

Manual: FDO91 Manual

Chapter 19: File Transfer Protocol defines file transfer protocol atoms and provides information for handling file transfer.

Last updated: March 1996

CHAPTER 19

File Transfer Protocol

The File Transfer protocol (protocol ID 7) consists of atoms that provide error-free transfer of files between the online service client application and the host system. File uploading and downloading are supported; in addition to single file transfers, batch downloads can also be performed. Downloads can be interrupted and resumed at a later time. Data compression is supported using a run-length encoding scheme.

File Transfer Protocol Atoms

The File Transfer protocol atoms are described in alphabetical order in the rest of this chapter.

atom\$xfer_abort

0 (\$00)

Description

atom\$xfer_abort terminates the current file transfer when the member clicks the cancel button in the File Transfer dialog box while the file is uploading. This atom is sent by the client.

Syntax

```
atom$xfer_abort
```

Return Value

Unchanged.

Example

The following is an example that terminates the current file transfer:

```
atom$uni_start_stream <00x>
.
.
.
atom$xfer_start_object <00x>
atom$xfer_bool_mail <yes>
atom$xfer_atr_request_id <123932171>
atom$xfer_atr_title "Updated Phone List"
atom$xfer_atr_file_size <25600>
atom$xfer_atr_file_name <mtfglf.xls>
atom$xfer_atr_create_date <508741200>
.
.
.
↵ atom$xfer_abort
atom$uni_end_stream <00x>
```

atom\$xfer_atr_create_date

23 (\$17)

Description

atom\$xfer_atr_create_date is sent by the host and indicates the date the file was created.

Syntax

```
atom$xfer_atr_create_date <create_date>
```

<create_date> A 32-bit integer indicating the creation date of a file attached to a message in number of seconds since midnight January 1, 1980 GMT.

Return Value

Unchanged.

Example

The following is an example that indicates the date the file was created:

```
atom$uni_start_stream <00x>
.
.
atom$xfer_start_object <00x>
atom$xfer_atr_library "Applications"
atom$xfer_bool_mail <yes>
atom$xfer_atr_request_id <123932111>
atom$xfer_atr_title "PK Zip"
atom$xfer_atr_file_size <42166>
atom$xfer_atr_file_name <pkzip.exe>
↔ atom$xfer_atr_create_date <508741200>
.
.
atom$xfer_end_object
.
.
atom$uni_end_stream <00x>
```

atom\$xfcr_atr_end_object

19 (\$13)

Description

atom\$xfcr_atr_end_object ends the current XFER object. This atom is sent by the host and triggers the client to display a dialog box asking the member where to save the file.

Syntax

```
atom$xfcr_end_object
```

Return Value

Unchanged.

Example

The following is an example that ends the current XFER object:

```
atom$uni_start_stream <00x>
.
.
.
atom$xfcr_start_object <00x>
atom$xfcr_atr_library "Applications"
atom$xfcr_bool_mail <yes>
atom$xfcr_atr_request_id <123932111>
atom$xfcr_atr_title "PK Zip"
atom$xfcr_atr_file_size <42166>
atom$xfcr_atr_file_name <pkzip.exe>
atom$xfcr_atr_create_date <508741200>
.
.
.
⇒ atom$xfcr_end_object
.
.
.
atom$uni_end_stream <00x>
```

atom\$xfer_atr_file_name

18 (\$12)

Description

`atom$xfer_atr_file_name` is sent from the host to indicate the name of the attached file to be transferred.

Syntax

```
atom$xfer_atr_file_name
```

Return Value

Unchanged.

Example

The following is an example that indicates the name of the attached file to be transferred:

```
atom$uni_start_stream <00x>
.
.
atom$xfer_start_object <00x>
atom$xfer_bool_mail <yes>
atom$xfer_atr_request_id <123932171>
atom$xfer_atr_title "Updated Phone List"
↪ atom$xfer_atr_file_size <25600>
atom$xfer_atr_file_name <mtfglf.xls>
atom$xfer_atr_create_date <508741200>
.
.
atom$xfer_end_object
.
.
atom$uni_end_stream <00x>
```

atom\$xfer_atr_file_size

16 (\$10)

Description

atom\$xfer_atr_file_size is sent from the host to indicate the size, in bytes, of the attached file to be transferred.

Syntax

```
atom$xfer_atr_file_size <size>
```

<size> The size of the file to be transferred.

Return Value

Unchanged.

Example

The following is an example that indicates that the size of the file to be transferred is 25600 bytes:

```
atom$uni_start_stream <00x>
.
.
.
atom$xfer_start_object <00x>
atom$xfer_bool_mail <yes>
atom$xfer_atr_request_id <123932171>
atom$xfer_atr_title "Updated Phone List"
⇒ atom$xfer_atr_file_size <25600>
atom$xfer_atr_file_name <mtfglf.xls>
atom$xfer_atr_create_date <508741200>
.
.
.
atom$xfer_end_object
.
.
.
atom$uni_end_stream <00x>
```

atom\$xfer_atr_library

21 (\$15)

Description

atom\$xfer_atr_library is sent from the host to indicate the name of the software library that contains the attached file to be transferred.

Syntax

```
atom$xfer_atr_library <library_name>
```

<library_name> The name of the software library.

Return Value

Unchanged.

Example

The following is an example that indicates the name of the software library that contains the file to be transferred:

```
atom$uni_start_stream <00x>
.
.
.
atom$xfer_start_object <00x>
↵ atom$xfer_atr_library "Applications"
atom$xfer_bool_mail <yes>
atom$xfer_atr_request_id <123932111>
atom$xfer_atr_title "PK Zip"
atom$xfer_atr_file_size <42166>
atom$xfer_atr_file_name <pkzip.exe>
atom$xfer_atr_create_date <508741200>
.
.
.
atom$xfer_end_object
.
.
.
atom$uni_end_stream <00x>
```


atom\$xfer_atr_request_id

15 (\$F)

Description

atom\$xfer_atr_request_id sends the ID from the host to the client in response to a request for an attached file. The e-mail ID is used for text files and the file ID is used for attached files within software libraries.

Syntax

```
atom$xfer_atr_request_id <id>
```

<id> The ID of the attached file or software library.

Return Value

Unchanged.

Example

The following example sends the file ID to enable the transfer of an attached file:

```
atom$uni_start_stream <00x>
.
.
.
atom$xfer_start_object <00x>
atom$xfer_bool_mail <yes>
⇨ atom$xfer_atr_request_id <123932171>
atom$xfer_atr_title "Updated Phone List"
atom$xfer_atr_file_size <25600>
atom$xfer_atr_file_name <mtfglf.xls>
atom$xfer_atr_create_date <508741200>
.
.
.
atom$xfer_end_object
.
.
.
atom$uni_end_stream <00x>
```

atom\$xfer_atr_title

17 (\$11)

Description

atom\$xfer_atr_title is sent from the host to indicate the title of the message to which the file to be transferred is attached.

Syntax

```
atom$xfer_atr_title
```

Return Value

Unchanged.

Example

The following is an example that indicates that the title of the message is "Updated Phone List":

```
atom$uni_start_stream <00x>
.
.
atom$xfer_start_object <00x>
atom$xfer_bool_mail <yes>
atom$xfer_atr_request_id <123932171>
↵ atom$xfer_atr_title "Updated Phone List"
atom$xfer_atr_file_size <25600>
atom$xfer_atr_file_name <mtfglf.xls>
atom$xfer_atr_create_date <508741200>
.
.
atom$xfer_end_object
.
.
atom$uni_end_stream <00x>
```

atom\$xfer_bool_in_progress

10 (\$A)

Description

atom\$xfer_bool_in_progress is sent from the client to indicate whether a file download is in progress.

Syntax

```
atom$xfer_bool_in_progress
```

Return Value

Unchanged.

Example

The following example indicates that a download is in progress:

```
atom$xfer_start_object <00x>  
atom$xfer_bool_mail <yes>  
↔ atom$xfer_bool_in_progress <1>  
atom$xfer_end_object  
atom$uni_end_stream <00x>
```

atom\$xfer_bool_mail

22 (\$16)

Description

atom\$xfer_bool_mail is sent by the host to indicate whether mail system tokens or software library tokens should be used to view and download the file to be transferred.

Syntax

```
atom$xfer_bool_mail
```

Return Value

Unchanged.

Example

The following is an example that indicates that mail system tokens should be used to download and view the file:

```
atom$uni_start_stream <00x>
.
.
.
atom$xfer_start_object <00x>
atom$xfer_atr_library "Applications"
↪ atom$xfer_bool_mail <yes>
atom$xfer_atr_request_id <123932111>
atom$xfer_atr_title "PK Zip"
atom$xfer_atr_file_size <42166>
atom$xfer_atr_file_name <pkzip.exe>
atom$xfer_atr_create_date <508741200>
.
.
.
atom$xfer_end_object
.
.
.
atom$uni_end_stream <00x>
```

atom\$xfer_clear_rle_flag

8 (\$08)

Description

atom\$xfer_clear_rle_flag clears the RLE (run length encoding) flag. This atom is sent by the client when the member unchecks the file decompression checkbox within the Download Preferences dialog box.

Syntax

```
atom$xfer_clear_rle_flag
```

Return Value

Unchanged.

Example

```
atom$xfer_clear_rle_flag
```

atom\$xfer_finish_later

5 (\$05)

Description

atom\$xfer_finish_later interrupts the current file transfer when the member clicks the cancel button in the File Transfer dialog box while the file is uploading. This atom is sent by the client.

Syntax

```
atom$xfer_finish_later
```

Return Value

Unchanged.

Example

```
atom$uni_start_stream <00x>
.
.
.
atom$xfer_start_object <00x>
atom$xfer_bool_mail <yes>
atom$xfer_atr_request_id <123932171>
atom$xfer_atr_title "Updated Phone List"
atom$xfer_atr_file_size <25600>
atom$xfer_atr_file_name <mtfglf.xls>
atom$xfer_atr_create_date <508741200>
.
.
.
↪ atom$xfer_finish_later
atom$uni_end_stream <00x>
```

atom\$xfer_refresh_prefs

9 (\$09)

Description

atom\$xfer_refresh_prefs reloads all of the download preferences set by the member, such as automatic decompression and setting the number of downloads to retain. This atom is sent by the client.

Syntax

```
atom$xfer_refresh_prefs
```

Return Value

Unchanged.

Example

```
atom$xfer_refresh_prefs
```

atom\$xfer_set_rle_flag

7 (\$07)

Description

atom\$xfer_set_rle_flag sets the RLE (run length encoded) flag. This atom is sent by the client when the member checks the file decompression checkbox within the Download Preferences dialog box.

Syntax

```
atom$xfer_set_rle_flag
```

Return Value

Unchanged.

Example

```
atom$xfer_set_rle_flag
```


atom\$xfers_start_object

14 (\$E)

Description

atom\$xfers_start_object is sent from the host to indicate the start of an XFER object such as a file to be downloaded. The argument to this atom specifies whether the file will be downloaded immediately (Download Now is selected by the member), or the file will be added to the download manager queue (Download Later is selected by the member).

Syntax

```
atom$xfers_start_object (<0>|<1>)
```

(<0>|<1>) The status of the file to be downloaded. Valid values are:

<0> The file will be downloaded immediately.

<1> The file will be downloaded later.

Return Value

Unchanged.

Example

The following is an example that specifies that a file will be downloaded immediately:

```
atom$uni_start_stream <00x>
.
.
.
⇒ atom$xfers_start_object <00x>
.
.
.
atom$xfers_end_object
```

atom\$xfer_toggle_signoff

4 (\$04)

Description

atom\$xfer_toggle_signoff toggles the state of the sign off when finished flag. This atom is sent when the member checks the Sign Off After Transfer checkbox. This atom is sent by the client.

Syntax

```
atom$xfer_toggle_signoff
```

Return Value

Unchanged.

Example

```
atom$xfer_toggle_signoff
```