

Printed: Wednesday, February 1, 2006 8:03:12 AM

---

#### Mailstore:

[ The current version of this file can be found at  
"//depot/projects/eudora2/docs/mailstore.txt" ]

The mailstore is the object that manages the user's messages and mailboxes. It doesn't know about accounts, nor networks, nor protocols. It knows about messages and mailboxes.

The structure of a mailstore will be identical across platforms. The idea is that you can take your mail to another platform without a conversion step.

A mailstore will be a single folder, with all the structure inside of it. On MacOS, the mailstore could be a package, which would emphasize its opaqueness.

"There are no 'user-serviceable' parts inside".

Eudora will support (eventually) multiple active mailstores at once (think archiving).

Each message in the mailstore can be looked up using a Message-ID hash. Since Message-IDs are not unique, we actually use a pair of values (MessageID-Hash, sequence #). This is encapsulated in the class MessageID.

A mailstore will contain three top-level items:

- a) a text file named "mailstore.xml"
- b) A folder named "Messages"
- c) a folder named "Mailboxes"

The XML file will contain:

- 1) The version of the mailstore
- 2) A collection for the mailstore
- 3) The transaction queue for the mailstore (expensive operations that are currently queued for execution)

The "Messages" folder will contain all the messages in the mailstore, stored one per file. All messages in the mailstore will exist here. There will be a (TBA) mechanism for breaking large mailstores into several subfolders (inside the Messages folder) in order to reduce access time and stress on the file system.

The "Mailboxes" folder will contain mailboxes. Each mailbox will simply be a list of messages, plus some cached information for each message (subject, status, etc), and a small amount of mailbox-specific information (window position, selected messages, etc).

The mailbox heirarchy is maintained by parent <--> child links inside the mailboxes; there are no folders on disk inside the Mailboxes folder.

Note #1: A mailbox can only reference messages in the same mailstore.

Note #2: Every message must be in at least one mailbox.

#### Operations on a mailstore:

- 0) Create a mailstore
  - 1) Open a mailstore
  - 2) Close a mailstore
  
  - 3) Create a mailbox
  - 4) Delete a mailbox
  - 5) Rename a mailbox
  - 6) Enumerate the mailboxes in the mailstore
- We need to provide a tree-like structure here.

Printed: Wednesday, February 1, 2006 8:03:12 AM

---

- 7) Provide notifications when a mailbox is created/deleted/changed
- 8) Add a new message to the mailstore
  - You need a message body, a mime structure, a mailbox, and a couple other things.
- 9) Copy a message from one mailbox to another.
- 10) Transfer a message from one mailbox to another.
- 11) Delete a message from a mailbox.
  - These three operations will occur in two stages: First, the mailboxes will be updated. Then, on a background thread, the messages will be updated. This background updating engine will be "repurposed" from Windows Eudora 7.
- 12) Provide a notification when a message is added/deleted/changed
- 13a) Find all the messages with a particular Message-ID hash.
  - this will return a list of MessageIDs.
- 13b) Find the message with a particular MessageID
  - This will return a pointer to the message.
- 14) Enumerate all the messages in the mailstore.
  - This will mostly be used for integrity checking.

Note #3: When a message is copied from one mailbox to another, the actual message file is not copied; the message is merely marked as being in two mailboxes.

Note #4: When a message is deleted from the last mailbox that contains it, the message file is deleted from the mailstore. (See note #2)

For details, see "//depot/projects/eudora2/common/include/mailstore.h"