Appendix 6A
Message API

GetDirectedMessage 2

GetMessage 3

GetMessageMode 4

GetMessageTimeout 5

SendDirectedMessage 6

SendMessage 7

SetMessageMode 9

SetMessageTimeout 10

GetDirectedMessage

Description Retrieves a user broadcast message which has been stored in the workstation.

Syntax

```
UINT32
GetDirectedMessage (
  UINT32    maxMsgLen,
  UINT8   msg[maxMsgLen]);
```

Input *maxMsgLen* Maximum length of message to store in the message buffer (58 or greater).

msg Address of message buffer to store the message in.

Return values Length (in bytes) of the message retrieved; zero means no message is waiting.

Remarks This corresponds to **SendDirectedMessage** which sends the message directly to the destination rather than to a common server connection.

If the *maxMsgLen* is insufficient to retrieve the complete message, the portion of the message not retrieved may be lost. The caller may set *maxMsgLen* to 58, which will retrieve longer messages in smaller parts.

See also GetMessage
SendDirectedMessage

GetMessage

Description Retrieves a user broadcast message stored on a specified connection.

Syntax

```
UINT32
GetMessage (
  UINT32    connHandle
  UINT32    maxMsgLen,
  UINT8   msg[maxMsgLen]);
```

Input

connHandle Connection handle specifying the server which has the message stored.

maxMsgLen Maximum length of message to store in the message buffer (58 or greater).

msg Address of message buffer to store the message in.

Return values Length (in bytes) of the message retrieved; zero means no message is waiting.

Remarks **GetMessage** corresponds to **SendMessage**, which sends the message to a common server connection.

If *maxMsgLen* is insufficient to retrieve the complete message, the portion of the message not retrieved may be lost. Because newer NetWare servers support the storage and retrieval of messages longer than 58 bytes, the caller may set *maxMsgLen* to 58, thereby retrieving longer messages in smaller parts.

See also GetDirectedMessage
SendMessage

GetMessageMode

Description Returns the current message handling mode for a specific connection or for the default case.

Syntax

```
UINT32
GetMessageMode (
  UINT32 connHandle);
```

Parameters *connHandle* Connection handle to be queried for a mode, or zero for the default mode.

Return values The message handling mode. This value is defined as follows:

Value	Server	Client
0	Hold client msgs ON	Retrieve and display ON
1	Hold client msgs OFF	Retrieve and display ON
2	Hold client msgs OFF	Retrieve and display OFF
3	Hold client msgs ON	Retrieve and display OFF

Remarks The directed messages follow the default mode and are not affected by any specific connection's message mode.

See also SetMessageMode (sibling).

GetMessageTimeout

Description Returns the number of milliseconds configured for the message to be displayed before the client automatically clears the message.

Syntax

```
UINT32
GetMessageTimeout (
   void);
```

Input Nothing.

Return values The number of milliseconds configured for the message to be displayed before it is automatically cleared.

Remarks This feature is considered disabled (user must clear the message) if the timeout is set to zero.

This value defaults to zero for backward compatibility with previous functionality.

See also SetMessageTimeout

SendDirectedMessage

Description Sends a message directly to one specified workstation.

Syntax

```
UINT32
SendDirectedMessage (
  UINT32    msgLen,
  UINT8   *msg,
  UINT32    tranType,
  UINT8   tranAddr[32]);
```

Input *msgLen* Message length (maximum = 58 bytes)

msg Address of message to be sent

tranType Transport type for the network address specified

tranAddr Transport-specific network address of the destination workstation

Return values

```
0         Successful
non-zero  Non-successful
```

Remarks This routine is similar to **SendMessage** except that it allows only one workstation to be specified at a time. It also sends the message directly to the workstation rather than via a common server connection. This method of sending a message will store the message in the destination client rather than at the server. The destination workstation is specified by its network address and transport type.

The transport type and address must be specified as defined by the Transport Service Interface. The destination workstation does not send an acknowledgment so this is not a guaranteed message delivery.

See also **SendMessage**

SendMessage

Description Sends a message to one or more workstations via a common server connection.

Syntax

```
                UINT32
                SendMessage (
  UINT32    connHandle,
  UINT32    msgLen,
  UINT8   *msg,
  UINT32    listLen,
  UINT32    list[listLen]);
```

Input

connHandle Handle of connection which is common between the source workstation and those specified in the destination list.

msgLen Message length.

msg Address of message to be sent.

listLen Number of destination stations in the destination list.

list List of destination connections.

Return values

0 Successful

non-zero Failure

Remarks This method of sending a message will store the message at the server until the destination workstation retrieves the message. The destination workstation(s) are specified by their connection number on the server.

Messages longer than 58 bytes may be broken up into smaller

```
messages by either this routine (if the server version does not
support long messages) or by the server (if the destination
workstation does not support long messages).  No language
translation is performed on the string by either this routine or
```

the server. The destination workstation will interpret the characters of the message according to its local code page.

See also **SendDirectedMessage**

SetMessageMode

Description Sets the current message-handling mode for a specific connection or for the default case.

Syntax

```
UINT32
SetMessageMode (
  UINT32 connHandle,
  UINT32 msgMode);
```

Parameters

connHandle Connection handle for the mode to be set, or zero for the default mode.

msgMode The message mode to be set. This value is defined as follows:

Value	Server	Client
0	Hold client msgs ON	Retrieve and display ON
1	Hold client msgs OFF	Retrieve and display ON
2	Hold client msgs OFF	Retrieve and display OFF
3	Hold client msgs ON	Retrieve and display OFF

Return values The previous message mode for the given connection or the default mode.

Remarks The directed messages follow the default mode and are not affected by any specific connection's message mode.

See also GetMessageMode

SetMessageTimeout

Description Sets the number of milliseconds configured for the message to be displayed before the client automatically clears the message.

Syntax

```
UINT32
SetMessageTimeout (
   UINT32 timeout);
```

Input

timeout New message timeout (in milliseconds).

Return values The value of the previous timeout.

Remarks None.

See also GetMessageTimeout