report wizard. It sounds good and there is probably the making of a useful database module within StarOffice, but in the current version it is virtually unusable.

To explore the database features you choose New from the File menu and select Database. This presents a dialog where you can choose the database type. In the Windows version, there is a native StarBase type which stores data with a .RBX extension, but otherwise you can select from a number of types. Unless you choose the StarBase type, however, you cannot create a new database through StarBase, but only connect to an existing one. If you do choose StarBase you get a New data Source dialog which lets you specify a file name and start from scratch.

Once you have chosen a data type, a window opens with icons for Forms, Queries, Reports and Tables. Double-clicking any of these drills down into the relevant section, which if you are starting a new database means an empty window. One option in the database window is SQL, available from a right-click menu, which lets you execute an SQL statement against the current database. This seems to be intended for commands that update or insert data rather than return results. We tried a Select statement which returned 'Command successfully executed', without providing any data to view.

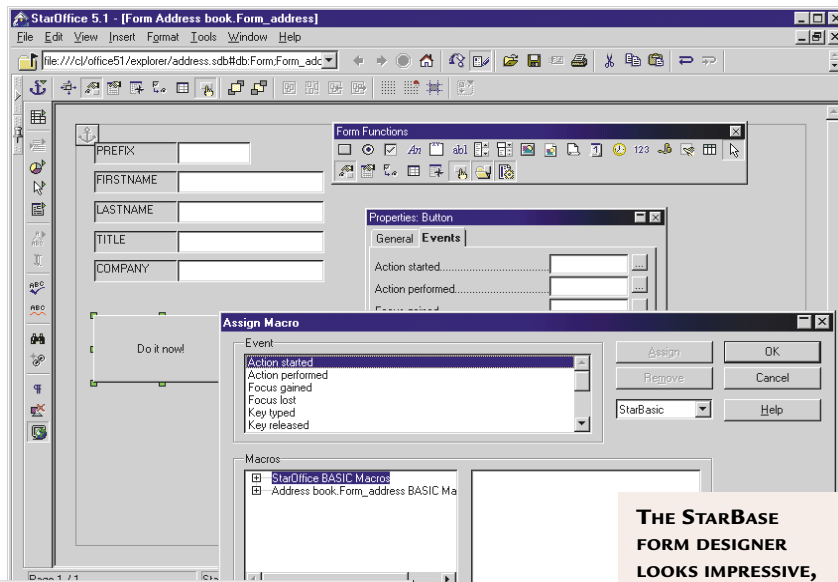
The StarBase database designers, for tables and queries, are reasonably intuitive assuming you are familiar with

Database designers for tables and queries are reasonably intuitive

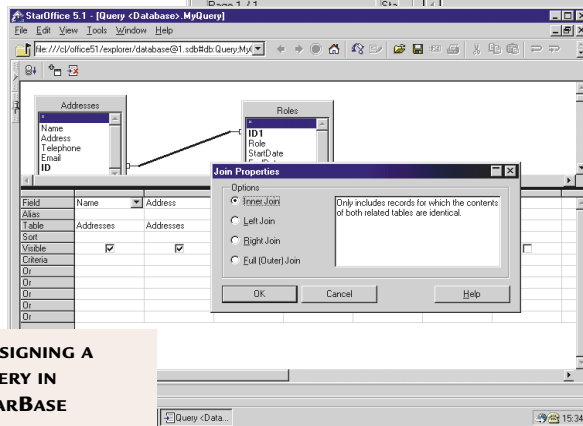
something like Microsoft Access. Field types for the native format include Text, Number, Boolean, Currency, Memo, Image, Decimal and Binary. You can define indexes including a primary key, set default values and mark fields as Required or Unique.

Making use of data was frustrating. There is a range of auto-pilots for

common tasks, and while this was successful in producing a simple report, on other occasions the auto-pilot failed to run. StarBase had difficulty displaying its form designer, and offered a blank text document without any database connection. We were able to get to the form designer with New - Form from the File menu. Forms have built-in controls for navigation, new and delete functions. You can also use the



THE STARBASE FORM DESIGNER LOOKS IMPRESSIVE, BUT PROVED FRUSTRATING IN PRACTICE



DESIGNING A QUERY IN STARBASE

Form

Functions

toolbar to add controls such as buttons. Controls fire a range of events such as Mouse pressed or Mouse moved, and you can link these to StarBasic macros.

Integration with StarOffice looks as if it is intended to be good, but it was hard to make it work. You can insert database fields in a StarOffice document, but it was quirky and frustrating.

Documentation, such as it is, is hard to find. If you are lucky it might pop up at the right time, otherwise it is a matter of searching the online help. There is no entry for Database in the main contents. StarBase is also unreliable. The interface is quirky, and we were afflicted from time to time by messages like: 'Error reading data from the database. Connection failed.' At the same time, this is clearly an ambitious product that has the makings of a standards-based and cross-platform database manager. Consider it a work in progress.

PCW DETAILS



Price Free download from www.sun.com/products/staroffice

Contact Sun 01276 451440

www.sun.com

Good points An ambitious product with potentially good connectivity and integration

Bad points Buggy and scarcely documented

Conclusion Do not use for real work yet

Beginner's guide

Database management is about organising information. It is not strictly correct to call an application like Microsoft Access a 'database'. Instead, it is a database management system (DMS). To confuse matters further, the same database may sometimes be accessed by more than one DMS. Many applications include some degree of database management, even when that is not their main focus. For example, email clients and PIMs (personal information managers) usually have an address book facility. You might also keep a list of addresses in Word, or a list of products and prices in Excel, in which case these too act as database managers.

These simple databases are usually a single list of records, where each record has a number of fields such as forename, surname, postcode and so on. A list like this is called a table. The database manager will make it easy to search the list, sort it in different ways, and create reports for printing out the information. Databases like these are called flat-file and they are very common.

The problem with a flat-file database is that it is inefficient and inflexible for certain types of data. The main problems come when you have a lot of repeated data, or when the data simply does not fit the flat-file structure. For example, if you had a database of vehicles you would probably find the same names cropping up frequently in the

Manufacturer field: Ford, BMW, Vauxhall and so on. It would be more efficient to have a code for each manufacturer, so you can use the code instead of the full name, and a second table listing the manufacturers and their codes. At this point it becomes a relational database, so called because you have more than one table and the tables have a relationship, in this case a common field called, for example, 'ManufacturerCode'. Instantly the database becomes more powerful. For instance, you can now easily add information for each manufacturer like contact details, brief history, current turnover or whatever.

Some information simply cannot be represented in a flat-file database. A classic example is a list of students and classes. Each class involves more than one student, and each student attends more than one class. There is no sensible way to store this data in a single table. In a relational system, it takes three tables, one for students, one for classes, and one to store the links between them.

Relational databases are vulnerable to corruption. The system depends on valid links. Duplicate or missing entries in any of

the tables destroy the integrity of the data. This is called Referential Integrity, and most database managers have built-in features that enforce it, preventing errors of this kind.

The other challenge with relational data is extracting and updating the data successfully. The industry standard is a language called SQL (Structured Query Language), and most systems allow you to use it, although there are annoying variations between different implementations.

Another key feature is transaction management. On systems with several

tables, it is often important to treat a series of updates to individual tables as a single transaction,

which can be completed or cancelled.

The relational system is mature and well understood, but it is not the only way to manage data. Another important development is the object database, which fits well with modern object-oriented programming. There are specialist object database managers, and relational database vendors are working to adapt their systems to store objects easily and efficiently.

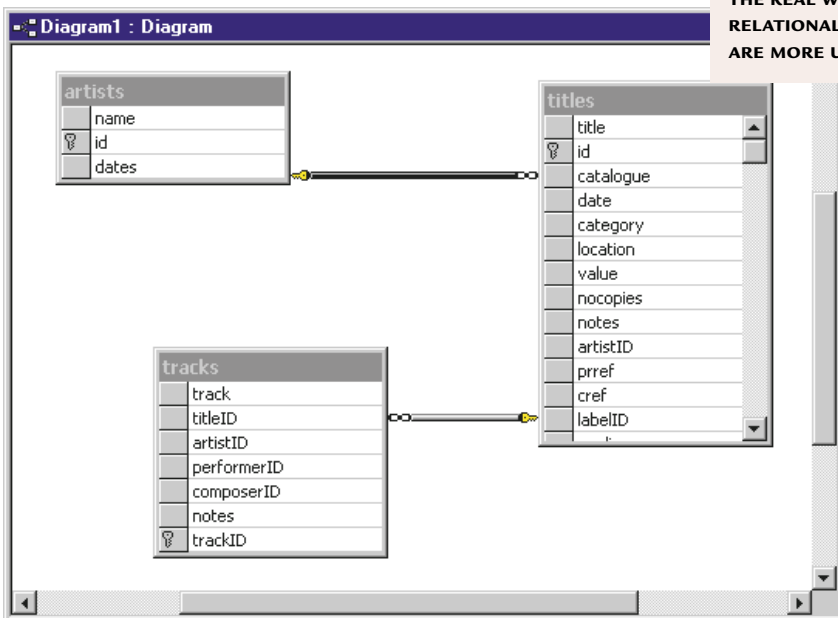
Finally, there is an important distinction between file-based and client/server database management. These apply when more than one workstation has access to a shared database.

In a file-based system, a separate database manager runs on each workstation, reading a common database file on a file server. Client/server systems, on the other hand, have a single database management application running on the server, and the workstations communicate with this application rather than reading the data directly.

A variant on this is a multi-tier system, where clients communicate with intermediate software, which in turn communicates with the server. Such systems increase the complexity, but because they are more adaptable and scalable they are gradually replacing file-based solutions.

Some information cannot be represented in a flat-file database

IT WOULD BE SIMPLER IF ALL DATABASES WERE FLAT-FILE, BUT IN THE REAL WORLD RELATIONAL DATABASES ARE MORE USEFUL



Web-enabled databases

Publishing data on the web enables access from anywhere and use of a universal, cross-platform client. There are strong advantages for public-facing sites and private intranets, and anyone considering a new database project should look carefully at a web-based solution.

Web pages are constructed using HTML, which is plain text marked up with tags to give it rich formatting. Database managers, on the other hand present dynamic information, which may change minute by minute as records are added, modified and deleted. There are two options for publishing data on the web. The simplest is to convert data into HTML and post it on a web server as simple HTML. This has three substantial disadvantages. First, the data is only as up-to-date as the last time the page was uploaded,

although you can overcome this to some extent by automating the process. Second, it is hard to avoid pages that are too large, or require the user to navigate numerous pages before finding the required data. Third,

pages are read-only. Even so, for small read-only data that changes rarely, uploading HTML pages is effective.

Another option is to have software that generates HTML on-the-fly, in response to queries entered on web forms. A script or executable runs on the web server, where it accepts queries or updates, accesses data programmatically, and generates results

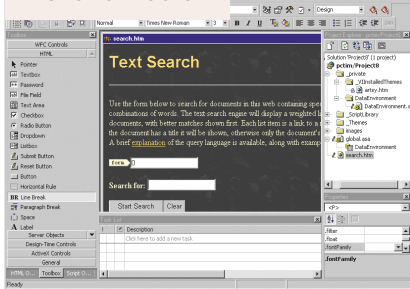
which are fed back to the browser as HTML. Perl is popular for this work, since it is web-savvy, runs well on Unix-like operating systems as well as Windows, and has several libraries available for database work. You can use something like mSQL as the database engine and get all this up and running for free, aside

from ISP costs, or connect to any database server via ODBC or other library code. The snag is that it is fairly low-level stuff, not to be undertaken casually.

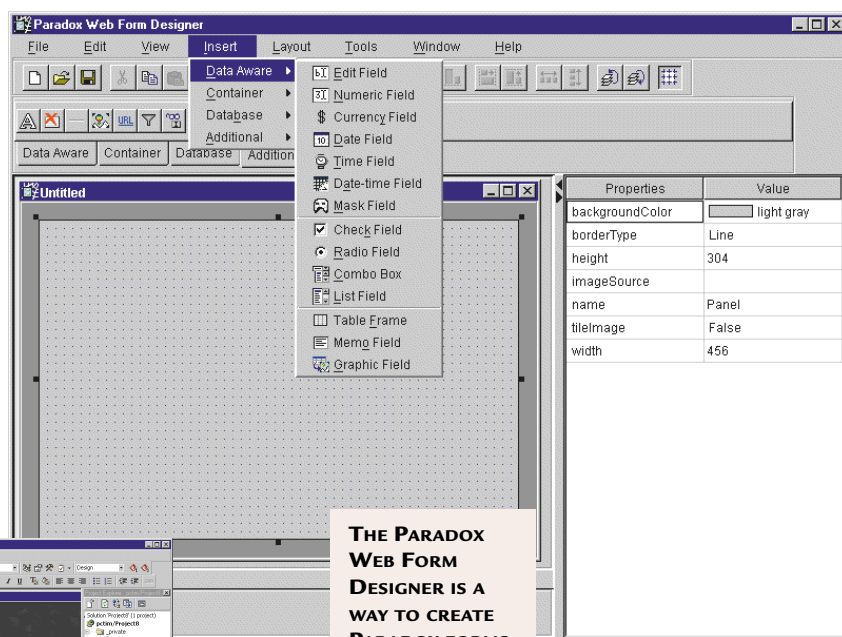
A number of approaches make it easier. One is ASP (Active Server Pages), a feature of Microsoft's Internet

Some tools make web publishing easy. Prime examples are the database products that have an inbuilt web server, such as FileMaker Pro and Corel Paradox. Using these systems, you need do little more than check a box to have a database accessible by a web

VISUAL INTERDEV IS FOR DEVELOPING DYNAMIC WEBSITES, BUT IT IS NOT THE EASIEST OF TOOLS



THE PARADOX WEB FORM DESIGNER IS A WAY TO CREATE PARADOX FORMS AS JAVA APPLET



Information Server that parses HTML pages to run server-side scripts before delivering them to the

browser. These scripts, which can be written in Visual Basic Script or JavaScript, can access databases via ADO (Advanced Data Objects), which includes drivers for SQL Server, JET, or any ODBC data source. Microsoft's Visual InterDev is a tool for working with ASP, as is Macromedia's Drumbeat, which contains many pre-written scripts for rapid development.

Microsoft FrontPage has database features which rely on ASP technology. Overall, ASP is easier than Perl, but still not for beginners. A similar approach is taken by Allaire's Cold Fusion which, unlike ASP, runs on several platforms including Windows, Solaris and Linux.

browser over TCP/IP. But going beyond what the system provides is complex.

Web compromises

Before getting totally carried away by web database publishing, it is worth looking at the compromises involved. Things you can take for granted in a conventional application are either difficult or not available, things like pixel-precise control of the display, modal dialogs, and rich clients that have access to all the features of the OS.

Traditional applications have to be completely redesigned to fit a page-based interface rather than one based on WIMP (windows, icons, menus and pointers). Also, instead of a single persistent connection between a user and a database server, the web model is stateless. All the web server sees is intermittent requests for data along with parameters passed from the browser. Web applications have to overcome this by using some technique to track a user's activities between logging in and drifting off the site, usually involving cookies,

Overall, ASP is easier than Perl, but still not for beginners

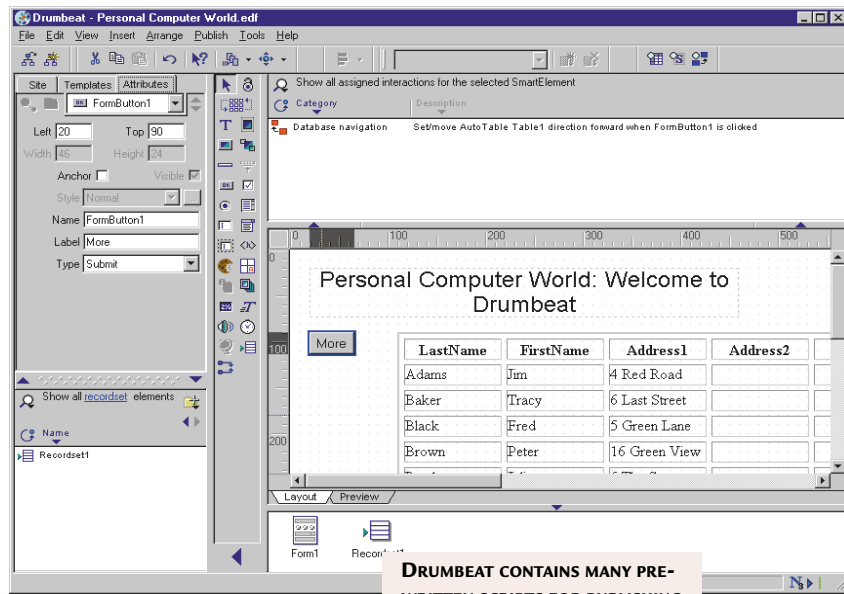
which are tracking codes stored on the client, or by using extended parameters to exchange identification codes with each web request.

Managing data through a web browser and web server is more complex than the traditional approach, so the compelling attractions of the web do have a cost.

One way to obtain more capable clients is to run Java applets or ActiveX components on the client machine. In this scenario, the browser is used to host an application that has its own means of making data connections and interacting with the user.

Microsoft's Data Access Pages work like this using ActiveX, or you can build data-aware Java applets using any number of Java development tools including Inprise JBuilder, Symantec Visual Café or IBM VisualAge. The weakness of the ActiveX approach is that you are limited to Microsoft Internet Explorer 5.0 running on Windows. Java solutions can work well, although it is more restrictive on clients than a system that delivers pure HTML.

There is a place for both client-side and server-side processing in web database applications, but in general it pays to exploit the web model to its fullest by keeping application logic and data access components on the server and supporting the widest possible range of clients. That way, you can exploit browser features like client-side JavaScript and Dynamic HTML while also providing a pure HTML interface for universal access, perhaps from PDA clients as well as full-featured web browsers.



DRUMBEAT CONTAINS MANY PRE-WRITTEN SCRIPTS FOR PUBLISHING DATABASES ON THE WEB

Server components

At the high end, web database publishing and old-style database applications are coming together. You can do this by using server-side components which can be accessed by conventional clients such as Windows applications, or by web applications. The idea is to reduce scripting to a minimum, on the grounds that script runs slowly and is hard to maintain, particularly when mixed with HTML in a web page. Application logic and data access is encoded into components, and for maximum scalability these can be managed by a transaction server. Transaction servers are intelligent about matters like sharing a single database connection between several users, or keeping an object instance in memory so it

can be re-used. You can also define a sequence of operations as part of a transaction, so that if need be it can be undone as a whole.

The two dominant technologies for server-side components are Java and COM (Component Object Model). Java Servlets are web server extensions that run either on a Java web server or on any web server supported by JRun, an add-on from Allaire designed for this purpose. Servlets can do data access using Java technologies such as JDBC (Java Database Connectivity). Most leading server databases have JDBC drivers, or you can use a JDBC-ODBC bridge to access ODBC data sources. Servlets can be used in tandem with JSP (Java Server Pages), a Java-based equivalent to Microsoft's ASP.

THERE ARE SEVERAL PERL LIBRARIES AVAILABLE FOR DATABASE WORK

The COM solution exploits the ability of ASP to instantiate COM objects on the server.

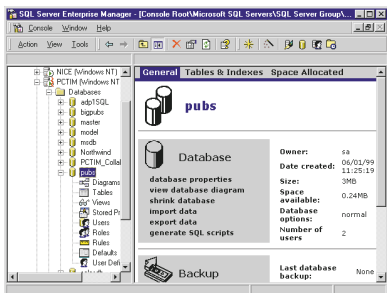
COM is a Microsoft technology and is perceived as Windows-only although some effort has been made to support it on other platforms. The advantage of COM for Windows developers is that you can adapt or create Visual Basic, Visual C++ or Delphi code, using familiar data access techniques, and compile it as COM components. You can host these in Microsoft Transaction Server, also known as COM+, which is an integral part of Windows 2000.

There may seem to be a bewildering array of options for web deployment of database solutions, but that is because it is the current hot spot, in contrast to the moribund desktop database arena.



High-end server databases

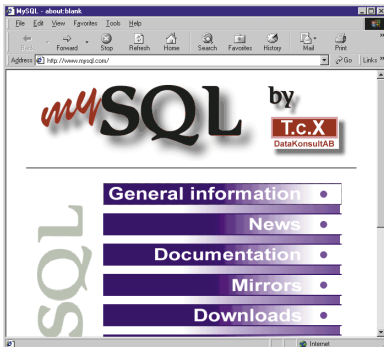
Although the main focus of this group test is on desktop database managers, it is important to keep half an eye on where you might go if you want to scale up to a server database engine. The drive towards web development means that may be soon. Using a server database does not necessarily mean abandoning your favourite desktop system. Server database systems do not work on their own, but require some kind of client, and with the partial exception of FileMaker Pro, all the desktop systems have connectivity features that let them act as clients to each of the server systems described here. Having said that, it is more common to find custom-built clients for this kind of database. All of them are also well suited to form the database layer of a web application.



Microsoft SQL Server

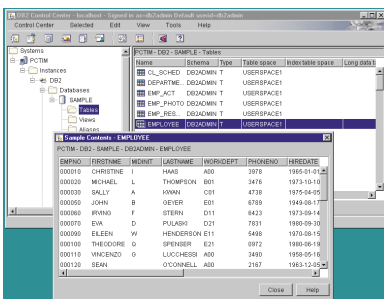
Microsoft may dominate client OSs, but in server databases it has been playing catch-up for many years. Ironically, one factor holding it back is that it only runs on Windows NT. It is still NT-only, but since version 7.0 SQL Server has been gaining ground as an obvious choice for the next step up from Microsoft Access or other file-based systems. The appeal of version 7.0 is partly that it is easier to use, since you can now use Access to manage an SQL Server database. There is a version for Windows 95 and 98. Another irritation, that of having to set the maximum size of a database, has been removed. Best of all, a cut-down version can be freely deployed, setting it apart from most other databases of this type.

SQL Server has been criticised in the past for its lack of scalability, but the Enterprise edition supports up to 32 processors and databases of a maximum theoretical size of over a million Terabytes, so that is no longer valid for most firms.



MySQL

MySQL is ideal for use on websites. Supported platforms include Linux and Windows and, apart from the Windows version, it is free for non-commercial use or as the database for a website. There are extensions for Hypertext Pre-Processor, which is another way to do server-side scripting, as well as for Perl. Java developers can use a JDBC driver. The main snag with MySQL is that you need to be happy working with command-line tools, since it does not come with graphical administration utilities. Having said that, there is enthusiastic user support for MySQL so a browse around the web might turn up some handy Java utilities.

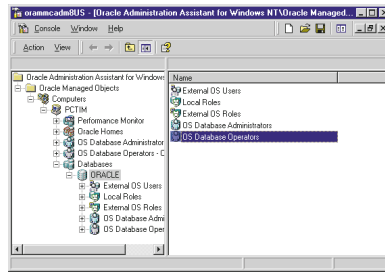


IBM DB2 Universal

IBM's server database runs on a wide range of systems including Windows 95 and NT, Linux, Solaris and AS/400 right up to high-end systems such as the OS/390. A personal Developers' Edition is available for free download. DB2 is solid, mature database technology that integrates with a range of IBM's hardware and software. There is support for replication, data warehousing, data mining, online analytical processing (OLAP), and large objects such as graphics or multimedia data. DB2 extenders are specialist extended data

types with many potential applications. DB2 is also characterised by strong Java support which translates to good web application support as well.

For Windows developers, the tools are not as easy to use as those for SQL Server, and while there is an ODBC driver there is no native OLEDB provider, for those wanting to exploit the latest Windows database API. It is also more expensive to deploy. On the other hand, DB2 scores well on the factors that count most, high availability and scalability.



Oracle 8i

Oracle is a dedicated database company and 8i runs on NT, and several Unix platforms including Solaris, AIX, HP-UX and Linux. The company has a curious relationship with Windows, taking every opportunity to berate its quality while providing strong support for the platform.

Oracle has gone even further than IBM in integrating Java into the database server. Oracle 8i has a Java Virtual Machine that runs within the server address space, for efficient data access from Java objects. Stored procedures can be written in Java. Three JDBC drivers are tuned for different purposes.

Oracle 8i comes in Standard, Enterprise, Workstation and Lite versions, the last being aimed at laptop or PDA users who want to synchronise data with corporate servers. Like DB2, Oracle is excellent for those wanting to move beyond Windows, but is more expensive to deploy than SQL Server.

PCW DETAILS

- Microsoft SQL Server**
Contact www.microsoft.com/sql/
- MySQL**
Contact www.mysql.com
- DB2 Universal**
Contact www.ibm.com/data/
- Oracle**
Contact www.oracle.com

Editor's Choice

The challenge with desktop database products is to provide a full range of features for managing relational data without losing the non-expert user in a jungle of incomprehensible dialog boxes. While all but one of the products covered here are competent, only two are outstanding.

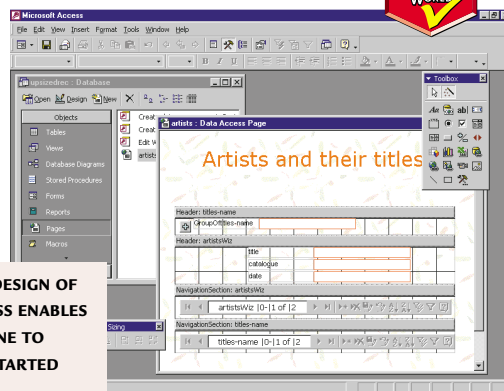
➔ **The Editor's Choice** is Microsoft Access. Although it could be better for novices, Access is sufficiently well-designed that anyone can get started, and sufficiently powerful that it can accommodate most desktop tasks. Plus points are the ability to use SQL Server as a native database source, which makes this the easiest way yet to migrate to a client/server system, the use of the widely understood Visual Basic for program code, and the comfortable fit with other Microsoft technologies such as Active Server Pages for intranet and web deployment.

➔ **Highly Commended** is FileMaker Pro from FileMaker. Its most impressive

feature is the thought that has gone into making the business of database management a common sense matter that most people can easily accomplish, at least for simple flat-file data. It is a little less powerful than Access, particularly in the scripting area, but there is an upgrade path to the server version, and FileMaker's web deployment feature is the easiest way yet to publish dynamic web pages.

➔ **Corel Paradox 9.0** deserves a special mention. This is a frustrating product that cannot be recommended except to developers, but with its extensive web integration features, it points the way forward to database management that is cross-platform and database-independent. The future of data management is web-based, and advanced features like these will matter

THE DESIGN OF ACCESS ENABLES ANYONE TO GET STARTED



increasingly as the era of desktop databases comes towards its close.

For information on ASP, ADO and COM, explore www.microsoft.com or try to grab an MSDN library CD set for online documentation.

Information about ColdFusion and JRun is at www.allaire.com. Both have cut-down versions for free download.

The best starting point for Java technology is www.java.sun.com.

Table of features

MANUFACTURER	MICROSOFT	LOTUS CORPORATION	FILEMAKER	COREL	SUN MICROSYSTEMS
PRODUCT NAME	ACCESS 2000	APPROACH MILLENNIUM	FILEMAKER PRO 5	PARADOX 9.0	STARBASE
Tel no	0345 002000	0870 600 6123	0845 603 9100	0800 973189	01276 451440
URL	www.microsoft.com	www.lotus.com	www.filemaker.com	www.corel.com	www.sun.com
Price (ex VAT)	£326.43 (£277.81)	SmartSuite £439.45 (£374)	£233.82 (£199)	£146.88 (£125)	Free download
FEATURES					
Client/server option	✓	✗	✓	✗	✗
Export to HTML	✓	✓	✓	✓	✗
Publish dynamic web pages	✓	✗	✓	✓	✗
JDBC driver	✗	✗	✗	✓	✗
Integrated web server	✗	✗	✓	✓	✗
Max database size	2GB	2GB	2GB	Not stated	Not stated
Integrated security	✓	✓	✓	✓	✓
Native data format	MDB or SQL Server	dBase	FileMaker	Paradox	StarBase
Connect to ODBC	✓	✓	Import only	✓	✓
ODBC driver	✓	✗	✓	✓	✓
COM automation server	✓	✗	✓	✓	✗
Programming language	Visual Basic	LotusScript	✗	ObjectPal	StarBasic
Script editor	✓	✓	✓	✗	✗
Runtime version available	✓	✗	✓	✓	✗
SQL support	✓	✓	✗	✓	✓
Linux version	✗	✗	✗	✗	✓
Macintosh version	✗	✗	✓	✗	✓