

Manual: FDO91 Manual

Chapter 12: Chat Protocol defines chat protocol atoms and provides information for chat functionality.

Last updated: August 1996

CHAPTER 12

Chat Protocol

The Chat protocol (protocol ID 11) consists of atoms that handle real-time conferencing and messaging within people connection rooms, forum conference halls, and auditoriums. The Chat protocol also provides the following functionality:

- Opening and closing chat rooms
- Adding and removing members from chat rooms
- Managing chat preferences
- Sending chat messages

Chat Protocol Atoms

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

atom\$chat_add_user

1 (\$01)

Description

atom\$chat_add_user notifies the host when a member enters a chat room. The member's name is added to the local list of members currently in the chat room. If the member enables arrival notification within chat preferences, other members in the chat room will be notified that the member has arrived. This atom is sent by the client.

Syntax

```
atom$chat_add_user <name>
```

<name> An ASCII string (null-terminated) representing the screen name of the member who is entering the chat room.

Return Value

Unchanged

Example

The following is an example of how to add a member to a chat room:

```
atom$var_number_set <A, 1>  
atom$var_number_save <A, 65537>  
atom$man_update_display  
atom$chat_room_open  
↪ atom$chat_add_user <54x, 41x, 44x, 75x, 6ex, 6ex>
```

atom\$chat_end_object

2 (\$02)

Description

atom\$chat_end_object terminates the current chat object in context. For example, each screen name on the member list for a current chat room is a chat object. This atom is sent by the host.

Syntax

```
atom$chat_end_object <object>
```

<object> An ASCII string representing the name of the object to be terminated.

Return Value

Unchanged

Example

The following is an example of how to terminate the current chat object in context:

```
atom$chat_add_user <54x, 41x, 44x, 75x, 6ex, 6ex>
atom$mat_relative_tag <19>
atom$act_set_inheritance
.
.
.
↵ atom$chat_end_object
```

atom\$chat_message

5 (\$05)

Description

atom\$chat_message contains the text string of the chat message that is sent when a member enters text and selects the Send button. This atom is sent by the client.

Syntax

```
atom$chat_message <text>
```

<text> A text string that represents the chat message.

Return Value

Unchanged

Example

The following is an example of how to send the text string of a chat message:

```
atom$uni_start_stream
atom$de_start_extraction
atom$de_validate
↪ atom$chat_message <"Hello">
.
.
.
atom$uni_end_stream
```

atom\$chat_refresh_pref

6 (\$06)

Description

atom\$chat_refresh_pref reloads all the chat-related preferences set by the member, such as member arrival notification, member exit notification, double-spacing of incoming messages, alphabetizing the member list, and enabling chat room sounds. This atom is sent by the client.

Syntax

```
atom$chat_refresh_pref
```

Return Value

Unchanged

Example

The following is an example of how to reload all chat-related preferences:

```
atom$uni_start_stream  
atom$uni_set_context_relative <1>  
↔ atom$chat_refresh_pref  
atom$man_end_context  
atom$uni_end_stream
```

atom\$chat_remove_user

3 (\$03)

Description

atom\$chat_remove_user is sent by the client when a member leaves the current chat room. The member's name is removed from the local list of members currently in the chat room. If the member enables exit notification within chat preferences, other members in the chat room will be notified that the member has left. This atom is sent by the client.

Syntax

```
atom$chat_remove_user <name>
```

<name> An ASCII string (null-terminated) representing the screen name of the member who is exiting the chat room.

Return Value

Unchanged

Example

The following is an example of how to remove a member from a chat room:

```
atom$var_number_set <A, 1>
atom$var_number_save <A, 65537>
atom$man_update_display
atom$chat_room_open
atom$chat_add_user <54x, 41x, 44x, 75x, 6ex, 6ex>
.
.
.
↵ atom$chat_remove_user <Akoontz>
```

atom\$chat_room_open

0 (\$00)

Description

atom\$chat_room_open is sent to the client when a member enters a chat room. This atom precedes **atom\$chat_add_user**, which adds the member's name to the local list of members currently in the chat room. If the member enables arrival notification within chat preferences, other members in the chat room will be notified that the member has arrived. This atom is sent by the client.

Syntax

```
atom$chat_room_open
```

Return Value

Unchanged

Example

The following is an example of opening a chat room:

```
atom$var_number_set <A, 1>  
atom$var_number_save <A, 65537>  
atom$man_update_display  
↔ atom$chat_room_open  
atom$chat_add_user <54x, 41x, 44x, 75x, 6ex, 6ex>
```

atom\$chat_show_arrival

4 (\$04)

Description

atom\$chat_show_arrival displays a notice from the OnlineHost announcing the arrival of a member to other members in a chat room. This atom is sent only if the member has enabled arrival notification within chat preferences. This atom is sent by the client.

Syntax

```
atom$chat_show_arrival (<yes>|<no>)
```

<yes> Arrival notification is enabled.

<no> Arrival notification is not enabled. (Default)

Return Value

Unchanged

Example

The following is an example of how to announce the arrival of a member to a chat room:

```
atom$chat_show_arrival <yes>
```