

Utility Functions

Before you begin basic ATCP usage, it might be wise to learn what utility functions I've included, and how to use them.

Utility: idATCP

Not a function, but it contains the ATCP plugin's ID, which you should use in OnPluginBroadcast to make sure it's an ATCP broadcast you're receiving, and not from some other unrelated plugin.

Utility: EnableModuleATCP(string)

Use to enable an ATCP module, such as "map_display" for Client.Map, or "ping" for Client.Ping. Should only be used in response to a broadcast where msg is 0.

Utility: SendATCP(msg)

Attempts to send an ATCP message to the server. (It validates the message first to be sure you're sending a valid command and valid parameters)

Utility: UnpackATCP(atcp)

Splits the text from a msg(1) broadcast into a message name and the message content (i.e. "Room.Exits", "n,ne,e,s,w").

Receiving

The name and content should be obvious; Module is the name you should pass to EnableModuleATCP() to enable usage of that message.

See [roomname.xml](#) for example usage of one of these messages, Room.Brief.

Name: Client.Compose

Content: title \n text

Module: *composer*

Indicates entry into an Achaean composer, where title is a suggested title of a client-side window, and text is the text that was already in the file being edited.

Name: Char.Name

Content: name \n name with title

Module: *char_name*

Pretty obvious: includes the shortname of the adventurer connecting, as well as the long name-with-title sort of name.

Name: Char.Vitals

Content: vital statistics

Module: *char_vitals*

Includes a line defining the character's current vital stats (those shown on the prompt, including experience), as well as their maximum values. It IS affected by recklessness, masochism, whatever that one affliction is.

Name: Client.VoteReminder

Content: label ID \n vote url

Module: *topvote*

I'm not quite sure what label ID actually does, so you might want to ignore that. However, I believe this is what powers the Nexus vote button, so it shouldn't be hard to figure out.

Name: Room.Brief

Content: room name

Module: *room_brief*

Pretty obvious. This is used in the example [roomname.xml](#) plugin.

Name: Room.Exits

Content: list of exits

Module: *room_exits*

It doesn't list hidden exits, or that wouldn't be any fun. The exits are formatted like "ne,e,w,nw". This is the message that powers Trevize's nexus-style compass plugin.

Name: Client.Ping

Content: N/A

Module: *ping*

This is only sent in response to a 'ping' message sent by the client. Its sole purpose is really just to figure out how much latency there is between you and the server, as you can measure the time it takes between sending the 'ping' and receiving this message.

Name: Client.Map

Content: map_url

Module: *map_display*

Sent along with Room.Brief, to my knowledge. Contains the url of a map-image of your current surroundings, which you can see for yourself by logging into the Flash client and clicking "Map" on the left. It includes coordinates in the URL just like the ones on the MAP, so you could parse those out if you wanted. Other than that, it's probably not too easy to write a plugin that actually shows that image in the client (though not impossible).

Sending

There are only a handful of messages you can send, and you should use them only in situations where they're really called for.

Message: olesetbuf

Parameters: text

Module: *composer*

Should only be used after a Client.Compose message has been received. This command is used when the composer is being closed and the text saved to the file; should be followed up with a '*s' command, which is NOT ATCP. That is to say, send this message with SendATCP(), then send *s with just Send(), which treats it as a normal command.

Message: keepalive

Parameters: N/A

Module: *keepalive*

Does nothing except tell the server "I'm still here, even if I'm not really typing much". Best used periodically (depending on your timeout) after Client.Compose and before closing the composer.

Message: voted

Parameters: N/A

Module: *topvote*

In terms of Nexus, it tells the server that you clicked on its vote button. Its use is somewhat limited in other clients, but kindly avoid using it when you haven't actually voted...

Message: ping

Parameters: current average_ping

Module: *ping*

Use this to request a Client.Ping message from the server. If that's all you want, a 0 might work as the parameter, but if you want to actually give meaningful information that's up to you.