



triCerat Simplify Printing v3

Sadly for support people everywhere, the paperless office just never materialized. We still need paper output, and we need it reliably. Printing is one of the consistent headaches for anyone supporting a network, and even more of a headache for anyone using Terminal Services. triCerat, Inc, has created Simplify Printing v3, a tool based on patent-pending ScrewDrivers technology that has helped thousands of customers solve their printing problems on platforms ranging from Windows NT, Terminal Server Edition (TSE) to present-day Terminal Services with MetaFrame XP.

To make it more clear how triCerat can simplify printing--and why this is necessary--let's take a look at how printing works in Microsoft server operating systems from NT 4 to the present day.

How NT Printing Works

The process of printing is a bit more complex than it may appear from the outside. The NT printing model creates and sends print jobs in several stages, first rendering application data for graphical output, then getting the data to a printer, and finally helping a printer manage multiple print jobs. The pieces of NT-based printing responsible for handling all this are the Graphics Device Interface (GDI), the printer driver, and the print spooler.

Let's first consider how NT-based printing works outside a terminal session. The *Graphics Device Interface* (GDI) is the portion of Win2K the operating system that begins the process of producing visual output, whether that output is to the screen or to the printer. Without the GDI, WYSIWYG output would be impossible. To produce screen output, the GDI works with the video driver; to produce printed output, the GDI works with the printer driver. The GDI operates both in user mode and in kernel mode, at different stages of the print process.

The operating system needs some way to communicate with the print hardware. One way to do this would be to build the capability into the OS itself, but doing that would require changing the OS every time you changed printers. Therefore, NT printing uses *printer drivers* to act as the go-between between the print hardware and the operating system. The print drivers that you can use depend on the OS architecture, so not all print drivers will work on all platforms. Although any Win32-based print client can print to a NT-based print server without having previously installed a local printer driver—they'll just download it from the print server and automatically install it—you'll have to make sure the drivers are available for the clients connecting to that print server.

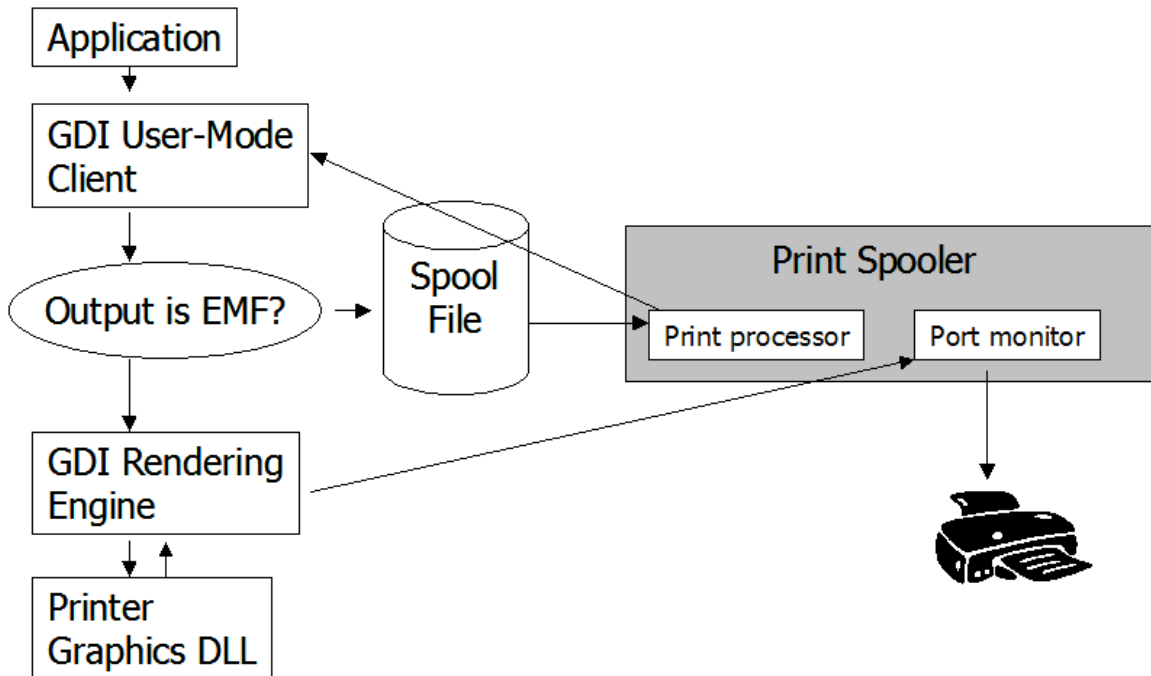
Print drivers have two pieces: a graphics DLL and a printer interface DLL. These two pieces support the jobs of the driver:

- The graphics DLL helps GDI render the print job by provide printer-specific drawing capabilities that GDI does not support. The graphics DLL also sends the rendered image to the print spooler.
- The printer interface DLL exposes the configuration information exposed through a printer's Properties sheet: number of copies, image resolution, color support, paper orientation, and so forth.

The *print spooler* receives, processes, schedules, and distributes print jobs to the appropriate print device. The print spooler is actually several pieces: the print router, the local print provider, the remote print provider, the print processors and the port monitor.

When a client computer prints from a NT-based print server, the client's *print router* calls the *server's print router*, which passes the print request to the appropriate *print provider*. the local print provider if it's a local job, and either the Windows or NetWare print provider if sent over the network. To find the right print provider, the print router polls the Windows print provider, which finds the printer name and sends a remote procedure call to the print router on that printer's print server. That local print provider then writes the contents of the print job to a spool file and tracks administration information for that print job. The *print processor* works with the printer driver to "de-spool" spool files during playback, making any necessary changes to the spool file.

The *port monitor* is the final link in the chain getting the print job from the client application to the print device. This piece sets up communication with the print device and transmits the print job to the printer port, whether that printer port is a parallel or USB connection on the local computer, or a network share representing a printer on a print server.



To sum up, the process normally works like this:

1. The application calls the user-mode component of GDI.
2. The GDI user mode piece creates the and makes it into an EMF record (since EMF is the default data type).
3. The graphics DLL in the printer driver sends that record to the print processor in the spool file.
4. The print processor sends the now-non EMF record back to the GDI kernel-mode component for formatting.
5. With the help of the printer graphics DLL, the GDI kernel-mode component formats the print job appropriately for the printer and converts it to RAW data.
6. When the print job is formatted, the print driver sends it to the port monitor in the spooler.
7. The port monitor sends the print job to the port driver managing the port the printer is connected to, and the job prints.

Printing works in much the same way from a terminal session. Both ICA and RDP 5.x and later can map client printers--both local and network--to terminal sessions. These mapped printers use virtual ports named for the session they're associated with (e.g. TS001). When you print to a mapped printer, the print job is created on the terminal server using the terminal server's locally installed drivers, is spooled to the virtual port for that terminal session, and then passed via virtual channels to the client machine hosting the terminal session. From there, the print job proceeds as though it had originated on the client computer instead of the terminal server.

Copyright 1997-2003 triCerat inc. ALL RIGHTS RESERVED

SCREWDRIERS IS PATENT PENDING ALL RIGHTS RESERVED

Problems of Printing and Terminal Services

The NT printing model can be a problem when you print from a terminal session for three main reasons:

- Driver management
- Driver name mapping
- Printer setting management

Driver Management. The terminal server must have drivers for all client printers because driver processing takes place on both the terminal server and the client computer. This in itself isn't so much of a problem, except that just about all of the print process takes place in the system kernel--even the GDI is partially in the kernel. All pieces of the kernel share memory space, which means that a badly written printer driver can stomp on other core pieces of the operating system and cause it to crash. A crashed terminal server tends to annoy those trying to use it. This means that you *must* test printer drivers before letting them onto the terminal server. Since Win32 operating systems are smart enough to install print drivers if they're not already installed, this means that you have to move faster than the terminal server's operating system to make sure that you install a tested driver.

Name Mapping. The print model--specifically, the part where the print processor de-spools the spool file and makes any necessary changes based on the driver information--expects the client printer drivers and the terminal server printer drivers to have the same name. The way that the print model works assumes that all the print job processing happens on a single machine, not on one terminal server and one client computer running the terminal server. Print driver names do not always match even in Win32 operating systems--the driver for the Windows 98SE client, for example, may not have the same name as the driver on the Win2K-based terminal server. If the driver names don't match, then the print job won't make it to the printer. To correct mismatches, you'll have to edit NTPRINT.INF, a text file on each terminal server, to tell it how to substitute driver names, and edit the Registry to point the terminal server to the section in this INF file explaining the substitutions.

Printer Setting Management. The problem of making sure that the mapped printer information is up to date on the server has two pieces. First, if you change computers while in a session, the "autocreated" printer inventory may not reflect the change in what's available. For example, Terminal Services is a popular tool among engineers in some organizations. The engineers log onto the terminal server, do some work, and then at the end of the day disconnect their sessions--not terminate, disconnect, so that all their work remains in memory. They go home, talk to the spouse, scratch the kids and ask the dog about school, and then reconnect to the terminal session to work some more until they need to print. RDP re-inventories the printer mappings when you reconnect a session. ICA does not. Engineers using ICA cannot print to their home printers when working at home unless they terminate their session and re-start it.

The second issue has to do with how autocreated printer settings are stored for use in a terminal session. Printer configuration of mapped client printers is done separately for local use and terminal session use. That is, when you start a Terminal Services session with a new local printer connected to your client computer and printer redirection enabled, this creates a redirected print queue in your session. The print queue uses a set of default printer settings determined by Terminal Services, not the local printer settings defined offline. Changes that the user makes to the printer settings from within the terminal session are cached on the client computer for subsequent reconnections, but this cache will reflect only the changes made within the session. (You also have to be careful about how you make those changes to ensure that they're cached, but that's another matter.) It is therefore entirely possible for a printer to have two sets of settings: one used within terminal sessions, and one used when printing from locally run applications. For users expecting that the changes they made on the local printer to be reflected in their terminal session, this could be a problem.

MetaFrame Printing Functionality

These are known problems, and MetaFrame XP has done some work to address two of them. First, you can use tools in MetaFrame XP to send tested drivers to other MetaFrame XP servers, so that once you know which drivers are safe to use you can easily distribute them. You can also edit NTPRINT.INF from the GUI, instead of having to manually edit the text file, and send the printer name mapping information to all MetaFrame XP printers instead of having to edit it on each terminal server.

Another solution to the driver problem is to use the Universal Print Driver introduced with Feature Release 1 for MetaFrame XP. The UPD isn't a full-featured printer driver (in its first incarnation it supported black and white only and a maximum resolution of 300dpi; the version in FR3 supports color and up to 600dpi), but can at least get the print job to the clients without having to have a local copy of the client's printer drivers installed on the terminal server. *It should be noted that the UPD is not fully compatible with a large number of printers; you'll need to test every printer you plan to support before relying on this method.* You can choose either to use the UPD for all print jobs or restrict its use to those times when an appropriate driver isn't already installed on the terminal server.

However, driver distribution tools and the UPD still leave you with the same dilemma that you have with unmanaged drivers. You can elect to install, test, and replicate printer drivers and name mappings to all MetaFrame servers, or you can have a reduced-functionality printer driver that may or may not work with your current or future printers. *All of this needs to be repeated for any printer model or driver changes.* Finally, both options are available only to those using MetaFrame XP. Those using MetaFrame 1.8, or native Terminal Services, don't have these tools.

How Simplify Printing v3 Fixes Printing Problems

triCerat's solution to printer driver problems is to avoid the whole situation entirely and not use drivers on the terminal server at all. ScrewDrivers technology built into the Simplify Printing v3, creates *virtual* printer drivers on the server using the driver information on the client.

It works like this: Simplify Printing v3 has two components: a server component, resident on the terminal server, and a client component, resident on each client launching terminal sessions. (The install the client component for the client component is very simple even for task-based workers, and you can download free deployment tools from the triCerat Web site if you need help with deployments.) These two components communicate through virtual channels, working with either ICA or RDP.

When a user launches a terminal session or reconnects to an existing session, the server component initializes and queries the client component for printer information. The client component then inventories the printers, collects each printer's configuration information, and sends this information to the terminal server. The server component also creates a virtual port for each session, identified by the display protocol used for the connection and the session ID (e.g., SDI006 to connect to Session 6 via ICA or SDR003 to connect to Session 3 via RDP). When users print to a mapped printer from their terminal session, Simplify Printing v3 formats the job using the configuration information drawn from the client and outputs it into triCerat's proprietary triMeta[®] format. The print router then directs the job to the virtual printer port for the session to go to the client's computer for spooling and processing, just as in the normal NT print model for terminal services. It actually works very like the standard model of printing from a terminal session, except that the driver is the virtual printer driver instead of one installed on the server.

Note: Because Simplify Printing v3 printing technology uses virtual channels to process print jobs, users will need permission to use virtual channels.

As you can see, Simplify Printing v3 reduces the complexity of the print process considerably. As the administrator, you no longer have to worry about installing or updating printer drivers on the terminal servers...*ever again*. If users change their printer drivers or printer settings on their local computer, then their virtual drivers automatically to reflect the change. Users don't have to take any special steps to print to their printers--they just print as they normally would when running an application locally, which reduces training time and should reduce the volume of calls to a support center.

Simplify Printing v3 vs. so-called “Universal Print Drivers”

Simplify Printing v3 may appear to be yet another so-called universal print driver, but it's not.

You'll recall that drivers are the software that allows an operating system and a piece of hardware to talk to each other. The virtual printer drivers used in Simplify Printing v3 are actually recreated on the server each time that a session begins or is reconnected. When a user connects to the terminal server, the Simplify Printing v3 server component queries the client component to get the current printer settings. All settings configured from the Printers tool in the control panel are reflected in this query. Because all printer functionality depends on the client-side driver, Simplify Printing v3 does not reduce that functionality at all.

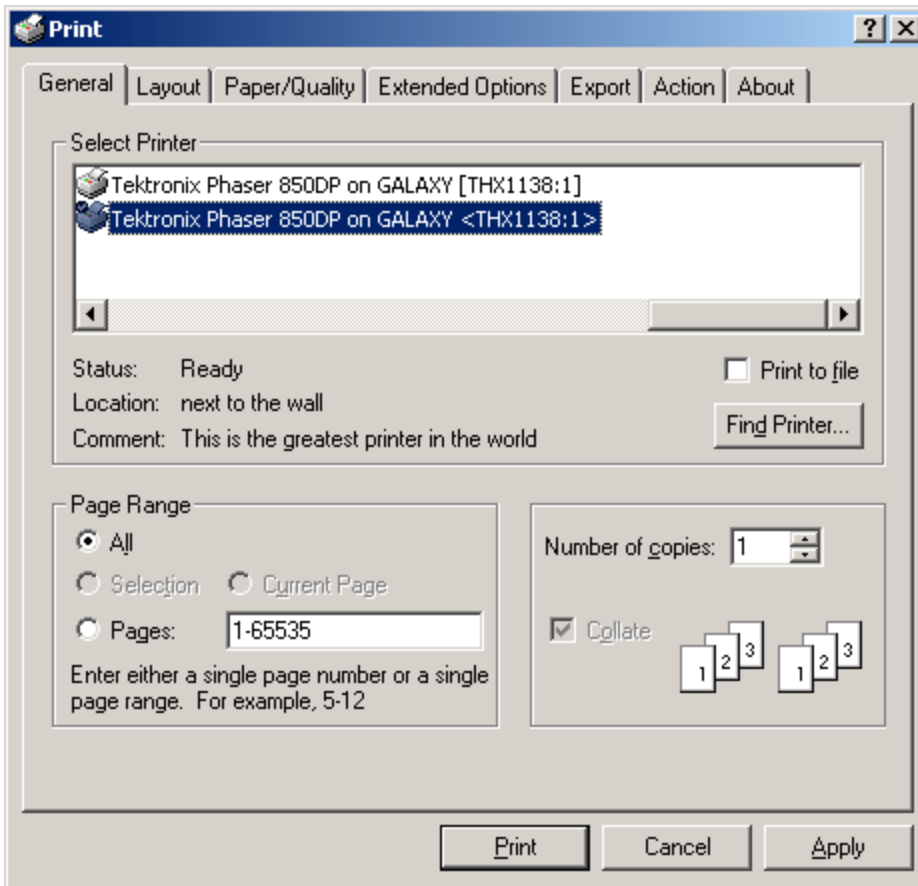
A “universal printer driver”, in contrast, is a generic printer driver that provides least common denominator features. A UPD can provide some output for many printers, but because it needs to be generic, it is not possible to provide the features users expect.

The end result of using Simplify Printing v3 is that the mapped printers in a client's terminal session are always up to date, reflect the current printer configuration, and support every feature controlled through the Win32 operating system. And you can accomplish this without ever installing any drivers on the terminal servers.

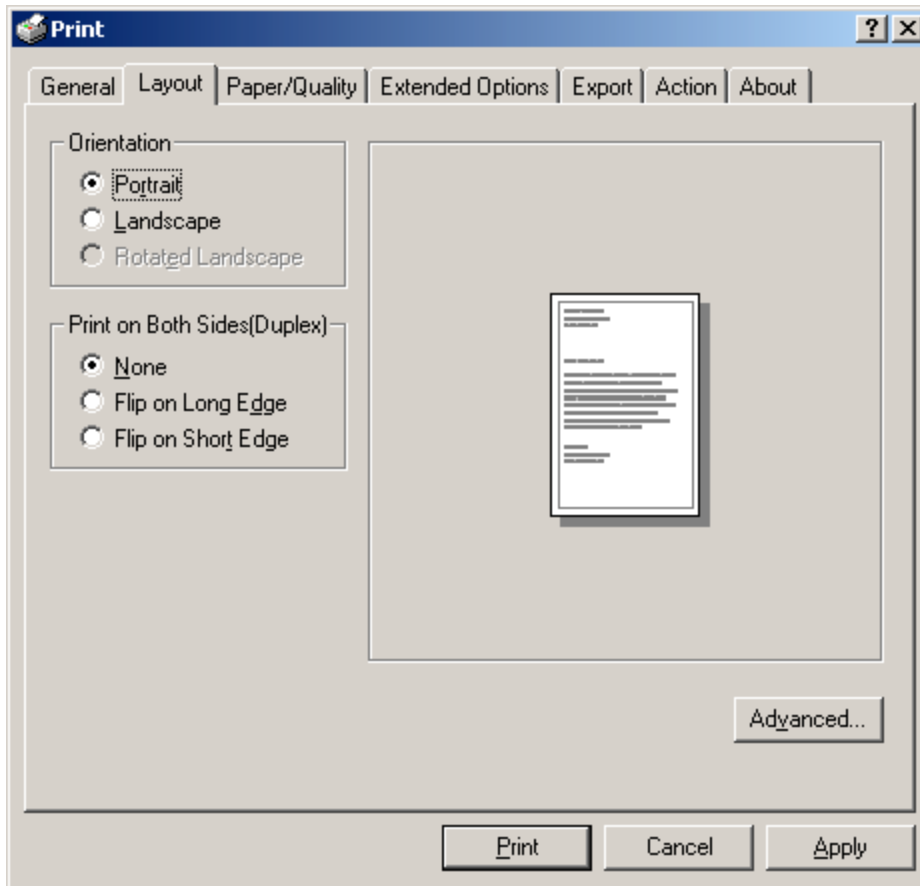
Note: Simplify Printing v3 allows you to completely disable printer mapping for your display protocol.

Printing With Simplify Printing v3

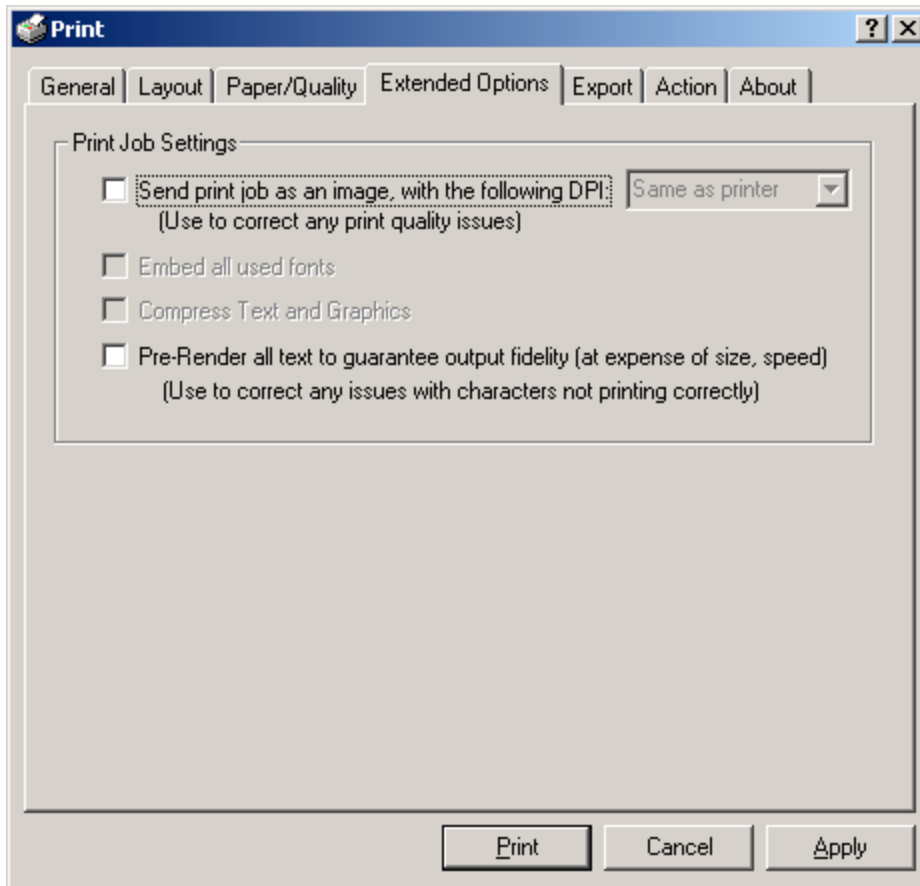
From the client perspective, using Simplify Printing v3 is completely transparent and seamless. When you start to print from within a terminal session, you'll see a dialog box like the one below.



For most print jobs, you won't need to go beyond the General tab. From here, you'll choose the printer to use (or whether to print to a file for later use—see the Export tab for more detail over file exports), the number of copies to make, and the range of data to print.

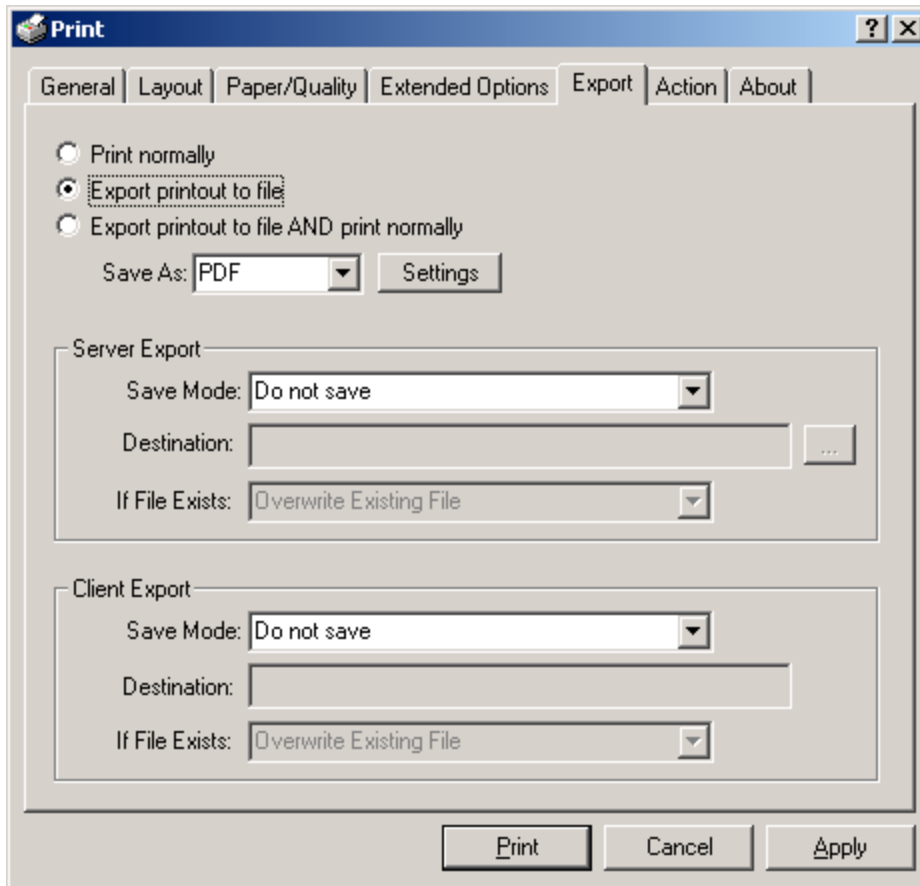


The Layout tab controls paper orientation (portrait or landscape) and, for double-sided printers whether the job should use duplex printing for the current print job. Paper/Quality controls the paper tray to use for the current job, and allows you to specify the paper type and color depth.



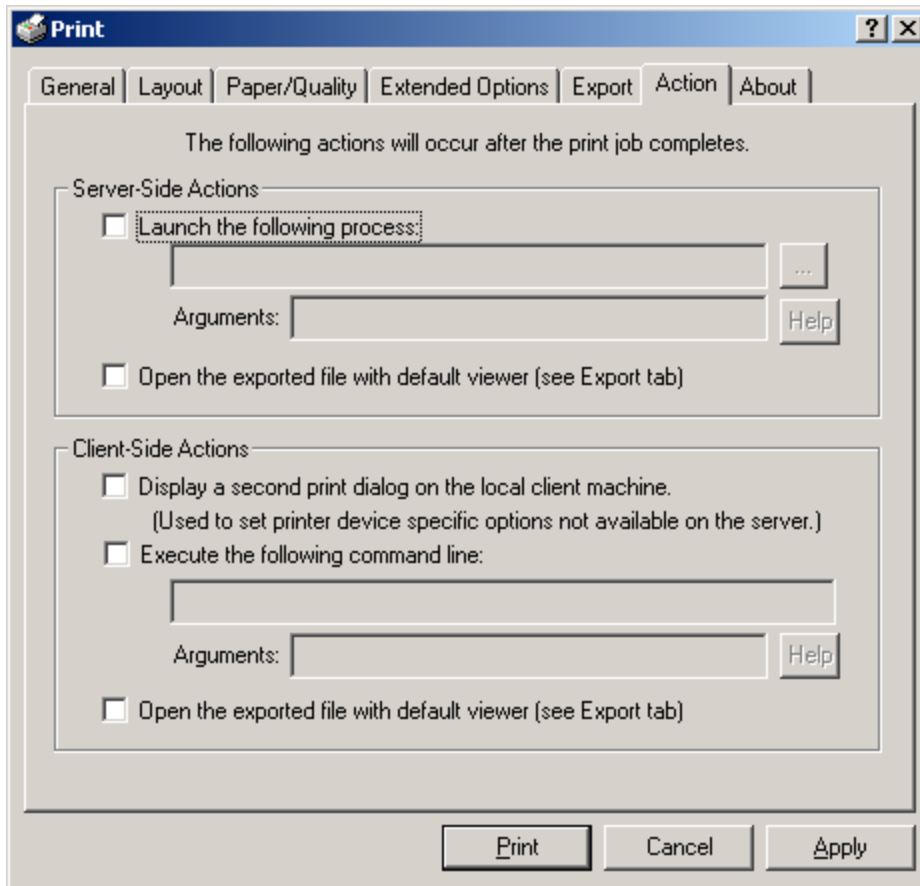
Extended Options can be helpful for troubleshooting and are rarely used. If a user is having trouble getting a print job to print correctly, you can choose to print it as an image or pre-render it. Printing the job as an image turns the entire job into a bitmap. Pre-rendering all text renders each individual letter as a bitmap. (Incidentally, the former option is equivalent in some ways to saving the file as a bitmap and the latter to saving it as a PDF, then printing the saved file.)

Note: Print as an image or pre-render text only when a document is not printing as expected and you're researching the cause. For example, pre-rendering text can help avoid output problems caused by insufficient printer memory, that result in images not printing at all or the printer using its default font instead of the font you selected. Both techniques force the printer to print exactly what it's given, instead of interpreting instructions. However, doing so may affect the size of the print job significantly. Only choose to pre-render text if the job is not printing correctly, and consider reducing the dpi and compressing text when printing as an image.



If you'd like to export a print job for archive and subsequent reprints, turn to the Export tab. Creating either PDF or bitmap files may take longer than only printing the document. To both save a print job as a file for later printing and create a hard copy now, choose to both print and save the job. You can export print jobs to a file either on the terminal server or, through virtual channels, on the client.

The PDF writer in ScrewDrivers will create valid PDF documents, but it does not attempt to replicate the features available in Adobe's Acrobat products. If you need to provide PDF features such as annotation or text search, consider obtaining the Adobe tools to run alongside Simplify Printing v3.

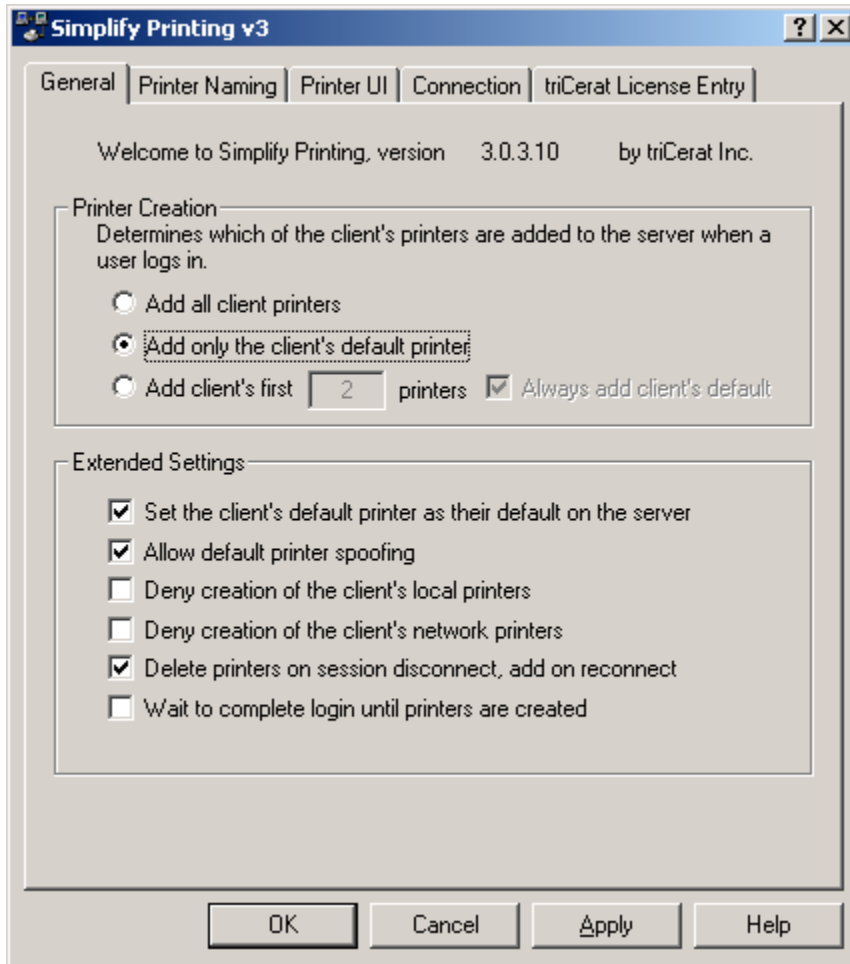


Edit the settings on the Action tab to launch applications after printing—for example, to open a print job saved as a file in a bitmap viewer. *This feature allows administrators to easily incorporate Terminal Server originated print jobs into remote office local area network workflows.*

The client option to display a second print dialog box on the local machine (outside the terminal session) is for setting printer settings not already configured on the terminal server.

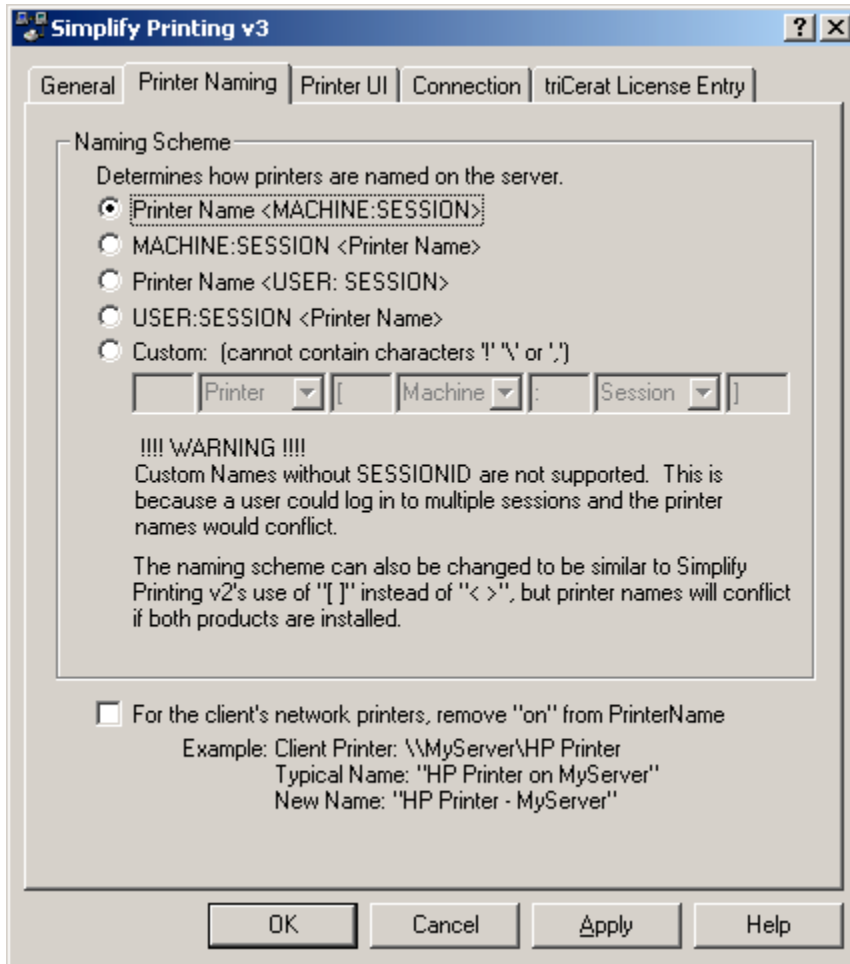
Managing Server Settings

The server-side configuration options offer a fine level of control over the printing process. Where applicable, the server-side settings override the client-side settings, allowing administrators to maintain desired standards.

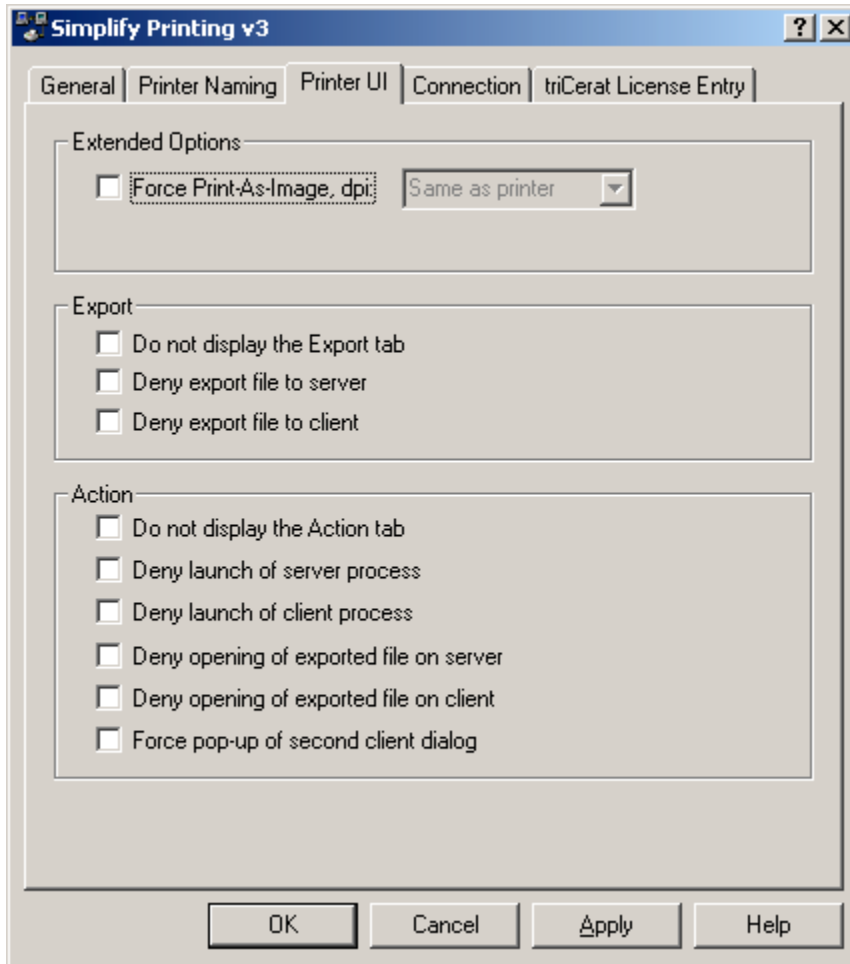


The General tab (shown here with the default settings) controls printer building for each client. As discussed earlier, clients can control the order in which their printers are built. The General tab further controls that order, allowing you to specify that only the default printer should be created, or only the first two on the Preferred or Available list. Reducing the list of printers makes it easier to select the appropriate device.

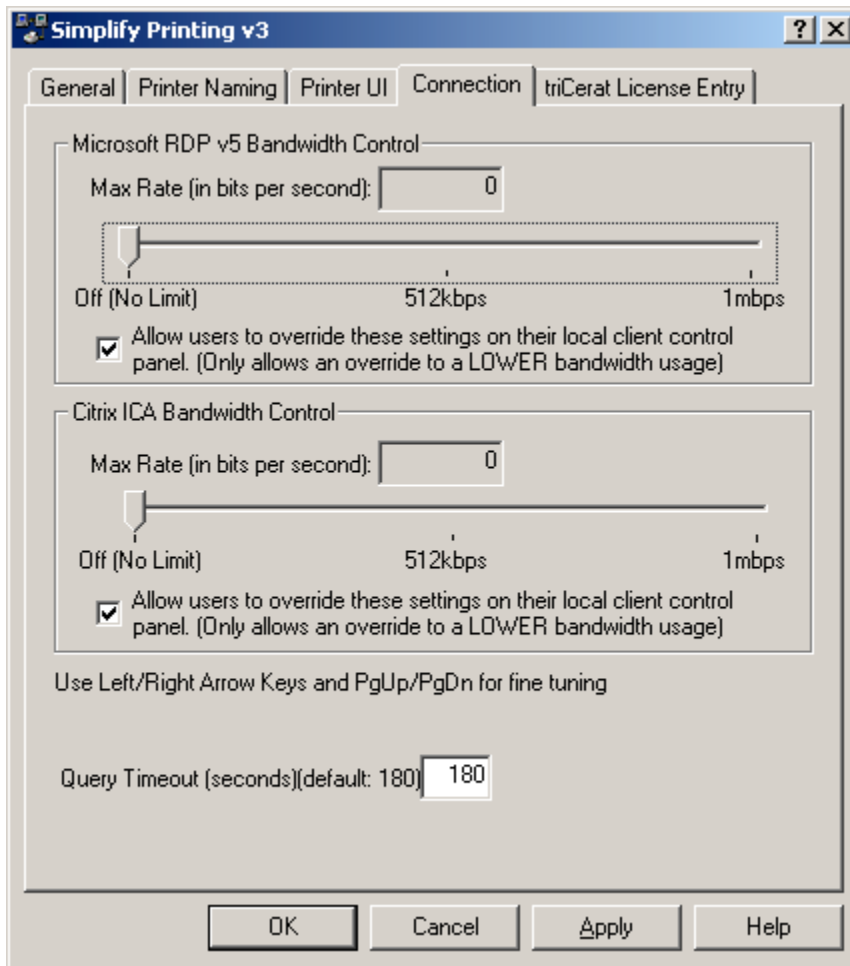
The Extended settings below are fairly self-explanatory. Check the appropriate boxes to control which printers are added, how the default printer for the terminal session is identified, and what happens when users disconnect from a session and reconnect to it later. Older applications that are Terminal Services “unfriendly” may require printers to be specifically assigned before execution by the user. In that rare instance, you can force that published application to become available only after Simplify Printing v3 printing building is finished. (By default, printers are added to the session concurrent with other login events).



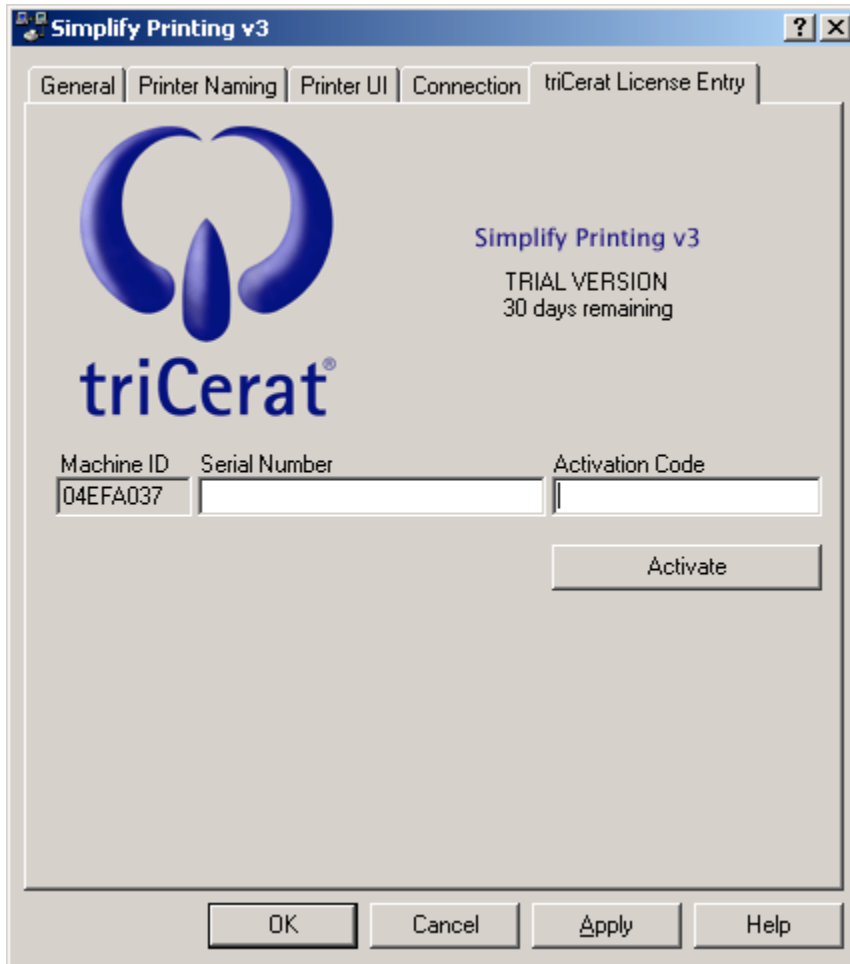
The Printer Naming tab controls how mapped printers are identified in the terminal session. Notice that the name must include the session ID, to make sure that all printers have unique names.



The settings on the Printer UI tab control what the user sees when printing from the terminal session, specifically whether the Export tab (controlling file exports) and Action tabs appear. You can either hide the tabs entirely or disable certain options on them. For example, to prevent users from exporting to a file saved on their client computer, you'd check the Deny Export File to Client box.



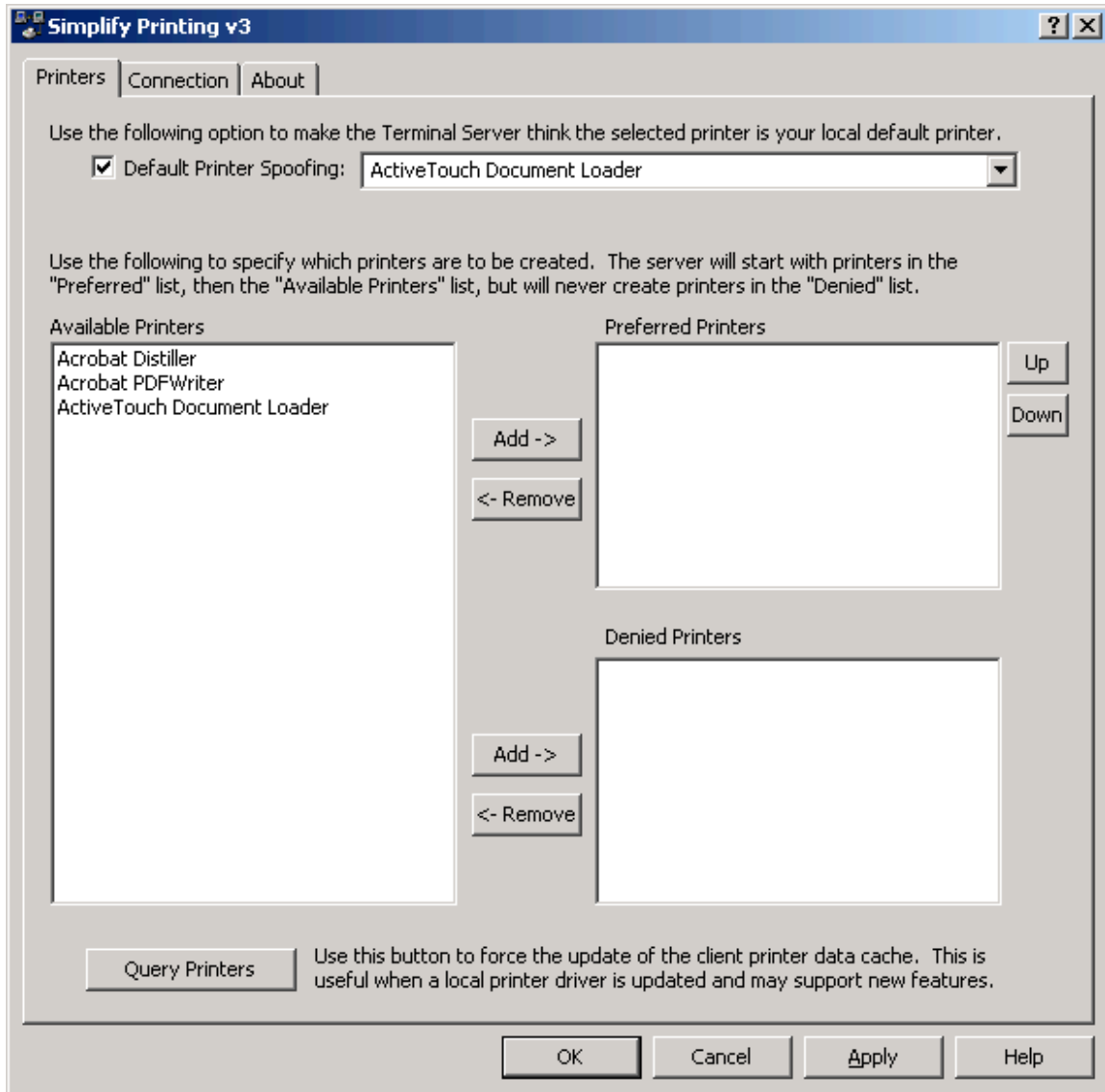
As discussed earlier, clients can limit the amount of bandwidth available to a print job for the protocol they're using. This tab is for setting your own server-wide limits. As you can see, these override the client-side settings. Even if you permit client settings to control, a client can never use more bandwidth than that permitted here, if you set up restrictions. Notice that you establish limitations for RDP and ICA separately.



The final tab on the server management tool is for entering the serial number and activation code you get from triCerat when purchasing the tool.

Managing Client Settings

All of these features are optional; you can simply not install the client side control panel if you do not want your user base interacting with the product. Clients can configure their printer settings with the Simplify Printing Client v3 tool found in the Control Panel after installation. This tool is pretty self-explanatory but a few settings require some explanation for less-experienced users.



The Printers tab is for printer management. The option for printer spoofing allows the user to specify an alternative default printer for terminal sessions. To spoof a printer, make sure that the box is selected and choose the appropriate printer from the drop-down list.

Client Printer Spoofing

You might choose to spoof a printer for several reasons. One option would be to configure printer settings separately for local use and use within a terminal session. As discussed earlier, one of the advantages of using Simplify Printing v3 instead of autogenerated printers is that changes made to the local printer are reflected in the printer that's opened within the session. However, if you actually want alternative settings you'll need to create a second printer for the same print device and make that is the default for terminal sessions. Creating a second printer to represent the same print device is fairly common—you'd do this on a print server to configure different sets of permissions for the same printer, or configure the printers differently depending on user groups. Same print device, different name.

Another possible reason to spoof the default printer might be location. It is entirely possible that you don't always want to print by default to the local printer. Perhaps the print job actually belongs in another office from the one you're sitting in. In that case, you'd choose the other printer and make that the default for terminal session printing.

The printers tab allows users to specify which printers should be built into the session and the order in which this should occur. By default, Simplify Printing v3 will inventory all printers and build drivers for each. Perhaps you have a large number of printers defined on your workstation. If you know that you'll never use a particular printer in a terminal session, add it to the Denied Printers list. And if you know that you absolutely want a printer available, then add it to the Preferred Printers list. All printers not absolutely denied will be inventoried and built into the session.

The Connection tab on the client management tool controls the maximum throughput allowed for print jobs. Normally, there's no limit—a job uses the bandwidth it needs. But a large print job (or one printing across a connection with a lot of competition for bandwidth) might benefit from bandwidth throttling.

Note: The server administration tools also have bandwidth controls. The client tools can optionally override the server controls, but only downward—they can never provide *more* bandwidth for print jobs than the server allows.

Summary

Terminal Services printing is extraordinarily complex to manage and can easily consume the majority of an administrator's time. Fortunately, triCerat has designed software that makes Terminal Services printing hassle free and virtually automatic.

To get a free 30-day trial of Simplify Printing v3, visit www.tricerat.com.