

4th Dimension, MS Access, and FileMaker Pro: A Comparison



4D TECHNOLOGY WHITE PAPER

Many individuals and organizations need database applications. Their needs can range from a simple database that tracks songs in an MP3 collection all the way to a multi-user client/server application that supports hundreds or even thousands of users as well as millions of records. The range of needs is quite broad so no database is ideal in all circumstances. However, most database applications fall in the middle range of system requirements. For instance, most applications have or will have multiple users. Most handle more than one file or table of related information. Most manage at least thousands of records. It is this middle ground of applications that are targeted by 4th Dimension, MS Access, and FileMaker Pro.

Personal computers have empowered individuals to do much on their own. More specifically, modern development tools make designing databases much easier than in the past. However, as many individuals begin to create database systems, they find that designing a database can be complex, regardless of the tool. The three products compared here attempt to minimize the effort needed to build a database.

On the other hand, users of a database application don't care how difficult it is to create. They frequently rebel against quickly designed applications that make their jobs more difficult. It often seems to users that developers tend to minimize development time at the expense of usefulness.

In database design, difficulty is an extremely subjective attribute. Most database developers use time as a measuring stick for comparisons. New database designers tend to focus on how "quickly" they can learn a database development system. While this is a completely valid component of measuring a database, the ease of learning a product is only one factor in evaluating a database design system. More experienced database developers consider other factors in their evaluations. How quickly can a database be developed? Once the database is completed, how quickly can a new user

learn the system? How quickly can an experienced user navigate through the application? How much time is needed to maintain and support the system? How useful is the resulting application? When evaluating tools for designing databases, more and more organizations are seeing the need to evaluate all of the attributes of a database design system as well as the attributes of the resulting database applications.

This document examines the following products: MS Access, FileMaker Pro, and 4th Dimension. Each product has its own strengths and weaknesses. In this document, we will focus on the ability of these three products to meet the needs of a middle range of system requirements. Specifically, we assume the following requirements: multiple users (2 to 100), multiple tables (2 to 50) and at least thousands or even tens of thousands of records.

TERMINOLOGY

Although each of the products examined in this white paper work with the same fundamental database components, they sometimes use different terminology. Some of the products mentioned here use the term "File" when referring to a database table. This white paper uses the term "Disk File" to indicate an individual document in the OS. Below is a brief chart that summarizes some of the differences in nomenclature between the products.

Item	MS Access	FileMaker Pro	4th Dimension
Table	Table	File	Table
Field	Column	Field	Field
Form	Form	Layout	Form
Automation	Macros/VBA	Calculations /Scripts	Methods
Relationship	Relationship	Relationship	Relation
Structure	Relationship Window		Database Structure
List of records	Datasheet view	Browse	Output form
Query using form		Find	Query by example
Query using editor	Query Builder		Query Editor
Selected records/rows	RecordSet	Found Set	Current selection

There are a few gaps in the above terminology. MS Access does not have an automatic way to query using an existing form whereas FileMaker Pro has no query builder other than using forms. FileMaker Pro also has

no mechanism for viewing multiple table definitions and their relationships in a single screen.

In addition to the items in the chart above, this white paper uses the term application to refer to everything that makes up a database program including table definitions, forms, programming, relations, etc. It also uses the term data to refer to the information entered by users.

CREATING A DATABASE

It is important to note that creating a database always involves the same fundamental steps. Regardless of how a database product approaches each step, developers of a database need to do the following in creating a database:

- Define tables of information and the fields in those tables.
- Design forms for on-screen use, including data viewing and editing.
- Design reports, labels, and other output, using forms or some other mechanism.
- Write scripts, macros, or methods to automate functions and cause records/rows to be processed.
- Customize the interface to make the system as easy to use as possible.

Defining Tables/Columns and viewing a database structure

MS Access uses a spreadsheet-like interface for defining the fields in a table. Each row in the interface represents one column in the table. Various field-level attributes can be specified. This method is fairly convenient, but is a separate operation from defining relationships. Users must use the Relationships window to see an overview of the tables that make up the application.

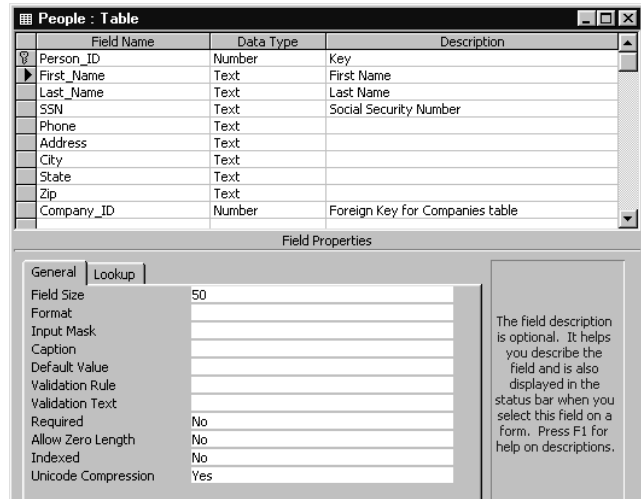


Figure 1: Defining fields in MS Access

MS Access provides an overview of the tables, columns, and relationships of a database through the Relationships window (Figure 2). After defining the tables in a separate window (Figure 1), the developer manually chooses the tables that should be included in this view. Dragging between fields creates relationships. To edit field or table definitions, the developer uses the table design window.

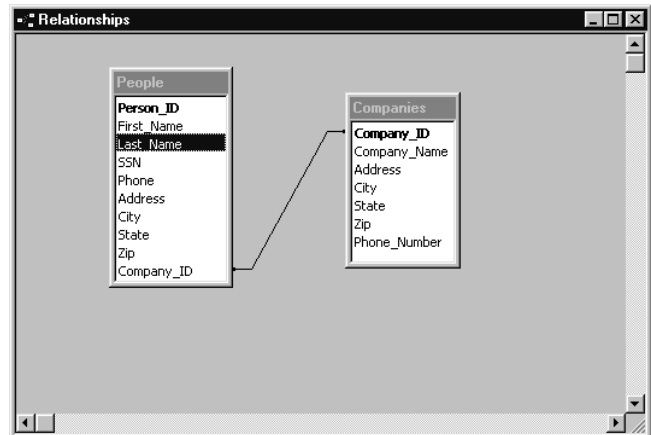


Figure 2: MS Access Relationships Window

FileMaker Pro uses a single dialog for defining fields. As with MS Access, each row represents a field in the table. Unlike MS Access and 4th Dimension, FileMaker Pro includes "calculations" as a type of field. Calculations are defined using a formula editor and are treated similarly to other fields. FileMaker Pro takes care of evaluating the calculation whenever needed. This is convenient, but can also be a problem. Since calculations are evaluated by FileMaker Pro, developers do not control when the

calculation is evaluated, which can hamper performance.

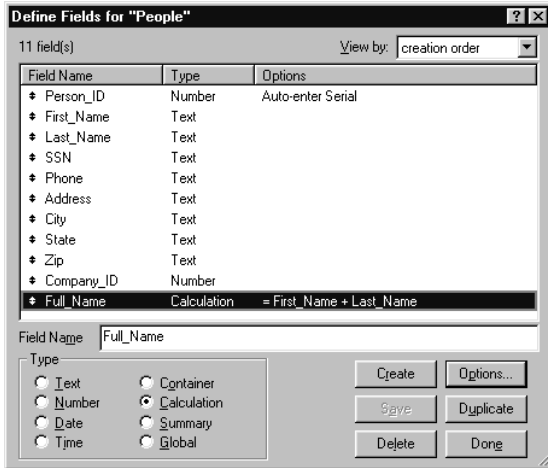


Figure 3: Defining fields in FileMaker Pro

FileMaker Pro places a lot of emphasis on field creation. This is where you define fields, calculations, validation, formats, etc. (Figure 3). While both MS Access and 4th Dimension have field level attributes, most attributes in those products can also be set on a per form basis. In order to get form-specific behavior from a FileMaker Pro field, a calculation must be created that determines which form is currently being displayed (Figure 4).

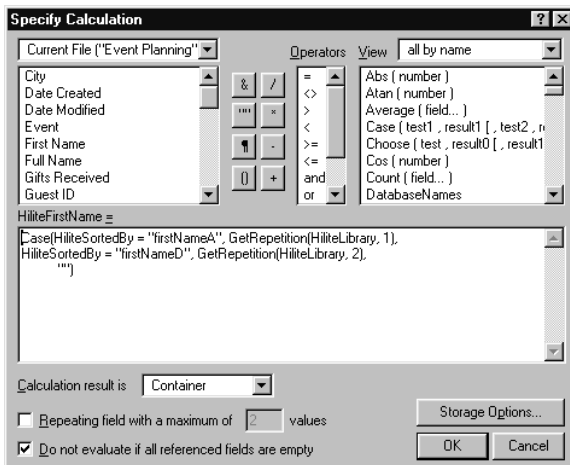


Figure 4: FileMaker Pro calculation editor

FileMaker Developer 5.5 introduces a new capability called the Database Design Report. This feature outputs an overview of an entire FileMaker Pro application as either a FileMaker Pro database or as XML output. Although a welcome addition for developers, this new capability lacks the interactive nature of the other two database applications. FileMaker Pro's "one table per disk file" approach is the source of this and other issues. For example, if even one of the files becomes misplaced,

it can play havoc with the entire application. Changing FileMaker Pro's multiple file structure is one of the most common requests from developers.

With no database overview, relationships are created in a series of dialogs (Figures 5 and 6) and can only be defined "one way." For example, to relate a People and a Company table in both directions, the developer must define two relationships each in a different file.

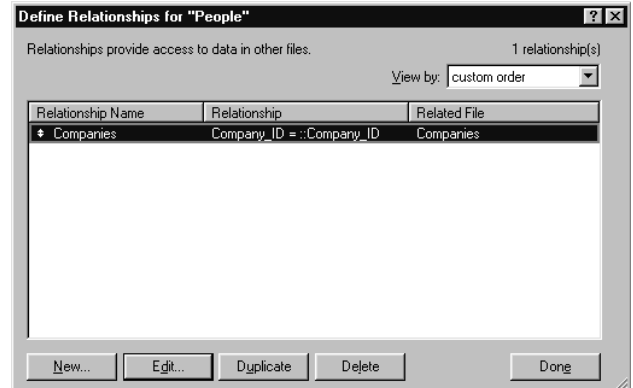


Figure 5: FileMaker Pro Define Relationships window

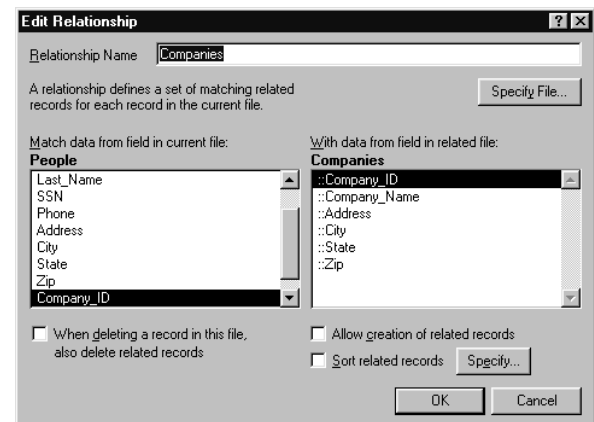


Figure 6: FileMaker Pro Edit Relationship window

Creating tables and fields in 4th Dimension involves the Database Structure window and the Field Properties dialog (Figure 7). A single click defines and positions a new table in the structure window. Selecting a table allows access to the Field Properties dialog where developers add and modify field definitions.

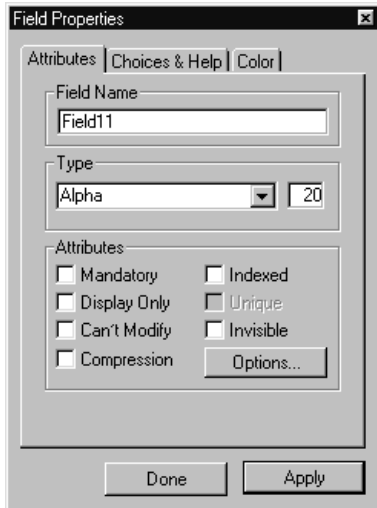


Figure 7: Defining fields in 4th Dimension

Since 4th Dimension uses the Database Structure window to define tables and fields (Figure 8), an overview of the database is created interactively. Like MS Access, dragging from field to field creates relationships.

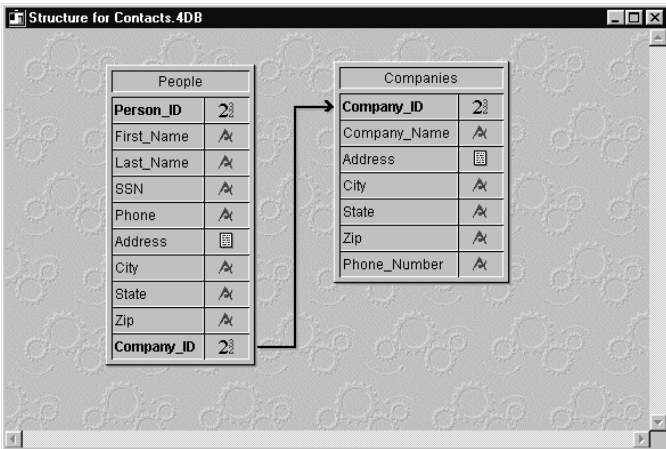


Figure 8: 4th Dimension's Database Structure window

Forms

A user most frequently interacts with a database through forms (screens). Forms provide an interface for users to add, modify, delete, examine, and manipulate data. From a database user's perspective, a database application is composed mostly by the sum of its forms.

Creating and modifying forms is one of the most common activities in designing a database application. Controlling the look, placement, and behavior of form objects is a significant task. Developing a database application requires some ability to understand screen layout, user interface design, and the most appropriate interaction between users and the data.

All three of the products provide a "Form Wizard" to aid in the creation of new forms. A form wizard is a screen or series of screens that lead the developer through the steps necessary to create different kinds of forms. Although this can be a useful learning tool, most developers usually start with blank forms or use the wizard to create simple forms as starting points. In either case, most of the work is done by manually editing forms. Below is a brief description of the form editing process for each of the products.

MS Access

Each form is edited in a separate window (Figure 9). A small toolbox window or palette displays icons that represent the type of form object to be placed (Figure 10). Placing objects is similar to working in a drawing program. Choosing the correct icon and then drawing a rectangle on the form creates a new form object.

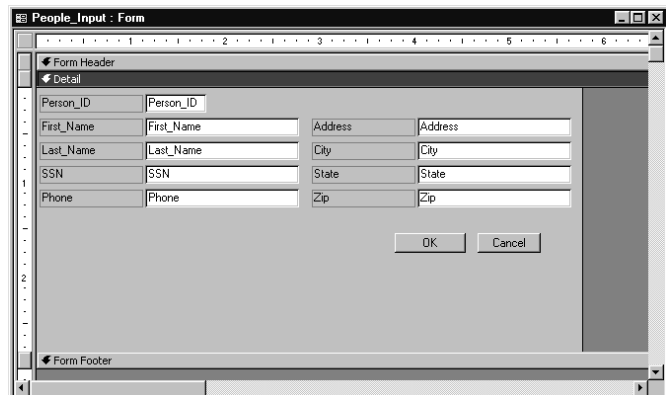


Figure 9: The MS Access Form Editor



Figure 10: The MS Access Toolbox

After an object is placed, a Properties window with several tabs (Figure 11) provides a way to set the various attributes that make up the form object. If more than one form object is selected, the Properties window shows fields for the attributes that the selected objects have in common. If the selected objects have different values for the same attribute, the value is shown as blank. This is a fairly convenient method for editing one or more form objects.

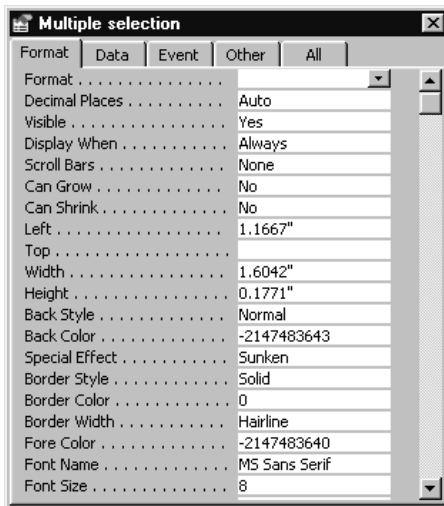


Figure 11: MS Access Properties window

In MS Access, the order that users tab through fields can be set in a special tab order window where the fields are moved up and down in a list (Figure 12).

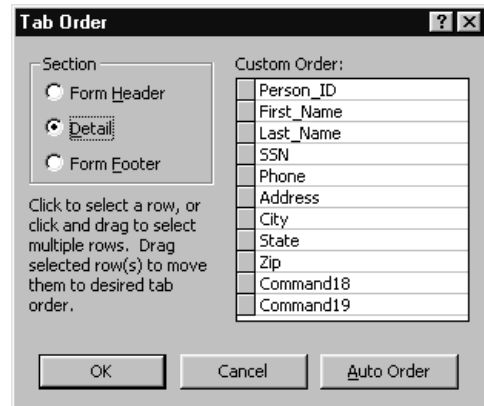


Figure 12: Setting the order that users tab through fields

FileMaker Pro

In FileMaker Pro, only one form per table can be edited at a time using FileMaker Pro's "Layout" mode (Figure 13). Developers place form objects using a palette of choices that are part of the form design window. Options and attributes for each of the objects are set by controls on the palette, items in the menus, items in pop-up menus, or by using attribute-specific windows.

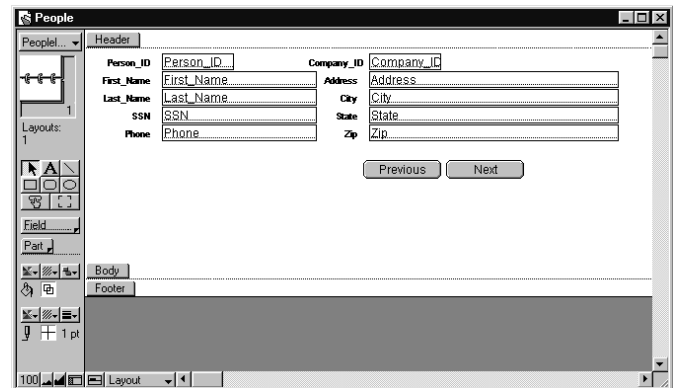


Figure 13: FileMaker Pro form editor

4th Dimension, MS Access, and FileMaker Pro: A Comparison

In FileMaker Pro, setting the tab order involves typing a number into small arrows that appear next to each field (Figure 14).

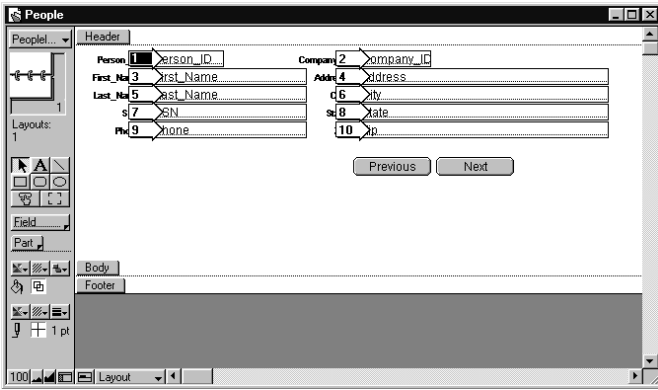


Figure 14: Setting the tab order in FileMaker Pro

4th Dimension

Like the other two products, 4th Dimension allows form editing in a draw-like environment (Figure 15). Objects are created by either selecting the object type from the Tool palette (Figure 16) then drawing in the form, or by dragging the object type from the icon to the form. In either case, the attributes of the resulting object can be set using the Properties List.

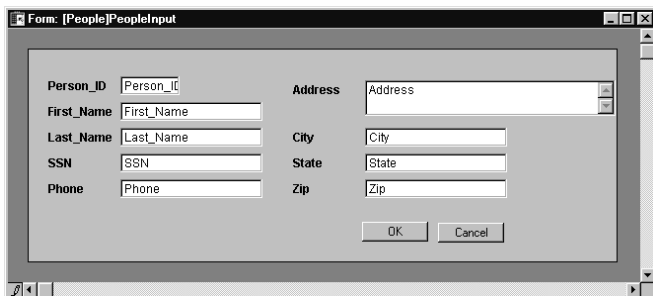


Figure 15: The 4th Dimension Form Editor



Figure 16: The 4th Dimension Tool Palette

The 4th Dimension Properties (Figure 17) List provides a single convenient window for setting all of the attributes of the selected form objects. The list changes based upon the selected objects. Form-level attributes can be set by selecting no objects.

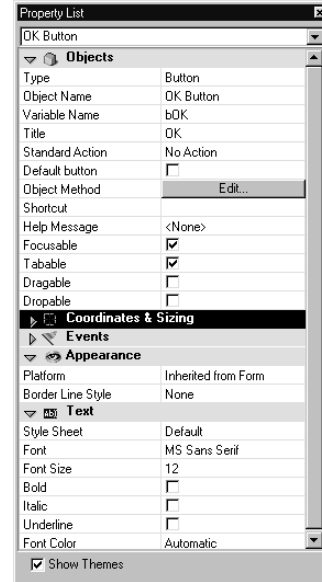


Figure 17: 4th Dimension Properties List

Like FileMaker Pro, setting the tab order in 4th Dimension is accomplished without leaving the form editor. However, the interface is more intuitive and easier to use. A line with directional arrows indicates the order of the fields (Figure 18). The developer simply drags from field to field to set the order, or can set the tab order of several objects by selecting them as a group.

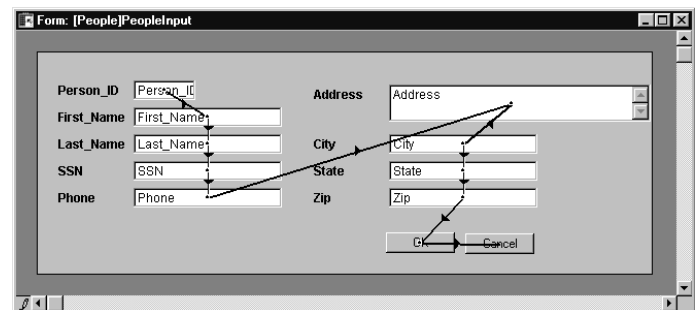


Figure 18: Setting the entry order in 4th Dimension

FORM OBJECTS

Form objects are the building blocks of forms. Form objects represent the answers to developer's questions: *Does the user need to choose from among several items? Will the user need to type information? Should the user press a button to start a task?* Developers ask these questions as they design forms and answer them with form objects.

All three of the products examined here provide varying types of form objects. Each type is designed to meet a particular interface need. Products with more object types tend to be more flexible, resulting in applications that are easier to use and more powerful. Below is a comparison of the form objects available in each product.

Form object type	MS Access	FileMaker Pro	4th Dimension
Text Box	Yes	Yes	Yes
Buttons	Yes	Yes	Yes
Checkbox	Yes	Yes	Yes
Radio Button	Yes	Yes	Yes
Toggle Button	Yes		Yes
Group Box	Yes		Yes
Combo Box	Yes		Yes
Popup Menu		Yes	Yes
Graphical Popup Menu			Yes
Hierarchical Popup Menu			Yes
List	Yes	Yes	Yes
Hierarchical List			Yes
Tab Control	Yes		Yes
Subform/Portal	Yes	Yes	Yes
Indicators			Yes
Button Grids			Yes
Splitters			Yes

As shown, 4th Dimension provides the greatest number of form object types. In addition, all three of the products listed above support plug-ins of one kind or another. In 4th Dimension and MS Access, plug-ins can be used to create new object types as well as new commands. In FileMaker Pro, plug-ins can only be used to create new calculation functions.

Interface Flexibility

In addition to a variety of form objects, all three of the products provide a degree of interface flexibility or customization. FileMaker Pro is the most limited in this area. With FileMaker Pro you can disable menu items and change form design but the basic user interface remains the same. You cannot replace menus or menu items.

In FileMaker Pro, each form is an integral part of a table. This makes using fields on forms simple and direct. When a record is opened the fields respond automatically. FileMaker Pro "knows" that the form belongs to the table, which simplifies several database operations. However, FileMaker Pro has some limitations with respect to forms. For example, only one form can be open for a table at a time. There is no way to have sepa-

rate, active windows for the same table.

MS Access allows greater flexibility in interface control. Unlike FileMaker Pro, MS Access can accomplish the following user-interface tasks:

- Open a window in a specific location.
- Change the entire menu bar on the fly.
- Format items conditionally based on content.

However, this flexibility comes at a price. In MS Access, each form is completely independent of any particular table. This means that the relationship between the form objects and the underlying data must be defined on a per form basis. This can be a cumbersome process.

4th Dimension has the best of both worlds. Like FileMaker Pro, each form belongs to a table. This makes adding and using fields in a form very easy. Unlike MS Access, the relationship between the form and the record/row is automatic. But unlike FileMaker Pro, 4th Dimension provides extensive control over the interface. Some of 4th Dimension's interface capabilities include:

- Position, size and type control of windows
- Multiple active forms per table
- Conditional visibility on a per object basis
- Multiple data entry pages per form
- Complete control of menu bars, menus, and menu items
- End-user controlled form-object resizing
- Form object specific Tip Help
- Custom tool bars
- Style sheets that control appearance independently for each platform
- Automatic object resizing based on window resizing
- User defined keyboard equivalents for menu items, buttons, fields, etc.
- Multiple custom validation messages per field

Security

Security is another area where the three products differ. In FileMaker Pro, access privileges are set on a per file basis. To change a password for a FileMaker Pro solution involves opening and changing the password in each separate table. In addition, FileMaker Pro only supports passwords. There are no user name/password combinations as with other applications.

The security model in MS Access is more traditional. You can define users with passwords and groups of users. Privileges are assigned to both the users and the groups. This model is fairly flexible.

4th Dimension, MS Access, and FileMaker Pro: A Comparison

4th Dimension approaches security in almost the same fashion as MS Access. Users are defined with passwords. Groups are defined as a set of users but can also contain other groups. One significant difference in 4th Dimension is the ability to use SSL encryption and authentication between 4D Client and 4D Server. This feature insures that information passing between the user and the database is not intercepted nor altered.

INPUT MASKS

Input masks control the characters that can be typed into an editable field or text box. For example, when a user types into a SSN field an input mask might require that only numbers be typed and that there must be nine of them. Both 4th Dimension and MS Access support input masks. FileMaker Pro does not.

Reports

The term *reports* can refer to many different things but generally refers to data organized and formatted for printed output. Although reports are the most common kind of database output, there are many types of reports. Traditional reports are often tabular or grid-like. Detail reports include one record per page. Some reports include subtotal sections that summarize the prior rows. Some reports include a header section followed by a repeating subsection such as an invoice. The degree of flexibility provided for report creation translates directly into the quality of reports.

Reports in MS Access are created identically to forms. Developers place report components using a graphical interface. The resulting report is stored as part of the database application. An MS Access report can be tied to an MS Access Query to provide a means to find and print the correct rows.

FileMaker Pro reports use the form editor itself. In fact, the difference between a form and a report is merely a matter of use. Developers specify the fields and other items that make up the report just as they would a form. Like MS Access, the resulting report becomes part of the database application.

Like MS Access and FileMaker Pro, 4th Dimension uses the form editor to create reports. The process of specifying headers, detail, and subtotals is very similar to the other products. Literally any kind of report can be generated using the form editor. 4th Dimension also provides a built-in graphing tool that allows 2-D and 3-D graphs to be included side by side with other information in a report.

Although designing reports using forms is powerful and flexible, it requires design-level access and the presence of the database developer. Yet database developers can rarely foresee all of the reports that are needed in an application. Invariably, users require new reports over the life of a database. With all three of these products this means creating the report off-line and then installing a new version. With FileMaker Pro and MS Access, installing a new version is cumbersome. Something else is needed.

Unlike MS Access and FileMaker Pro, 4th Dimension provides a simple end-user report generator. Called Quick Reports (*Figure 19*), the report generator allows end-users to create their own tabular type reports that include common items such as subtotals, counts, etc. as well as formatting options and control over appearance. The 4th Dimension Quick Report system can be added to any database application and allows output to printers, disk files, and graphs.

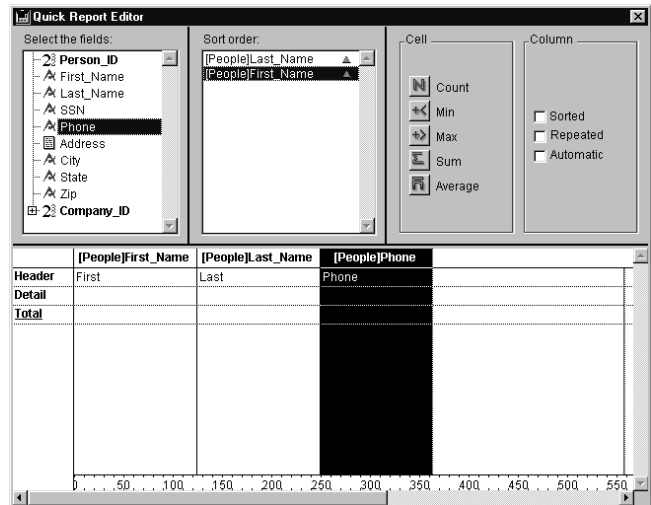


Figure 19: The 4th Dimension Quick Report Editor

Labels

Although labels are not as common as reports they are still an important form of output in many databases. Unlike reports, labels are printed several times across as well as down a page. All three of the products compared in this document allow the developer to create mailing labels using forms. In FileMaker Pro and 4th Dimension, developers design a form that represents one label. A control line sets the width of the label. During printing, the database places as many labels as will fit across the page. In MS Access, the width of labels is defined using page setup.

Regardless of the product, defining labels using a form requires design-level access to the database structure. But what if end-users need the ability to design labels? Like the Quick Report window, 4th Dimension's Label Editor (*Figure 20*) is a simple interface that allows end-users to be self-sufficient.

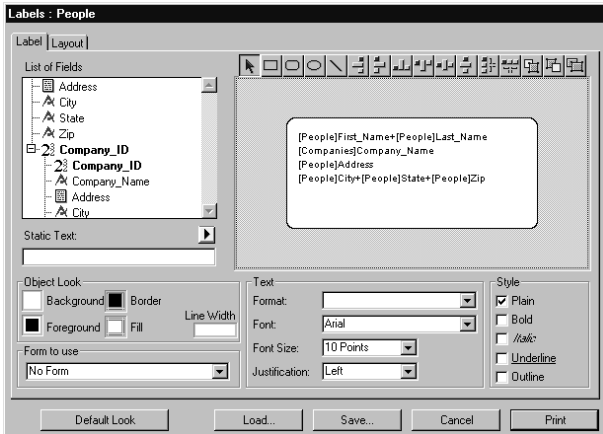


Figure 20: The 4th Dimension Label Editor

Automation

Most database applications require some amount of automation. In fact, a database would be a fairly limited tool if it provided no way to allow users to perform multiple steps, functions, or processes in succession. Automation is one area in which these products differ substantially since each has a different approach.

The built-in automation mechanism in MS Access is macros (*Figure 21*). An MS Access macro is defined in a spreadsheet-like interface similar to MS Excel macros. Choosing from a series of macro steps and then filling in the blanks creates each macro. This method is similar to creating scripts in FileMaker Pro. Although some operations can only be performed by macros, they are somewhat limited, which leads most developers to use Visual Basic for Applications in automating MS Access databases. Even MS Access's Database Wizard, which uses templates supplied by Microsoft, tends to create applications with no macros and much VBA code.

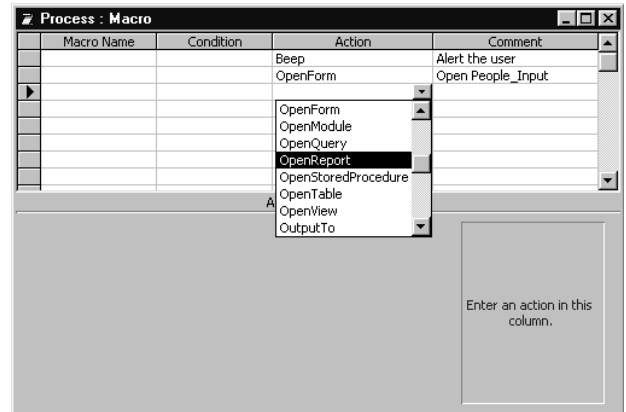


Figure 21: MS Access macros are defined in a spreadsheet-like interface

Visual Basic for Applications is the programming environment for all MS Office applications. It is a deep, complex product but it is very powerful and provides control over most aspects of an MS Access database. When VBA is used in conjunction with MS Access, new programming objects are added to manipulate the database. Working with VBA in conjunction with MS Access can be bewildering. Since VBA is a complete programming environment that is not specifically targeted for use with MS Access, a developer that has Visual Basic experience has a head start. However, a developer who isn't as familiar with VBA must learn a new programming environment that is not tailored to designing databases.

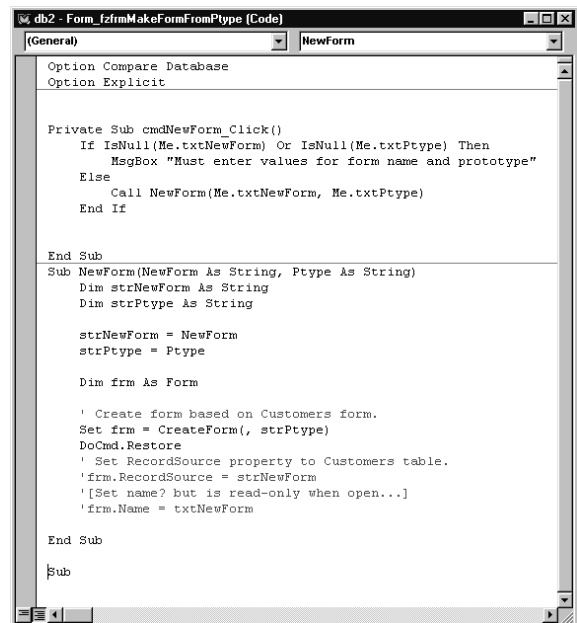


Figure 22: Visual Basics for Applications code window

FileMaker Pro approaches the task of automating a database using calculation fields and scripts. Calculation fields are defined during table creation and can be likened to the formula in a cell of a spreadsheet. Whenever one of the source values used in the calculation changes, FileMaker Pro re-evaluates the field's value. Calculations are used extensively in FileMaker Pro and are often used in conjunction with scripts.

Defining a FileMaker Pro script (Figure 23) is very similar to creating an MS Access macro. But since FileMaker Pro has no programming language, much more emphasis has been placed on scripts.

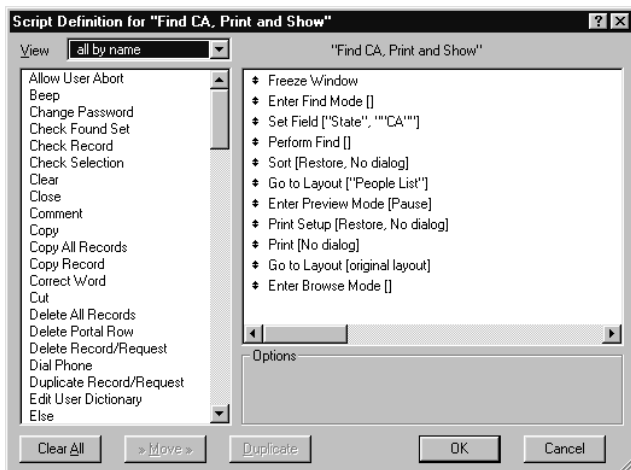


Figure 23: A FileMaker Pro script is similar to an MS Access macro

Like FileMaker Pro, 4th Dimension's automation mechanism is built-in and specific to creating database applications. However, 4th Dimension's method-based programming language is far more flexible than FileMaker Pro scripts.

Automating a 4th Dimension database is done by creating methods. Programming a method is similar to creating VBA code but unlike the MS Access/VBA relationship, 4th Dimension's methods are built-in and very tailored to database automation.

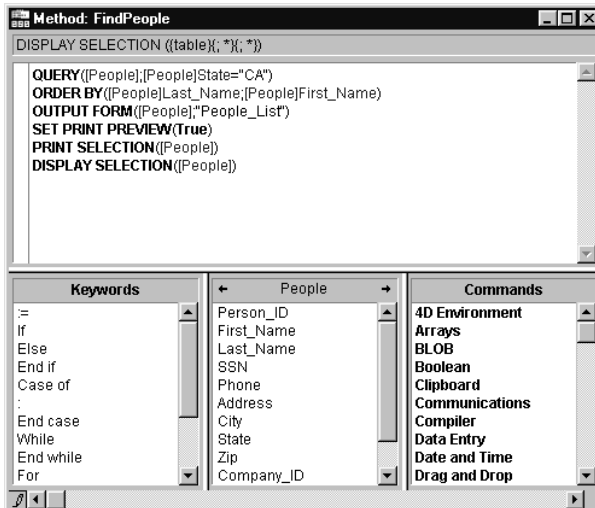


Figure 24: The 4th Dimension method editor

The 4th Dimension programming language is very readable. 4th Dimension methods are used for a variety of purposes. In one method you can automate, calculate, and validate data. Like FileMaker Pro scripts and VBA Code, 4th Dimension methods can be attached to form objects. (In the 4th Dimension form editor, form objects with attached methods are indicated visually).

EVENTS

The term "event" is used to describe something that triggers a database operation. FileMaker Pro's events trigger scripts. In MS Access, events cause VBA code to execute. In 4th Dimension, events cause methods to run. Regardless of the terminology, the idea is the same; something happened or is happening and therefore some automated steps need to occur.

Probably the most common events are clicking a button or choosing a menu item. When the user causes one of these events, the script/module/method indicated runs and performs some action. This action can open a different form, update several records, query another table, or perform any number of steps supported by the database product.

As with other database features, the number and types of events that these three products address varies greatly. FileMaker Pro is fairly limited. Its lack of events has resulted in the creation of plug-ins that perform similar functions. Both MS Access and 4th Dimension are far more mature in this area. Both provide a variety of events, some of which are triggered by users and some that are caused by other operations.

As with other database features, the number and types of events that these three products address varies greatly. FileMaker Pro is fairly limited. Its lack of events has resulted in the creation of plug-ins that perform similar functions. Both MS Access and 4th Dimension are far more mature in this area. Both provide a variety of events, some of which are triggered by users and some that are caused by other operations.

Like other aspects of database design, flexibility in event handling allows developers to tailor a database interface to meet users' needs. The result is a simple interface that responds the way users expect, reducing training costs and mistakes and increasing productivity.

	MS Access	FileMaker Pro	4th Dimension
Form Opening	Yes		Yes
Form Closing	Yes		Yes
Object Clicked	Yes	Yes	Yes
Object double-clicked	Yes		Yes
Printing Form/Region	Yes		Yes
Field/Text Box Modified	Yes		Yes
Getting Focus (field entered)	Yes		Yes
Losing Focus (field exited)	Yes		Yes
Drag Over	Yes		Yes
Object Dropped			Yes
Keystroke (each character as user types)	Yes		Yes
Resize Form	Yes		Yes
Resize Form Object			Yes
Timer Events	Yes	Apple Events or batch files on FM Server only	Yes
Error Events	Yes		Yes
Form Activated/Deactivated (user click in other window/form)	Yes		Yes
Menu Item chosen		Script Menu	Yes
Close Box clicked		Yes	Yes
Open Database	Yes	Yes	Yes
Exit/Quit	Yes	Only as file windows are closed	Yes
Server Startup			4D Server
Server Shutdown			4D Server
New client connection			4D Server
Client disconnection			4D Server

VARIABLES

Although database applications deal in tables and fields, there are times in a database application where other information needs to be temporarily stored and manipulated. FileMaker Pro provides the concept of global fields. A global field is a memory-based storage cell associated with a table. Each user can have a different set of global values. The value in each global is persistent while the file is open. Each global contains a single value.

MS Access can store non-field values in an unbound control (form object) while the form that contains the control is open. The value stored is independent from other controls on other forms. Using VBA allows developers to define independent variables with different types.

4th Dimension also lets you define variables wherever needed. The values in 4th Dimension variables can be specific to the method, to the process or even interprocess in

scope. Furthermore, 4th Dimension can define a separate set of variables on 4D Server.

DEBUGGING

Debugging is the process of testing an application and fixing any problems. This process is usually assisted by a debugger. A debugger is a special interface that allows the developer to step through an automated process and view various pieces of information as things change. Most debuggers let you view the method/module/script that is currently running and the values in fields and variables. Usually a debugger will provide an "evaluation" area where the developer can type in an expression and view the result. In addition, most debuggers provide a means to set "breaks" in the code so that execution pauses at a certain point.

Debugging in MS Access is done through Visual Basic for Applications. VBA provides several different dockable windows or panes to use during debugging. These include:

- The Immediate window which provides an area to evaluate expressions.
- The Watches window where the developer can enter expressions that pause execution of the database at a certain point.
- A Locals window that provides a hierarchical view of the various database objects, their values, and their types. It also shows the name of the currently executing code.

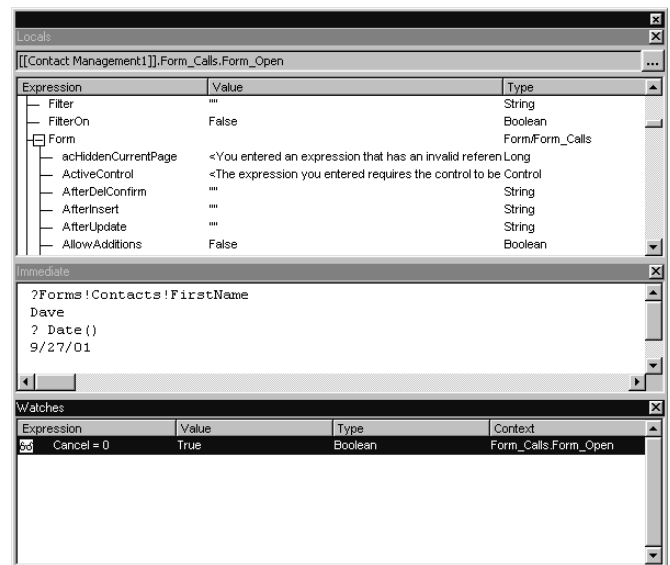


Figure 25: Visual Basic debugging panes

4th Dimension, MS Access, and FileMaker Pro: A Comparison

In addition, VBA provides

- The code window that displays the code that is currently executing.
- The Call Stack window that shows the hierarchy of calls.

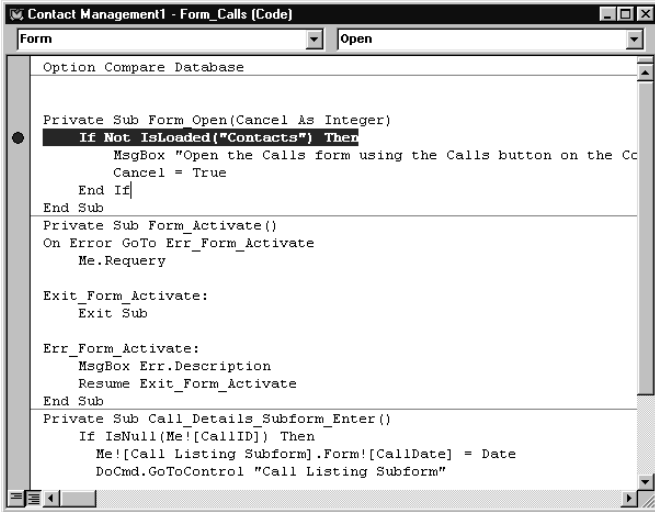


Figure 26: VBA code window in trace mode

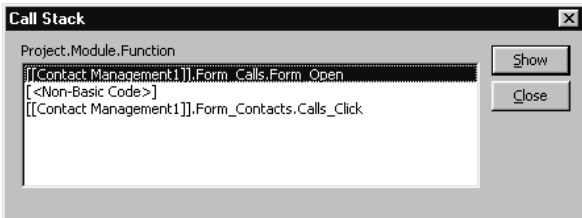


Figure 27: VBA call stack window

In general, MS Access does a pretty good job of helping developers debug their applications. The tools are a bit scattered and are limited, but for the most part debugging is supported.

FileMaker introduced a debugger in the 5.5 version of FileMaker Developer. Although this is a welcome addition to FileMaker Pro, it is fairly limited. The FileMaker Pro debugger shows the contents of the currently executing script as well as the calling chain, referred to as Active Scripts in the debugger.

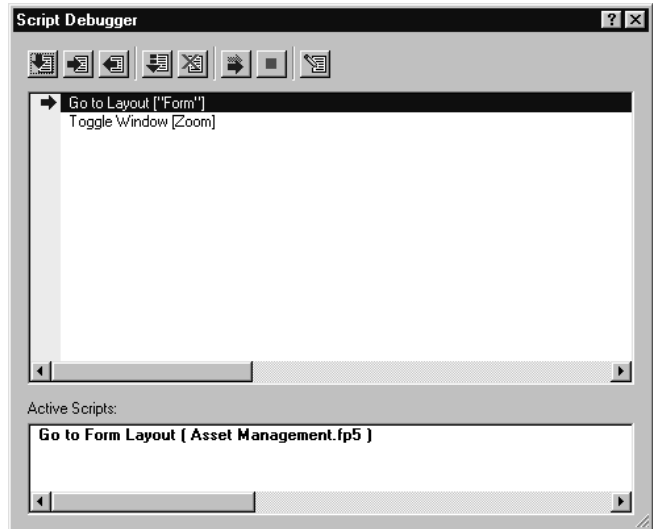


Figure 28: The FileMaker Pro debugger

Of the three products, 4th Dimension provides the most comprehensive debugger (Figure 29). The 4D Debugger is divided into 4 panes: Watch pane (upper left), custom watch pane (middle right), call chain pane (upper right) and source code pane (bottom).

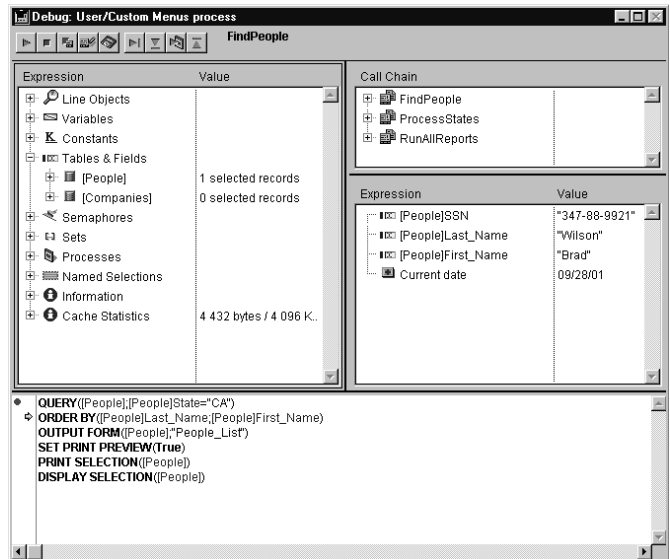


Figure 29: The 4th Dimension debugger

The Watch pane provides a comprehensive list of every object in the database application along with their current values. It also provides information on the state of memory and other application wide settings. The Custom Watch pane allows developers to type any valid expression and have it evaluated by 4th Dimension immediately. The Call Chain pane shows the chain of calling methods and allows the user to view any level of the chain by double-clicking. The Source Code pane shows the currently executing code and can be used to set break points where execution is interrupted and the debugger is displayed.

RELATIONAL ABILITIES

Access has good relational abilities. Tables can be joined using standard SQL statements. The result is a standard SQL RecordSet, which represents rows of data (not necessarily records).

FileMaker Pro's relational abilities are more limited. FileMaker Pro's relations are pre-defined equations that require an exact match between two tables. Since relations are the only direct way to access data from one table to another in FileMaker Pro, developers have come to rely on them when several records of information in another table are needed. The resulting related records are displayed in a "portal" which is a grid-like interface object that appears on a form. Because FileMaker Pro relations and portals are one of the few relational abilities in FileMaker, they are often used in a manner beyond how they were intended. They have become Band-Aids for much that is lacking in FileMaker's relational abilities as well as other areas. For instance, since FileMaker Pro lacks the ability to query one table from another, relations take on this role.

4th Dimension's relational abilities have aspects of both MS Access and FileMaker. Like MS Access, 4th Dimension can use information in related tables, and even unrelated tables, at runtime. Many operations have the option of considering related information if need be. But also like FileMaker Pro, 4th Dimension includes "Subforms." This flexible interface object allows a listing form to be included in another form. It can display information from related tables or even unrelated tables. However, unlike FileMaker Pro, 4th Dimension's subforms do not have to rely on a pre-defined "relation" and can display any records that are needed.

MULTIPLE USERS

This document assumes the need for an application that allows multiple users to access the data at the same time. Even single-user applications are usually designed as if they will one day be used in a multi-user environment.

Standard MS Access uses a file server to support multiple users. The database application, data, and/or MS Access itself can be placed on a file server that is accessible to several users. When users need to use the shared information, the file server provides the proper disk files through the network. The more portions of the database that are placed on the file server, the slower access is. Most database developers that use a file server put only the data on the server.

Access can also act as a "front-end" to a separate MS SQL Server database or any ODBC data source. This option is more expensive and more complex than the scope of this document allows. Like MS Access, 4th Dimension can also behave as a front-end to an MS SQL Server database. Please see the white paper entitled "4th Dimension and SQL Server" for more information about connecting 4th Dimension to MS SQL Server.

Each copy of FileMaker Pro has built-in peer-to-peer multi-user support. Through a proprietary protocol, FileMaker Pro can host a small number of users and/or tables without a central server. In very small applications, this can be useful. However, FileMaker Pro is not multi-threaded. This means that the hosting FileMaker Pro database can only be doing one thing at a time. If a user on the host machine performs any operation, all other users will have to wait until the operation is completed.

FileMaker Server allows a greater number of users and tables to be hosted. FileMaker Server is multi-threaded so the server doesn't force client requests to wait while it processes others. This is the most efficient FileMaker Pro setup for multi-user databases and is similar in concept to 4D Server. However, unlike 4D Server, FileMaker Server only processes standard FileMaker Pro data operations on the server.

A multi-user 4th Dimension database is served using 4D Server. Users connect in much the same way as FileMaker Pro clients. However, 4D Server allows users to choose to run any operation on the local 4D Client or on 4D Server. The 4D Server interface provides information on the status of the various users and allows information on each of the processes to be examined using the Runtime Explorer.

Even though a 4th Dimension application can be designed using the stand-alone version, 4D Server provides a comprehensive multi-developer environment. The 4D Server design mode allows several developers to collaborate on a database application and provides object-level locking for all methods, forms, menus, etc. Also, you can use the debugger on both the 4D Client side as well as 4D Server.

4th Dimension, MS Access, and FileMaker Pro: A Comparison

In 4th Dimension you can create login documents that bypass the various login screens and simplify logging into a 4D Server. In FileMaker Pro you create a separate "opener" file that contains a script which opens a hosted database application to do something similar.

APPLICATION AND DATA IN A MULTI-USER DATABASE

One of the most significant differences between the three products examined here is the way they store the database application and the database data. For the sake of this paper the "Application" of a database is everything that makes up a database program including table definitions, forms, and code (scripts/methods/macros). The database "Data" is all the information entered by users.

FileMaker Pro has one disk file for each table in the application. Each file stores a single table definition as well as forms, scripts, and relationships. In addition, each of these files stores the corresponding table data. This configuration is very problematic. In order to change a database's functionality, the designer must know in which file the form/script/relationship is stored. And since there is no FileMaker Pro "overview" capability, it is very difficult to get a complete picture of the overall application.

Even more important, once an application is in use the data is stored in the same file as the table definitions, forms, scripts, etc. This leaves developers in an undesirable position. They can either make changes while the system is live (not recommended) or they can work off-line on a copy of the system. If they work on a copy of the system, installing the changes involves getting all users out of the application, exporting the data from the affected files, replacing the affected files then importing the data. In the very least this is cumbersome.

MS Access can store the application and the data in separate disk files. To achieve reasonable performance, most developers place only the data file on the file server and provide a copy of the structure on each user's computer. Developers can work on the application off-line and can install new versions without exporting and importing. However, since each user usually has a separate copy of the structure, developers must update each machine with the changes. Many developers use other distribution products to accomplish this, which can add significantly to the cost. Another option is to have MS Access act as a client to an MS SQL Server back-end database. This option is quite a bit more complex and costly and not covered in this document

Like MS Access, 4th Dimension also separates the data from the structure of an application but both are served using 4D Server rather than a file server. 4D Server provides high-

er performance than a file server does because queries and other operations are performed on the server without affecting the network. When installing new versions of the database, users logout and the structure is replaced. When users log back into the system they are automatically updated with the new structure. As each user visits portions of the application, that portion is downloaded to the client machine and cached for even greater performance. This is only one area where 4th Dimension shows a significant advantage over the other two products.

TRANSACTIONS

Transactions are a way to group multiple data changes (additions, modifications, and deletions) into a single unit that can then be committed in its entirety or rolled back (cancelled). Transactions are important for keeping data consistent in a database. For example, a traditional structure for an invoice involves two tables, one for the invoice master data (such as invoice date and invoice number) and one to hold the line items in the invoice (such as the item and quantity purchased). Each invoice is made up of one invoice master record and any number of line item records. It is quite simple to see that there would be problems if the database allowed line items to be saved without the invoice master and conversely, there would be no purpose to saving an invoice master without the line items that go with it.

This is where transactions become important. With a transaction, the invoice master as well as all line items would be stored in a temporary buffer. At the end of data entry the entire transaction, both invoice and line items, is committed to the database.

Both MS Access and 4th Dimension support transactions while FileMaker Pro does not. With FileMaker Pro, all changes to a record's data (including the creation of the record itself) are committed as soon as the client exits the record.

FINDING INFORMATION

Finding information in a database is probably the most common task. In many products the act of finding is referred to as a query. The word query means something slightly different in each of the above products but it almost always involves finding records/rows that meet specified criteria.

In Access, queries are stored objects. They are defined in an interface that help you visualize the tables of your database and then saved in the database structure. Each query describes the tables where the information will be found, the columns that will be included in the result, and/or the order. Even though a query in Access is defined prior to its

use, it can include placeholders that are "filled in" when the query is used. An Access Query results in a SQL statement that can be viewed in a separate window. The result of an Access query is a RecordSet. The information in a RecordSet does not necessarily represent records in the underlining table.

In FileMaker, queries are performed using Find mode. From a user's perspective, Find mode involves entering values into a "Find Request" using an existing form. Fields that are filled in become part of the criteria for the Find Request. If the query requires a logical OR, multiple Find Requests are filled out in succession. The result of a FileMaker query is a found set. Users can also omit records from the found set. Once a set of records is found, multiple operations can be performed without re-querying the data.

In 4th Dimension, queries are performed in a variety of ways. Like FileMaker, 4th Dimension's Query by Example function allows users to fill in information on a data entry form and use the information as criteria for a query. 4th Dimension also provides a Query Editor. This one screen interface provides a simple criteria builder and takes into account the various relationships between all of the tables in the database. Similar to FileMaker Pro, 4th Dimension queries results in a current selection, which can be used in subsequent operations without re-querying the data. However, unlike FileMaker Pro, 4th Dimension allows new criteria to be applied to the current selection, further refining the selected records. In addition, 4D provides a simple means for users to manually select any number of rows/records and then change the selection to include the records chosen. This manual approach to row/record choosing is a very convenient addition to formal queries.

CROSS-PLATFORM

One area where these products differ greatly is in their support for different operating systems or platforms. MS Access, as a Microsoft product, only runs under various flavors of Windows. FileMaker Pro runs under both Windows and Mac OS as does 4th Dimension, 4D Client, and 4D Server. However, 4th Dimension's cross-platform abilities far exceed FileMaker Pro's and make the use of a single application under several OS's a breeze.

4th Dimension's cross-platform abilities include:

- Style sheets that control font, size, style, etc. on a per OS basis.
- Appearance control settings that let forms appear and function natively on each platform.
- A compiler that transparently generates machine-specific

code for each platform and combines them into a single file for easy distribution by 4D Server.

CONCLUSION

The key to a good database system is balance. A good database strikes a balance between the needs of developers and the needs of end-users. A good database offers flexibility and ease of development. A good database provides a comprehensive set of features without making database development nor database use more difficult.

FileMaker Pro is fairly easy but limited. MS Access provides greater capability in some areas but less in others and is much more complex. 4th Dimension provides the greatest balance. It has much of the ease of development of FileMaker yet has a larger feature set than either of the other two products.