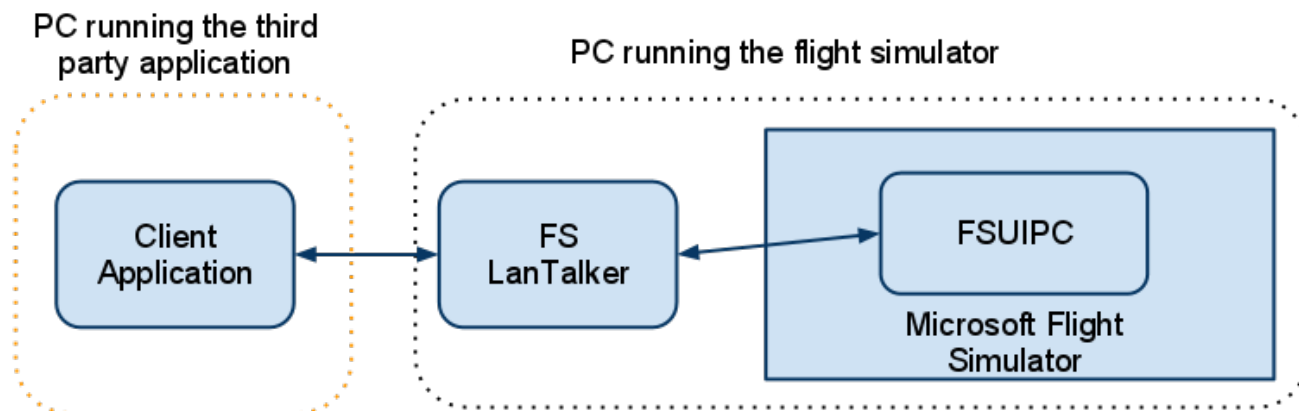


# HowTo FSLanTalker2

Last Update: 21 July 2011



## Connecting to FSLanTalker2

Start a TCP connection on port 3000 like

```
this.tcpClient = new TcpClient();
try
{
    tcpClient.Connect("127.0.0.1", 3000); }
catch
{
    MessageBox.Show("Connection Error"); }
```

The server answers with the current used protocol version.

**TalkerV:n**

n = version number

You must answer with **TalkerV:n:OK** or request another version with **TalkerV:NOK:x**

You will receive **TalkerV:x:OK** or **TalkerV:NOK**. The server closes the connection after the NOK message was send.

## Read data from the FS

To read data from the FS via the FSLanTalker send the message READ with the desired offset

separated by the “.”.

Example to read the latitude:

Client -> Server

**READ:0560**

You can read several offsets separated by an “|”:

Client -> Server

**READ:0560|READ:0568|...**

The server answers with the data read from FS. There is no conversion of the data.

Answer for single offset 0560:

Server -> Client

**24475928671485952**

Answer for offset 0560 + 0568:

Server -> Client

**24475928671485952|346146811405139968**

If something went wrong, the server sends a “NOK” message.

## Write data to the FS

To write data to the FS via the FSLantTalker send the message WRITE with the desired offset + data separated by the “.”

Example to pause the flightsim

Client -> Server

**WRITE:0262:1**

You can write several offsets separated by an “|”:

Client -> Server

**WRITE:0262:1|WRITE:05DE:1|...**

If the server was able to process the write request, it will answer with an “OK”. If something went wrong, the server sends an “NOK” message.

## Mixing read write request

You can mix read/write requests:

Client -> Server

**READ:0560|WRITE:0262:0|READ:0568|WRITE:05DE:1**

The servers answer could be:

Client -> Server

24475928671485952|OK|346146811405139968|NOK

## AI Traffic

You can request different information about the AI traffic around you:

Client -> Server

**TRAFFIC:POSITION** - returns all AI flights position

Server -> Client

Answer: **ID:CALLSIGN:LATITUDE:LONGITUDE:HEADING:ALTITUDE**

ID - FS ID of the AI plane

Callsign - The callsign used in the FS (COA121, DAL534, ...)

Latitude - The latitude of the AI plane

Longitude - The longitude of the AI plane

Heading - The heading of the AI plane

Altitude - The altitude of the AI plane in feet

## Other Commands

Client -> Server ; Server -> Client

**CLOSE** - The connection will be closed after the message was send.

Server -> Client

**REQ:NOK** - If the request could not be processed.