



DataPipe Technical Q & A

EHS Information Management Software

2/1/2008

M. Douglas

 **DataPipe™**
The Complete Information Management Software Solution



**OCCUPATIONAL HEALTH
AND MEDICINE**

**ENVIRONMENT AND
WASTE MANAGEMENT**

**INDUSTRIAL HYGIENE
AND SAFETY**

Microsoft®
CERTIFIED

Partner

ISV/Software Solutions

Table of Contents

ARCHITECTURE..... 1

DATABASE SETUP & CONFIGURATION..... 3

DATABASE UTILIZATION & BENCHMARKING 5

SECURITY 6

DISASTER RECOVERY 10

MISCELLANEOUS..... 10

ARCHITECTURE

Q. What type of architecture does the .NET version of DataPipe use?

A. DataPipe is based on Microsoft's .NET technology. Please refer to the document titled **DataPipe Architectural Overview** for a complete overview of DataPipe's architecture. The document is freely available upon request.

Q. What are the system requirements for DataPipe?

A. Here again, please refer to the document titled **DataPipe Architectural Overview**, where client and server requirements are outlined.

Q. What features are built in to DataPipe to improve performance?

A. DataPipe has been designed from the ground up to offer the best performance possible, taking in to consideration network bandwidth, the web server and the database. DataPipe can address bottlenecks in an IT environment to maximize performance depending on how it's deployed. The various DataPipe deployment options are discussed in the **DataPipe Architectural Overview**. If you decide to deploy DataPipe via Auto-Deploy, for example, application code gets downloaded and is persisted on the client machine. The next time you run DataPipe, it will check if there are any newer versions of requested application code available on the server. If so, the newer code is automatically downloaded. If not, it uses what was downloaded last. So, initially the application code gets downloaded but subsequently the only thing going across the network is data. Also, application code is only downloaded as it's needed. You aren't forced to download the entire application when you first run it. The intent is to absolutely minimize network traffic, offering the best performance possible. This typically produces much better network bandwidth utilization than a straight HTML-based solution, in which case an entire form's content (user interface as well as application logic) needs to travel across the network every time that form is requested.

Q. We don't have the resources to install and host the DataPipe web/application and database servers internally. Can DataPipe be hosted by an Application Service Provider (ASP)?

A. Yes. A trend in recent years has been to outsource the application server hardware and software to an ASP to address the limited IT resources available to host applications internally in an organization. There are several ways we can address this but it's important to have a firm understanding of what using an ASP implies.

If you're using an ASP you're going to want to make sure that you have adequate bandwidth available to go out to the Internet to connect to the ASP. Applications hosted internally in an organization typically have more available bandwidth than what's available to them once they go outside the company's network boundary. Bandwidth is typically the bottleneck when using internet-based applications. We have a reliable and highly professional ASP that we've worked with in the past and can make recommendations regarding the different packages they offer. If you have a preferred ASP, you're also welcome to use them. We will work with you and them to try and address any potential hosting problems in using them to host DataPipe. The ASP will typically provide you with access to a Windows Server that has Internet Information Server (IIS) on it. A database

will also need to be installed at the ASP. This is one of the issues you'll want to consider with an ASP. They're hosting the database that contains your DataPipe data. It may not be as easy to get at the data if it's residing at an ASP as opposed to within your organization. There are several other issues you'll want to consider before deciding that using an ASP is the best way to host DataPipe. We can work with you to help determine what options offer the best comfort level for everyone.

Q. Is there any way to determine if DataPipe will run in our environment?

A. We offer a **DataPipe System Check Wizard**, which is an ASP.NET application that needs to be installed and accessed from a browser. There are five tests that the wizard performs in order to determine if the basic pieces are in place to run the full version of DataPipe. It checks that the browser it's being run from is Internet Explorer 5.01 or above, that the .NET Framework has been installed on the server, that the .NET Framework has been installed on the client, that it can connect to the database and that it can issue a simple SELECT statement against the database.

Q. The DataPipe Architectural Overview discusses an Auto-Deploy option. What exactly is Auto-Deploy and how does it work?

A. Auto-Deploy (also referred to as No-Touch Deployment and Zero Deployment) is a Microsoft technology that's part of the .NET Framework which allows for simple deployment of DataPipe updates. Please refer to the following articles that discuss how the technology works: [No-Touch Deployment in the .NET Framework](#) and [.NET Zero Deployment](#).

Q. Our IT Department hasn't deployed the .NET Framework 1.1 (service pack 1) on our computers? How can we learn more about .NET?

A. The .NET Framework is an integral Windows component for building and running the next generation of software applications and Web services. Refer to the following articles for information on the technology: [.NET Technology Overview](#), [.NET Frequently Asked Questions](#), [An Overview of Security in the .NET Framework](#) and [Analysis of .NET Security Architecture](#).

Q. What's the best way to secure our database? Likewise, how should I secure our IIS server?

A. Both of these questions are really outside the scope of DataPipe itself as there are many resources regarding industry best practices for setting up and securing any servers that are to host DataPipe components (web server, database server, etc.). Following are some references published by Microsoft that offer a good starting point for security and administration of the relevant servers:

- [SQL Server 2000 SP3 Security Features and Best Practices: Security Best Practices Checklist](#)
- [Best Practices Analyzer Tool for Microsoft SQL Server 2000 1.0](#)
- [Securing Your Database Server](#)
- [Checklist: Securing Your Database Server](#)
- [SQL Server 2000 Operations Guide](#)
- [Internet Information Services \(IIS\) Security Center](#)
- [Microsoft TechNet Security Center](#)
- [Microsoft Security Guidance Center](#)

DATABASE SETUP & CONFIGURATION

Q. What databases does DataPipe support?

- A.** DataPipe has been tested and verified for use with the following DBMSs:
- SQL Server (SQL Server 2000 on Windows 2000 Server)
 - Oracle (Oracle 8.1.7 and Oracle 9i on Windows 2000 Server)
 - IBM DB2 (DB2/NT 8.1.0 on Windows 2000 Server)

Q. How are the database and the necessary DataPipe objects (tables, indexes, triggers, etc.) initially created?

A. The initial database and object creation is done via a Data Definition Language (DDL) script that Knorr Associates supplies.

Q. Can the database be created by one of our DBAs and the installation scripts then run against the empty database?

A. Yes. You do not need to use the database creation script that we provide. In fact, we generally discourage using the one we supply as it is a bare minimal database creation statement. Organizations typically have standards regarding database creation and configuration and we encourage you to implement your own standards as long as they don't present any limitations with using DataPipe.

Q. How are database schema changes delivered? Are they done via installation program or scripts?

A. If there is a database schema change required as part of a DataPipe update, it is supplied as a DDL script. Unless otherwise noted, the scripts that we distribute always assume that any data in the database is to be retained.

Q. What method does your product use to connect between the Web/Application Server and the database server (e.g. JDBC, ODBC, OleDB)?

A. We use 100%-pure .NET data providers offered by DataDirect Technologies (<http://www.datadirect-technologies.com>). There is no database vendor client piece required on the Web/Application Server as the necessary drivers are installed as part of the DataPipe installation.

Q. Are there any known issues with using a DNS Alias name for the connection string instead of the server name?

A. No, and it's worth noting that specifying a port number to make this work is optional. This is a great way to handle being able to switch between different database servers without requiring a change to any DataPipe configuration strings. Simply change the DNS value and everyone will automatically use the new server.

Q. We support three database environments: development, integration test and production, and would like to be able to easily transfer data between the

environments. Is the migration of database objects using the native database tools supported?

A. Migration of database objects and data using the dbms migration tools shouldn't be a problem as long as you keep a few things in mind. The key thing to remember is that in order to "switch" the database that you're using for DataPipe, you'll need to change the connection string so that the web server knows which database to use.

The default connection string in the config file on the IIS Server is a plain text file and is easily changed. If you've decided to implement user-specific connection strings these are stored encrypted in the database. These get generated from within DataPipe so if you're simply copying the data from one server to another they'll need to point to the new server. An easy way to avoid having to modify the user-specific connection strings is to use a DNS Alias for the database server, which allows you to map a DNS entry to a particular server. The connection string simply points to a DNS entry. If you wanted to change which database is used, you'd simply need to update the DNS table and the connection strings would work unaltered. If you didn't use a DNS Alias, you'd need to generate appropriate UPDATE statements to avoid having to manually modify the encrypted user-specific connection strings from within DataPipe after the database is transferred. Ideally you would generate these for each database so that if you transfer the data from one database to another you'd simply need to run an UPDATE statement to make use of the proper connection string. Also, if you use SQL Server's Import/Export Wizard or comparable tools in Oracle or DB2, you'll need to assure that the appropriate objects (users, etc.) are carried across the different environments appropriately.

Q. Are all database objects (i.e. tables, indexes, etc.) owned by the database owner?

A. DataPipe makes use of non-owner-specific SQL statements. As such, whatever User ID you use to connect to the database needs to be aware of the tables that they need access to without having to specify an owner name with a database object. The way we've implemented this varies by database. In SQL Server, anything owned by the **database owner** (dbo by default) is available to other users (assuming they have the necessary permissions) without having to specify the owner in the SQL Statement. In Oracle, **Public Synonyms** are used to accomplish the same thing. DB2 has a similar capability via **Aliases**. The DDL script we provide to generate the database objects has **dbo** as the owner by default. If you'd like the owner to be a different user, a global search and replace in the script file before it is run against the database can accommodate such a change.

Q. Is the database connection string externalized from the code? Where is it stored and is it encrypted? What tools are available to encrypt/decrypt?

A. The database connection string is definitely externalized from any DataPipe code. There are actually two levels of connection strings that can be used for DataPipe. The first connection string resides on the IIS Web Server in a web.config file. It is a plain text file. By default, these config files aren't viewable from a browser. This setting should be left as is for obvious security reasons. This first connection string should have the User ID of a database user with very restricted access to the database. In fact, the only permission this user should have is to be able to connect to the database and issue a SELECT statement to one database table, that being the table of valid DataPipe users. Each user can also have a specific database connection string. This allows you to control access in the database down to the user level. Keep in mind that if each user has a different connection string, connection pooling will not take place and you lose any performance benefit that you might gain from using pooling. Once DataPipe has confirmed that the credentials supplied are

those of a valid DataPipe user, we connect the user to the database using their specific connection string if they have one defined. A user-specific connection string is optional. The user-specific connection string is stored in the database table of valid DataPipe users and is encrypted using a Rijndael symmetric encryption algorithm. Additional security is implemented in the application log on process, where the user must supply a user name and password. The password is not sent over the wire in clear text for obvious security reasons. In the case of the password, it is hashed using MD5 so that the contents can't be determined as it travels from the client to the server. Beyond the password and connection string, you should consider everything else going over the wire to be going in the clear and fair game for snooping. If you're not working in a secure environment and you're concerned about your EHS data going over the wire in the clear we recommend the use of SSL or VPN.

Q. Does the software support connecting to a Named Instance of SQL Server?

A. DataPipe does support the use of SQL Server Named Instances.

Q. Are any special components or software required on the database server?

A. DataPipe intentionally uses a very well known set of database objects and generic SQL statements to alleviate the need for any special components or software in the database. As such, there are no special components or software required on the database server.

Q. Does DataPipe provide archive/purge functionality?

A. This would be a function of the database itself or a routine that would need to be developed. It can be done as an External Process or a DTS (SQL Server) or a comparable utility and scheduled as a job if desired. There is an easy way to determine the age of records within DataPipe as every table has a MOD_DATE_TIME field, which gets stamped whenever someone adds or updates a record.

DATABASE UTILIZATION & BENCHMARKING

Q. In order to determine potential resource utilization and assure that the database is initially created with the proper parameters and that it can meet future growth needs, can you provide some database sizing information?

A. We'd certainly be glad to work with you regarding this. Because we use variable length fields, which take up only as much room as the data users supply, we can't easily provide such sizing information as we don't know how much data your users will typically supply per record in normal usage. We have the ability to generate size estimates but you'd need to supply us with some information. What you'd need to provide is an average length for each column, the percent of the records that will be nulls for that column (0, 25, 50 or 75) and the percent of the records that are distinct values for that column (25, 50, 75 or 100). Then, if you tell us the number of records you're estimating on maintaining in that table we can generate a size estimate. This would need to be done for each table you're interested in estimating. We'll gladly provide spreadsheets of the database table fields and types and lengths with empty columns for the information you need to provide. Additionally, the databases themselves offer varying degrees of size estimating tools. These need to be run after the database and tables have been created though.

Q. What tools do you provide to benchmark transaction loads and stress test the database server?

A. We offer performance monitoring capabilities within DataPipe that generate statistics on network traffic (to determine if bandwidth is an issue), server processing time (to determine if the IIS Server is an issue) and database processing time (to determine if the database is an issue). Also, in our development environment we've used Microsoft's Application Center Test and other tools to do various internal benchmarking. SQL Server, Oracle and DB2 offer their own benchmarking, performance monitoring and load testing capabilities. Your DBA would be the best person to talk to about this. There are also numerous third party tools available as well as dozens of books on the topic of performance monitoring.

Q. What is the maximum number of users supported by DataPipe? What is the number of concurrent users supported by DataPipe?

A. DataPipe doesn't have an inherent limit on the number of users that it can support. It was designed to be able to scale up and out to accommodate future usage growth. The number of users is based on your software license agreement. We license DataPipe by the number of concurrent users. So if you have a 50-user license, you could set up an unlimited number of valid DataPipe users. It's just that only 50 of them would be able to use it at one time. There is also a mechanism to assign users to one of 10 different user priorities. Certain usage slots can be left available for users with higher-level priorities.

SECURITY

Q. Does the software connect to the database and process data with an account other than system administrator or database owner? Our DBA will create an account that the application will use to query/update data. The system administration and owner accounts and passwords will not be available to the application in the integration test and production environments. What permissions does this account need?

A. The User ID and password specified in the connection strings must obviously be a valid database user. The User ID specified **DOES NOT** have to be a system administrator or the owner of the database. In fact, we highly discourage such practice, as it is a best practice to only grant someone access to the things they need. We encourage defining a user in the database with very restricted access (something like DPDEFAULT) that will be the User ID specified in the configuration file. An additional database user (something like DPUSER), or users, should be defined for use in user specific connection strings. The DPUSER user account should be given whatever permissions you determine it needs in order to use the application effectively. You should only grant permissions to the objects that the user will need to use. When we supply the database DDL script, we include a set of statements to grant SELECT, INSERT, UPDATE and DELETE permissions to all DataPipe tables to a database ROLE to which the DPUSER database user belongs. This script is supplied to be used as a starting point. You do not need to run this script. You are encouraged to customize or assign your own permissions to make the user access within the database as restrictive as is necessary.

Q. We want to use Windows Groups to authenticate users of the system. These Windows Groups will be defined in SQL Server and added to database roles. What database roles are required and what permissions should they have?

A. No database roles are required, but making use of them is an encouraged practice. This makes administering your database users much easier. DataPipe also has the ability to define aliases for its users, so you can define one set of DataPipe permissions (and an appropriate connection string) for an alias and assign the alias to a group of users. If you want to make a change, you'd only have to change it in one place and all users with that alias will automatically have the change applied to them.

Q. Does DataPipe support Windows Authentication? If not, how are users authenticated?

A. This question is applicable in two places: 1) connecting to the database and 2) logging in to DataPipe. In connecting to the database, Windows Authentication is supported in SQL Server, Oracle and DB2. Regarding logging in to the application, DataPipe maintains its own set of valid users in a database table and does not support the use of Windows Authentication.

Q. Is there a system administrator or 'super user' role supported in DataPipe?

A. Yes. DataPipe System Administrators can have access to the administration forms that allow them to configure application users, permissions and numerous other configuration settings.

Q. How many levels of authorized access to stored data can be created by the system administrator? How is the integrity of each level enforced—for example, what types of password and logon security are implemented (e.g. encrypted passwords, timed password-change prompt)?

A. There are several levels of access to anyone trying to get at DataPipe data residing in a database: 1) the database security itself, 2) the network and operating system security, 3) DataPipe security and possibly others, depending on your IT infrastructure. The database security is specific to whatever database you're using for DataPipe's data store. The same is true for your network security. For specific questions on these it's best to talk to your DBA or System Administrator or refer to the DBMS/OS/Network documentation. DataPipe security is completely subservient to any database, network or operating system security. If there is a particular security setting defined for a database user within the database, DataPipe doesn't in any way override it. Passwords for DataPipe logons are hashed using MD5 so that as the password goes over the wire it is not transmitted in the clear. Like connections strings, passwords are encrypted when passing to and from the database, and stored encrypted in the database, using a Rijndael symmetric encryption algorithm. If you want to limit and/or keep a record of failed DataPipe logon attempts, this can be accomplished by means of config file settings. You may choose to force DataPipe to close if any attempt to run it results in more than a specified number of consecutive logon failures. You may choose to log (to a server Application event log) either every logon failure or every failure that resulted in the forced closing of DataPipe. If you want to implement a custom restriction on password content, applicable to all DataPipe password changes, this can also be accomplished by means of a config file setting. There are two options available: writing a "Regular Expression" pattern that all passwords must match, or defining a minimum count of digits (0-9), a minimum count of upper case characters (A-Z), a minimum count of lower case characters (a-z), and/or a minimum count of special

characters (your choice) that all passwords must satisfy. There are several ways to customize, on a user by user basis, what you want a user to do regarding their DataPipe password and the usage of DataPipe itself. The System Administrator can specify a user's session duration (so that if there is a period of inactivity it will automatically log them off and free up a user slot), or temporarily disable a user account (prevent that user from logging on to DataPipe) by setting it as inactive, or end a user's session immediately (by the User Manager). The System Administrator can also specify whether or not the user is allowed to change their own password, how long a password is valid for (the user will be prompted to change it every time they log on within two weeks of its expiration), the minimum number of characters that the password must be and if they want to force the user to change their password the first time they run DataPipe. A System Administrator can change any DataPipe user's password without needing to know the old password, though individuals can change their own passwords only if they know their old passwords. In addition to the above basic user information, DataPipe allows a System Administrator to totally customize which forms a user is allowed to access and what they can do within each form. For example, some users may have access to the HR form, while others may not. You can specify if a user is allowed to add, edit or delete data, lock records to prevent editing, run reports and just about anything else related to data on a form, giving you unlimited control over what you want to allow users to do within DataPipe on a user by user and form by form basis.

Q. Can the database be set up to allow any or all of the data to be encrypted? If so, which data and by what method?

A. The only values that are encrypted in the database by DataPipe itself are the user passwords and user-specific connection strings, if used. DBMSs also offer their own encryption capabilities in different ways. SQL Server 2000 offers encryption via SSL by making use of their Super Socket Net-Library. Oracle offers the DBMS_Obfuscation_Toolkit as well as the Encryption Wizard for Oracle. DB2 offers EDITPROC on their mainframe but no comparable functionality for UDB. Unfortunately, there are limitations as far as which field types can be encrypted, indexes not being easily decrypted and other issues that make using any of these encryption mechanisms unpractical for DataPipe and other software applications that haven't been specifically designed to accommodate encryption at the database level. Encrypting data in the database is a rather extreme security measure and can lead to difficulties in other areas. If you use any third party reporting or other tools to query data from the database, for example, how are they going to be able to decrypt the encrypted data they get back from the database? If someone has cracked in to your database you've probably got bigger problems than their ability to read the data. At this point, they've probably gotten through a firewall, a Windows Domain Controller or Active Directory, a server hosting SQL Server and the SQL Server database. Certainly a better security perimeter should be set up. Encryption within the database also adds a serious performance hit on the database server. Disk space is another issue with data encryption because encrypted fields are larger than unencrypted fields. They're a little larger for textual data and a lot larger for numeric and binary data. The bottom line is when planning your disk capacity for database encryption you'd need to anticipate a tripling or quadrupling in size. This would also require that the database fields needed by DataPipe to be quadrupled in size. Having said that, with things like HIPAA and FISMA becoming a reality, we understand that there may be rare cases where the information stored in the database is deemed sensitive enough to warrant such a level of security. Unfortunately, we do not offer an option to encrypt anything stored in the database other than user passwords and connection strings. There are several third party vendors that may be able to address this, but we've not worked with any of them and would question the feasibility of their usage with DataPipe.

Q. Does your product include an SSL encryption option to protect transmitted data? What about VPN?

A. DataPipe can run over an SSL connection. The ability to do this is typically by installing a digital certificate and configuring the IIS Server. Talk to your System Administrator or refer to the operating system documentation for how to configure SSL. There's nothing that really needs to be done within DataPipe other than change configuration URL settings from http to https where appropriate. Since DataPipe uses industry standard protocols, it should work fine over a VPN connection.

Q. Does your product use any permanent cookies or special client-side scripting (e.g? JavaScript, VBScript, Java applet, ActiveX controls)? If so, what?

A. DataPipe makes no use of browser cookies, VBScript, Java Applets or ActiveX controls.

Q. Does your product include a method for auditing users to support control of data access (e.g. an access audit or audit log on the Web/app or database server)? If so, what method?

A. Auditing is available in the DBMSs supported by DataPipe. SQL Server offers Auditing as well as C2 Auditing through their SQL Profiler tool. Oracle offers auditing via its Oracle Audit commands as well as system triggers, fine-grained audit and system logs. IBM DB2 offers numerous options as outlined in their Administration Guide: Design and Implementation. See the chapter titled Auditing DB2 Activities. There are also third party tools available for auditing, including: SQL Power Tools (<http://www.sqlpower.com>) and Lumigent (<http://www.lumigent.com>), among others. DataPipe also comes with a User Manager form, which allows you to monitor in near real-time who is currently logged on, how long they've been logged on, how much time is remaining in their current session, their user priority and more.

Q. Can you provide any information specific to the requirements of Federal Information Security Management Act (FISMA) or the Health Insurance Portability and Accountability Act (HIPAA)?

A. Regarding FISMA and HIPAA and many other similar federal- and state-mandated requirements, a software vendor whose application is being used in an applicable environment can't really say that they are "FISMA-" or "HIPAA-compliant." Most of these regulations require procedures and policies that, for the most part, fall outside of the control of any software application vendor. Securing data requires keeping the DBMS and other servers physically inaccessible from unauthorized users, enforcing policies with regard to who has what access to that data and much more. All of which are outside of our control. There are numerous firms that provide HIPAA and similar consulting services to help determine compliance. If there's a particular FISMA or HIPAA standard you're interested in, please inform us and we will work with you in trying to determine if there might be any compliance issues regarding DataPipe's involvement with the standard.

DISASTER RECOVERY

Q. Are there any special requirements to set up a warm or hot standby database server to be activated for disaster recovery? There are known issues in some products as the storage of configuration information – like server name – is maintained in a database table.

A. There are several possibilities here, depending on how everything is configured. If the DBMS is on a different server than the IIS server, and there is a primary and a standby dbms and you want to switch users to the standby if the primary goes down, the simplest way to accommodate the switching is to use a DNS Alias in your connection string. To switch databases you'd simply need to change a DNS entry. There would be no changes required to any application configuration files or to any entries in the database. If the use of a DNS Alias is not an option, you'd simply need to modify the web.config text file on the IIS server so that its connection string points to the standby server. Additionally, if users have their own connection string, you'd need to issue an SQL UPDATE statement(s) to the standby database table to modify users' encrypted connection strings appropriately. Ideally, you'd want to generate the UPDATE statements ahead of time, as you'll need to generate the encrypted, user specific connection strings from within DataPipe. While they can be generated manually from within DataPipe, it would be more efficient to have them generated ahead of time so that switching users to the standby takes a minimal amount of time. Remember that the data in a warm standby server is only as current as the last time it was transferred from the live production server. This may or may not be acceptable. Depending on how current you want the data in the event of a disaster, you may want to consider using the Replication or Failover Clustering technologies offered in SQL Server, or similar technologies offered in Oracle or DB2, which offer more real-time failovers, but are much more costly to implement and maintain.

MISCELLANEOUS

Q. Is any ad-hoc reporting functionality built into the software?

A. Yes. The Ad Hoc Report Designer is included as are the Advanced Report capabilities and an interface to Crystal Reports™. Refer to the appropriate sales literature for further explanation of these options.

Q. Is any searching capability built into the software?

A. Absolutely. There are several ways to search for information within DataPipe. Key Lookup allows you to quickly find the first record matching the value you provide. This offers extremely fast searching, especially on fields that are indexed; it is not available for binary or large text fields. You have the option to limit the fields a user can use for sorting records and for Key Lookup to only those that are indexed. Filters allow you to search any and all fields (subject to database restrictions). Additionally, filters can be defined and saved as part of Ad Hoc reports.

Q. We have data that needs to come from our Human Resource system (SAP, PeopleSoft, etc.), instruments, third party labs, PDAs and potentially other sources that we haven't necessarily thought of yet. Is there any importing functionality

built into the product? Also, we want these imports to be run on a predefined schedule. Is a scheduling engine included?

A. How best to implement import functionality depends on what needs to be imported in to DataPipe and what, if any, additional processing needs to take place. Import functionality can be provided in DataPipe via a Special Import or an External Process. What gets imported and to what degree it gets validated needs to be defined and implemented. Special Imports can't be scheduled. They must be initiated from within DataPipe by a user. External Processes can be scheduled. Since these are essentially stand-alone Windows applications, the ability to have them run on a schedule is freely available with Windows as well as other third party scheduling tools. Another option is to use functionality provided by the different databases or a third party tool. For example, you could use SQL Server's Data Transformation Services. The DTS could then be set up to run at a certain interval by means of a SQL Server Job, which could do additional processing via BAT or Script files. There are many options. We need to have an understanding of what you're trying to do and work with you to come up with the best approach.

Q. Is the user interface browser-based or client-server-forms-based?

A. It can be either, as well as a hybrid approach, depending on the deployment mechanism used. Please refer to the **DataPipe Architectural Overview** document.

Q. What version of the .NET Framework is required for DataPipe?

A. DataPipe requires version 1.1 service pack 1 of the .NET Framework. There were some bug fixes in 1.1 to handle asynchronous web service calls correctly. As a result, not all of DataPipe would function properly in version 1.0. If you're running DataPipe you can determine the version of the Framework that it's running under by checking the Help>About dialog box.

Q. Does DataPipe need to be installed at the root of a web site or can it be installed in a virtual directory?

A. DataPipe can be installed at the root of a web site or in a virtual directory.

Q. Does DataPipe make use of any COM-based DLLs?

A. No. DataPipe was written using 100% Microsoft .NET technology. As such, it requires no COMbased DLLs. It does have DLL files, but these are .NET assemblies and run under the .NET Framework.

Q. Does DataPipe use COM+ in any way?

A. No. The .NET Framework's architecture takes care of a lot of the things that COM+ was need for in legacy applications, including object pooling and the like. Database transactions are handled in the database, not through COM+.