Manual: FDO91 Manual
Appendix A: Contains examples of MIP and List Manager atom streams.
Last updated: May 1996

APPENDIX A
Examples

This appendix contains examples of MIP and List Manager atom streams. Most examples contain at least two atom streams showing the communication that occurs between the host and the client when an action is performed by a member.

The heading for each example indicates the example number and a description of the action that produced the atom streams. Each atom stream is introduced with a line stating what token is sent and in which direction the atom stream is going.

Example 1: Sending an E-mail Message

m4 Client to Host

```
atom$uni_start_stream
atom$mip_start_message
atom$mip_attr_addressee_type <0>
atom$mip_addressee "John Smith"
atom$mip_addressee "John Doe"
atom$mip_attr_addressee_type <1>
atom$mip_addressee "Archie"
atom$mip_subject "Next Year's Goals"
atom$mip_header_complete
atom$uni_end_stream
```

AT Host to Client

```
atom$uni_start_stream
atom$mip_header_accepted
atom$uni_end_stream
```

m5 Client to Host

```
atom$uni_start_stream
atom$mip_text "This is the first line of my message."
atom$mip_text "This is the second line of my message."
atom$mip_start_block
atom$mip_attr_data_type <259>
atom$mip_data <Magic Cap object data goes here...>
atom$mip_end_block
atom$mip_end_message
atom$uni_end_stream
```

AT Host to Client

```
atom$uni_start_stream
atom$mip_message_accepted
atom$uni_end_stream
```


Example 2: Requesting a List of New E-mail

m7 Client to Host

No atoms are sent - only the token...

AT Host to Client

```
atom$uni_start_stream
atom$lm_start_list
atom$lm_attr_list_type <0>
atom$lm_start_list_entry "Next Year's Goals"
atom$lm_attr_list_entry_id <2233440>
atom$lm_end_list_entry
atom$lm_start_list_entry "Subject of message 2"
atom$lm_attr_list_entry_id <2233441>
atom$lm_end_list_entry
.
.
```

More list entries here...

```
.
.
atom$lm_end_list
atom$uni_end_stream
```

Example 3: Requesting a Specific E-mail Message From a List

Assume that the e-mail list shown in Example 2 has just been received and one of its messages is requested...

m3 Client to Host

```
atom$uni_start_stream
atom$mip_message_id <2233440>
atom$uni_end_stream
```

AT Host to Client

```
atom$uni_start_stream
atom$mip_start_message
atom$mip_attr_message_type <0>
atom$mip_message_id <2233449>
atom$mip_message_datestr "94-08-03 11:42:15 EDT"
atom$mip_author "John Doe"
atom$mip_attr_addressee_type <0>
atom$mip_addressee "GlendaW"
atom$mip_addressee "YouWho"
atom$mip_subject "Yesterday's Meeting"
atom$mip_text "This is the first line of the message."
atom$mip_text "This is the second line of the message."
atom$mip_start_block
atom$mip_data <Magic Cap object data goes here...>
atom$mip_end_block
atom$lm_end_message
atom$uni_end_stream
```


Example 4: Requesting a List of News Articles

Ct Client to Host

```
atom$uni_start_stream
atom$de_data 0x00050001              ;See Note below
atom$uni_end_stream
```

AT Host to Client

```
atom$uni_start_stream
atom$lm_start_list
atom$lm_attr_list_type <5>
atom$lm_start_list_entry "Whitewater claims second
  Treasury official"
atom$lm_attr_list_entry_id <123-1>
atom$lm_end_list_entry
atom$lm_start_list_entry "Shuttle mission scrapped with
  two seconds to liftoff"
atom$lm_attr_list_entry_id <123-2>
atom$lm_end_list_entry
.
.
```

More list entries here...

```
.
.
atom$lm_end_list
atom$uni_end_stream
```

Note: In the initial request (Ct packet), a 32-bit data value is provided to specify the article request. The high-16 bits of this value specify the maximum number of articles desired; less than the maximum may be available, but more than the maximum will not be offered. The low 16-bits indicate which collection of articles is being requested. Currently, the following collections are available:

1 Top News

2 Sports News

3 U.S. Business News

4 World Business News

Example 5: Requesting a Specific News Article From a List

Assume that the article list shown in Example 4 has just been received and one of its articles is requested...

t6 Client to Host

```
atom$uni_start_stream
atom$de_data <123-1>                ;article ID
atom$uni_end_stream
```

AT Host to Client

```
atom$uni_start_stream
atom$mip_start_message
atom$mip_attr_message_type <4>
atom$mip_message_id <2233449>
atom$mip_message_datestr "94-08-04 14:05:09 EDT"
atom$mip_author "America Online"
atom$mip_subject "Whitewater claims second Treasury
  official"
atom$mip_text "Treasury Department General Counsel Jean
  Hanson resigned Thursday under pressure ..."
.
.
```

More text here...

```
.
.
atom$lm_end_message
atom$uni_end_stream
```


Example 6: Requesting a Stock Quote

pq Client to Host

```
atom$uni_start_stream
atom$de_data "AMER"
atom$uni_end_stream
```

AT Host to Client

```
atom$uni_start_stream
atom$mip_start_message
atom$mip_attr_message_type <3>
atom$mip_message_datestr "94-08-04 13:31:46 EDT"
atom$mip_author "AOL Stock Service"
atom$mip_subject "AMER - 59 at 13:12"
atom$mip_text "AMER - AMERICA ONLINE INC\n"
atom$mip_text "pr: 59 at 13:12\n"   ;current price
atom$mip_text "ch: Down 5/8\n"      ;change today
atom$mip_text "th: 59 3/4\n"        ;today's high
atom$mip_text "tl: 57 3/4\n"        ;today's low
atom$mip_text "to: 59 1/2\n"        ;today's open
atom$mip_text "yc: 59 5/8\n"        ;yesterday's close
atom$mip_text "vl: 95,200\n"        ;today's volume
atom$mip_text "yh: 92\n"            ;yearly high
atom$mip_text "yl: 43 1/2\n"        ;yearly low
atom$mip_text "pe: 77.6\n"          ;P/E ratio
atom$lm_end_message
atom$uni_end_stream
```

Note: The text of a pcquote (compressed) message may also contain lines beginning with "la:" (last ask), "lb:" (last bid), and "yi:" (yield). Programs accepting stock data should be prepared to accept, and ignore, lines that are not recognized (i.e., new data elements that are added to the host after the client software was written). Clients should also be able to handle the case where some of the normal data items are omitted.