Performance and Implementation for Large Scale Environments

1. Introduction

From beginning, Green Screens Terminal Server is built with performance, large scale and cloud support as main focus. However, there are elements which might affect product performance. This document is made to explain key internal features, performance affecting elements, base measures and recommendations.

2. Product internal key features

To create a real web based 5250 terminal, two key features are required: full asynchronous communication and EBCIDIC to UNICODE conversion.

Web based terminals use one of 4 base techniques to convert 5250 data into format compatible for browser representation which can be used by one of the two techniques:

- **one-direction** – http request-response; not all terminal features available, requires additional round-trips for every req./res. Increasing response latency. Performance is significantly affected when SSL/TLS is used as number of round-trips increase reducing response time.

- **full-duplex** – WebSocket, all terminal features available, no additional round-trips, operates over existing channel.

2.1. 5250 to HTML rendering

Oldest technique. Generate html page at the server side and deliver html page representing terminal UI. Average data size is **25KB / req**.

2.2. 5250 to JSON

Technique used to reduce amount of data that is sent through network and to reduce performance requirements at the server side. Average amount of data is **8KB/req**.

2.3. 5250 to binary - WebSocket

Latest technique which has the same operating mode as 2.3 except using binary data transfers instead of JSON format additionally reducing data transfer size from 8KB/req. to avg. **1.2KB/req.** No other product using this protocol.

2.4. Green Screens Product

Green Screen Terminal server is the only product on the market using 2.3. combined with fast compression reducing data transport size to average **650byte/req.** which is 2.5x smaller than original terminal data stream.

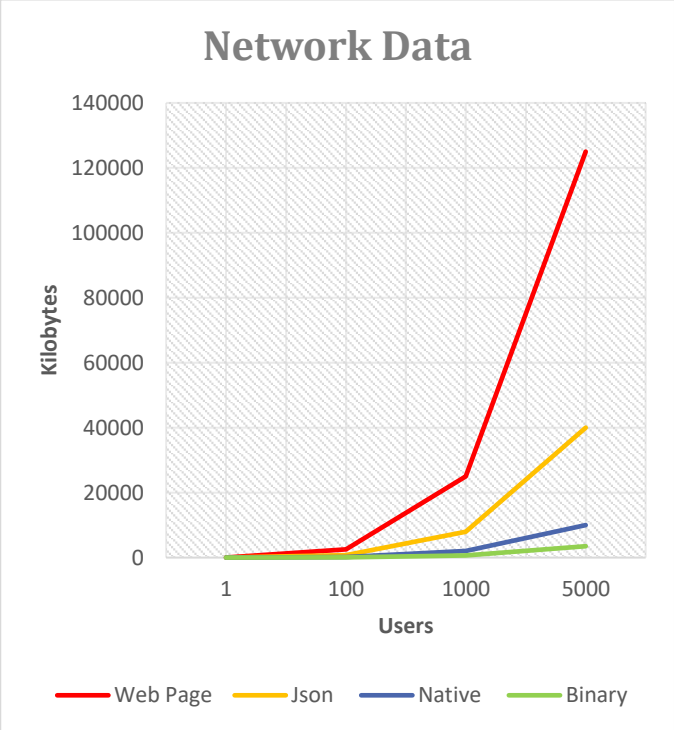
Compression algorithm used is optimized for server-side environment allowing compression rate of 450MB/sec.

*Green Screens Terminal server **does not use server-side UI rendering** or JSON encoding for terminal data. Binary data is sent to the browser and processed by the browser to generate UI within workstation.*

Table and graph below show data transfer rates and network bandwidth requirements based on used techniques. Green Screen Terminal server is the only product using Binary data format shown in the last column.

Users	Html	Json	Native	Binary
1	25KB	8KB	2KB	650Byte
100	2,5MB	800KB	200KB	65KB
1000	25MB	8MB	2MB	650KB
5000	125MB	40MB	10MB	3,25MB

**Data for HTML/JSON might be GZ-Compressed reducing size by 40-60%*



What needs to be taken into consideration is that Green Screens Service sits between workstations and IBM I server handling 2-sides. For every transport type, native terminal data size must be taken into calculation.

For 5000 workstations, this will add ~10MB for native 5250 protocol and 3.5MB for Green Screens Binary format in total for data processing.

2.6. Internal processing

Green Screen Server use event-based networking. That means, server use limited calculated shared number of processing threads based on available processor physical threads instead of creating a thread for every user instance.

This significantly reduce amount of memory usage and increase throughput as processor is doing significantly less memory copy operations by sharing one thread across multiple user requests.

3. Performance affecting elements

In this chapter we will describe elements that influence product performance.

3.1. Java Virtual Machine

Even Green Screens Terminal server is written for compatibility with Java 8, installer for MS Windows is distributed with embedded Open Java JDK12. Linux default images for Docker and non-Docker environment are prepared with Open Java JDK12.

As product is written in Java, for excellent platform supportability, some customers might prefer using existing older versions of Java JVM which it is not recommended.

From Java 11 and newer, new encryption algorithms for TLS are available as ECDSA, Chacha and Poly. Improved internal Java System as pauseless garbage collector and performance and memory efficiency optimizations.

If Java older than version 11 is used, some of advanced features will not be available within Green Screen Server reducing performance, security and stability.

It is important to note that Java 8 build older than 246 is not supported due to the missing functionalities in older Java builds.

3.2. Network Bandwidth

Green Screens Server use in avg. **650 byte / req.** which makes this product best in the class bringing fast response even over mobile or satellite networks.

All network devices have MTU (Maximal transaction unit) – size of data packet where average is 1500bytes. Compared to other formats, Green Screens Data is 99.9% below 1200 bytes, 90% below 1000bytes, 70% below 750bytes, 20%below 500bytes.

Every transaction is sent with a single network packet only compared to other solutions. Native Terminal data is usually sent in 2 packets for 24x80 and 4 packets for 27x132 screen size. Other data formats require even more packets.

As TCP is reliable networking, requiring feedback transmission acknowledge signal before sending next packet, data transmission time is in average 3x longer than Green Screens Data format. For 8KB JSON, it is 4-7x longer and for HTML format it is 10-20x longer depending if compression is used or not. In real numbers it means 5ms vs 20ms vs 40ms vs 70ms response times in average.

Some network appliances have smaller MTU, additional data splits might happen further reducing network transport times. As Green Screens Data is smaller than average MTU, it is very unlikely data split will happen.

Network packets might be lost (which is common) and in some networks, network loss goes up to the 8%. This requires retransmissions, additionally increasing response times. If that happens, retransmission requirements for Green Screens Data format are less common due to the FEC.

FEC (Forward Error Correction) is a redundant data allowing network equipment to recover damaged packets. Similar principle as QR-Code works. As Green Screens Data is half the MTU size in average, network appliance might use higher FEC redundancy levels within single packet reducing retransmissions to the minimum in a case of data error.

For large amounts of workstations, and increased peak usage, network QoS setup, other data transfers, file downloads might affect network latency and reduce response time also.

3.3. Network Encryption

Green Screen Server when used in internal network infrastructure, does not require TLS as all data is encrypted and sent through WebSocket natively. Removing the need for TLS in internal network positively affect server-side performance and response times. However, some features as WebAuthn and biometric security requires TLS. When accessing through Internet, TLS encryption is mandatory where performance and processor usage depends on TLS version used.

Green Screens Server might be used directly with provided TLS 1.2 support and future TLS1.3 which is significantly faster by avg. 30%, or through proxy / load balancer SW/HW which supports TLS1.3.

By default, http2 is enabled in Green Screens Server (available only through TLS) which improves resource transports between browser and Green Screen Server.

If Green Screens Server implemented TLS is used, it is recommended to use at least Java 11, as it brings faster ECSDA, Chacha and Poly algorithms.

As Green Screen Server sits between workstations and IBM I, only one side (browser to green screen server) is required. If encryption is used on both sides, performance will be reduced by half as processor must handle TLS channels on both sides.

3.4. Network elements locations

Location of network elements, as location of Green Screen Server, workstations and IBMi affect network response latency. If there are connected offices / branches, additional influence in response times might be affected by public networks.

3.5. Operating system

Performance on non-GUI environments are significantly better due to the fact that processor does not have to spend processing power for handling GUI messaging and processing.

Creating minimal Linux distro with only kernel and base services might use only 200MB of RAM for OS compared to 1-2GB of ram required for GUI based OS, which also influence requirements for processing power.

Additionally, some OS specific scheduled tasks or back services might cause “processor hiccups” as Disk Optimizer, Indexing service, auto updates, etc.

3.6. Processor type

We made base server code data processing measures for terminal screen and spool to PDF generation on dual core Intel i6200U processor. We intentionally used mid-level processor made for laptops for internal testing and product optimizations.

In internal test measures, we got approx. 8000 raw transaction / second, which gives in average 4000 raw transactions / second / core for terminal data processing.

For PDF generation, we got approx. 5000 1-page PDF documents which gives average of 2500 documents/core.

Those numbers show how mid-level processor can operate as a Green Screens Server and handle large number of workstations. Today Intel servers came in configurations from 8 - 24 cores, 16-48 threads which might produce more than 200.000 transactions.

3.7. Shared server usage

Green Screen Terminal service performance might be affected if it is used on server shared with other services as MS SQL server or RDP based thin clients. This might significantly affect processing power and response times.

3.8. Workstation machine

Green Screen Server does not generate UI, only prepares 5250 data steam into lighter format suitable for browser side processing. Workstation computer browser will process binary data and generate terminal UI.

Modern techniques are used as WebAssembly to decode, decompress and preprocess binary data into Web UI allowing up to 20x better performance than standard JavaScript code.

Measured time on average laptop is about 25 ms compared to 8ms for native clients. However, this is not visible to the user, as human being vision response processing time is about 60ms. End user hardly can notice difference between native client and web based Green Screens Terminal.

What must be taken into consideration is that Green Screen data is hashed and compressed, while native data stream is not. Removing compression brings near native rendering performance.

However, network latency, and background services as disk optimizer, file indexer, automatic updates and other background running programs might affect browser performance and thus times on rendering UI.

Browser used are also important, all modern browsers based on WebKit engine will have excellent performance, second with similar performance is Firefox and non WebKit Edge, while Internet Explorer shows performance degradation from 50% to 200%.

3.9. IBM I Server usage

IBM i, network response time might be affected by IBM I system if it is overloaded with batch jobs or if end user or service program is executing complex unoptimized analytical SQL etc.

3.10. Workstation usage estimates

To estimate usage, average terminal usage and peak terminal usage measurements are used. Average parallel terminal usage is about 20%. For 1000 workstations it makes 200 parallel simultaneous requests. Based on such numbers, usage and server processing requirements can be extrapolated.

3.11. Double shifts

If a company with large scale number of workstations use double shifts, it is expected that maximum number of workstations will be 50% of total number of workstations. In organization of 5000 workstation this makes 2500 workstation per shift or based on 3.10; average of 20% of total requests will give 500 simultaneous requests.

3.12. Time zone

In a large organization working across multiple time-zones it is expected that workstation usage will overlap. How much overall average usage will increase depends on number of people working in different time-zones overlaps during work-time.

3.13. Cluster mode

Green Screen Server is licensed per IBM I machine / user. Licensing does not limit number of Green Screen Server instances used. In increased number of workstations, it is always possible to split workload between multiple instances. Thus, for an example, 5000 workstations can be spread across 5 Green Screens Servers leaving 1000 workstations per Green Screen Server instance.

Based on 3.10. this gives approx. 200 simultaneous requests / Green Screens Server instance.

Available options are to allow regions / branches or time-zones to use different Green Screen Servers directly or to use single entry point and redirect users to different servers through load-balancer SW or HW equipment.

3.14. Cluster Node Sync

When running multiple Green Screens Servers (nodes) in clustered configuration, data between nodes must be synchronized.

All secondary nodes will synchronize with master node by exchanging live workstation statistics, initial config and node workstations data.

Periodical workstations sync is insignificant. While initial sync transfer is in range of 0.5MB for configuration data, list of all live workstations depends on number of live sessions.

In environments with 5000 workstations, we measured about 6000 workstations can be synchronized in less than 1sec. During initial sync of 0.5MB of initial data, Green Screens Server will be paused and reloaded after the sync.

3.15. VPN Broker

Green Screen Server have VPN broker implemented supporting remote Green Screens VPN service. This allows to run Green Screens cloud instance while IBMi is located in different network hidden behind a firewall. Opening VPN tunnel is secured with TLS 1.3 secured layer.

Even implemented broker can handle thousands of requests, it has an impact on performance and memory usage so this should be taken into consideration.

Using HW VPN appliance also affect any network latency. Impact is significantly lower than in other products due to extremely small bandwidth requirements for Green Screens Data stream format.

3.16. Virtualization

Running in Virtualization environments as Docker, VMWare, etc. brings excellent resource usage distribution and control between different containers, but have a network latency and transfer rate impact. As Green Screens peek data transfer rate for 5000 workstations is in average 3.5MB compared to other solutions, impact is not significant compared to containerization benefits.

3.17. UI Modernization

Modernization engine for 5250 terminal screens are based on HTML templates and browser UI library. All UI processing is made at browser side, Green Screen Server is only responsible to deliver HTML and JavaScript resources to the browser. Slowdowns might happen on workstation side for the reasons as described in 3.8.

3.18. Disk influence

Disk usage does not affect Green Screen Server usage.

4. Conclusion

By presented numbers, statistics, and average expected parallel usage, single mid-range server will satisfy 5000 workstations for terminal usage.

Important segment that must be taken into account is average parallel usage and peak usage. Those numbers are usually much smaller than total number of workstations.

Calculating this number with TLS encryption used or not, network bandwidth and workstation performance it is possible to extrapolate avg. HW requirements.

However, resource requirements depend on many factors as spool2pdf service intensity usage, file transfers used, OS used, TLS and algorithm type used, JVM version, clustering, VPN etc.

Green Screens Server allows running multiple instances to spread load between instances without additional licensing cost which opens possibility to use 1 high-end server and one mid-range backup server or multiple small servers to distribute load.

Benefit of multiple small servers removes the need for a backup server as there are more nodes in use, but requires fast load balancer in front or different branches, regions can access to different set of servers.

More important than performance is a question about backup system or multi-node setup choice.

Only way to find an optimal configuration in large environment is to actively monitor and measure all affecting factors during the extended time period.

5. Recommendations

Please note that recommendations given here are for information only.

- Use of non-GUI env. with minimal image setup
- Use multiple small nodes to spread load
- If TLS used, use TLS1.2 or TLS1.3 with load balancer
- Internal network TLS not required (no WebAuthn)
- Use modern or WebKit based browsers
- Run GS Server with Java 12 or newer
- Use dedicated server
- If possible, avoid VPN

As described in 3.2. Network bandwidth, Green Screens handle two sides of the network: browser -> GS Server and GS->IBM i. While GS to browser data is highly optimized, IBM I data to GS is not causing multiple packet transmissions. Keeping Green Screens Server closer to the IBM I in the same network will decrease network latency.

Use implemented VPN Broker only if it is the only possibility or reduce only for special requirements.