GX and Windows Remote Desktop (Terminal Server)

1. Introduction

This document describes some issues with GX and the Windows Remote Desktop feature. In particular, it explains why a "classical" Remote Desktop configuration is **not** the most efficient way to drive the specialised Global Application Explorer interface.

2. Overview

Unlike all other Windows applications, Global 3000 and other applications developed using the Global Development system run under the Global System Manager (GSM). GSM provides a fully protected multi-user, multi-tasking run-time environment providing many powerful features. In the context of the Windows Remote Desktop environment, the most obvious feature provided by GSM is a fully multi-user, multi-tasking kernel that provides Application Server functionality. The Application Server "exposes" any number of TCP/IP sockets and makes them available for subsequent connection by Global thin-clients, normally running on remote PC's. The internal GSM name for this unique interface is the "NETWORK console" interface. The only networking requirement for the GSM NETWORK console controller is for WinSock V2.0, or later, to be installed on the server computer that is running the Global Application Server (GLOBAL.EXE). **No other Windows software is required.**

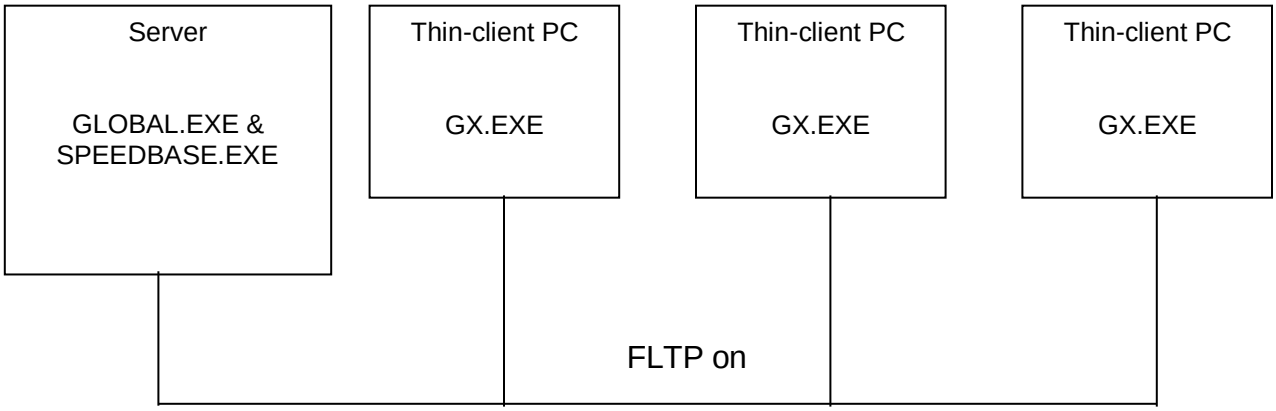
The preferred thin-client for Global applications is the Global Application Explorer (GX.EXE). The interface between the Global Application Server and the Global Application Explorer utilizes the unique Field Level Transaction Protocol (FLTP™) communication protocol. This protocol has been designed to deliver the maximum possible performance from the smallest communications channels in existence. Its major design goal was to permit fully featured GUI Applications to be deployable using G2 Wireless standards, which operate at a nominal 9.6Kb.

FLTP is extraordinarily efficient. It uses less bandwidth than any competitive protocol while delivering a full Graphical User Interface (GUI). It is so powerful that half a dozen or more users can simultaneously share a single dial-up modem. Even the slowest fixed-line modem connection can now deliver performance otherwise seen only in LAN configurations.

FLTP's extreme efficiency results from it's SetCode™ Instruction Set. The instruction set contains only 18 operations, most of which send no more that 2 bytes of data to the client. As a result, FLTP uses less than a quarter of the bandwidth needed to run equivalent serial screen applications. Surprisingly, FLTP manages this frugality without resorting to compression techniques. It simply avoids sending any information until absolutely necessary, and then only re-transmits altered data. Consequently, FLTP needs only minimal server processing time to assemble and transmit information.

3. GX and "bare" Windows Networks

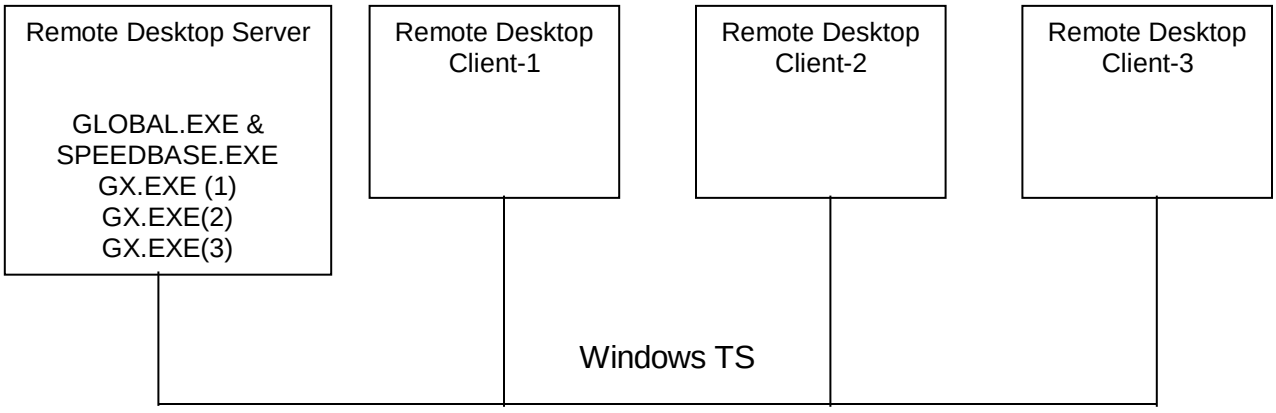
A typical "thin client" configuration on a LAN or WAN consists of the Global Application Server (GLOBAL.EXE) running on a central server with a Global Application Explorer (GX.EXE) running on each remote PC. This configuration is represented in the following diagram:



In this configuration no software, other than Winsock, GX.EXE and the NETWORK controller within GLOBAL.EXE is required to connect the thin-client PC's to the server.

4. GX and a "classical" Remote Desktop

The standard way to run applications under Windows Remote Desktop can be illustrated thus:



However, because of the specialised nature of the interface between GLOBAL.EXE and GX.EXE(s) the Remote Desktop merely acts as another software layer above the "network aware" FLTP interface and although the above configuration will function correctly it is **not** the most efficient way of driving GX thin clients. Consider what is involved when sending a window to a remote PC:

- GLOBAL.EXE sends a "window command" to GX.EXE via the FLTP protocol using an intra-computer TCP/IP connection on the server;

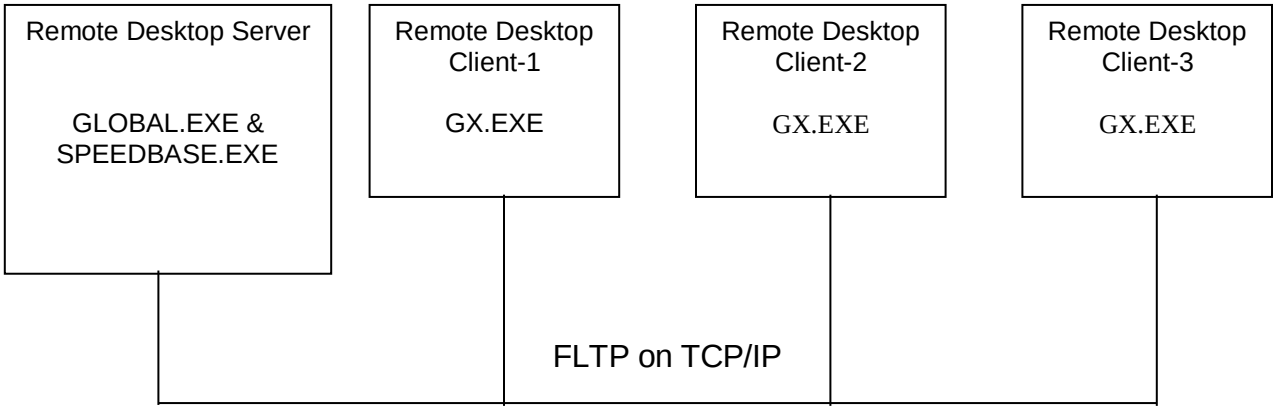
- The Remote Desktop functionality updates the Remote Desktop client with the modified window image.

Contrast this with the simpler mechanism that achieves the same effect in the configuration shown in section 3:

- GLOBAL.EXE sends a "window command" to GX.EXE via the FLTP protocol using an inter-computer TCP/IP connection between the server and the thin-client PC;

5. Recommended GX and Remote Desktop Configuration

The recommended way to run Global applications that utilise GX under Windows Remote Desktop can be illustrated thus:



Note the similarity between this configuration and the non-Remote Desktop configuration shown in section 3.

6. GX and Remote Desktop Bandwidth Requirements

The bandwidth requirement of GX, for a typical Global application, is typically 10% that of Windows Remote Desktop. This difference can be attributed to the finely tuned FLTP protocol, which is highly optimised for the connection between the Global Application Server (GLOBAL.EXE) and the Global Application Explorer (GX.EXE).