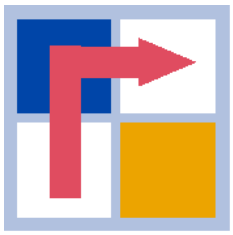


S R S

Point. Click. Extract.
That's it.

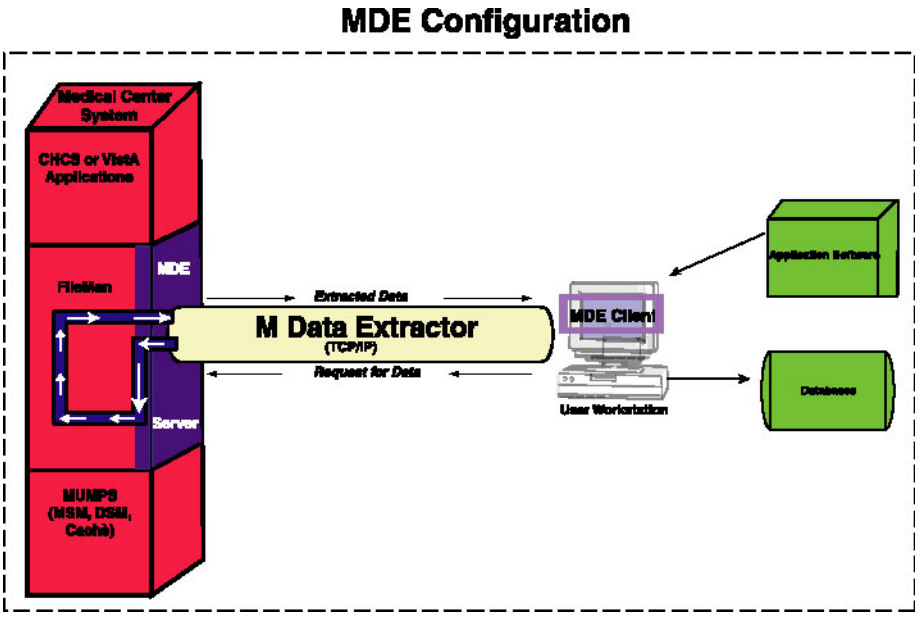
The M Data Extractor (MDE)

White Paper

www.mde.srs-inc.com

Executive Summary

The M Data Extractor (MDE) is an intermediary (middleware) COTS software utility located between M-based systems using FileMan and any relational database, such as Microsoft SQL Server and ORACLE 8i.



MDE extracts and migrates M-based data to relational databases, data warehouses, and data marts. This extracted data is available for combining multiple department/site data, clinical analysis, performance goal measurement, online analytical processing (OLAP), decision support, and ad hoc reporting.

Introduction

The difficulty in analyzing very large data sets or in extracting and migrating large volumes of data for analysis is not confined to M-based systems using FileMan. Other transaction processing systems suffer similar shortcomings.

The source of the difficulty is that data structures like FileMan are designed to provide the discrete detail required for the daily functioning of real-world businesses; that is, OLTP (on-line transaction processing). By way of contrast, large-scale enterprise-wide data analysis (OLAP) requires an overview of all data.

The following table summarizes the differences between OLTP transaction processing systems and OLAP relational data warehouse systems:

Characteristic	***OLTP Transaction Processing***	***OLAP Relational Data Warehouse***
Type of user	All users	Clinicians, healthcare providers, managers, analysts, physicians, epidemiologists
Goal	Record Keeping	Information analysis
Update frequency	Continuous	Periodic
Level of data detail	Detailed	Summary or detailed
Database architecture	Transactional	Analytical
Database structure	Fixed	Dynamic

The challenge for an M-based system using FileMan is how to effectively and efficiently achieve the OLAP analysis and reporting within the framework of an OLTP system.

The MDE meets this challenge by addressing two principal issues: (i) how to extract and migrate large amounts of M data from FileMan data structures to relational databases, and (ii) how to conduct large-scale data analysis on data located in M-based systems.

This White Paper presents solutions to both these issues.

Issue One: Large-Scale Extraction and Migration of Data From M-based Systems

Clinicians, analysts, providers, and managers require timely access to enterprise-wide data for patient care and decision support. This data needs to span many departments, sites of care, and facilities to provide the depth and breadth for functional analysis. But, meeting this requirement in M-based systems is constrained by M's inherent hierarchical data organization and storage.

Several approaches have been developed in an attempt to remedy this constraint. They include:

- Creating interface engines,
- Writing M routines,
- Adding an SQL layer, and
- Developing custom interfaces.

Each of these approaches has demonstrated certain value in limited contexts but each has also experienced severe limitations:

- Interface engines often are expensive to build, costly to maintain, and incomplete in their ability to access and retrieve all data from an M-based system. Several interface engines seek to wrap FileMan data as objects, but this approach has not proven more effective, efficient, or easy to use.

- Writing M routines to call specific pieces of M data requires the technical expertise of an M programmer, which can create technical dependencies. Programming time is expensive, often not readily available when needed, and risky if a programmer becomes otherwise committed on a temporary or indefinite basis. Most importantly, M code does not automatically adapt itself to changes in FileMan data structures, so technical support is always an issue.
- Adding an SQL layer to FileMan data structures permits external ODBC-compliant applications to access M data. However, SQL does not inherently know where the M data is located within FileMan. Consequently, using SQL requires the creation and maintenance of costly and complex "data mappings" to enable SQL to be used to locate M data within FileMan data structures.
- Developing custom interfaces through the use of data mappings requires significant programming time to keep data mappings current with changes, updates, or additions made to M-based application. Only with keeping data mappings current can external SQL systems recognize additional FileMan data elements. This expense can be substantial.

None of these approaches has met the test of providing an enterprise-wide solution that:

- Seamlessly spans functions, departments, sites, and facilities,
- Is inexpensive to develop, deploy, and maintain,
- Is capable of accessing all data stored within an M-based system, and
- Is up and running within hours of installation

Issue Two: Large-Scale Data Analysis in M-based Systems

M-based systems are substantially impacted by FileMan's methods for locating, acquiring, and analyzing information. While data are available for daily operations, the same data are made relatively inaccessible for in-depth analysis and ad hoc queries and reports.

As an example, FileMan analytical and reporting capabilities, while powerful, are not easy to learn. Also, they consume a significant portion of system resources when complex queries are executed. Additionally, operational data is usually not in the form suitable for even basic analysis. Often, data contains internal codes, unneeded groupings of data items, redundant data fields, and requires a person highly familiar with the file structure to locate and prepare data for analysis.

Although many facilities follow FileMan standards for storing data in data dictionaries, many others make minor modifications for FileMan to meet their specific data needs. While very helpful on an individual basis, this flexibility can make meaningful data analysis difficult, particularly across functions, departments, or facilities.

For example, consider the database definition of a lab test such as potassium. There may be differences in the way this test is named at various facilities. There can also be multiple forms of these tests at a single site, as in the following example from the Pittsburgh VAMC:

Test	Location of test in FileMan
Potassium	Field 6 in File 63
Potassium (BG)	Field 645059 in File 63
Potassium (CX7-OUT)	Field 646074 in File 63

The location of these tests could be different at different sites. Consequently, it is necessary to use a data extraction tool that works in an integral manner with FileMan and outputs data in a correlated fashion.

One approach to improving the accessibility of data for analysis is use of ODBC mapping directly from within the M system. However, a direct ODBC approach does not provide an efficient solution for large-scale data analysis and reporting for the following reasons:

- The data remains in its hierarchical form, making performance slow.
- There is no aggregation or cleaning of data, two tasks that greatly improve the ease and speed of data analysis.
- There is no segmentation of domain-specific data elements, making analysis more complex.

While M-based systems using FileMan comprise the largest unified collection of operational and clinical data, the use of this data for analysis and reporting has been limited to date.

The Solution: The MDE (M Data Extractor)

The MDE is specifically designed to address the limitations of M-based systems using FileMan: (i) conducting large-scale extraction and migration of data, and (ii) conducting large-scale data analysis. MDE is unique in its approach to extracting and migrating M data from FileMan data structures to relational databases such as Microsoft SQL Server and ORACLE for analysis and reporting.

MDE's unique approach embodies:

- Natively reading the FileMan data structure,
- Automatically creating and executing SQL statements to build table(s) for the target relational database,
- Automatically inserting data into the table(s) on the target relational database, and
- Enabling the aggregation of data from different/multiple FileMan data sites to a single target relational database.

The central values of the MDE are that it:

- Requires no changes to be made to the FileMan system,
- Requires no changes to be made to the M code,
- Does not use SQL, ODBC, or data mappings to produce data extractions,
- Does not add any layer of technology to the FileMan data structure itself,
- Uses a familiar easy-to-understand Window GUI interface for data extraction process,
- Uses a simple installation process that opens the FileMan-based M System within 60 minutes

Most importantly, since MDE has no third party dependencies, such as data mappings, MDE is always upward compatible with changes in FileMan data structures. The MDE automatically recognizes and adapts itself to any updates or modifications made to FileMan data structures. Therefore, MDE assures users of data validity and integrity.

MDE is a natural "pipe" that seamlessly connects an M system to a relational database for use in a data warehouse, data mart, Web or any analytical system based on a relational database. Installing an MDE "pipe" onto each M system server quickly creates an infrastructure to achieve seamless sharing of M data across and to:

- User communities,
- Geographically separated sites,
- Functional departments,
- Facility complexes,
- Governmental agencies, and
- Private sector organizations.

For example, VA's VistA data and DoD's CHCS data can be extracted, migrated, and combined with each other in a common relational database. This capability enables both the VA and DoD to more efficiently and effectively achieve their shared mission of providing better healthcare services to active duty service personnel and to veterans.

Similarly, communities of users (such as VAMCs within a VA VISN, or MTFs within a DoD Region) can seamlessly share data within or among their organizational structures. The aggregation of a community's data can be used to:

- Enable local MTFs and VAMCs to more readily monitor and evaluate patient treatment courses, patient treatment pattern comparisons, and clinical outcome studies through timely access to CHCS and VistA data migrated to SQL/Server or other relational databases.
- Facilitate clinical, managerial, and administrative reporting and analysis by MTFs and VAMCs through COTS software tools and applications applied to CHCS or VistA data migrated to accessible data warehouses.
- Provide TRICARE and VA clinicians and clinical managers with the means to obtain current and complete data for conducting clinical risk management studies, developing clinical pathway guides, and ensuring patient compliance with clinical pathways.

- Permit the creation of searchable data archives, through use of XML with SQL Server or other relational databases, which archives are composed of CHCS or VistA data on data warehouses for use throughout the MHS or VA.
- Facilitate local MTF and VAMC initiatives in Web-based primary care to track and monitor patient treatments, including -- for example -- allergies, laboratory tests, prescriptions, radiology results, and medications, and to encourage patient participation in healthcare.

MDE installs in two components:

- **The MDE Server**. An M application comprising approximately 36 routines (less than 500k), which is installed on the M System server.
- **The MDE Client**. A 32-bit Windows application (.exe) which runs on Windows XP and Windows 2000.

The MDE has a straightforward process for its use:

1. A user at the MDE Client logs into the M System through the user's FileMan Verify and Access Codes.
2. Communication between MDE Client and MDE Server is established and maintained via TCP/IP.
3. Once the MDE Client is connected to FileMan, the user is presented with a GUI tree view of the FileMan data structure. The user points and clicks to choose files and fields within the files for data extraction. Field filters are available to limit the amount of data extracted.
4. Extracted data is stored in a temporary file located either on the MDE Client or a server other than the M system server.
5. SQL statements are automatically created and executed on the relational database to create tables.
6. Extracted data are automatically loaded into the relational database's table(s), the temporary file is deleted, and the MDE process is completed.

The Solution: MDE's Features and Functions

Feature/Function	Description
COTS technology	MDE does not require changes to FileMan data structures or M programming code. M data can be extracted within hours of MDE implementation.
All M data is made open	MDE natively reads the FileMan data dictionary; so M data immediately becomes fully accessible.
FileMan changes require no user action	MDE automatically detects all changes and updates to FileMan as it natively reads the FileMan data structure.

Feature/Function	Description
Create and save user-defined data extraction sets	MDE enables a user to create data extraction sets and save them for reuse or for sharing among a user community. Also, the MDE facilitates specialized data extractions for specific projects, such as reporting on *Network Performance Measurements*, *Balanced Scorecard* applications, and Pharmacy data warehouses.
Enterprise-wide catalog	MDE has an electronic "file cabinet" to store user-defined data extractions for reuse and sharing among a community of users. Also, the catalog is called by MDE's automated data extraction scheduler for unattended automatic extraction.
Automated data extraction scheduler	MDE enables a user to set the date, time, and frequency for automatically running a data extraction. No attendant is necessary to initiate data extraction during off-peak time.
Sub file-field support	MDE extracts all FileMan files and fields within files, and multiples, including sub-files (multiples).
Support non-standard FileMan files	MDE extracts non-standard FileMan files, such as a laboratory file.
Automatically resolve pointers or views internal ID number	At user's option, MDE can resolve and return the actual value of a field's data, such as the name of a patient. Alternatively, MDE can return the internal ID number of a field for use in data warehouse/mart creation or in linking tables within a relational database.
Connect site number with internal ID number	MDE enables data to be extracted across the enterprise (multiple care sites) while maintaining the integrity of identifying the location from which the data was acquired.
Automatically extract, create table, and deposit data into a relational database	In one automatic process MDE extracts data, creates SQL table(s), produces keys and indices, inserts data, and deposits data into a relational database.
FileMan file naming conventions preserved	MDE retains the naming of a relational database's table(s), column(s), and field(s) as the same given within FileMan. This naming aids a user to know the relationship between FileMan data structures and their counterparts in the relational database.

Feature/Function	Description
Microsoft DTS support	MDE supports Microsoft's DTS (Data Transformation Services).
Multiple site connection	MDE Client can connect to multiple FileMan sites to concurrently extract data. MDE combines this extracted data while uniquely identifying the data from each FileMan data structure.
VA KIDS installation	MDE incorporates the VA KIDS package as its installation process.
Additional filtering on top level fields	MDE enables up to seven field filters to be applied at the top level file, which permits a user to refine data extractions, limit the amount of data extracted, and target more specifically the data to be extracted.
Automated data transformation	MDE transforms certain FileMan-based values, such as date/time, into a format acceptable to the target relational database.
Automated data aggregation	MDE extracts like data-records, such as lab, from multiple sites and automatically aggregates and cleans them before depositing them into a relational database without any duplicate records.

The Solution: MDE's Benefits

The MDE has proven itself in many implementations, so the benefits listed below are neither hypothetical nor conceptual but, instead, they are demonstrable:

- MDE provides a seamless M-based data infrastructure that crosscuts departments, sites, facilities, government agencies, and private sector organizations to facilitate data sharing and analytical collaboration.
- MDE organizes FileMan data in a relational form and thereby facilitates the use of third-party analytical tools.
- MDE opens M data to Web access and graphical display of analytical results.
- MDE unifies patient data identifiers, allowing integration of data from disparate FileMan data structures and concentrates data from various sites to produce composite reports of all relevant fields of interest.
- MDE groups data in whichever way is most suitable for analysis.
- MDE strips off unneeded fields, making analysis easier and faster.
- MDE removes redundant data fields.
- MDE simplifies access to data so that knowledge of FileMan is not necessary.
- MDE offloads the slow, computationally intensive FileMan reports from primary hospital data systems to SQL-equipped peripheral systems.

- MDE allows the use of commercial-off-the-shelf (COTS) tools that non-programmers can use to generate reports, for example:
    - Natural language specifications and graphical diagrams.
    - User control over the type and appearance of data on reports, using functions that sort and filter the data.
- MDE enables use of data warehousing and data mart OLAP tools that:
    - Structure large amounts of data into a format suitable for use by healthcare analysts and epidemiologists who wish to identify and analyze, for example:
        - Those Persian Gulf veterans who have never been seen in the Persian Gulf Clinic.
        - The numbers of World War II veterans taking various types of hypertensive medications.
        - The outcomes of treating specific diseases with different approaches.
        - The efficacy of laboratory tests for determining disease prognoses.
        - The total costs incurred for the same illnesses treated at different sites.
        - Forecasting of trends of clinical resource consumption.
    - Combines data from different departments, such as clinical and financial, to create insightful data "views."
    - Uses executive information systems for decision support.
- MDE provides scalability. Just as M is highly scaleable, so is the MDE.