Chapter 2
Private and Global Resource Scope

Abstract

This chapter is a preliminary look at how resource scope issues are handled with Client32. Scope algorithms may change in later releases of Client32. Private scope is not yet supported.

Introduction

The *scope* of a resource defines the realm in which that resource is visible. The resource types that Client32 monitors separately for scope are connections, drive root mappings, drive relative mappings (e.g., *cd*'s), and printer redirections.

A realm is generally a process group, for example a DOSBOX (under Windows) or a Task Switcher DOSBOX (under DOS). Windows programs are processes within the system process group.

There are three kinds of scope: global, private, and inherited.

Definitions

Global
Global scope means that a resource is *shared*, allowing access from any process group on the system. Any changes made to a global resource are visible from any other scope.

For example, if drive mappings were of global scope and a new drive mapping was made in one DOSBOX, then that drive would also be visible in any other DOSBOXes on the system. If the drive were later deleted, it would disappear from the other DOSBOXes as well.

Inherited
Inherited scope means that when a resource is created, it is initialized to match what the system process group represents for that resource, after which it is treated like a private resource, and any changes do not affect any other resources.

Private
Private scope means that when a resource is created, it stands alone and does not inherit attributes from any other resource. For the case of a whole process group being launched as private, no resources are initialized other than the "F:\LOGIN" drive. Changes made to private resource are visible only in the process group that created the resource.

Goals for a Scope Algorithm

The Client32 Requesters needs a scheme for setting the default scope values and managing scope of different system resources. Some factors in determining this scheme:

1. Support a majority of user desires for resource scopes. The Client32 resource scopes, listed by resource type, are:

Resource Type	Scope
Connections	Global to system
Drive roots	Inherited, global within process group
Drive paths	Inherited, global within process group
Printers	Inherited, global within process group

2. Give users the option to specify that *all* resource scopes be either global or private. This is an advanced configuration option that changes the default, making all connections, drive roots, drive paths, and printers either global or private, system-wide. The client must be reloaded for the change to take effect.

3. Support launching a process group or a process privately such that any associated resources are kept independent. This could be initiated by the user by using the GUI provider to drag and drop an icon on an icon that represents private-ness. A script could be associated with that icon that indicates resources that are needed and their respective redirections to the network; otherwise the process would only receive the standard "F:\LOGIN" drive resource.

4. Consider backward-compatibility issues. The following grid shows default scopes for Netx, VLMs, and Client32.

Resource	Global	Inherited	Private
Connections	CN	V	
Drive roots	V	CN	
Drive paths		CNV	
Printers	NV	C	

   Key: C = Client32, N = NETX.COM, V = VLMs

Identifying Resource Scope

In order for the Requester to keep track of which resources are associated with which scope, all APIs involved with a tracked resource must use the calling process and process group as parameters. These values are then converted to a scope (usually from within the "HandleToXXX" function) using the **SCOPEGetResource**. This scope is then assigned to the resource if it is being created, or is used to check for valid access to that resource.