

MAPI Sample Message Store

Overview and release notes, 5/13/95

This document gives a brief overview of the sample message store service provider (SMPMS.DLL) that is part of the MAPI SDK.

Architecture

Disk layout. The sample message store is based on the operating system's file system. Each folder is a directory and each message a file. Additional files hold properties of folders and of the message store itself. A brief taxonomy follows:

*.FLD	Folder
*.MSG	Message
FOLDER.PRP	Folder properties (any folder)
MSGSTORE.PRP	Message store properties (root folder only)
RECFLDR.STG	Receive folder settings (root folder only)
CONTENTS.TBL	Folder contents table cache
OUTGOING.TBL	Outgoing queue table cache (root folder only)
*.TMP	Message in composition
*.RRT	Read receipt

Entry identifiers. A sample store entry identifier contains the path of the item to which it refers, relative to the store's root directory. This enables a sample message store to be copied using the filesystem with relative impunity (except to a subdirectory of another sample message store).

The sample message store does not distinguish between short- and long-term entry identifiers.

Message on storage. The **IMAPIProp** interface for all objects relies heavily upon the **IMessage** interface implemented in **IMSG.DLL**, which in turn is based upon OLE docfiles (the **IStorage** interface).

Interface lifetime and cross-referencing. The sample message store links child objects back to their parents for leak detection and invalidation when the parent object is closed before the child. Opening a table on an object (such as a folder) and then closing the folder invalidates the object, but not the table. This method is slightly more sophisticated than simply incrementing the object's reference count when the table is opened.

Contents and hierarchy tables. The **IMAPITable** interface for contents and hierarchy tables relies upon the **IMAPITableData** interface in **MAPI32.DLL**; the size of a folder is therefore limited to the amount of RAM available for the contents table. Initially, a contents table is created by scanning the directory for **.MSG** files, extracting some properties from each file, and writing a row for each message to the table data object. To speed up subsequent open operations, the table is then written out to a **.TBL** file, and the **.TBL** file is carefully kept up to date by subsequent changes in the folder.

Hierarchy tables are not cached; the sample message store uses notification internally to keep them up to date while they are open.

Supported features

Basic operations. The sample message store supports reading, creating, modifying, and deleting messages; browsing, creating, and deleting folders; submitting outbound messages to transports; and associating all types of attachments with messages. Multivalued properties are supported for all objects. The sample message store can be used as the default message store for a MAPI client.

Copy operations. The sample message store supports moving and copying messages and folders. These operations are delegated to MAPI support functions unless an operation occurs entirely within one sample message store.

Event notification. The sample message store supports event notification on tables (again, courtesy of **IMAPITableData**). Object notifications and new-message notifications are not supported.

Multithread safety. On 32-bit Windows platforms, all sample message store interfaces protect themselves from concurrent access by multiple threads.

Cross-platform support. The sample message store source code generates executable files for Windows 3.X, Windows 95, and Windows NT 3.51 and later from a single set of source files.

Unsupported features

Searches. (The **IMAPIFolder::SetSearchCriteria** method).

Categorization of contents tables.

Folder-associated information. (The **MAPI_ASSOCIATED** and **DEL_ASSOCIATED** flags).

IMAPIStatus interface.

Other methods including **IMAPIFolder::OpenProperty** and **IMsgStore::CopyTo**.

PR_ACCESS_LEVEL doesn't reflect open mode.

MAPI_E_FOLDER_CYCLE isn't detected.

Special behaviors of the **PR_ACCESS_LEVEL**, **PR_MESSAGE_SIZE**, **PR_NORMALIZED_SUBJECT**, and **PR_MAPPING_SIGNATURE** properties.

Cannot delete folders containing open messages.

Cannot copy or move open messages.

Configuration

The sample message store has two configurable properties: the path to the store's root directory and a password.

Known problems

No error message appears if a user types the wrong password when logging onto a sample message store.

PR_SUBJECT/PR_NORMALIZED_SUBJECT/PR_SUBJECT_PREFIX handling not up to spec.

Read-only bits in **PR_MESSAGE_FLAGS** aren't protected correctly.