M-Files System Architecture and Technical Handbook

Version: 45
Authors: Jussi Salmi
Samppa Lahtinen
Mika Javanainen

1. General

M-Files encapsulates a great set of technical features and connectors to other systems.

This document is intended to cover a deeper technical overview of M-Files and the target audiences are administrators and other technical persons such as system specialists and resellers. The reader of this document should be familiar with the concepts of Microsoft Windows and networking.

2. M-Files Server Architecture

M-Files Server is the backbone of the M-Files system. It stores all the objects (documents, customers etc.), controls the access to the objects, tracks the changes of the objects (version history) and handles all the connections to the other systems. In other words M-Files Server stores and handles all the information in the M-Files system.

Refer to M-Files Technical Data Sheet for system requirements and supported operating systems and database engines.

2.1 Backend Architecture (on-premise servers)

M-Files Server runs as a Windows service. This means that M-Files Server is started automatically by the Windows Service Control Manager during the startup of Windows. Note that M-Files will run even when nobody is logged on to the Windows system. By default the M-Files Server service uses the identity of the local system account. This should be taken into consideration when planning the security settings of the folders the M-Files Server accesses. On the M-Files Server computer the required access rights must be given to the local system account. If the resource to be accessed by M-Files Server is located on the network, the required access rights must be given for the computer that runs the M-Files Server service.

The figure below illustrates the main components of M-Files system on the server side as well as the protocols used to communicate between the systems.

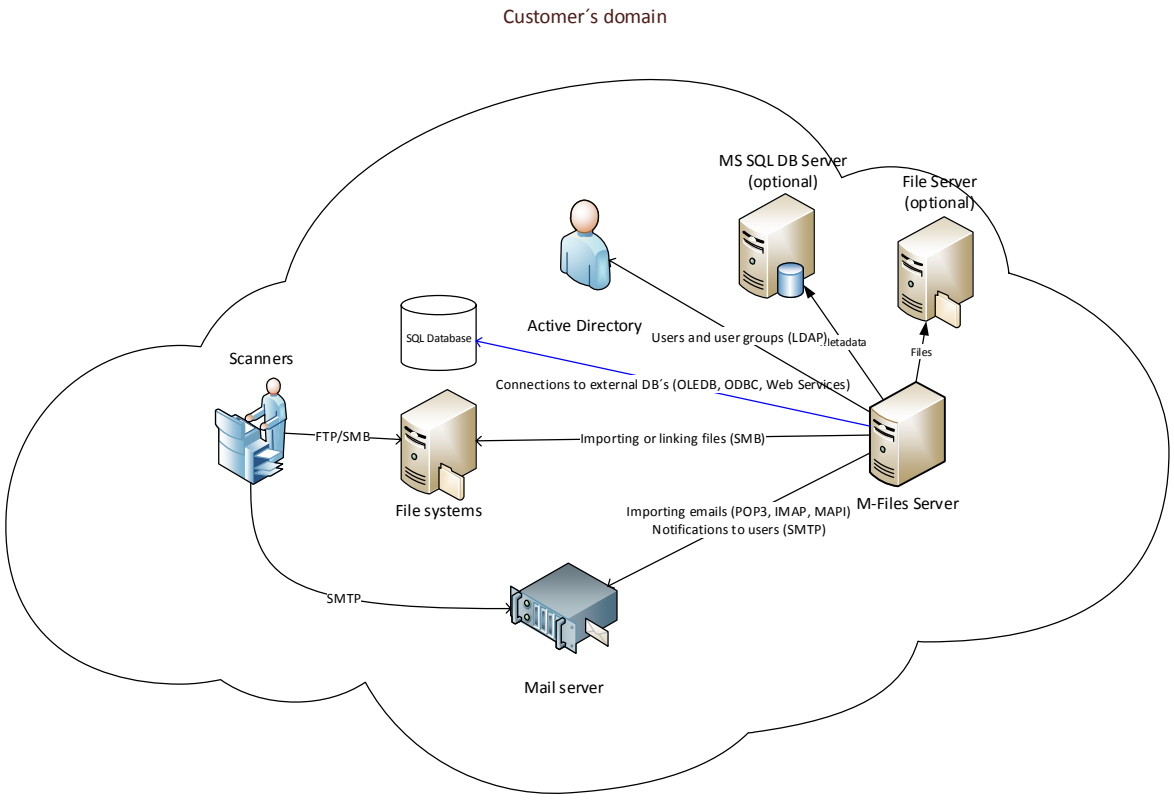


Figure 1 M-Files Server Architecture (on-premises)

M-Files can use either the embedded Firebird SQL Server or Microsoft SQL Server as a database engine. The database engine can be selected for each document vault separately.

2.1.1 Firebird Database

M-Files Server comes with embedded Firebird database engine. As the database engine is tightly embedded to M-Files Server, it does not disturb other applications and other applications cannot see or use this instance of the database. The database engine is a part of M-Files Server so it is installed during M-Files Server setup. Also when M-Files Server is uninstalled, the database engine will be removed from the system too.

The Firebird database instance of M-Files Server does not prevent other applications to install their own instances of the database engine. This means that there can be several Firebird database engines on the same computer at the same time.

2.1.2 MS SQL database

Alternative database engine, typically used in larger implementations is Microsoft SQL Server. There are two ways to configure M-Files document vault to use MS SQL Server:

1. All vault data is stored into a single SQL database in MS SQL Server
2. All vault data except object files are stored into a single SQL database in MS SQL Server. Object files are stored on file server.

MS SQL database engine can run on the same or different server with M-Files Server. M-Files Server software communicates with the MS SQL database engine via OLE DB protocol.

MS SQL database and M-Files Server software are standalone, which means that the uninstallation of M-Files Server software should not affect the database engine and vice versa.

2.2 Backend Architecture (M-Files Cloud Vault Server)

M-Files Cloud Vault server is hosted by M-Files Corporation in Windows Azure. The figure below illustrates the main components of the system as well as the communication protocols between different services. M-Files Server in Windows Azure can communicate with customers´on-premise systems. The connections between Windows Azure and customer´s domain are tunneled using Azure Connect technology.

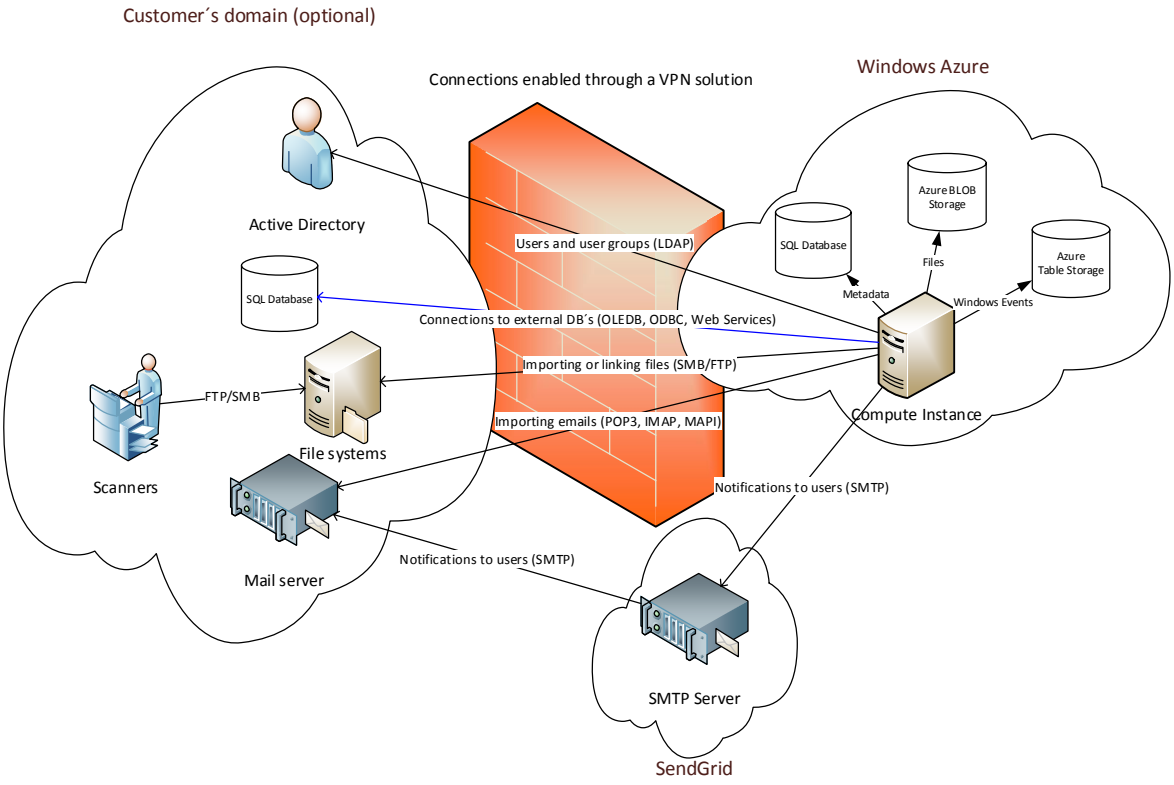


Figure 2 M-Files Server Architecture (Cloud Vault)

2.3 M-Files Administrator

M-Files Administrator is an MMC snap-in application that is used to manage M-Files Server. Most administrative operations are done using M-Files Server Administrator. The application can be installed on the same computer with the M-Files Server or on any other computer in network meeting the system requirements.

M-Files Administrator can be used to manage several servers by registering the needed servers to the application.

M-Files Administrator communicates with M-Files Server using RPC, RPC over HTTP(s) or LPC calls.

3. Networking

M-Files Server can be accessed only by using Remote Procedure Calls (RPC), Local Procedure Calls (LPC), or RPC over HTTP(s). M-Files Cloud Vault server can be accessed only through RPC over HTTPS. This is to ensure that all traffic between client and server computers is encrypted.

3.1 Networking to M-Files Server in on-premise installation

M-Files Server listens the TCP port 2266 in the server computer. The 2266 port is also officially registered to M-Files, so there should not be other software using this port. This is also the only port M-Files Server uses. In normal configuration of M-Files, only by allowing requests from client computers over port 2266 should be enough in terms of firewall configuration. There is no need to open Windows file sharing ports, because M-Files does not share any folders or files. All the files and other traffic are transferred using this single port.

Notice that the standard TCP/IP traffic is not encrypted. It is recommended to encrypt the traffic from public domain to the on-premise M-Files server. This can be done by allowing connections to the M-Files servers from outside the domain only through VPN, or by using RPC over HTTPS protocol.

3.2 Networking to M-Files server in cloud deployment

M-Files Server listens the TCP port 4466 in the server computer. The 4466 port is also officially registered to M-Files, so there should not be other software using this port. This is also the only port M-Files Server uses. When using RPC over HTTPS protocol, technically client computers connect M-Files Cloud Server using port 443 (SSL) and the traffic is redirected to M-Files server through IIS proxy server. Hence, only inbound HTTPS traffic to port 443 is allowed from client computers to M-Files Cloud Server.

Typically there is no need to add any additional rules to client computers´ firewall since all traffic from client computers to the server is tunneled through the standard https port (443).

3.3 Firewall configuration on M-Files server

To use M-Files, no additional ports such as Windows file sharing ports should be opened, because M-Files does not share any folders or files. All the files and other traffic are transferred using the single port.

When the clients use the M-Files Server, they open the connection to the server and use the functionality the server provides. This is always the direction of the communication, so the server never calls the clients. This is essential information when planning network issues and firewall configurations. M-Files has been designed to work in all kinds of networks. The only requirement is that there is a connection between clients and server. This means that for example Network Address Translation (NAT) and Virtual Private Networks (VPN) work without any special configurations.

3.4 Client Access to the server

M-Files users can access M-Files Server though native Windows client, M-Files Web Access, M-Files Mobile Access, or Native M-Files Mobile Applications. Native Windows client (M-Files Client) is the most versatile UI and also offers best user experience.

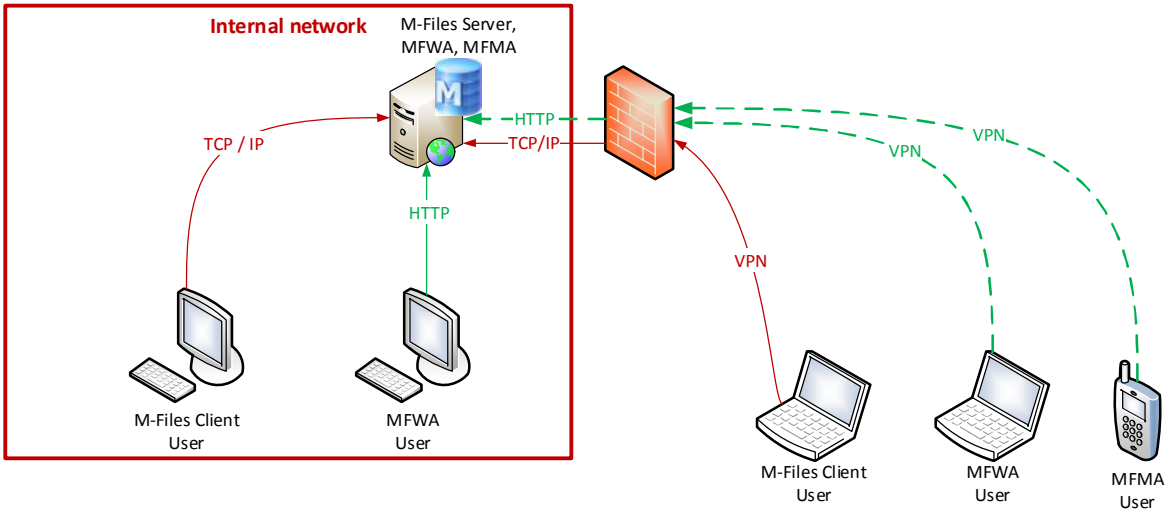
Web Access is a good option for non-Windows users and for those who access the system irregularly.

3.4.1 Networking from client to on-premises server

There are a few ways to configure client connections in on-premise deployments.

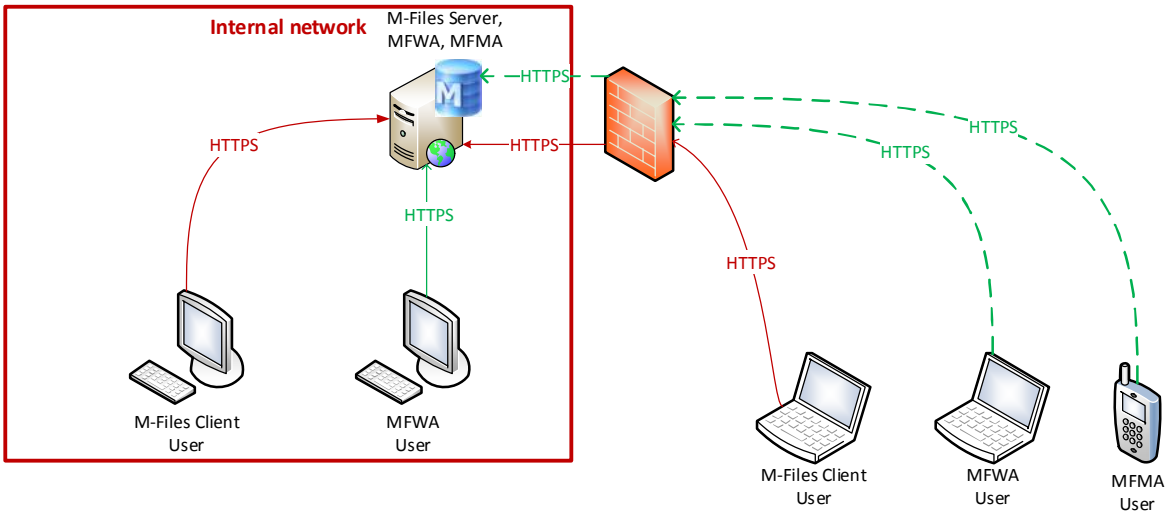
Standard setup

The option illustrated below is the most secure and also the best option performance-wise. In this option M-Files server listens port 2266 and client software calls it over TCP/IP protocol. Web Access listens at standard http port (80). All traffic inside the domain is unencrypted and the connection from outside the domain are encrypted using VPN clients.



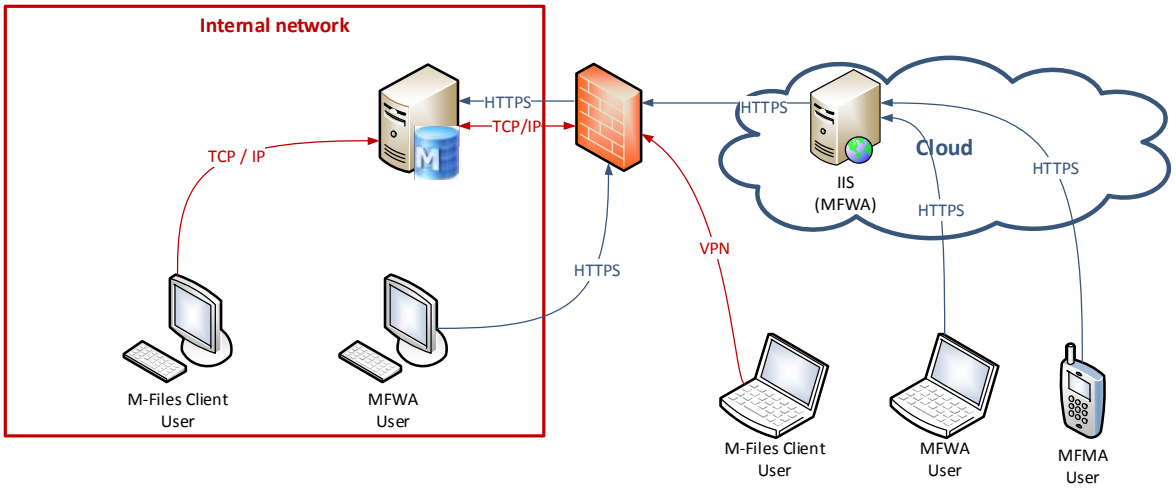
HTTPS setup

This setup is recommended for organizations that do not have resources to organize VPN connection to remote users and if M-Files needs to be accessible from anywhere. All traffic inside and across the domains is encrypted using HTTPS.



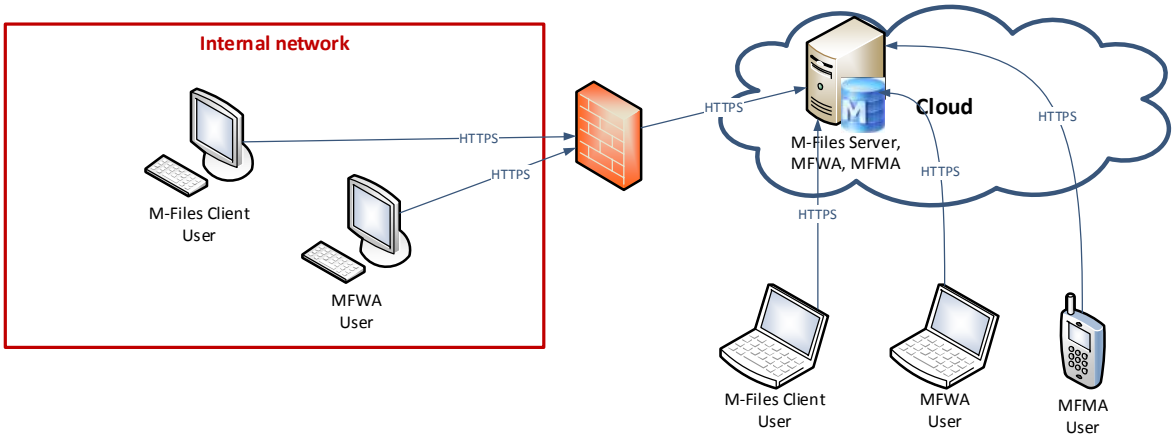
Hybrid setup

The option illustrated below is recommended when M-Files Web Access or Mobile Access needs to be accessible through public Internet without VPN connections and such. This is a good option to consider especially if it is important to ensure that remote users do not have direct access to M-Files Server. This setup might be also considered when willing to enable M-Files Mobile Access to those users whose smartphones do not support VPN connections.



3.4.2 Networking from client to cloud server

M-Files Cloud Vault server allows only calls through encrypted HTTPS protocol.



4. M-Files data storages and backups

This chapter describes the different components of M-Files system from backup perspective. The content of this chapter only applies to M-Files Servers that are hosted on-premises. M-Files Cloud Vault servers are maintained and backed up by M-Files Corporation.

M-Files Server stores server-specific and document-vault specific data.

4.1 Server-specific data

M-Files server-specific data is always stored to the embedded Firebird SQL databases regardless of deployment or configuration type chosen.

Most critical server-specific data can be backed up using M-Files Server Administrator to master database backup file. This data includes:

- License details
- Login accounts and their associated license types and roles
- Passwords of M-Files login accounts
- Notification settings
- Web Access settings

In addition to these settings, the system may have been configured using different Windows registry key values. These settings can be exported to a text file and backed up separately.

4.2 Vault-specific data

Vault-specific data can be stored into either embedded Firebird database, to MS SQL database, or to MS SQL database, and to file system.

4.2.1 Firebird vaults

When using the standard embedded Firebird database engine, vault metadata is stored in the database and object files are stored on file system. M-Files Server locks the database files of the online vaults, and these files should not be accessed directly.

These vaults can be backed up using Scheduled Jobs in M-Files Server Administrator. This tool creates a single backup file of each document vault. Therefore, administrators do not need to backup object files from the file system separately.

M-Files supports taking full backups and differential backups of the document vaults. Full backups contain all file and metadata of the vaults and differential backups contain changes in metadata and file data from the latest successful full backup.

4.2.2 MS SQL Server vaults

When using MS SQL Server as database engine, administrators can choose to store file data either on file system or into the database. Accessing M-Files vault databases directly using MS SQL Server Management Studio or similar is not recommended.

When using the first option, administrators, must backup both the MS SQL database and files on the file system separately.

When using the latter option, only one database per vault needs to be backed up.

MS SQL Server is backed up using the tools provided by Microsoft or any compatible tools from 3rd Parties. File data can be backed up using any compatible backup system.

4.2.3 Secondary data

Regardless of the database, M-Files server creates also secondary data for each vault. This data is stored on the hard drive of the M-Files Server. It is not mandatory to backup this data as this content

can be always re-created from a working backup file. Rebuilding the search index of a large document vault can take a significant amount of time, however.

Index Files

When information is stored to the M-Files Server, the information is indexed for searching. Indexing produces own special files and these files are used only by M-Files Server.

Thumbnail Pictures

Thumbnail pictures are created for any file type recognized by M-Files Server. The users can switch the display mode of the views to Thumbnails or Large Thumbnails (Vista). These thumbnail pictures are automatically generated by M-Files Server and stored to Thumbnails sub folder of the document vault folder.

Viewer Files

M-Files Server generate a temporary viewer file when file version is viewed by a user first time. This speeds up viewing the file by the same or different user next time.

4.3 Best backup practises

Taking regular backup of the M-Files system is important. The recommended backup interval depends on the use case and criticality of the system and the recommended minimal level is described in this chapter.

M-Files backups are taken for two reasons:

- To protect against hardware disaster
- To protect against logical errors

Build-in document control features, including version history and soft-deleting eliminate the need of restoring backups in multiple scenarios. These features do not, however, protect against logical errors made by administrators or due to system errors.

In case of hardware failure, the database corruption is not necessarily noticed in the same day. Therefore it is important to store enough restore points of the system. We recommend designing the backup system in such way that the database can be restored to reflect the situation of any of the previous 14 days:

- Take a full backup of the master database every day. Create 14 distinct backup jobs for master database backups to ensure that you can store 14 separate master backup files. Name each backup file as “M-Files Master Backup <X>”, where X represent the sequence number of the job (i.e. 1 to 14).
- Take a full backup of each document vault at least once a week. Create two distinct backup jobs. Name the backup files as “M-Files <VAULT NAME> FB <Y>” where Y represents the sequence number of the full backup job (i.e. 1 or 2).
- Take a differential backup of each document vault every day except during those days when you take the full backup. Name the backup files as “M-Files <VAULT NAME> DIFF <Y>_<Z>” where N stands for sequence number after the full backup (that is, you end up having the following naming conventions: 1_1, 1_2, 1_3, 1_4, 1_5, 1_6, 2_1, 2_2, 2_3, 2_4, 2_5, 2_6).

5. M-Files Client Architecture

M-Files Client consists of several software components. These components are installed to the Windows operating system. M-Files Client has been integrated very tightly to the Windows, so it provides familiar user interfaces and users do not need to learn new ways to operate with documents.

The main components of M-Files Client are:

- M-Files Client service
- M-Files file system driver
- M-Files user interfaces

M-Files Client service is a Windows service like M-Files Server. This service is started during system startup and provides the connection to the M-Files Server. The identity of this service is local system account.

M-Files file system driver is a virtual hard drive installed on the operating system. This is a very essential part of the client components, because this provides a virtual hard drive for the users. By default the drive letter is M. When users use M-Files and access the server, they browse this drive. The drive lists objects like documents and document files. When a user wants to open a document file, it is opened directly from the M-Files drive. Because the drive functions like any other standard drive, all the applications work normally. There is no need to do any custom modifications for applications. In other words we can say that if an application works against computer's C drive, it works also with M-Files.

In addition to the M-Files drive and its base functionality, M-Files Client provides a set of advanced user interfaces. When a user accesses the M-Files drive using Windows shell, M-Files Client extends the basic functionality of this. All needed commands are added and M-Files shows extra panes on the shell. Also extended search capabilities are shown.

6. Other modules related to M-Files

6.1 M-Files Web Access

The M-Files Web Access web application resides on the M-Files Server computer. It is also possible to configure Web Access to another IIS server for instance in DMZ or in cloud.

M-Files Web Access is implemented using Microsoft .NET framework and it runs on the Microsoft Internet Information Services (IIS). In basic configuration the site of the M-Files Web Access uses standard TCP port for HTTP protocol (i.e. 80). It is recommended to configure the site to only allow encrypted connections through HTTPS protocol especially if the site is accessed from Internet.

M-Files Web Access uses standard HTML, DHTML, JavaScript and CSS technologies, so it can be accessed using standard internet browsers such as Microsoft Internet Explorer, Mozilla Firefox, Opera, and Safari.

6.2 M-Files Automatic Updates

M-Files provides functionality for automatic software updates. This means that the computers installed with M-Files can monitor M-Files Automatic Update Server hosted by M-Files Corporation. New versions of M-Files become automatically available for download through this service. Client computer can download the new version automatically and prompt the user to install the upgrade given that the user has rights to install new software on her computer.

Many organizations prefer to install software updates centrally and in this case it is recommended to disable automatic updates.

Client computers access Automatic Updates Server using the RPC protocol that is routed via HTTP protocol and TCP port 80.

6.3 Integration with Other Systems

One of the advantages of M-Files is that it can be integrated seamlessly with other systems. M-Files supports multiple integration methods, including:

- M-Files Client API (documentation installs with software)
- M-Files Server API (documentation installs with software)
- M-Files URL's (documentation available per request)
- M-Files Web Service API (http://www.m-files.com/MFWS)
- M-Files UI Extensibility Framework (http://www.m-files.com/UI_Extensibility_Framework)

6.3.1 External Databases

M-Files provides flexible methods for replicating and routing data from external databases. The only requirement is that the external database can be accessed OLE DB, ODBC, or Web Service interfaces. M-Files can read and/or write data to external databases.

A good example of an external database connection is a co-operation with a customer database. By importing those customer objects to M-Files, users can easily tag content to the existing records without having to duplicate customer data in M-Files database.

6.3.2 Existing Files and Folders

M-Files can import and link existing files from external locations. This functionality helps the start-up process of M-files, because all the existing files can be accessed through M-Files. After linking or importing the files to M-Files, the original external location can be isolated from the users and so they can start to use the added functionality that M-Files provides.

By importing or linking the existing files to M-Files, the organization gets a lot of advantages. For example version history starts functioning immediately, simultaneous modifications are prevented,

backups can be done using the methods that M-Files provides and M-Files' outstanding search capabilities are available.

6.3.3 Mail Servers

M-Files can use mail servers in two different ways; notifying users of events in the M-Files system and importing e-mails from desired mail boxes.

Notifications are used to notify the users of the events in document vaults via e-mail. There are several events in M-Files that can provide notifications. For example creating a new document or modifying an existing one causes an event. M-Files creates e-mail messages of events and sends these to the users who have subscribed to them. The e-mail messages are delivered using the standard SMTP protocol.

M-Files can be configured to monitor desired e-mail boxes. The monitoring is done by using the standard IMAP, POP, or MAPI protocols. When a new message appears to the monitored e-mail box, M-Files imports it and creates a new document out of it. This means that the e-mail is the document and the metadata of the document contains e-mail specific information.

6.3.4 Scanners

M-Files helps organizations to digitalize and index paper documentation. M-Files is compatible with any desktop and network scanner.

Network scanners are configured to produce a scanned image file to a network folder and M-Files is set to periodically monitor this folder and import its content as new documents to the system. M-Files OCR (payable add-on) converts the scanned images searchable PDF's and hence makes any scanned document searchable based on its content.

Many scanner devices also support metadata. M-Files can read these metadata XML files and import the contents of them as properties to the document.

6.3.5 M-Files Application Programming Interface (M-Files API)

M-Files Application Programming Interface is an ActiveX component that provides programming interface to access M-Files Server and Client. This enables 3rd parties to integrate their own systems with M-Files.

M-Files API documentation installs with the software and M-Files Corporation provides support and services for this kind of software development.

6.4 Windows Event Log

M-Files reports error messages and notifications to users via appropriate user interface when possible. Sometimes these messages cannot be shown to user (for instance in case where no one has logged on to the M-Files server computer). These errors are written to Windows Event Log of M-Files server and client computers.

M-Files administrators should routinely check Windows Event Log for error that may require configuration to the system. Some organizations configure event logs to notify administrators by email whenever a new error appears on the log.