Finger 3.1: a Windows Sockets Finger Client

by Lee Murach Internet: lee@nrc.com, CompuServe: 71161,651 Tel: (805) 484-2128

Overview

Finger 3.1 is a Windows Sockets finger client. You may use Finger to query for users on a remote host. The Finger 3.1 distribution builds two finger clients: Finger and MFinger. These clients have the same user interface, and differ only in their network interface; Finger uses Berkeley style synchronous blocking calls, whereas MFinger uses the asynchronous WS extensions.

Contents

makefile   The make file for building the distribution.
finger.c   M/Finger's user interface. This module processes all user input, and displays query results and errors.
dsplist.c   The 'display list' module that enables the network module to hand off the finger query results in a form that is meaningful to the user interface module.
netwrkm.c   Mfinger's network module. It isolates the network interface from the rest of the program, and uses asynchronous WS calls to query the remote host.
netwrkb.c   This is Finger's network module and has the same external call interface as netwrkm.c, but uses Berkeley style synchronous (blocking) WS calls.
*.ico   This is "dirty Bert," the finger icon, and is a matter of hysterical convention.
other.ico   Can be used in place of finger/mfinger.ico, if you prefer a more conventional icon.
*.def   The .def file describes executable output to linker.
finger.dlg   Contains definitions of dialog boxes.
finger.rc   Specifies resources.
readme.*   You're reading it now.
m/finger.exe   The executables.

Requirements

To run Finger, you'll need access to a remote host which runs a finger server.

Since finger dynamically links to the winsock DLL and builds with the winsock.h, .def, and .lib files, you'll need a TCP/IP implementation that provides a Windows Sockets interface.

The 3.1 distribution was built with the Microsoft C/C++ 7.0 compiler and the Windows[1] 3.1 SDK.

Release Notes

The netwrkm.c module now calls WSAAsyncSelect() with FD_READ | FD_CLOSE flags in order to detect the end-of-stream. Actually, the module considers the zero recv() return to be the definitive indication of end-of-stream.

Netwrkm.c now checks for the WSAEWOULDBLOCK error that connect() usually returns. This isn't actually an error, but merely indicates that the connection request is still pending completion.

The host dialog now (optionally) queries for a user login id, along with the host name/address.

The 3.0 Mfinger would crash if given a host IP address. This has been fixed.

The release now builds for Windows Sockets rev 1.1, by default.

[1]Windows is a trademark of Microsoft Corporation