

Unknown Facts about Win365 and AVD What Really Happens in the RDP Stack

Dr. Benny Tritsch | Independent Performance Data Scientist
info@eucscore.com | <https://drtritsch.com> | linkedin.com/in/drtritsch



Thank you to all our SPONSORS!





Bernhard Tritsch







-  Germany
-  Dr. Tritsch IT Consulting
-  He/Him/His
-  23 years in the program



Table of Content

- Classic RDP versus Side-by-Side (SxS) RDP
- Why remoting clients can exhibit screen artifacts
- How RDP bandwidth consumption depends on whether the desktop is visible on the endpoint device
- How localized performance counters silently break monitoring
- Conclusions

Classic RDP versus Side-by-Side (SxS) RDP



Remote Desktop Protocol

Classic RDP

- **Inbound Connections**
- Introduced: 1998 (RDP 4.0)
- Client: MSTSC (mstscax.dll) – retirement phase
- Refresh: With OS updates
- Evolution themes: Security, graphics & performance (UDP), connectivity, device & input
- Current focus: Security, cloud and cross-platform

Side-by-Side RDP

- **Reverse Connections (outbound)**
- Introduced: 2019 with WVD/AVD
- Installed as part of the AVD agent
- Client: Windows App
- Refresh: Every 4 weeks
<https://learn.microsoft.com/en-us/azure/virtual-desktop/whats-new>
- Based on URCP (Universal Rate Control Protocol) & WebSockets (TCP)
- RDP Shortpath (STUN, TURN)

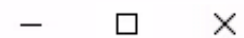
Side-by-Side Stack for RDP

- Per-session perf counter instances are named "RDP-SXS sessionID"
- SxS RDP can be installed in addition to the classic RDP protocol (RDP-TCP) and to other remoting protocols (HDX, Blast)
- Update without server reboot or user disconnect

The screenshot shows the 'Add Counters' dialog box in Windows Performance Monitor. The 'Available counters' list includes 'RemoteFX Graphics' and 'RemoteFX Network'. The 'Instances of selected object' list shows 'rdp-sxs 59' selected. To the right, a performance table for '\\W10N' is displayed, showing processor information and RemoteFX Network statistics for the instance 'rdp-sxs 59'.

\\W10N		Total
Processor Information		
% Processor Time		7.908
RemoteFX Network		rdp-sxs 59
Base TCP RTT		16.000
Base UDP RTT		0.000
Current TCP Bandwidth		36,725.000
Current TCP RTT		25.000
Current UDP Bandwidth		0.000
Current UDP RTT		0.000
FEC Rate		0.000
Loss Rate		0.000
Retransmission Rate		0.000
Sent Rate P0		0.000
Sent Rate P1		0.000
Sent Rate P2		30,099.795
Sent Rate P3		195,088.343
TCP Received Rate		0.000
TCP Sent Rate		0.000
Total Received Bytes		64,554,982.000
Total Received Rate		18,553.912
Total Sent Bytes		1,294,391,414
Total Sent Rate		222,252.343
UDP Packets Received/sec		0.000
UDP Packets Sent/sec		0.000
UDP Received Rate		0.000
UDP Sent Rate		0.000

Administrator: C:\windows\system32\cmd.exe



C:\Users\bt-admin>query session

SESSIONNAME	USERNAME	ID	STATE	TYPE	DEVICE
services		0	Disc		
console		1	Conn		
rdp-sxs191031...	btritsch	2	Active		
>rdp-tcp#2	bt-admin	3	Active		
31c5ce94259d4...		65536	Listen		
rdp-tcp		65537	Listen		
rdp-sxs191031003		65538	Listen		
rdp-sxs200326004		65539	Listen		

C:\Users\bt-admin>query session

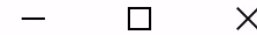
SESSIONNAME	USERNAME	ID	STATE	TYPE	DEVICE
services		0	Disc		
console		1	Conn		
rdp-sxs200326...	btritsch	2	Active		
>rdp-tcp#2	bt-admin	3	Active		
31c5ce94259d4...		65536	Listen		
rdp-tcp		65537	Listen		
rdp-sxs200326004		65538	Listen		

C:\Users\bt-admin>query session

SESSIONNAME	USERNAME	ID	STATE	TYPE	DEVICE
services		0	Disc		
console		1	Conn		
rdp-sxs200326...	btritsch	2	Active		
>rdp-tcp#2	bt-admin	3	Active		
31c5ce94259d4...		65536	Listen		
rdp-tcp		65537	Listen		
rdp-sxs200326004		65538	Listen		

C:\Users\bt-admin>

C:\windows\system32\cmd.exe



C:\Users\btritsch>query session

SESSIONNAME	USERNAME	ID	STATE	TYPE	DEVICE
services		0	Disc		
console		1	Conn		
>rdp-sxs191031...	btritsch	2	Active		
31c5ce94259d4...		65536	Listen		
rdp-tcp		65537	Listen		
rdp-sxs191031003		65538	Listen		

C:\Users\btritsch>query session

SESSIONNAME	USERNAME	ID	STATE	TYPE	DEVICE
services		0	Disc		
console		1	Conn		
>rdp-sxs200326...	btritsch	2	Active		
	bt-admin	3	Disc		
31c5ce94259d4...		65536	Listen		
rdp-tcp		65537	Listen		
rdp-sxs200326004		65538	Listen		

C:\Users\btritsch>query session

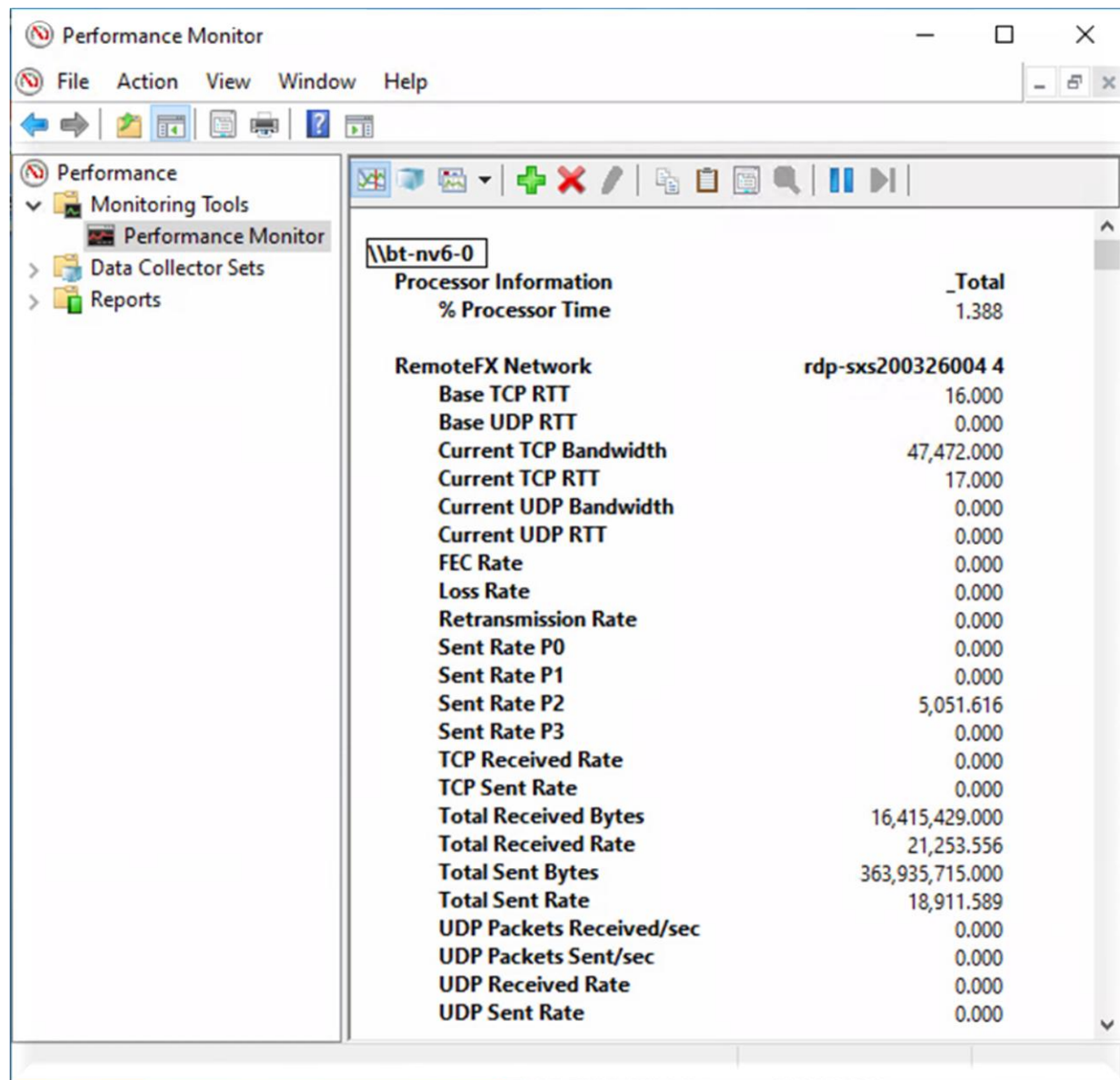
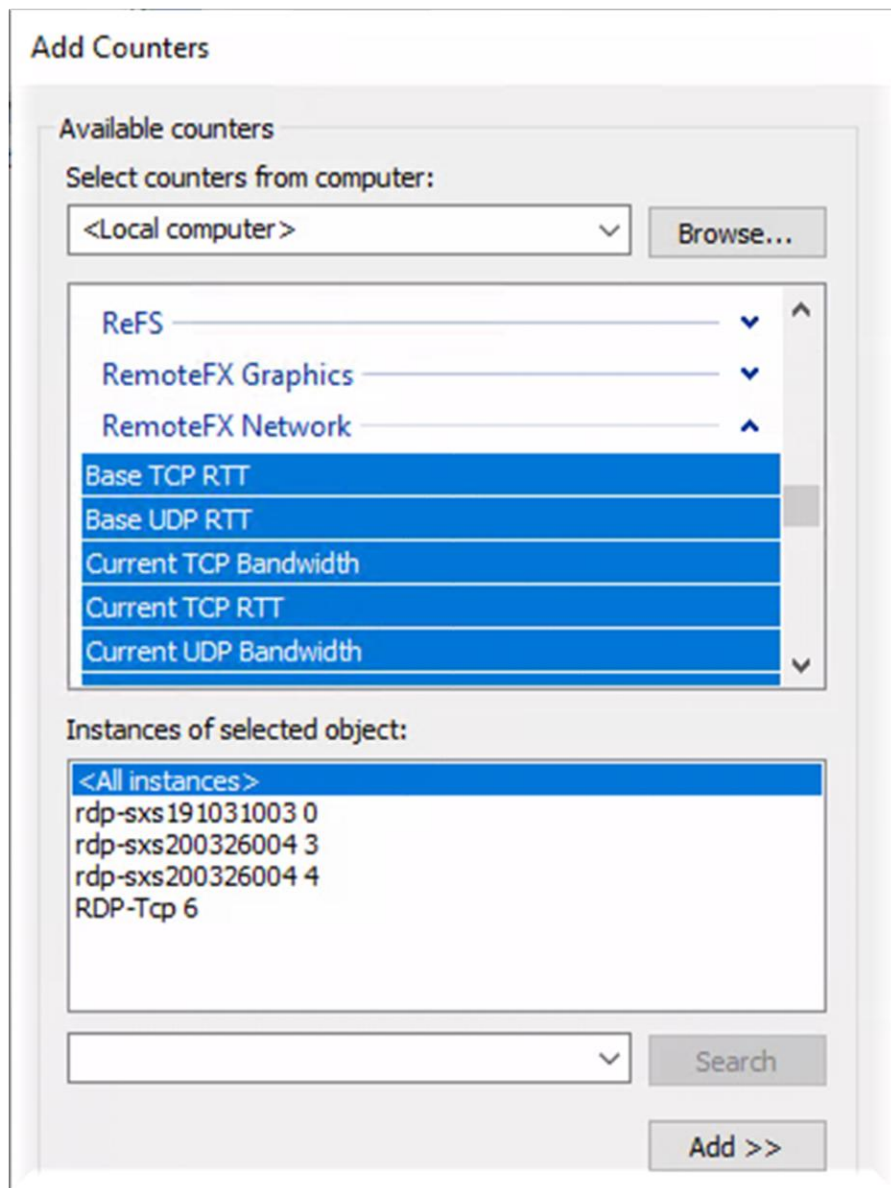
SESSIONNAME	USERNAME	ID	STATE	TYPE	DEVICE
services		0	Disc		
console		1	Conn		
>rdp-sxs200326...	btritsch	2	Active		
31c5ce94259d4...		65536	Listen		
rdp-tcp		65537	Listen		
rdp-sxs200326004		65538	Listen		

C:\Users\btritsch>

SxS Stack for RDP WVD VM

The screenshot shows the Windows Performance Monitor application. The left pane displays the 'Performance' tree with 'Monitoring Tools' expanded to 'Performance Monitor'. The right pane shows a table of performance data for the network interface '\\bt-nv6-0'. The table is organized into sections: Processor Information, RemoteFX Network, and a specific RDP session 'rdp-sxs191031003 0'. The data includes various metrics such as % Processor Time, Base TCP/UDP RTT, Current TCP/UDP Bandwidth, and Total Received/Sent Bytes and Rates.

\\bt-nv6-0	
Processor Information	
% Processor Time	43.546
RemoteFX Network	
rdp-sxs191031003 0	
Base TCP RTT	17.000
Base UDP RTT	0.000
Current TCP Bandwidth	38,936.000
Current TCP RTT	28.000
Current UDP Bandwidth	0.000
Current UDP RTT	0.000
FEC Rate	0.000
Loss Rate	0.000
Retransmission Rate	0.000
Sent Rate P0	0.000
Sent Rate P1	0.000
Sent Rate P2	116,850.052
Sent Rate P3	0.000
TCP Received Rate	0.000
TCP Sent Rate	0.000
Total Received Bytes	464,888.000
Total Received Rate	2,349.628
Total Sent Bytes	7,911,622.000
Total Sent Rate	118,648.236
UDP Packets Received/sec	0.000
UDP Packets Sent/sec	0.000
UDP Received Rate	0.000
UDP Sent Rate	0.000



Universal Rate Control Protocol

- <https://www.microsoft.com/en-us/research/publication/urcp-universal-rate-control-protocol-for-real-time-communication-applications/> (June 2013)
- <https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/msrtech-report.pdf>

URCP: Universal Rate Control Protocol for Real-Time Communication Applications

Di Xie ^{*†}, Sanjeev Mehrotra ^{*}, Jin Li ^{*}, Y. Charlie Hu [†]

^{*} *Microsoft Research* – {dxie, sanjeevm, jinl}@microsoft.com

[†] *Purdue University* – ychu@purdue.edu

Abstract

We present a Universal Rate Control Protocol (URCP), which provides applications with *fair and full link utilization* while operating at *low delay and loss levels* as needed by interactive real-time communications (RTC) applications across a range of complex networks in use today, such as Wi-Fi hotspots, 4G (HSPA+), and WiMAX. As opposed to many existing rate control techniques which are specially designed for differing network types and application requirements, URCP attempts to provide a universal framework for rate control by learning network parameters and then automatically adapting a utility maximization based rate control framework to achieve good performance across any network.

URCP is able to provide significantly improved performance over state-of-art rate control techniques for RTC applications, in terms of throughput, delay, and loss. On real-world network tests, URCP provides similar throughput to commonly used loss based schemes while achieving up to a 100x improvement in operating delay (compared to TCP NewReno on the Clearwire WiMAX network). Compared to existing delay based rate control protocols commonly used by RTC applications, URCP provides similar operating delay while achieving close to 2.5x improvement in throughput on noisy networks (compared to WebRTC on the T-Mobile HSPA+ network).

Results of Search in US Patent Collection db for:

URCP: 4 patents.

Hits 1 through 4 out of 4





Jump To

Refine Search

URCP

PAT. NO.

Title

- 1 [10,637,784](#)  [Equation-based rate control using network delay for variable bitrate scenarios](#)
- 2 [10,341,240](#)  [Equation-based rate control using network delay for variable bitrate scenarios](#)
- 3 [10,129,155](#)  [Delay based congestion control protocol co-existing with TCP](#)
- 4 [6,822,602](#)  [Method for generating and transmitting/receiving input codes in universal input device and apparatus thereof](#)

United States Patent

Zhao , et al.

10,129,155

November 13, 2018

Delay based congestion control protocol co-existing with TCP

Abstract

Technologies are described for performing hybrid rate control that switches between a delay-based mode and a passive loss-based mode for a flow of network traffic. The switching can be performed based on the presence of loss-based TCP network flows. For example, rate control can be performed for a flow of network traffic in a delay-based mode. When the presence of a loss-based TCP network flow is detected, the flow of network traffic can be switched from the delay-based mode to a passive loss-based mode and rate control can be performed in the passive loss-based mode. When the loss-based TCP flow is no longer detected, the flow of network traffic can be switched back to the delay-based mode.

Inventors: Zhao; Weidong (Bellevue, WA), Mehrotra; Sanjeev (Redmond, WA), Sadagopan; Jayashree (Bellevue, WA), Hagi; Costin (Sammamish, WA)

Applicant: Name City State Country Type

Microsoft Technology Licensing, LLC Redmond WA US

Assignee: Microsoft Technology Licensing, LLC (Redmond, WA)

Family ID: 62147924

Appl. No.: 15/357,984

Filed: November 21, 2016

Prior Publication Data

Document Identifier

US 20180145914 A1

Publication Date

May 24, 2018

Query Session

SxS RDP Session

[User Sessions]

Session1: SessionName=Services, UserName=N/A, SessionID=0, State=Disconnected

Session2: SessionName=Console, UserName=N/A, SessionID=1, State=Connected

Session3: SessionName=**rdp-sxs250225880#0**, **UserName=BernhardTritsch**, **SessionID=3**, **State=Active**

Session4: SessionName=31C5CE94259D4006A9E4, UserName=N/A, SessionID=65536, State=Listen

Session5: SessionName=RDP-Tcp, UserName=N/A, SessionID=65537, State=Listen

Session6: SessionName=rdp-sxs250225880, UserName=N/A, SessionID=65538, State=Listen

HDX Session

[User Sessions]

Session1: SessionName=Services, UserName=N/A, SessionID=0, State=Disconnected

Session2: SessionName=**Console**, **UserName=BernhardTritsch**, **SessionID=6**, **State=Active**

Session3: SessionName=31C5CE94259D4006A9E4, UserName=N/A, SessionID=65536, State=Listen

Session4: SessionName=RDP-Tcp, UserName=N/A, SessionID=65537, State=Listen

Session5: SessionName=rdp-sxs250225880, UserName=N/A, SessionID=65538, State=Listen

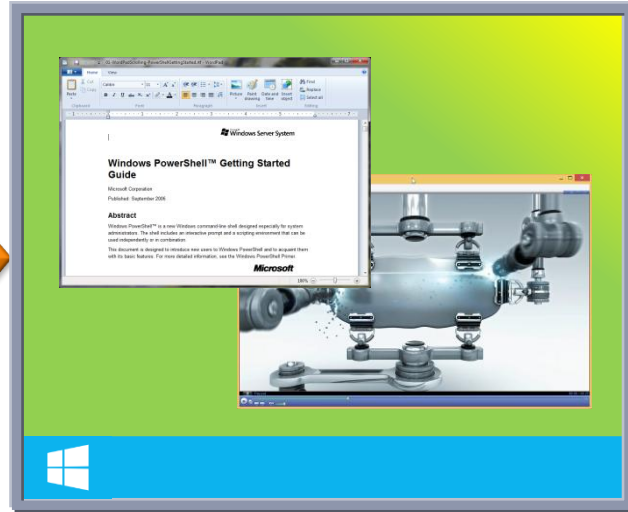
Silent Update of the SxS RDP Protocol

The image displays four overlapping screenshots of the Windows Remote Desktop connection quality status window, illustrating the protocol update over time. Each window shows connection details, network performance, and graphics settings.

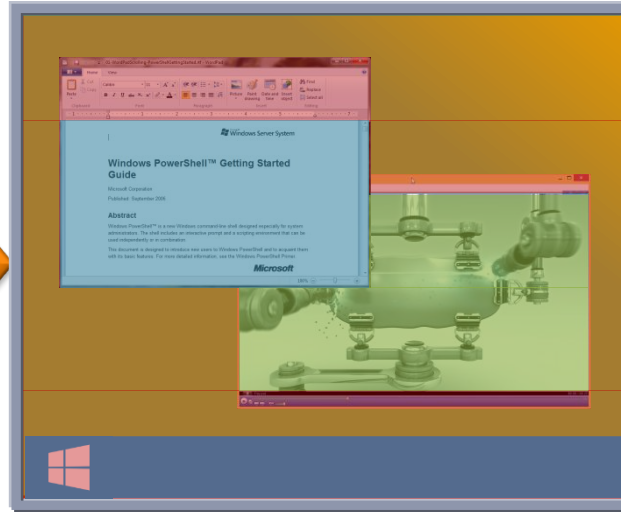
- Leftmost window (2024-10-14):** Shows a connection using UDP. Client version: 1.2.5709.0 (x64). Local OS: Windows 10 Enterprise x64 (10.0, Build 22H2). Transport protocol: UDP. Round-trip time: 10 ms. Available bandwidth: Greater than 113 Mbps. Frame rate: 6 FPS.
- Second window (2025-07-14):** Shows a connection using UDP. Client version: 1.2.6354.0 (x64). Local OS: Microsoft Windows 11 Enterprise x64 (10.0, Build 22H2). Transport protocol: UDP. Round-trip time: 7 ms. Available bandwidth: Greater than 513 Mbps. Frame rate: 0 FPS.
- Third window (2026-04-02):** Shows a connection using UDP - Multipath. Client version: 1.2.6980.0 (x64). Local OS: Microsoft Windows 11 Enterprise x64 (10.0, Build 22H2). Transport protocol: UDP - Multipath. Round-trip time: 16 ms. Available bandwidth: Greater than 168 Mbps. Frame rate: 2 FPS.
- Rightmost window (2026-06-04):** Shows a connection using Multipath Websocket. Client version: 1.2.7207.0 (x64). Local OS: Microsoft Windows 11 Pro x64 (10.0, Build 26200). Transport protocol: Multipath Websocket. Round-trip time: 87 ms. Available bandwidth: 1,85 Mbps. Frame rate: 0 FPS.

The status messages in the windows are: "Your connection quality is good and UDP", "Your connection quality is good and Multipath (UDP/WebSocket) is enabled", and "Your network might be slow. You may experience issues. Multipath (UDP/WebSocket) is enabled."

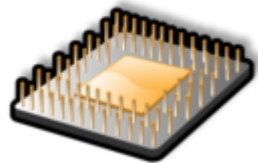
Render



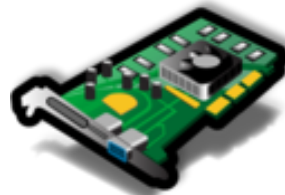
Capture



Compress



CPU
"Soft GPU"



GPU
vGPU

Identify different
capture areas:
Text, image,
video/animation

Compress

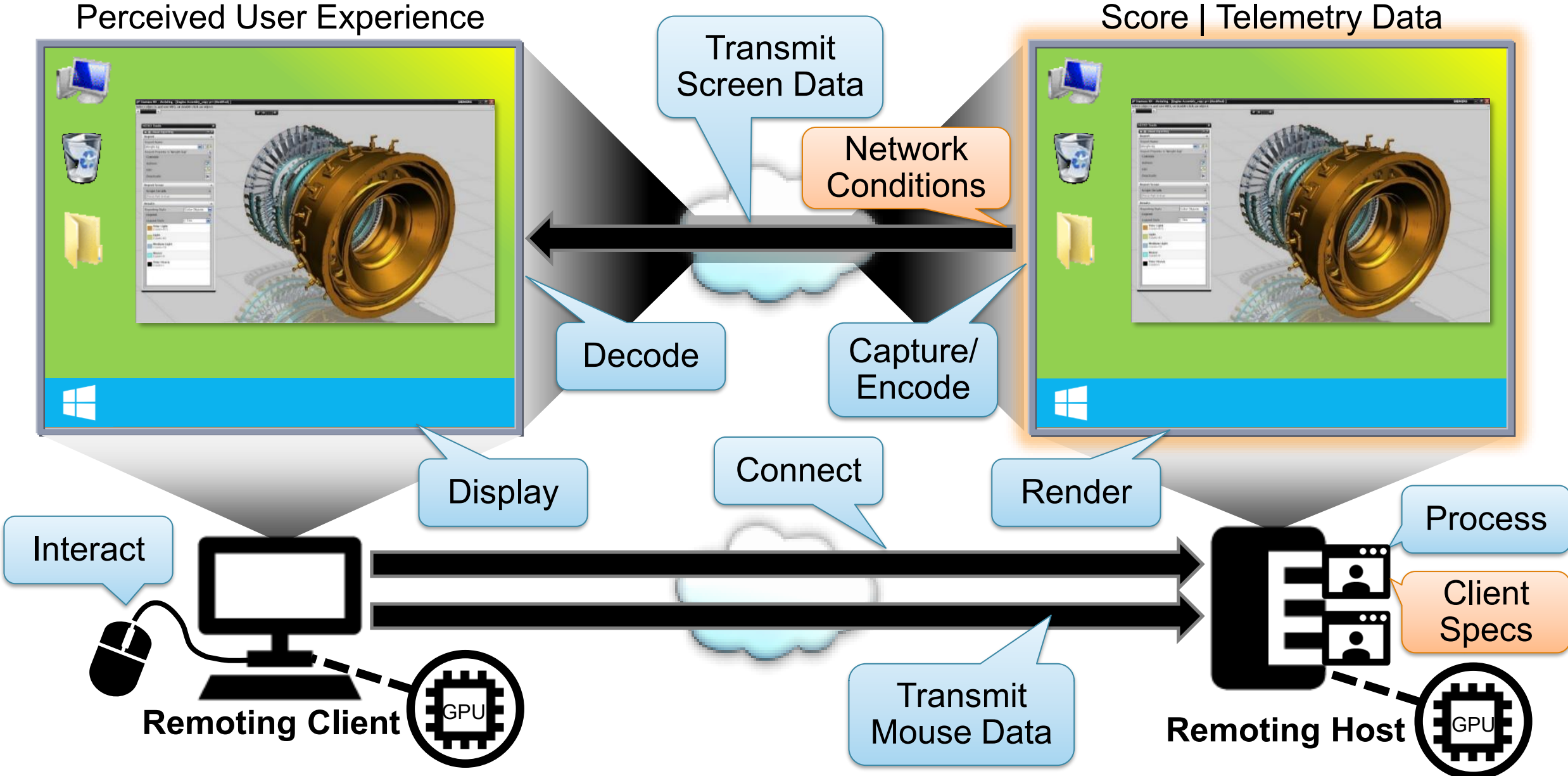
1. Text
2. Still Image
3. Video & Animation



Why remoting clients can exhibit screen artifacts

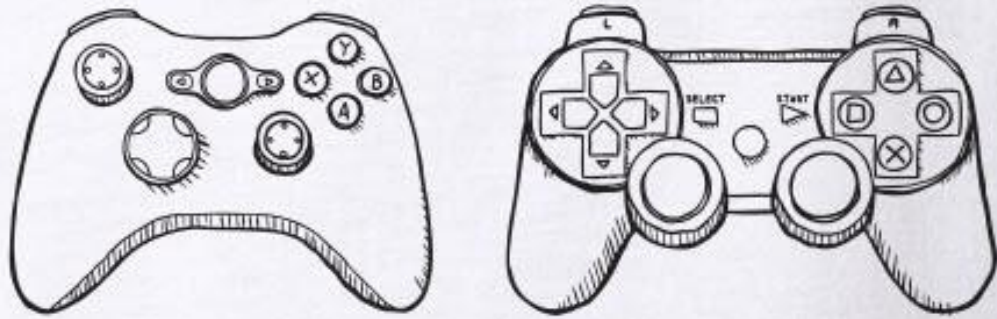


EUC in a Nutshell



Cloud Desktops

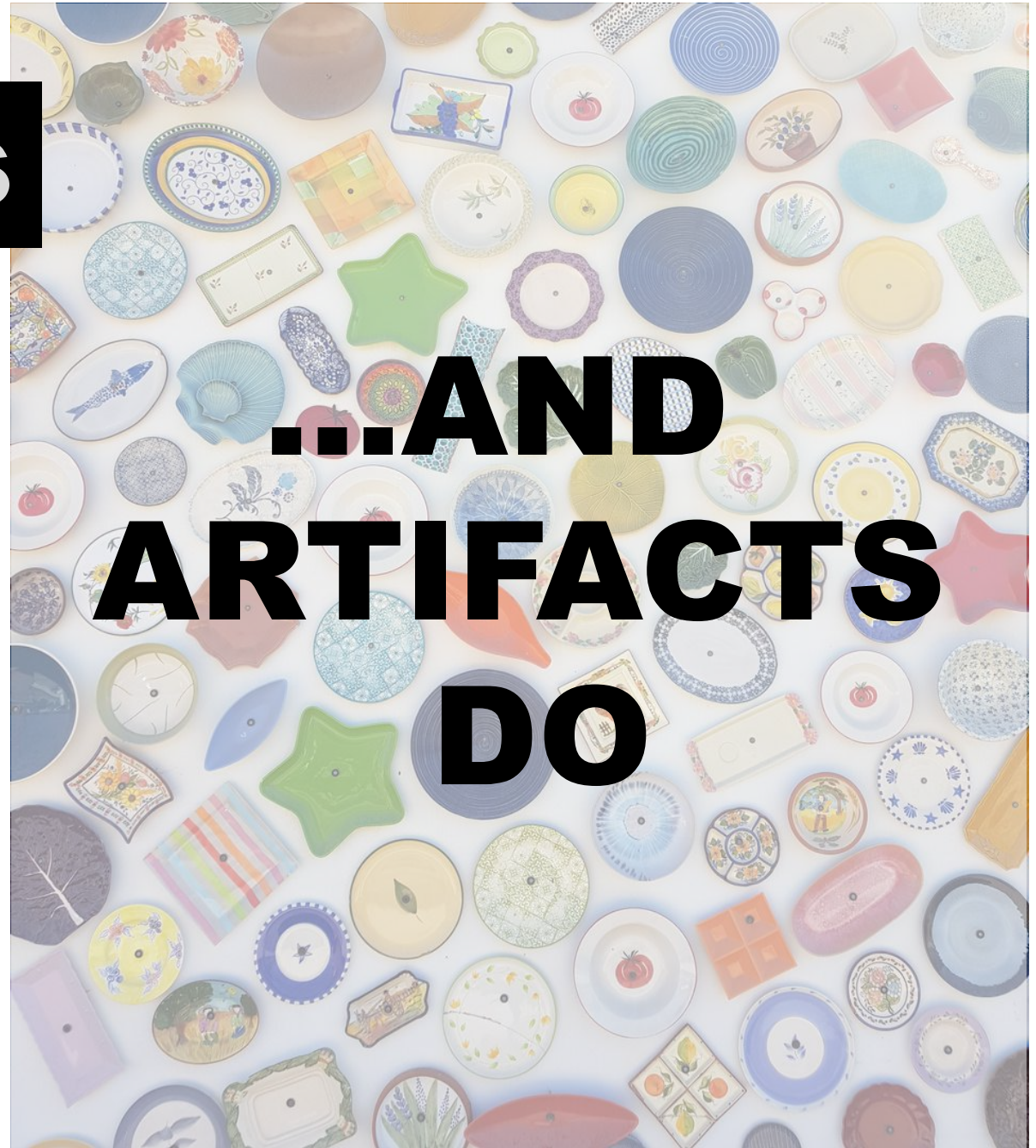
**DON'T MAKE
US VIOLENT**



LAG DOES

Patent
Patent

**...AND
ARTIFACTS
DO**

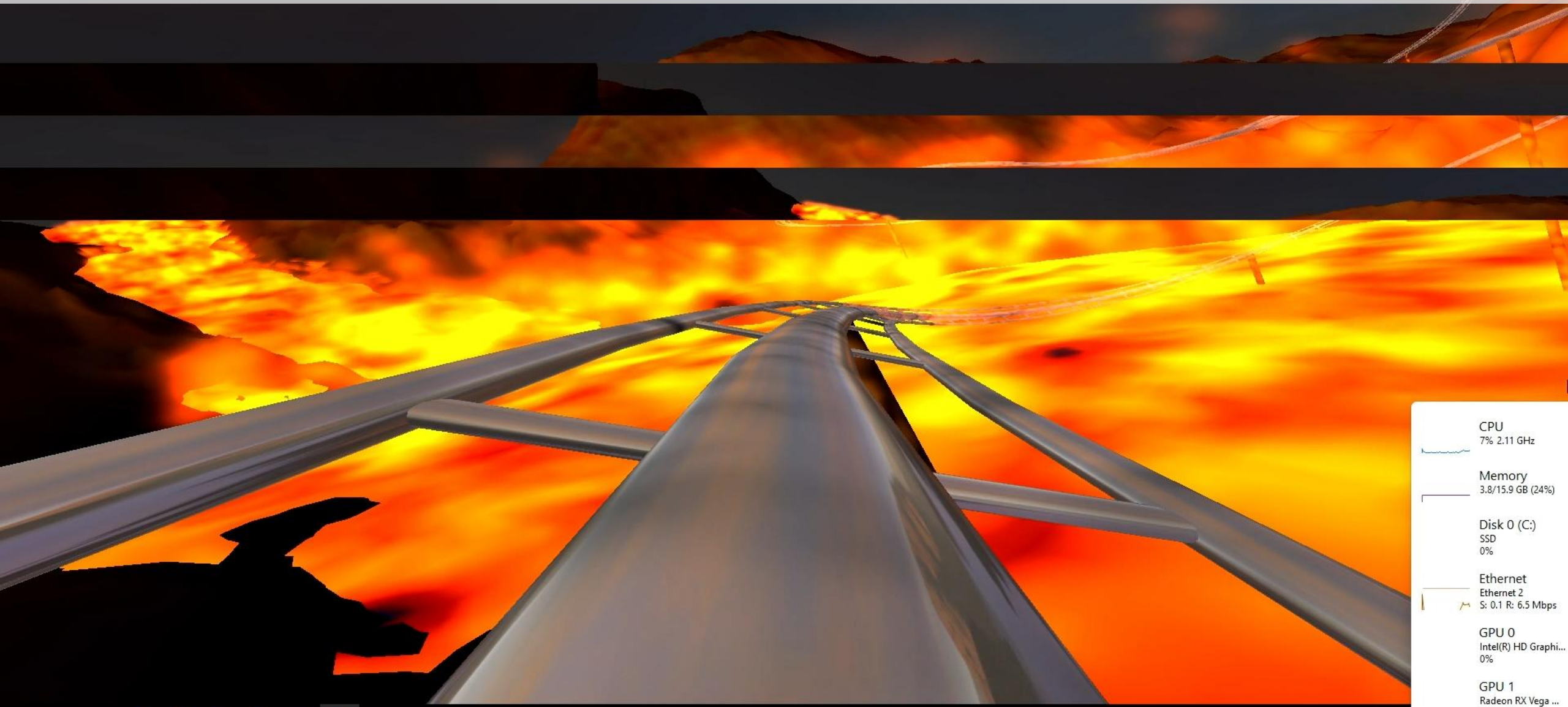


Block Artifacts or Tiling





Striping



CPU
7% 2.11 GHz

Memory
3.8/15.9 GB (24%)

Disk 0 (C:)
SSD
0%

Ethernet
Ethernet 2
S: 0.1 R: 6.5 Mbps

GPU 0
Intel(R) HD Graphi...
0%

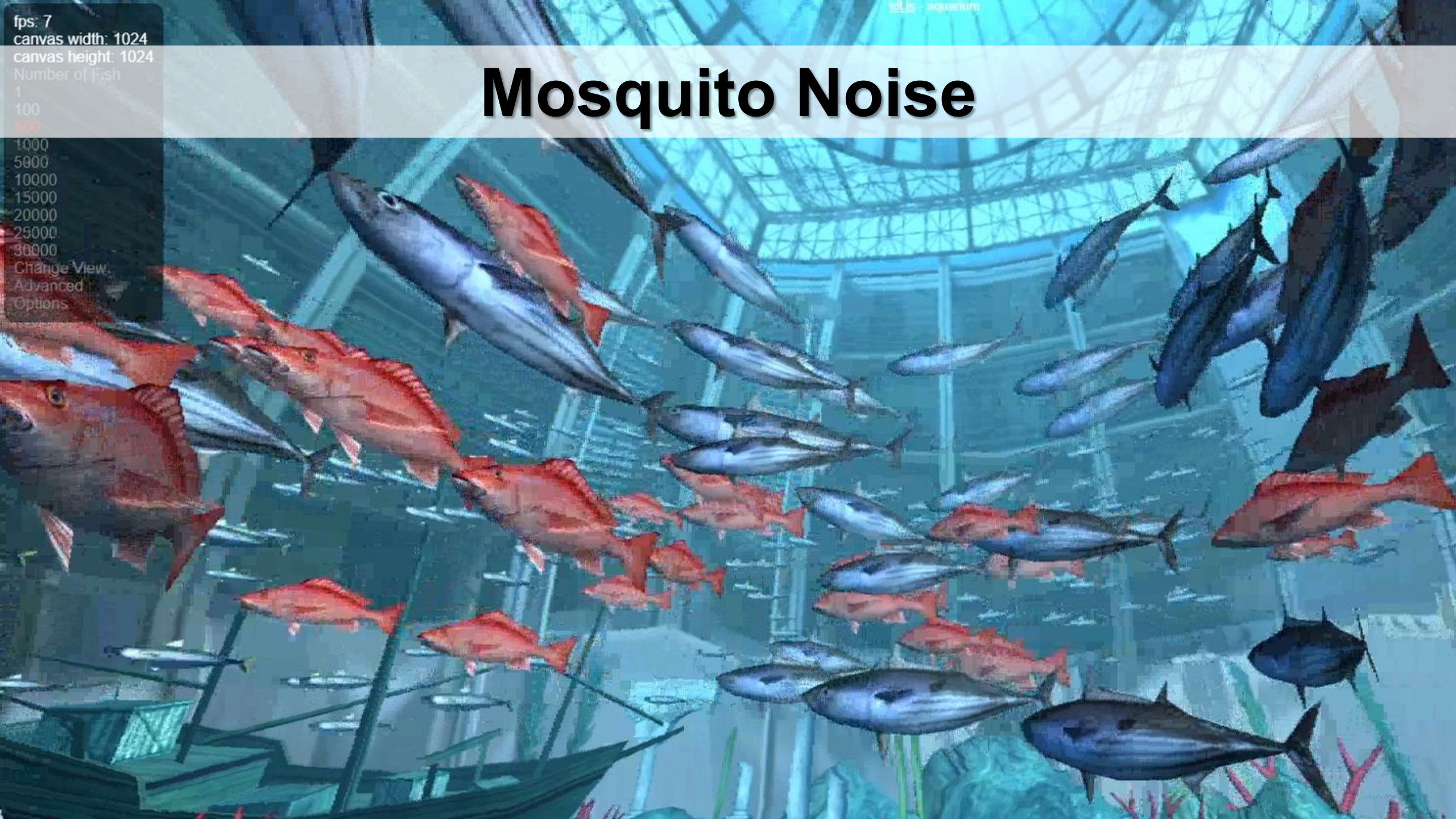
GPU 1
Radeon RX Vega ...
1% (47 °C)

Blurriness



fps: 7
canvas width: 1024
canvas height: 1024
Number of Fish
1
100
500
1000
5000
10000
15000
20000
25000
30000
Change View
Advanced
Options

Mosquito Noise



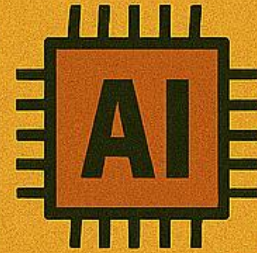
The Future of the RDP Protocol

- Better out-of-box automatic adaptation
- Codecs: H.264/AVC – H.265/HEVC – AV1 (Challenge: patents and CPU/GPU capabilities)
- HOT: Pluggable codecs to adjust quickly
- AI in remoting protocols – goal is to eliminate artifacts on the client
- DLSRR (Deep Learning-based Screen and Remote Rendering) with NPU + GPU

VDI SETUP

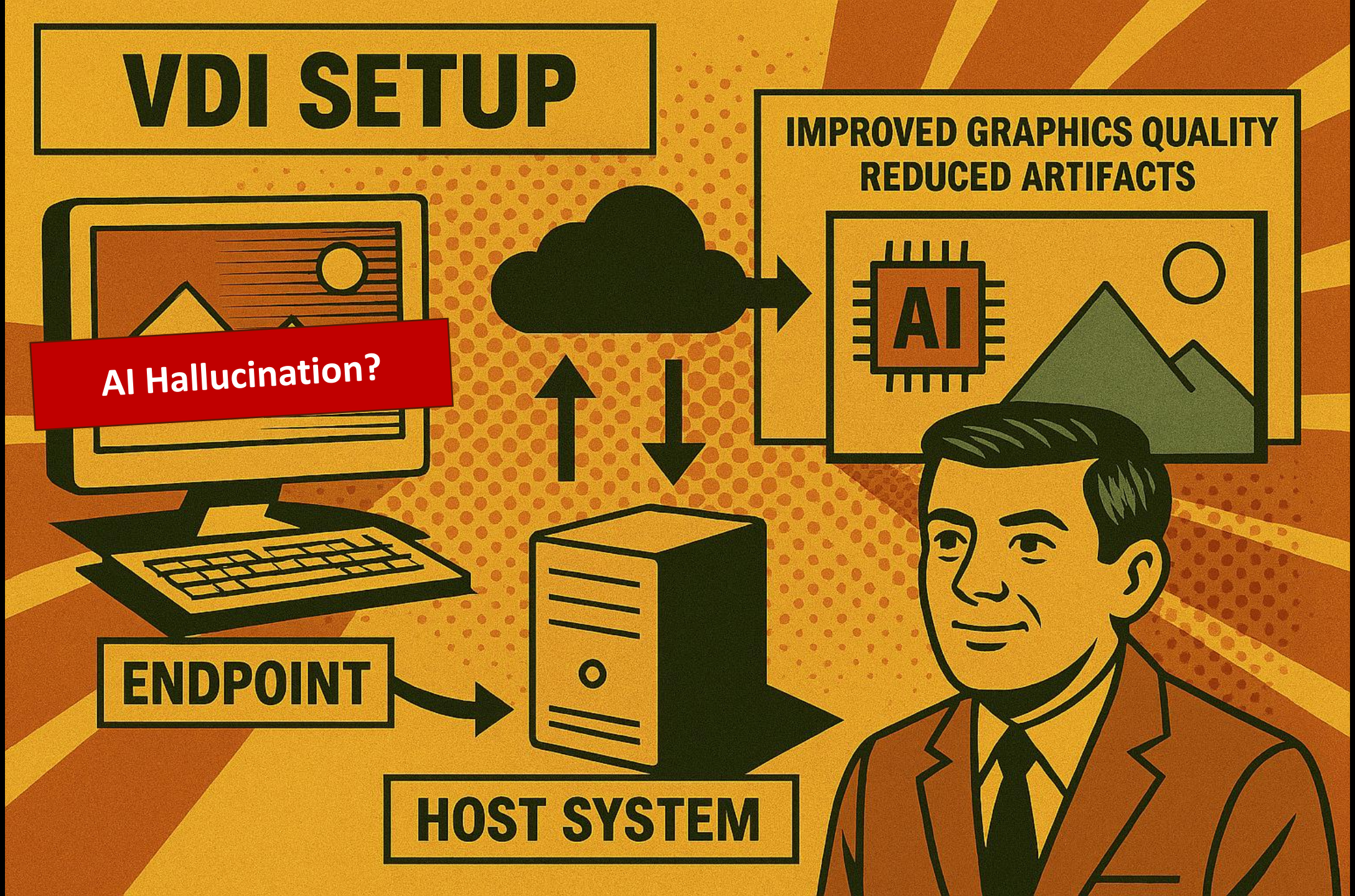
IMPROVED GRAPHICS QUALITY
REDUCED ARTIFACTS

AI Hallucination?



ENDPOINT

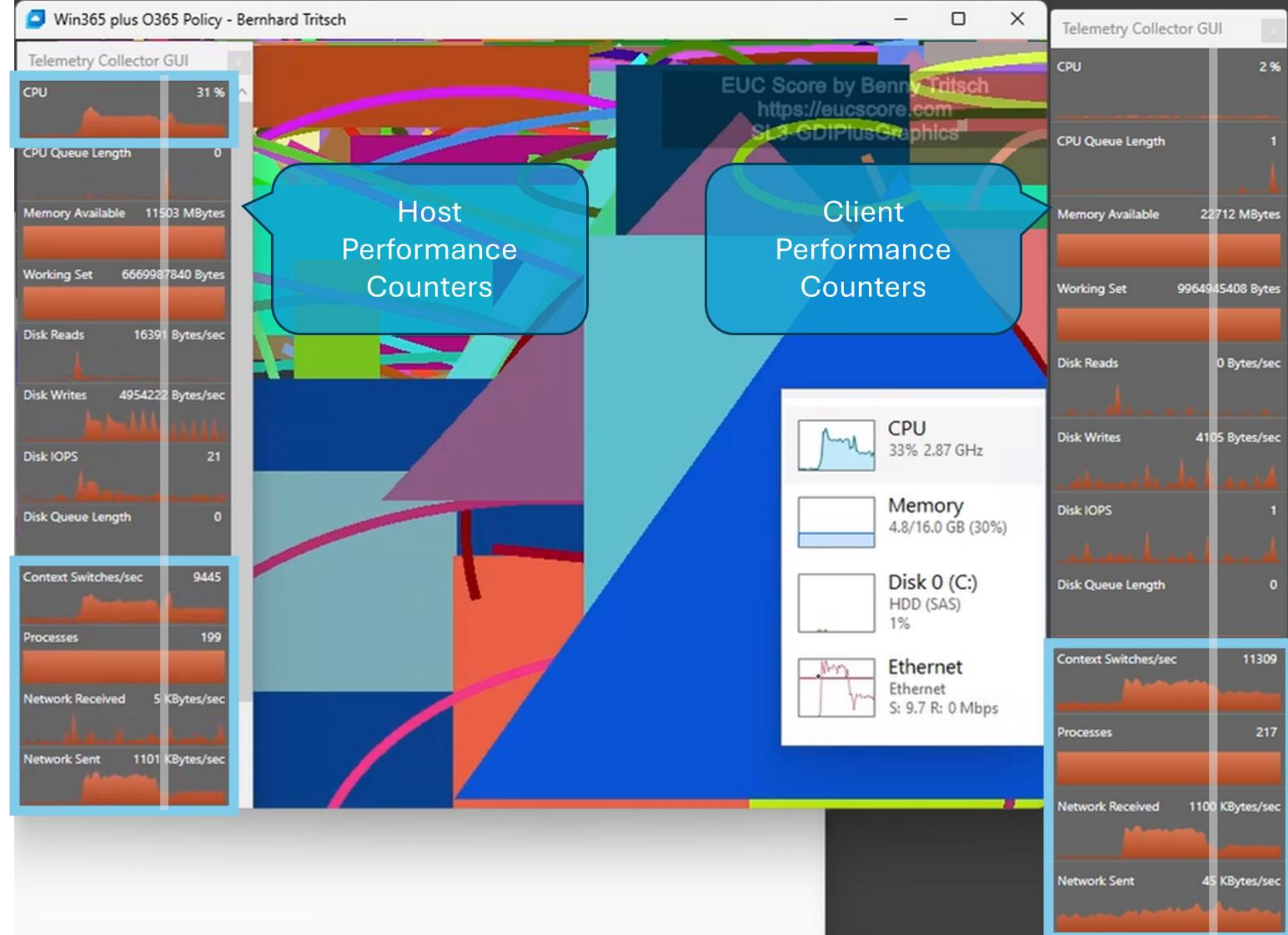
HOST SYSTEM



How RDP bandwidth consumption depends on desktop visibility



DrTritsch.com
How Screen
Resolution
Impacts AVD
Resource
Consumption -
And Why
Minimized
Windows
Behave
Differently

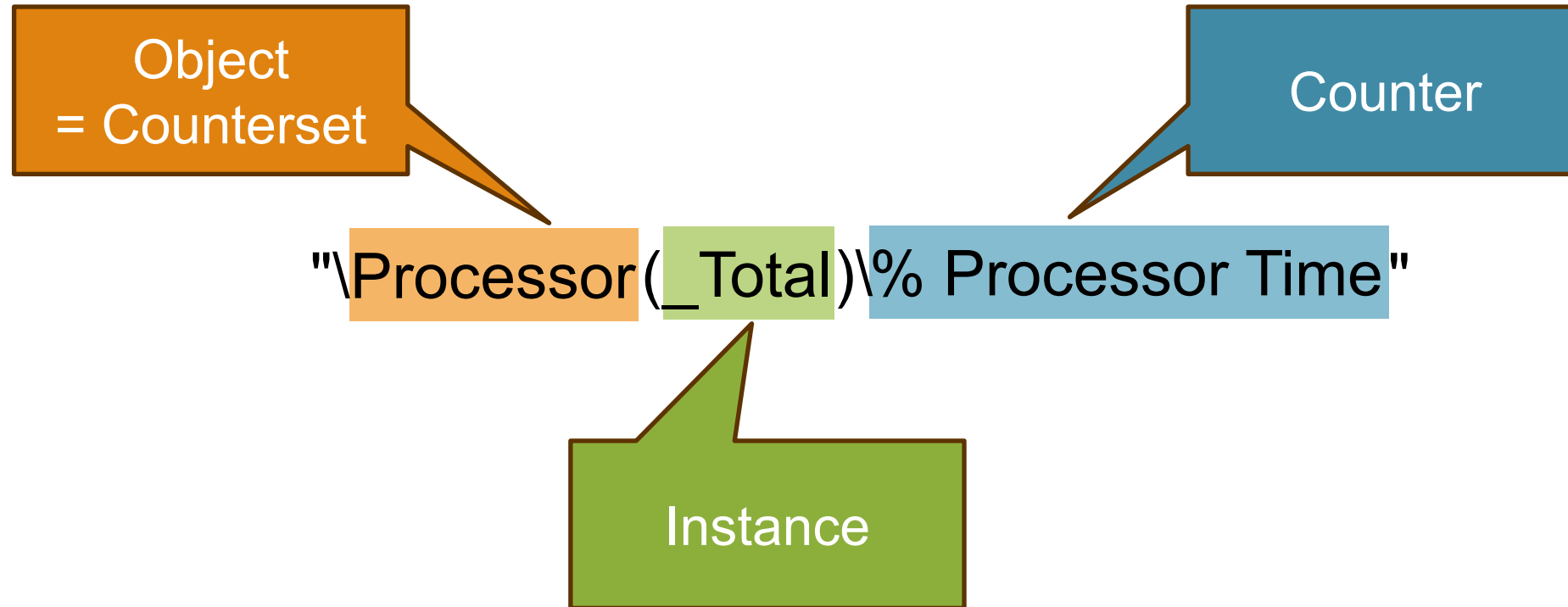


How localized performance counters break monitoring



Performance Counter Path Syntax

\\ComputerName\ObjectName(ObjectInstance)\ObjectCounter



A *counterset* is a grouping of performance data within a provider
A *counter* is the definition of single piece of performance data
An *instance* is an entity about which performance data is reported

Most Relevant EUC Performance Counters

Components	Performance Counters
CPU	\Processor(_Total)\% Processor Time \System\Processor Queue Length
Memory	\Memory\Available Mbytes \Process(_Total)\Working Set
Storage	\PhysicalDisk(_Total)\Disk Read Bytes/sec \PhysicalDisk(_Total)\Disk Write Bytes/sec \PhysicalDisk(_Total)\Disk Transfers/sec (IOPS) \PhysicalDisk(_Total)\Current Disk Queue Length
System	\System\Context Switches/sec \System\Processes
Network	TC::network received(_Total) TC::network sent(_Total)
GPU	TC::GPU load(_Total)\3D TC::GPU load(_Total)\Video Decode TC::GPU load(_Total)\Video Processing TC::GPU frame buffer(_Total)

Relevant RDSH/AVD/Win365 Perf Counters

\Terminal Services\Active Sessions	Number of active sessions
\Terminal Services\Total Sessions	Number of total sessions
\RemoteFX Graphics(*)\Graphics Compression ratio	Ratio of bytes encoded to bytes input
\RemoteFX Graphics(*)\Average Encoding Time	Average frame encoding time
\RemoteFX Graphics(*)\Frame Quality	Quality of the output frame
\RemoteFX Graphics(*)\Input Frames/second	Number of sources frames
\RemoteFX Graphics(*)\Output Frames/second	Number of frames sent to the client
\RemoteFX Graphics(*)\Source Frames/second	Number of frames composed by source
\RemoteFX Graphics(*)\Frames Skipped/second – Insufficient Client Resources	
\RemoteFX Graphics(*)\Frames Skipped/second – Insufficient Network Resources	
\RemoteFX Graphics(*)\Frames Skipped/second – Insufficient Server Resources	

Relevant RDSH/AVD/Win365 Perf Counters

\RemoteFX Network(*)\Current TCP Bandwidth	TCP Bandwidth detected in bits per second (bps)
\RemoteFX Network(*)\Current TCP RTT	Average TCP round-trip time (RTT) detected in ms
\RemoteFX Network(*)\Current UDP Bandwidth	UDP Bandwidth detected in bits per second (bps)
\RemoteFX Network(*)\Current UDP RTT	Average UDP round-trip time (RTT) detected in ms
\RemoteFX Network(*)\Loss Rate	Loss percentage
\RemoteFX Network(*)\Retransmission Rate	Percentage of packets that have been retransmitted
\RemoteFX Network(*)\TCP Received Rate	Rate in bps at which data is received over TCP
\RemoteFX Network(*)\TCP Sent Rate	Rate in bps at which data is sent over TCP
\RemoteFX Network(*)\UDP Received Rate	Rate in bps at which data is received over UDP
\RemoteFX Network(*)\UDP Sent Rate	Rate in bps at which data is sent over UDP

C:\Windows\system32\cmd.exe

```
C:\Users\benny.DARCOLABS>typeperf -q processor
```

```
\processor(*)\% Processor Time  
\processor(*)\% User Time  
\processor(*)\% Privileged Time  
\processor(*)\Interrupts/sec  
\processor(*)\% DPC Time  
\processor(*)\% Interrupt Time  
\processor(*)\DPCs Queued/sec  
\processor(*)\DPC Rate  
\processor(*)\% Idle Time  
\processor(*)\% C1 Time  
\processor(*)\% C2 Time  
\processor(*)\% C3 Time  
\processor(*)\C1 Transitions/sec  
\processor(*)\C2 Transitions/sec  
\processor(*)\C3 Transitions/sec
```

The command completed successfully.

```
C:\Users\benny.DARCOLABS>typeperf "\Processor(_Total)\% Processor Time"
```

```
"(PDH-CSV 4.0)", "\\MERLIN\Processor(_Total)\% Processor Time"  
"08/22/2024 09:13:53.884", "1.019657"  
"08/22/2024 09:13:54.900", "1.991526"  
"08/22/2024 09:13:55.912", "1.445304"  
"08/22/2024 09:13:56.918", "1.629610"  
"08/22/2024 09:13:57.928", "5.375511"  
"08/22/2024 09:13:58.942", "0.992495"  
"08/22/2024 09:13:59.956", "3.530964"  
"08/22/2024 09:14:00.957", "3.460074"  
"08/22/2024 09:14:01.969", "1.571992"  
"08/22/2024 09:14:02.977", "2.472840"  
"08/22/2024 09:14:03.989", "0.286005"
```

The command completed successfully.

```
C:\Users\benny.DARCOLABS>
```

PowerShell

Performance Counters

- `Get-Counter -ListSet "Processor"`
- `(Get-Counter -ListSet "Processor").Paths`
- `(Get-Counter -ListSet "Processor").PathsWithInstances`
- `Get-Counter -Counter "\Processor(_Total)\% Processor Time" -SampleInterval 2 -MaxSamples 3`
- `$CounterList = "\Processor(_Total)\% Processor Time", "\System\Processor Queue Length", "\Memory\Available MBytes", "\Process(_Total)\Working Set", "\PhysicalDisk(_Total)\Disk Read Bytes/sec", "\PhysicalDisk(_Total)\Disk Write Bytes/sec", "\PhysicalDisk(_Total)\Disk Transfers/sec", "\PhysicalDisk(_Total)\Current Disk Queue Length", "\System\Context Switches/sec", "\System\Processes"`
- `Get-Counter -Counter $CounterList -SampleInterval 1 -MaxSamples 20`
- <https://learn.microsoft.com/en-us/powershell/module/microsoft.powershell.diagnostics/get-counter>

Dealing With Localized Counter Names

- The most severe limitation of Get-Counter are the localized counter names
- There are two API functions you can use to convert localized counter names to id numbers and vice versa
 - Get-PerformanceCounterId takes a localized performance counter name and translates it to a language-agnostic id number
 - Get-PerformanceCounterLocalName does the opposite and translates the id number to the appropriate local name

<https://powershell.one/tricks/performance/performance-counters>

<https://powershellmagazine.com/2013/07/19/querying-performance-counters-from-powershell/>

Localized Perf Counter Names

The image displays three screenshots of the Windows Registry Editor, illustrating the process of localizing performance counter names. Each screenshot shows a different view of the registry tree and its corresponding data table.

Registry Editor (English)
Path: Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Perflib\CurrentLanguage

Name	Type	Data
(Default)	REG_SZ	(value not set)
Counter	REG_MULTI_SZ	1 1847 2 System 4 Memory 6 % Processor Time 10 File Read Operations/sec 12 File Write Operations/sec 14 File Co...
Help	REG_MULTI_SZ	3 The System performance object consists of counters that apply to more than one instance of a component proce...

Registrierungs-Editor (German)
Path: Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Perflib\009

Name	Typ	Daten
(Standard)	REG_SZ	(Wert nicht festgelegt)
Counter	REG_MULTI_SZ	1 1847 2 System 4 Memory 6 % Processor Time 10 File Read Operations/sec 12 File Write Operations/sec 14 File Co...
Help	REG_MULTI_SZ	3 The System performance object consists of counters that apply to more than one instance of a component proc...

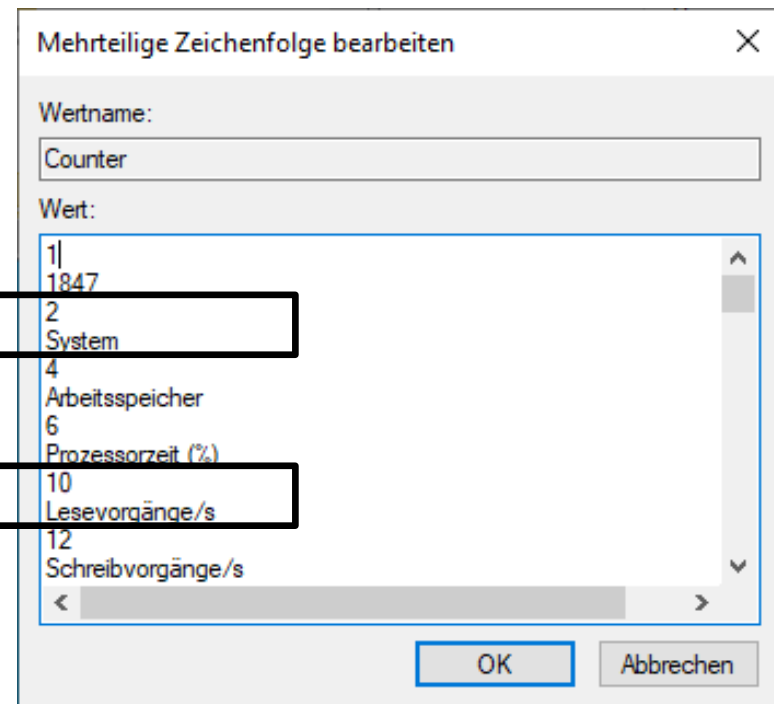
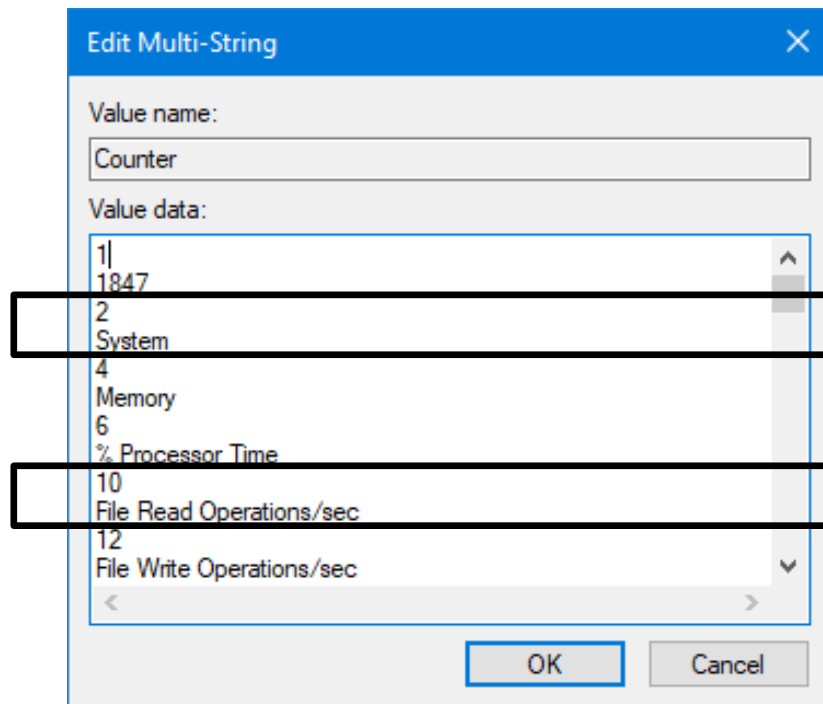
Registrierungs-Editor (German)
Path: Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Perflib\CurrentLanguage

Name	Typ	Daten
(Standard)	REG_SZ	(Wert nicht festgelegt)
Counter	REG_MULTI_SZ	1 1847 2 System 4 Arbeitsspeicher 6 Prozessorzeit (%) 10 Lesevorgänge/s 12 Schreibvorgänge/s 14 Dateisteuervorg...
Help	REG_MULTI_SZ	3 Das System-Leistungsindikatorenobjekt besteht aus Leistungsindikatoren, die für mehrere Instanzen eines Komp...

Localized Perf Counter Names

Perflib\CurrentLanguage\Counter
on a system with English (United States) language

Perflib\CurrentLanguage\Counter
on a system with German language



Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Perflib\CurrentLanguage

Localized Perf Counter Names

```
PS F:\> Get-Counter -Counter "\System\File Read Operations/sec"
```

Timestamp	CounterSamples
7/30/2020 2:24:42 PM	\\host1\system\file read operations/sec : 359.52894517603

```
PS F:\> Get-Counter -Counter "\2\10"
```

Timestamp	CounterSamples
7/30/2020 2:24:51 PM	\\host1\2\10 : 341.618616976008

- Unfortunately, many “localized” counter IDs do not work with Get-Counter
- And the decimal separator – period (.) or a comma (,) – depends on the locale...
- ...as does the 1000 separator (expect big fun with CSV files)

What Perf Counters are missing?

- GPU Performance Counters
 - 3D (rendering)
 - Video Decode
 - Video Encode (NVIDIA) / Video Processing (AMD, INTEL)
 - Dedicated Memory (Frame Buffer)
- Network Performance Counters
 - \Network Interface(_Total)\Bytes Received/sec
 - \Network Interface(_Total)\Bytes Sent/sec
- User Session ID – Auto-Detect

Conclusions



Conclusions

- Side-by-Side RDP is the new mainstream, classic RDP is legacy
- If remoting clients exhibit screen artifacts, AI may help
- RDP bandwidth consumption depends on remote desktop size and visibility on the endpoint device
- Noisy neighbors on multi-user AVD increase app response time
- Localized RDP performance counters can break TypePerf or PowerShell monitoring

Call to Action

If you want to learn more about
EUC Score, send me an email

info@eucscore.com



<https://eucscore.com>

NOTE: The EUC Score toolset is free for
community benchmarking tests when the
results are made freely available to the public



EUC Score Links

<https://eucscore.com>



Home Page

<https://eucscore.com/freeware>



Freeware Download

- Blog articles: <https://drtritsch.com>
- Toolset documentation: <https://docs.eucscore.com>
- Test Methodology: <https://eucscore.com/methodology.html>
- Simload Gallery: <https://eucscore.com/gallery.html>
- Test Results (Sync Player): <https://eucscore.com/results>
- Terminology (Glossary): <https://eucscore.com/terminology.html>
- Lab Equipment: <https://eucscore.com/equipment.html>

Bedankt voor mij

Tack för mig

Thank you

میرے لئے شکریہ

Tak for mig

Takk fyrir mig

Takk for meg

Danke für mich

Gracias por mi

Merci pour moi

Kiitos minulle