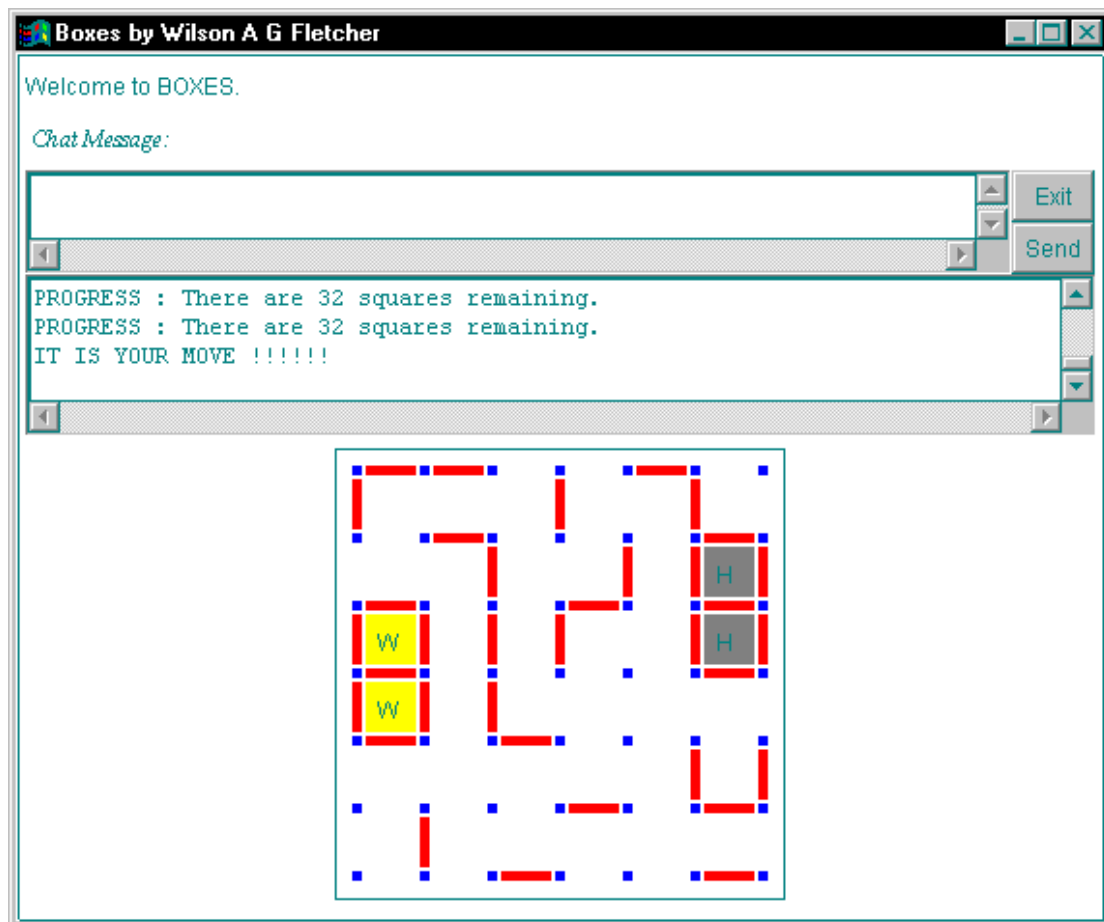


# Boxes

## Client User Manual

by

Wilson A G Fletcher



( Tested with Netscape 3.0 )

# Contents

<b>THE BOX CLIENT</b>	<b>3</b>
<b>Setting up the Client</b>	<b>4</b>
Place all the Client Files in the same directory	4
Set the CLASSPATH variable:	5
<b>Executing the Client</b>	<b>5</b>
Execute the client using the JAVA Interpreter	5
<b>THE BOXES CLIENT</b>	<b>6</b>
<b>Boxes Client Description</b>	<b>6</b>
<b>Operating the Client</b>	<b>7</b>
Connect to a known Boxes Server	7
Decide to Start a New Game	7
Decide to Join an Existing Game	7
Play the Game	8
The Client Window	8

# The Box Client

These notes instruct you how to install the client locally and describe basic operation of the client. If using the browser link then ignore the notes about installing the client.

In it's present state this software provides the following features:

- Has been tested on Windows95 and NT 4.0. Has been run locally on one machine and over a TCP/IP network with the server running on an NT 4.0 server. Has been tested in Netscape 3.0 and Microsoft Internet Explorer 3.02 and as a Java application.

- Allows multiple simultaneous games

- Works well in the way it is meant to at present. Many things could be improved, but then I would have to give up my job to find the time.

- Marks each completed BOX with the first Initial of the user who completed it and colours boxes in different colours.

- Displays the number of BOXES left to complete

- Prevents users moving out of turn

- Waits until other clients have approved a move before committing it.

- Allows the user to connect to any boxes server on a TCP/IP network as long as they know it's IP address and Port. (Note the server uses PORT 3606).

- Allows users to quit at will. If the remaining players reaches a minimum then the server forces all other players to quit. NOTE : In this version the MAX number of players = MIN number of players = 2.

- Allows the person who started the game to decide how big the grid is.

- Allows users to send broadcast messages to other users in the same game.

- User input is obtained with dialog boxes. Note the dialog boxes are only sort of pseudo modal.

- Makes use of sound when run as an applet to provide user feedback. ie. metallic ting noise if a mouse click is invalid. A latch noise if a mouse selection is valid.

In it's present state this software is limited in the following ways:

- The Prompts to users ie. Your go ! or Not your go are not overly visible until you are used to using the software.

- Only allows 2 users per game. The design should theoretically allow multiple users but it has not been tested for more than two users.

- When a game ends there isn't any fanfare announcing the winner. The game just ends and the client tool just doesn't let you move (because there

are not any more moves to make). NOTE: You can work out who won by counting the boxes with initials in them.

No online help is provided. This document could easily be converted and placed online.

NOTE: Once the game is finished you can continue to chat and maybe organise a rendezvous for rematch.

Users must reconnect to the server to play a second game.

The dialog boxes do not allow features such as the ability to tab between fields.

## Setting up the Client

### Place all the Client Files in the same directory

Create a directory on the HOST computer that is accessible from your Web server.

EG : mkdir boxes\client

Copy the following files to the new directory

BoxesClient.class  
MoveInfo.class  
Square.class  
BorderPanel.class  
DelayedExecute.class  
GameGridPanel.class  
LoginDialog.class  
NewOrJoinDialog.class  
InfoDialog.class<sup>1</sup>  
MultiLineLabel.class<sup>2</sup>  
BOX\_CONST.class  
GAME\_CONST.class  
SQUARE\_CONST.class  
appMSG.au  
ting.au  
bong.au  
toot.au

---

<sup>1</sup> Java in a Nutshell , David Flanagan , O'Reilly & Associates , 1996.

<sup>2</sup> Java in a Nutshell , David Flanagan , O'Reilly & Associates , 1996.

Boxes.html

**OR (If using boxes.jar)**

boxes.jar

Boxes.html

Set the CLASSPATH variable:

Before anyone can connect to the client you must have the CLASSPATH environment variable set on the host computer.

Set the CLASSPATH to the directory that contains the class required for the server. As well as the class for the JDK.

EG : CLASSPATH=c:\java\lib;c:\boxes\server;c:\boxes\client

**OR (If using boxes.jar)**

EG : CLASSPATH=c:\java\lib\classes.zip;c:\boxes\client\boxes.jar

## **Executing the Client**

Execute the client using the JAVA Interpreter

To execute the client start with the BoxesClient class. This is the main program in the Boxes client suite.

EG : java BoxesClient

The client will now execute. Operation of an application client is the same as operation of an Applet client. See the following pages for instructions.

# The Boxes Client

## Boxes Client Description

The Boxes client provides an interface for playing boxes over a TCP/IP network. The client can be run using a java enabled web browser such as Netscape 2.0 or as an application by using the java interpreter.

The client application allows users to connect to a server by specifying the server IP address and a port as well as the user name for connection ID. The name is not mandatory for the server, the server will always know where you are, the client will force you to type a name however.

The client gives users the option of creating their own game or joining an existing game. A list of existing games is provided by the server and displayed by the client. At present the user must type the name of the game they wish to join. In future versions these would be provided in a pull down list box to allow easy selection.

If the user specifies a JOIN request, then they must enter a game to join. If they specify a new game they must indicate the size of the game grid. Note the grid is a square so only one value is required.

Once connected users alternate. The server regulates which user may move next. The server does not validate moves apart from determining whether a particular user is allowed to move. It is unlikely the server will refuse a move on the basis that it is not that user's turn because the client will not let a user move if it is not their turn.

Once a user completes a box the first letter of their name is placed in the box. The user does not get another go if they complete a box, future versions could incorporate this feature.

The following pages describe how to connect to the Boxes client and initiate and play a game.

## Operating the Client

### Connect to a known Boxes Server

To play boxes you must connect the application to a boxes server session. When the game starts you will see a login dialog. This asks for the name of a server, the port the server is using and your name.

Enter this information into the dialog and click the OK button. Alternatively choose quit if you do not want to connect.

Note if running as an application the client will save the information you enter. It will be available next time to login.

EG : The Login Dialog is shown below

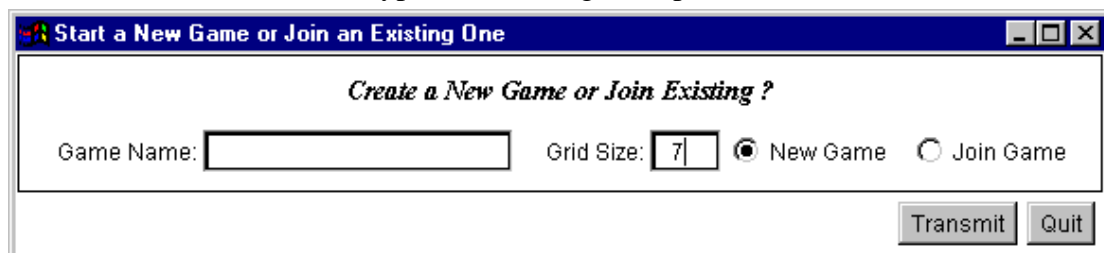
A screenshot of a Windows-style dialog box titled "Login Dialog". The dialog has a blue title bar with standard minimize, maximize, and close buttons. The main content area is titled "Connection Information" in italics. It contains three text input fields: "Name:" with the text "Wilson", "Host:" with the text "HAPPY", and "Port:" with the text "3606". At the bottom right of the dialog are two buttons: "Connect" and "Quit".

### Decide to Start a New Game

You will have two choices. You can JOIN an existing game or start a new one. If you want to start a new game choose the NEW radio button and enter in the size of the grid panel. (ie. the number of squares. NB the grid is a square so only one value is required).

You do not need to enter a game name as this is only for use when joining an existing game.

EG : The Connection type with NEW game options chosen

A screenshot of a Windows-style dialog box titled "Start a New Game or Join an Existing One". The dialog has a blue title bar with standard minimize, maximize, and close buttons. The main content area is titled "Create a New Game or Join Existing ?" in italics. It contains a "Game Name:" text input field, a "Grid Size:" text input field with the number "7", and two radio buttons: "New Game" (which is selected) and "Join Game". At the bottom right of the dialog are two buttons: "Transmit" and "Quit".

The above dialog requests a new game with a 3 X 3 game grid.

### Decide to Join an Existing Game

If games are running then they will be displayed in the text area in the middle of the screen. The top text area is for “chatting”. The word “GAMES :” will be followed by a list of games.

EG : The following output shows that there is one game running. The code 740 = WAIT\_LIST, the code 710 = NEW\_OR\_JOIN. These are the server talking to your client.

Server : 740

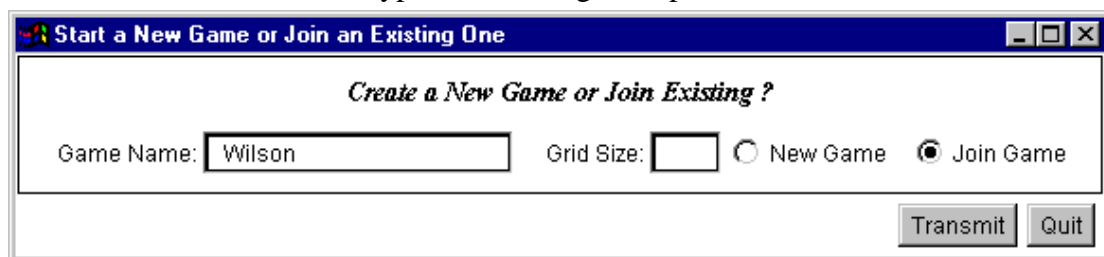
GAMES :

Wilson

Server : 710

Once you have decided what game to join. Fill out the dialog box. An example for this is shown below:

EG : The Connection type with JOIN game options chosen



The above dialog requests to join the game “Wilson”

## Play the Game

The game screen is now displayed. When it is your go you are prompted both in the message display area as well as at the top of the Window. Make a move by clicking the line between two dots. If it is not your go or you do not click in the correct location you will be chastised.

## The Client Window

The Window has 4 elements: These are:

1. The Chat Box : Type messages here and press the send button to broadcast the message to ALL OTHER PLAYERS. Message other users send to you are also displayed here.
2. The Communication and message Box. Shows what messages are being passed to and from the server. Also displays certain prompts such as “It is not your turn !!!!”.
3. The Boxes Grid. Click between the dots to connect them. Completed boxes are drawn with the initials of the user who completed the box displayed in the middle.
4. The exit button. Choose this to QUIT from a game.

EG. The following describes the elements of the client window..

