

**TCPIntf User's Guide
(TCP/IP Interface)
1.0b01**

TCPIntf is Copyright © 2007 - Pete Loveall AE5PL pete@ae5pl.net

Use of the software is acceptance of the agreement to not hold the author or anyone associated with the software liable for any damages that might occur from its use.

Table of Contents

| | |
|---|---|
| Section 1 - Introduction | 1 |
| Section 2 - Program Requirements and Description..... | 2 |
| Section 3 - Configuration Parameters | 3 |
| TNCSerialInterface Parameters | 4 |
| PortClassName=TCPIntf | 4 |
| SerialIPAddress= | 4 |
| SerialIPPort= | 4 |
| Section 4 - Recommended Configurations | 5 |
| Section 5 - Installation Instructions | 6 |

Section 1 - Introduction

TCPIntf was written to provide a universal interface between Java applications and a serial port. At the time of this document, IcomInterface and KISSInterface of javAPRSSvr can make use of TCPIntf. TCPIntf uses the standard OS file interface to access the serial port and is therefore under the constraints of the operating system (cannot set serial port parameters).

The TCPIntf source code is published for use by other developers to work with the TNCSerialInterface.

Section 2 - Program Requirements and Description

TCPIntf is designed to run on any OS with any recent Java Virtual Machine. The interface requires another application to provide the serial data (can be bidirectional) on a TCP port. The other application must be listening on that port BEFORE javAPRSSvr starts. There is no error recovery in TCPIntf for lost connections. This is designed for local connections only.

TCPIntf is comprised of a number of classes which Java looks at as objects. The main class is TCPIntf. This class is called at startup, sets parameters, and begins execution of the different support threads.

TCPIntf works in conjunction with the TNCSerialInterface to provide full, bidirectional communication with the serial port. All IGate and TNC logic is handled at in other classes which leaves the TCPIntf to concentrate on sending and receiving serial data.

Section 3 - Configuration Parameters

The configuration parameters reside in a configuration file which, by default, is called javaprssvr.cfg. You can use any text file if you pass the name into javAPRSSvr as a command line parameter.

The parameters are CASE SENSITIVE. Defaults are shown below.

NOTE: UNLESS YOU REQUIRE A SETTING OTHER THAN THE DEFAULT, DO NOT INCLUDE ANY PARAMETERS WITH DEFAULT SETTINGS.

List parameters may be defined on the line or may be defined in a text file. If defined on the line, each entry is separated by a semicolon. If defined in a file, each entry is put on a separate line. Do not put blank lines in the file. The file must have the extension .lst For instance, this would be the definition for hubs where you want to connect to first.aprs.net and second.aprs.net port 1313:

```
hubs=first.aprs.net:1313;second.aprs.net:1313
```

Or you could have the following 2 lines in hubs.lst:

```
first.aprs.net:1313  
second.aprs.net:1313
```

You would then put the following line in your configuration file:

```
hubs=hubs.lst
```

(R) at the beginning of the parameter description means that the parameter can be changed on-the-fly from the console with either the S or R commands.

TNCSerialInterface Parameters

PortClassName=TCPIntf

This is the name of the serial port class.

Use this parameter to specify the name of the serial port class. It defaults to the Random Access File interface.

SerialIPAddress=

This is the IP address of the serial data provider.

For example, 127.0.0.1 would make a serial connection to the local host.

SerialIPPort=

This is the TCP port of the serial data provider.

For example, 1234 would make a serial connection to port 1234.

Section 4 - Recommended Configurations

javaprssrvr.cfg:

<snip>

PortClassName=TCPIntf

SerialIPAddress=127.0.0.1:24580

<snip>

Section 5 - Installation Instructions

TCPIntf is included in all of the combined jar and exe files. Simply add PortClassName=TCPIntf to activate it.