Conquering Rogue Application Behavior in a Terminal Server Environment

Using Application Shaping™ to ensure reliable, consistent performance and application response times for improved end-user satisfaction

A white paper
by

www.rtosoft.com

Published November, 2003 (Rev 1.0)

RTO Software, Inc. • 5400 Laurel Springs Parkway Suite 108 • Suwanee, GA 30024-6106 • USA. +1.678.455.5506

Table of Contents

1. Problem Summary

Citrix® and Microsoft® Terminal Services have gained widespread acceptance, allowing IT departments to provide end users with access to applications from a central set of servers called Terminal Servers. Terminal Servers enable conventional server systems to operate in multi-user mode and extend applications to end-users through thin-client interfaces. This approach allows organizations to lower their total cost of ownership, minimizes the complexity of the end-user desktop, and centralizes software application management.

The challenge to any IT department is to provide the users in the organization with prompt access to new and updated business applications with acceptable levels of performance. The Terminal Server model delivers on its promise for rapid deployment of applications to all end-users, but the nature of the terminal server environment poses unique challenges in ensuring predictable, reliable performance and application response time.

A specific challenge, addressed in this white paper, is that application components, or instances of applications, can, at times, exhibit behavior that causes all the other applications on the terminal server to pause or run slowly. This negative behavior is often due to an instance of an application consuming high levels of CPU time. There are two categories. The first is legitimate applications that periodically require intensive CPU utilization, such as large spreadsheet macros and recalculations that can take upwards of 5 minutes. Another is when an instance of an application misbehaves and literally runs away with the CPU. In this paper, we will call these collectively "rogue applications" – applications that periodically require high CPU or run away with the CPU.

Applications can "go rogue" on a single-user desktop PC (outside a terminal server environment), and only that individual user is impacted. However, in a terminal server environment this can be disastrous, impacting all of the users on that server. Consider a 40 person call center where time is literally money, unable to process transactions because the terminal server is "hung" due to a rogue application.

In a typical scenario when an application "goes rogue" on a terminal server:

- All the users on that terminal server experience sluggishness or non-responsiveness with their applications.
- Several annoying minutes pass while they wait for it to "clear up."
- One or more users will call the IT helpline to complain about performance.
- IT then opens a case and assigns a resource to troubleshoot the problem.
- The Citrix/Network admin identifies the specific terminal server(s) with a problem.
- The Citrix/Network administrator then troubleshoots the server to determine the problematic application or process thread.
- The bad application/thread is manually terminated, restoring CPU bandwidth back to the users on the system.

This is obviously a frustrating experience for the users of the applications on the terminal servers, as well as *customers* who may be impacted by the poor system response. In addition, IT staff must respond to correct the problem, taking them away from planned projects. It is clear that there are significant costs to the organization due to rogue applications.

2. An Innovative Solution

Fortunately, there are software solutions to the problem of rogue applications. Two different approaches are commonly used; one is typically called CPU "clamping" or CPU "throttling," where a service runs on the terminal server and literally intervenes in the way the operating system runs to suspend the execution of application threads to limit offending applications to a preset percentage of CPU.

Another approach, developed by RTO Software and included in TScale®, is called Application Shaping™. Application Shaping automates two accepted technologies to provide network/Citrix administrators with pinpoint control of the applications. This allows them to overcome problems associated with rogue applications and provide consistent and reliable application performance and responsiveness.

3. Why is Application Shaping Necessary and Desirable?

A Terminal Server environment is typically characterized by many users using many different applications, with this mix of different applications sharing two or four CPUs on the server. In this environment, if one instance of an application "goes rogue" or starts to consume more than its "fair" share of the CPU, it will have a very detrimental effect upon the performance that the server delivers to the remaining users and applications.

For example, if a typical load on a production terminal server is 40 users of Outlook, 20 users of Internet Explorer, and 30 users of a CRM application load balanced between two CPUs on the server, the distribution would be as depicted in the table below:

Application	Users on CPU #0	Users on CPU #1
Outlook Users	20	20
IE Users	10	10
CRM Users	15	15

As long as each instance of each application is "well behaved," a typical modern dual processor server would in general be able to deliver acceptable response times and therefore an acceptable experience to the end users of the system.

However, if one instance if IE goes rogue (in some instances, this can occur just by loading a pop-up that contains animation), then that instance of IE can consume virtually all of one of the two CPUs on the server. This leads to two very undesirable consequences:

1. All 20 Outlook users, the other 9 IE users, and all 15 CRM users who share the CPU with the rogue process experience bad performance.
2. As that CPU starts to "hang" the users of the application on it log off, and log back on again. These logon's stress the farm, and create load on the other servers, reducing the response time and impacting the user experience across the farm.

Application Shaping is an automated approach to dealing with the negative impacts of rogue applications. Application Shaping has two components:

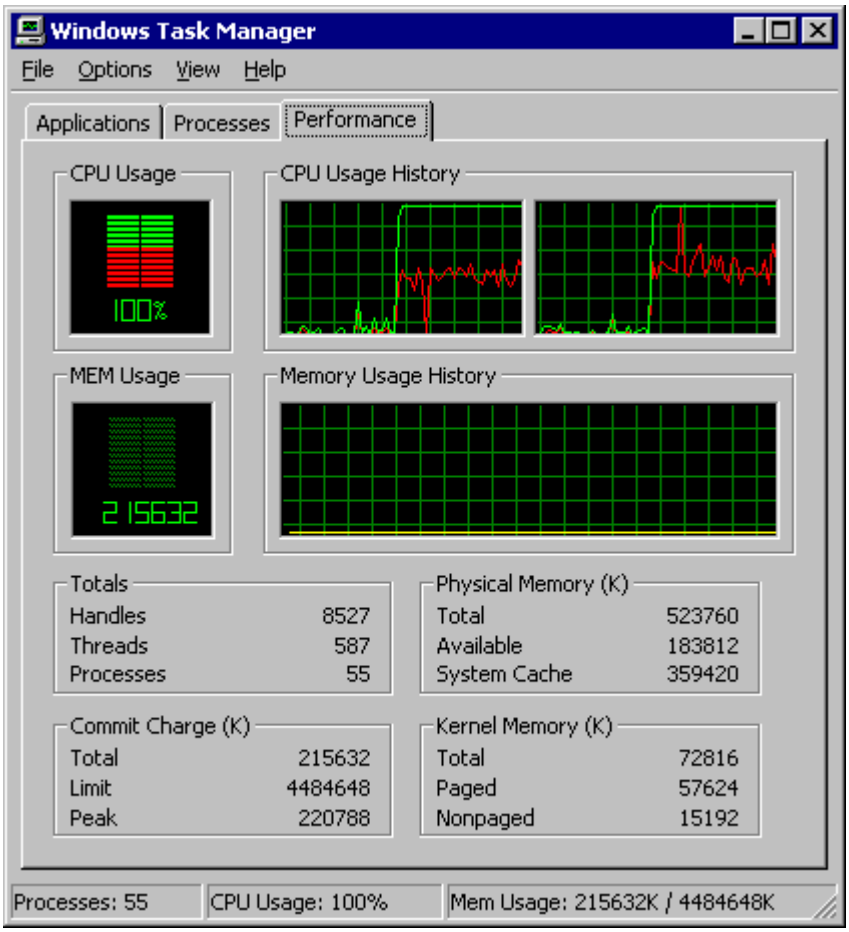
1. **Application Affinity™.** This function is used to cause all instances of an application to run on one of the CPUs on the server. If you have an application that frequently exhibits rogue behavior, this will ensure that if two instances go rogue at the same time, both CPUs on the server are not consumed, so that all of the users on that server are not inconvenienced.

2. **Application Priority™.** Administrators can define the circumstances in which a potential rogue application is only allowed to use CPU resources available after the CPU has serviced all other applications and users.

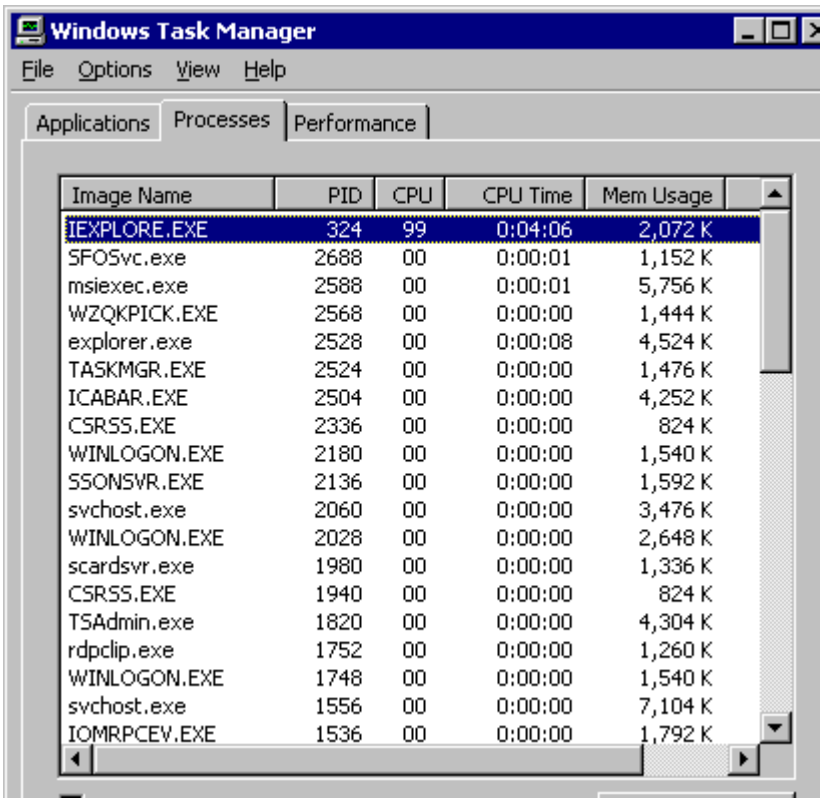
Together these two approaches allow you to eliminate the impacts of rogue applications upon your terminal servers.

4. How Does Application Shaping Impact User Experience?

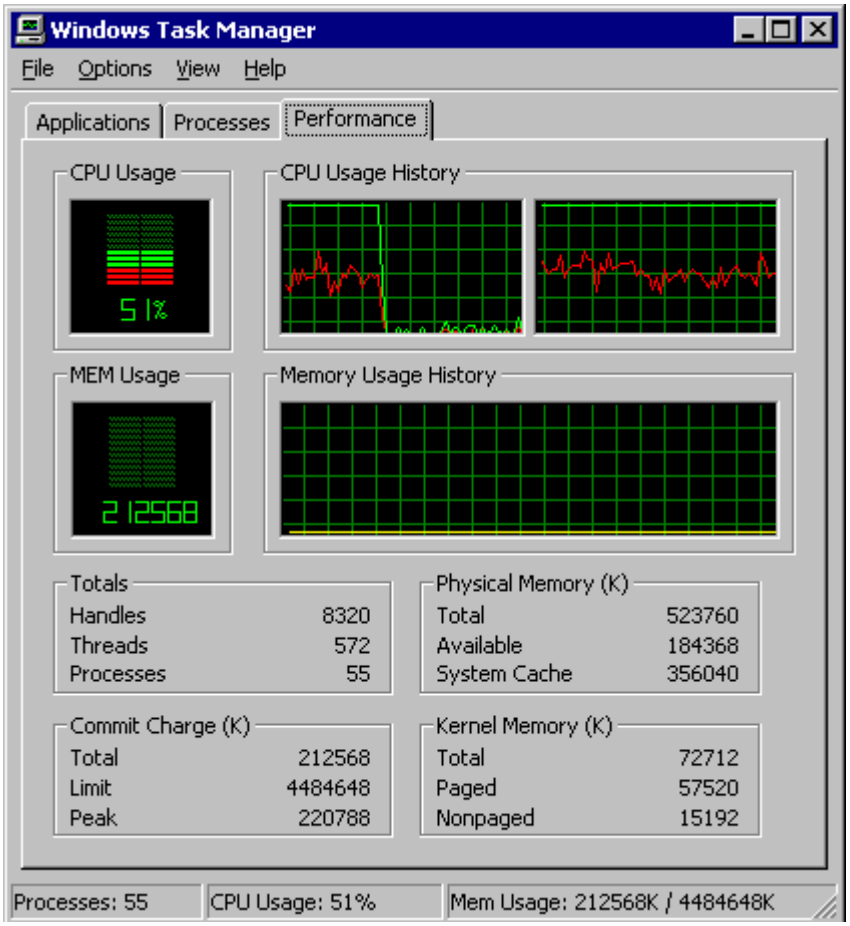
4.1. An application goes rogue on a terminal server. Assume this application is a multi-threaded application, and the threads are spread across both CPUs by the OS. Consequently, this one rogue application consumes 100% of both CPUs on this server.



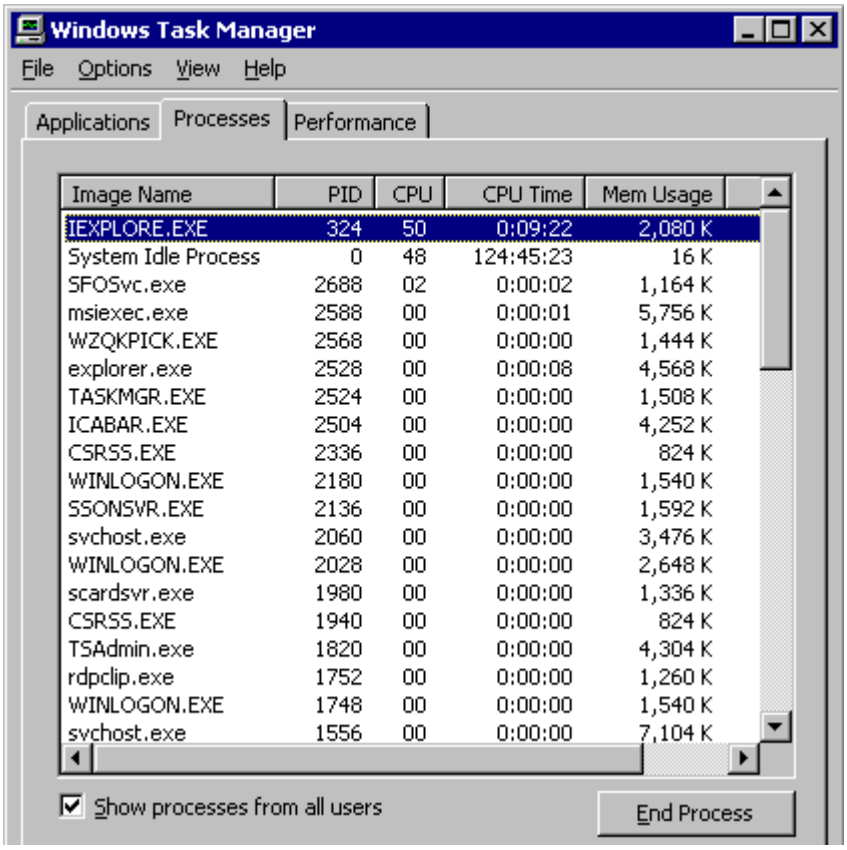
4.2. In this real world example, the rogue instance of the application is IEXPLORE.EXE and, as you can see from the Processes view in Task Manager, it is consuming 99% of all of the CPU resource on the server.



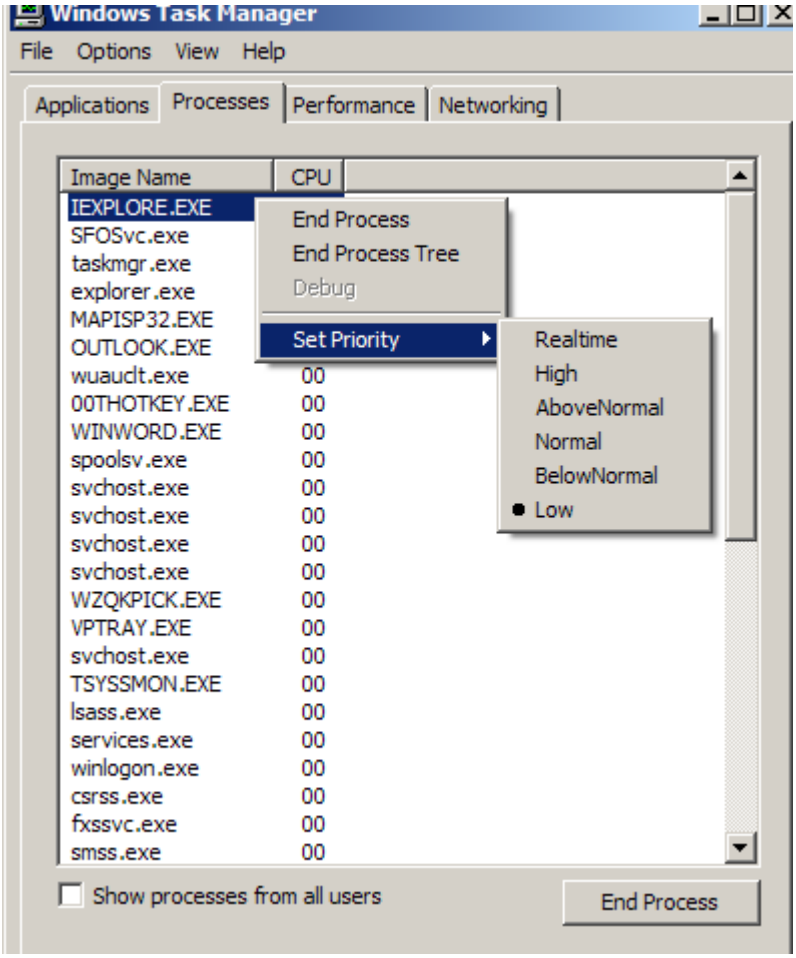
4.3. When Application Affinity is implemented (choosing CPU 1) for IE, it will tell the OS to move all of the threads for all IE processes over to CPU 1. CPU utilization for CPU 0 will then drop back to normal levels.



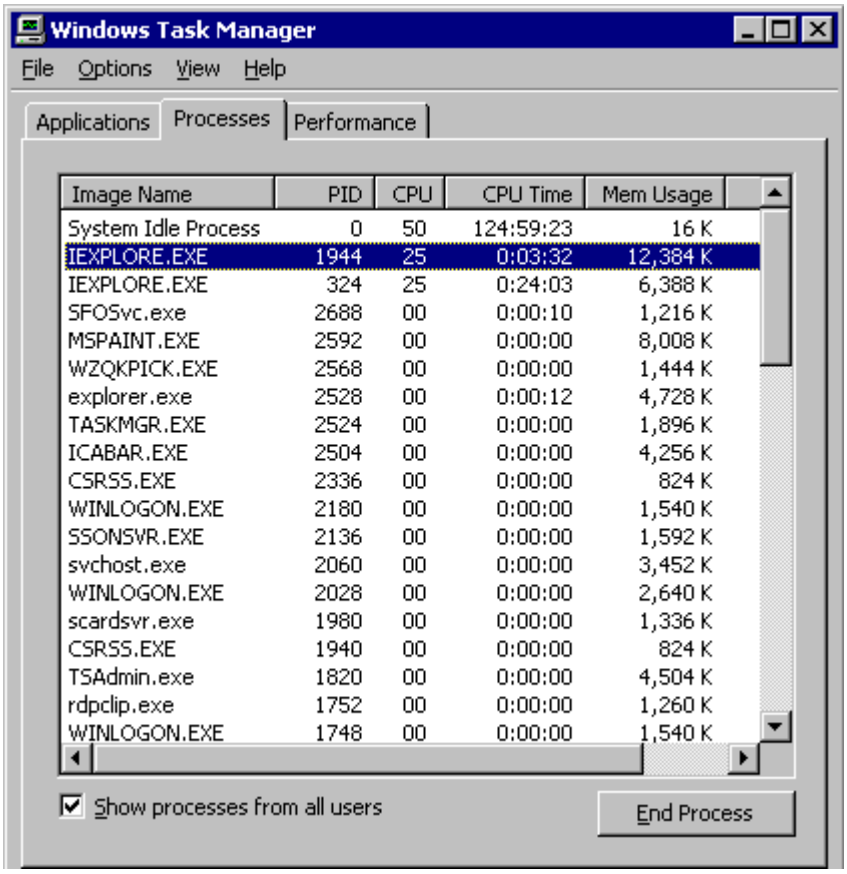
4.4. Now IEXPLORE.EXE is using 100% of CPU #1; therefore, Task Manager reports IE using 50% of the total CPU resource available in the server.



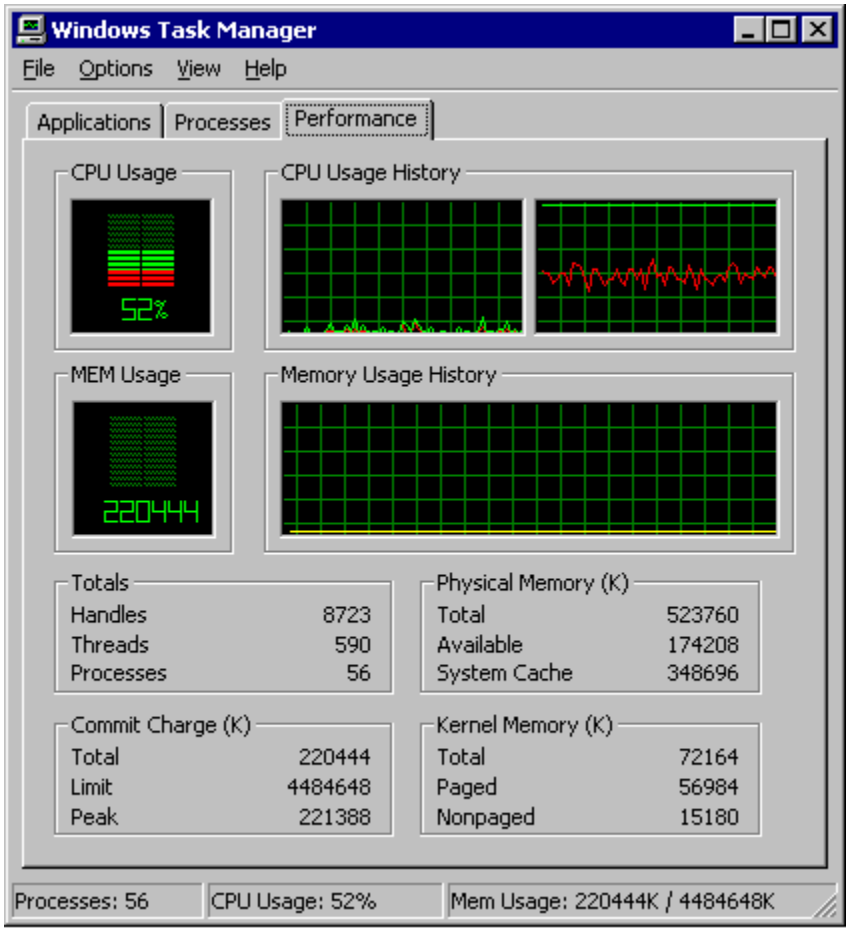
4.5. The next step is to implement Application Priority. This is the equivalent to manually right clicking a process in Task Manager and setting the Priority to Low. This causes the rogue process to use only CPU cycles that are available after every other application has consumed what it needs.



4.6. Now, having invoked Application Priority on IE, other applications can launch and run with acceptable performance. In the case depicted to the right, a second instance of IE is launched and is using 50% of CPU 1, which is 25% of the total CPU resource on the server as reported by Task Manager.



4.7. Note that Application Affinity and Application Priority together ensure that any instances of IE that need the CPU can run with good performance on CPU 1. All other applications can share CPU 0, and can also use time available on CPU 1. Shaping does not necessarily cause the total amount of CPU used on CPU 1 to be reduced below 100%, it simply ensures that every other process running on CPU 1 gets what it needs before the rogue process (in this case IE) gets the rest.



5. How is Application Shaping Implemented in TScale?

Application Shaping is comprised of two different features in TScale. These features are:

1. **Application Affinity**. Application Affinity allows the administrator to select which processors problem applications will run on, allowing you, for example, to force all instances of IE to run on the second CPU (CPU 1), while keeping the first CPU (CPU 0) isolated from rogue instances of IE when they occur. While Affinity will not stop instances of IE from launching on CPU 0, it will cause the OS to move the process over to CPU 1 as soon as it is up and running.

2. **Application Priority**. Application Priority allows administrators to set a threshold of CPU utilization for an application which, when reached, will cause TScale to reduce the priority of that process. Reducing the priority of the process has the effect of limiting the amount of CPU it can consume when other processes require CPU, but will allow the rogue process to use any remaining CPU bandwidth that would otherwise be allocated to the System Idle Process.

   Application Priority is triggered when an application consumes a CPU percentage above a set threshold for a specified period of time. (Both the CPU % and the time period are configurable by the administrator on an application specific basis.)

Application Shaping is extremely flexible and configurable, allowing administrators to

- setup "shaping policies" which includes the application(s) to be shaped, the time periods and thresholds
- create sets of users to which a policy applies
- create a schedule of time for which a policy applies

It is possible to individually specify shaping parameters for any number of applications, or to name a default "*.*" process so that any application reaching certain levels of processor usage will be subject to shaping.

Additionally, there is an easy mechanism to exempt specific users from shaping policies. So, an administrator can easily have a policy that is in force for "everyone," excepting just those few users who need to run an application at high levels of CPU utilization as a part of their job functions.

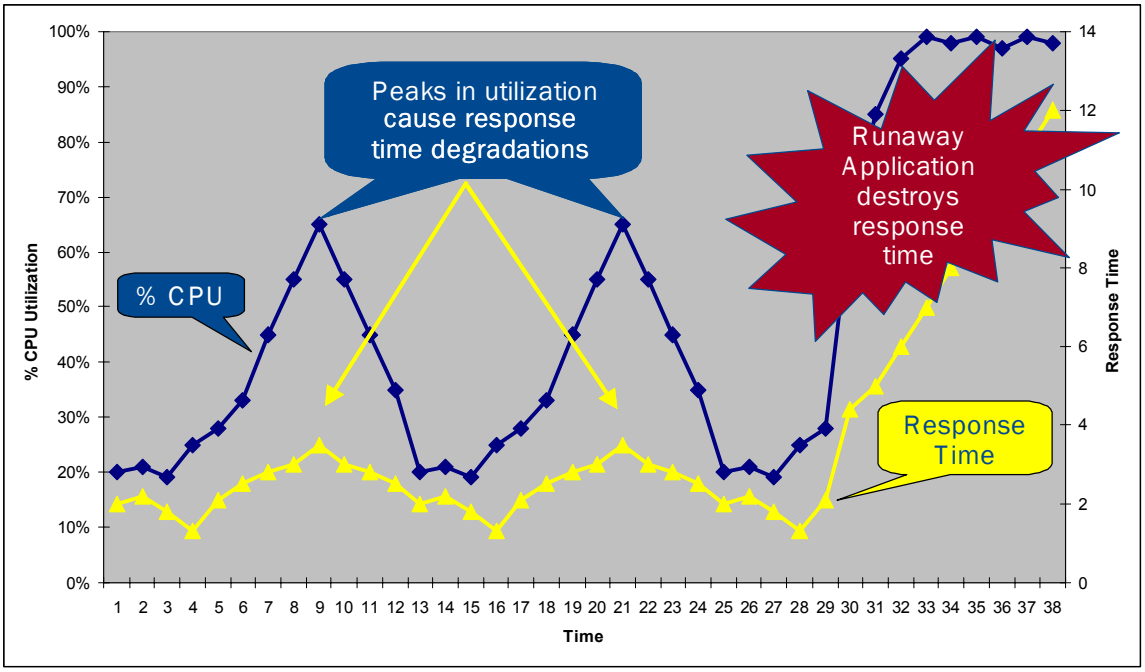
Note: *While we have used Internet Explorer (IE) as a real example, please consider two things:*

1) *TScale's Application Shaping obviously applies to any application, not just IE, including 32-bit, 16-bit and DOS applications (via NTVDM).*
2) *IE alone does not typically exhibit rogue behavior. It is the animations, pop-ups and controls frequently used by web designers/developers than can cause problems. Applying Application Shaping to IE controls these problems, and IE is simply used as an example.*

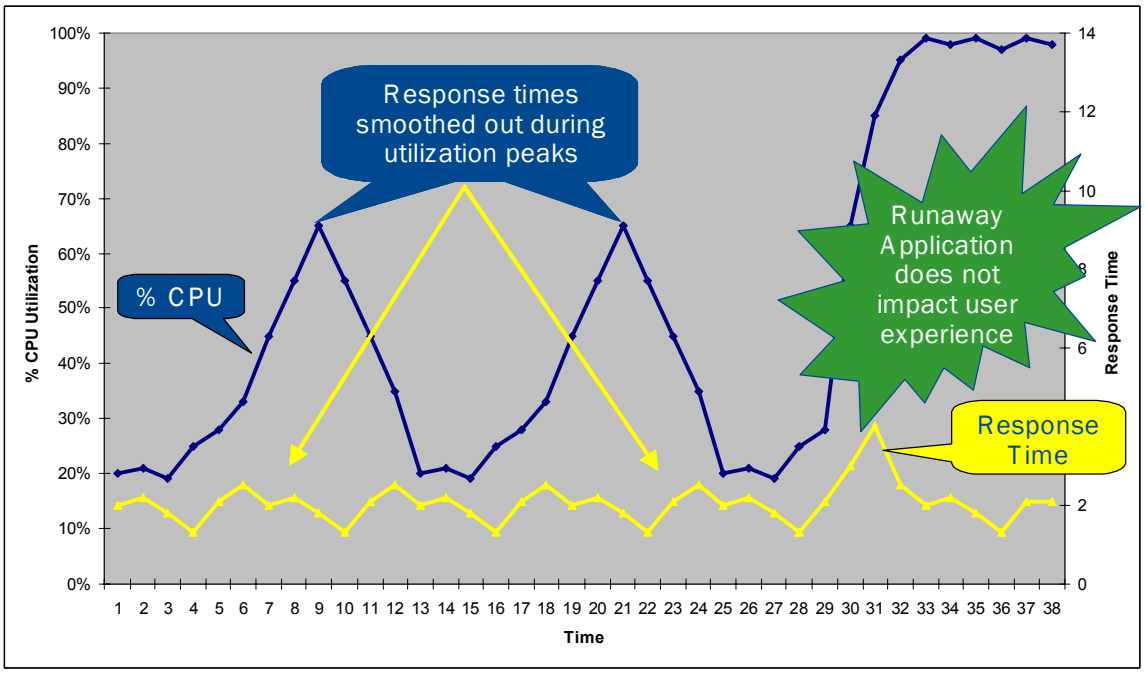
6. How Does Shaping Interact with the Rest of TScale?

When you add the benefits of Application Shaping to the benefits of the Applications Optimizations that have been in TScale since its inception you get the results depicted in the diagrams below. (To learn more about TScale's Application Optimization, please visit our web site – www.rtosoft.com – for more information and other white papers.)

Before TScale



After TScale



7. Conclusion

IT departments are responsible for providing consistent, reliable performance of key applications to the end-users. Organizations using Citrix/Microsoft Terminal Services in their business require robust tools to ensure peak system performance and prevent rogue applications from causing system freezes and sluggishness.

Only TScale combines the benefits of both Application Optimization and Application Shaping into one easy to use and easy to administer product. With TScale you can deliver the following benefits to the users of your terminal server farm:

- Applications will perform better during periods of high user load or high workload, resulting in better overall user experiences and less money spent on headroom and server redundancy.
- You will be able to comfortably support more concurrent users per server.
- Runaway (rogue) applications will be constrained from consuming all server CPU resources so users will enjoy a more consistent application operation.
- Administration and help desk resources will be better utilized, as it will no longer be necessary to manually terminate rogue processes.

About RTO Software

Founded in 2000, RTO Software is pioneering a new category of performance management tools that automatically, continuously and autonomously improve the capacity and scalability of Windows-based applications, servers and desktops. Based in Suwanee, Georgia, RTO's products are used in a wide variety of industries, including financial management, manufacturing, healthcare, telecommunications, and government.

For more information, or to obtain a free evaluation version of TScale, please visit RTO Software at www.rtosoft.com, or call at +1-678-455-5506.

A comprehensive applications gallery showing relevant TScale benefits for a variety of scenarios—including business, health care, and oil and gas industries—is available in the Products section of www.rtosoft.com.