Get predictable infrastructure performance

with JAR:Emulate

The risk of business interruption and pressure to increase speed to market in today's digital world means network and application failure is not an option. They need to work first time, every time, all the time.

The only way to get predictable infrastructure performance is to test before deployment. With JAR:Emulate you can simulate networks and emulate the real world conditions under which applications and platforms need to perform. Test specific scenarios and gain insight and data about where and when problems arise to identify effective resolutions before they become live issues.

Flexibility

Every business network and project requirement is different depending on factors such as location, connectivity, quality of service, number of services and the applications being used across it.

Reaching beyond simple delay, jitter and bandwidth emulation is essential for any business.

Real-life networks are a complex system of changing conditions, flexible routes and competing applications. JAR:Emulate provides the industry leading flexibility in building and modelling these complex real-life systems.

Whether you want to emulate point to point links, simulate complex data-centers with multiple gateways or extract out analytical information on video systems, with JAR:Emulate you can build your network in seconds.

Visual Design Process

Using a visual design process to build emulations, you can take control by linking together impairments in the order you require whilst directly controlling the flow of packets around your simulated network.

Available with multiple hardware options for 1Gbe and 10Gbe:

- Up-to 12 x 10Gbe or 24 x 1Gbe ports
- Flexible interfaces; RJ-45 or SFP+ (or a mixture)
- Flow packets between any port, no limitations.
- Up-to 1TB of storage for wire-rate recording
- Large delay capability up-to 30 seconds @ 10Gbe
- Fast visual emulation design creates networks in seconds.

JAR:Emulate Model: 1 G and 10 G

- **Line-rate 1Gbe and 10Gbe performance**
- **Up-to 24 ports (1Gbe) or 12 ports (10Gbe)**
- **Connect any-port to any-port, without limitations**
- **Delay up-to 3 seconds at 10Gbe (optional x10 increase available)**
- **Over 45+ impairments and tools**
- **Audio and Video modification and corruption**
- **Packet fragmentation and reordering**
- **Includes generic full packet inspection and filtering**
- **Packet modification, change any bits or bytes of any packet**
- **Load generators**
- **Network simulation - Create complex multi-hop networks using Virtual Routers for cloud, datacentre, customer and WAN simulation.**
- **Multi-user ready**

JAR:emulate
Network Emulation

Flexible Graphical User Interface

Our user interfaces run on Windows, Linux or Mac and provide a fast and responsive environment for you to build emulations.   The flexible UI enables you to drag and drop from the extensive list of impairments into your network to create specific scenarios to meet your testing needs.

Build network simulations through a visual design process.

Use JAR:Emulate to:

Trouble Shoot:
Investigate reported problems and test resolutions without disrupting production traffic.

Design Networks:
'What if' scenario building to evaluate network changes - evaluate different topologies and technologies before expensive implementation. For example evaluate WAN Acceleration Technology: Determine how different WAN optimisation products will perform under the best/average/worst conditions of your network.

Test Application Performance:
Determine how your software will perform for the end user on their network before deployment; trouble shoot and resolve issues before the need for re-work, release delay or failure.

Optimize Performance:
Adjust application and software settings under replicated network characteristics to optimize performance for different user groups.

JAR:emulate
Network Emulation

Features

Wire Rate

All core models provide full wire-rate but we understand that you might need 10Gb/s ports but only want to put 1Gb/s through the emulator. JAR:Emulate can be tailored to your individual throughput needs.

"Any Port to Any Port"™ capabilities.

JAR:Emulate ship with true "Any Port to Any Port"™ capabilities. Whether you are using 4, 8 or 24 ports they will all communicate with each other, without limitation.

Others emulate products may state you are buying a '4 port emulator' but often what you actually get is 2 'emulators' each with 2 ports inside a single box. There may be 4 ports on the outside, but what you have got is 2 independent emulators and you won't be able to send packets between each. So effectively you have got 2 emulators with 2 ports, not the 4 ports you need.

Fragmentation and Reordering

When packets cross network boundaries they can become fragmented, JAR:Emulate can fragment packets to simulate MTU changes down to 64 bytes. Packets can also be reordered either by displacing the packets "X" number of frames or time into the future.

Automation Ready

We understand that everyone's automation environments are different. JAR:Emulate can be controlled externally through a network of methods. You are provided with a CLI application (Windows, Linux or Mac) for controlling execution from the command line. Whilst developers can directly issue XML requests to the emulator to gain deeper and full control of its operation.

Load Generation

Very few networks are silent; there is always a certain amount of background traffic. Whether it is file servers, printers or other devices generating traffic, it is important to model this correctly.

JAR:Emulate provides a number of ways of generating this traffic including background traffic generation (contention), TCP load generators, PCAP file replay, etc.

Switch over and back-up circuits

Built on top of our fantastic filtering system, you can easily simulate multiple paths in any network and switch between them in real-time (or using an external trigger). This allows you to model a "good" and "poor" network and switch between them in real-time, furthermore you can create an unlimited number of paths.

Video and Audio

Dedicated impairments allow smart degradation of video (H.264), audio streaming, RTP, RTCP and much more. Understand how robust your AV equipment is to corruption of video or audio framing and view analytics information on video format internals.

Impairments

JAR:Emulate provides all the common impairments including delay, jitter, throttle, etc. Furthermore it also provides advanced impairments including fragmentation, video corruption, BER corruption, reordering and much more!

Comprehensive settings

Not only do we include 55+ impairments with our emulators, but each one of those impairments has extensive options and settings to uniquely control their operation. This gives you a unparalleled level of fine tuning; such as buffer and burst management on bandwidth throttles or affecting only B-Frames in H.264 video streams.

Filtering

It's very important to be able to identify and target certain streams of data on an emulated network. JAR:Emulate provides a host of built-in filters to identify packets for impairment or analysis.

Furthermore you can join filters together to create more complex filters (for example, you could find HTTP traffic going to Port 8080 on VLAN ID 90 with only 2 filters).

Flexible UI with Wizards and Easy Licensing

Our user interfaces run on Windows, Linux or Mac and provides a fast and responsive environment for you to build emulations. We've included some fantastic wizards to quickly create example networks and don't lock you down to restrictive licensing (install it on as many machines as you want!).

Virtual Routers

Routers effectively sit at the edge of most networks; they provide your ADSL connection in your business or accessibility to a cloud server through an ISP backbone.

JAR:Emulate contains full Virtual Router simulation, providing a method for simulating multi-hop WAN networks. These Virtual Routers provide DHCP or Static IP address assignment, automatic and manual routing table and multiple virtual interfaces. They will respond to both local and WAN pings and provide detailed analysis on packets received.

They also include OSPF to allow for the simulation of complex ring, double ring and double star topology networks with the ability to self-heal when injected faults are detected.

Technical Specification

	PRODUCT RANGE	
	1 G	**10 G**
Physical		
Built in Network Interfaces	4	2
Additional Network Interfaces	up to 24	up to 12
Standard Network Interfaces	GbE Copper	SFP+
Optional Network Interfaces	SFP+	SFP+
Max.packet Rate Per Port	11.84 million	59.2 million
Full wire rate (64 byte packets)	✓	✓
Storage Capacity (Optional Max)	8GB (64GB)	8GB (1TB)
Dual Redundant Power Supplies	◉	◉
Dimensions	2u Rack	2u Rack
General		
Timing Precision	0.01ms	0.01ms
Any Port to Any Port (tm) 4, 8, 24 ports? Packets can be sent between each and any port for complete flexibility	✓	✓
Live Changes Real-time modification of any impairment setting or network map	✓	✓
Maximum Impairments * Until performance is exhausted	Unlimited*	Unlimited*
Traffic Capture & Replay with Looping Option	✓	✓
Bi-directional, Independent Emulations	✓	✓
Timeline Schedule changes to emulation settings with no manual intervention required Option: Loop timeline for continuous playback	✓	✓
Modes of Operation		
Virtual Routing (Simulate routers / ADSL Gateways) Protocols: DHCP, ARP, ICMP, IGMP, etc DHCP: Enable / Disable Routing Table: Auto generate, manual Multiple WAN Links	✓	✓
Cloud / Data Center Simulation Simulate multi-hop routed networks with ease OSPF Support for self healing and Internet simulation	✓	✓
Bridged Mode Option to impair all traffic sent and received (protocol filtering available)	✓	✓
Delay Emulation (0ms to 30secs, 0.01ms)		
Fixed Latency	✓	✓
Variable Latency	✓	✓
Ramp	✓	✓
Normal / Gaussian	✓	✓
Sinusoidal Wave	✓	✓
Jitter 0.1 ms to 100 ms or 0.1 to 100% of constant delay	✓	✓
Timing Constraints (Specify start and duration of impairments activity) Start / Duration 0.1 ms to 360,000 ms (in 0.1 ms increments)	✓	✓
Bandwidth Emulation		
Constant Throttle	500 Bps to 1 Gbps	500 Bps to 1 Gbps
Random Range (min to max with time constraints)	501 Bps to 1 Gbps	501 Bps to 1 Gbps
Random Range Duration 0.1 ms to 10 minutes* (in 0.1ms increments)	✓	✓

Use JAR:Emulate to accurately simulate:

- VoIP
- Microwave
- T1
- WiFi
- Satellite
- OC-3
- E1
- WiMAX
- RoIP
- Cloud
- 3G
- T3
- DSL
- GPRS
- Dial Up
- XDSL
- E3
- 4G
- ASDL

✓**Provided as Standard**   ◉ **Optional**
*Depending on Storage Capacity

JAR:emulate
Network Emulation

	PRODUCT RANGE	
	1 G	10 G
Background Traffic Generation		
Fixed Data Rate: 500 bps to 40Gbps Percentage of available link: 1 to 99% Generate broadcast packets Range (min to max with time constraints)	✓	✓
Range Duration 0.1 ms to 360,000 ms (in 0.1ms increments)	✓	✓
Reordering		
Time Based Re-order Displace packet from 0.1 to 500 ms	✓	✓
Position Base Re-order Displace packet up to 1,000,000 places	✓	✓
Corruption		
Bitflips Start and end position (first byte to last byte), 1 to 100%	✓	✓
Byte Overwrites Start and end position (first byte to last byte) 1 to 100%	✓	✓
Ethernet Fragmentation MTU: 64 to 9,000	✓	✓
Bit Error Rate (Per) Simulation x bits in y received (1 bit to IE-14)	✓	✓
Duplication		
Simple (single duplication) Packets received on link will be immediately duplicated once	✓	✓
Timed (duplicated every x seconds) Single duplication after specified delay (1 to 1,000 ms)	✓	✓
Complex (multiple, timed duplication) Specified multiple duplications after specified time delay (1 to 1,000 ms)	✓	✓
Loss		
Standard Drop x packets in Y recieved (1x to 99,999,999y)	✓	✓
Percentage Drop 0.1% to 100% (in increments of 0.1%)	✓	✓
Outage Drop all packets recieved on specified link	✓	✓
Drop Evenly Packets will be dropped regularly throughout emulation	✓	✓
Drops in Bursts Packets will be dropped in continuous groups	✓	✓
Timing Constraints Start / Duration 0.1 ms to 360,000 ms (in 0.1ms increments)	✓	✓
Modification		
Generic Packet Modifier Modify up to 6 bit / byte sections per packet	✓	✓
Frame Check Sequence Modification Transport, corrupt and modify the FCS	◎	◎

Emulate Real Network Conditions

- Busy peak times
- Long distance
- Jittery Connections
- Duplication of Data
- Fragmentation of data
- Traffic Re-routing
- Limited bandwidth
- Information corruption
- De-sequencing of data
- Network traffic bursts
- Partial or total outage
- Back-up Switch overs

✓**Provided as Standard**   ◎ **Optional**
*Depending on Storage Capacity

JAR:emulate
Network Emulation

Model: 1 G and 10 G

		PRODUCT RANGE	
		1 G	**10 G**
TAP (Test Access Point) devices -			
Extract analysis information from any part of the emulation			
Bandwidth Graph		✓	✓
Show bandwidth utilisation - Export, clipboard, peak and averaging, etc.			
Packet Rates		✓	✓
Show packet utilisation			
RTP Analyser		⊙	⊙
Output detailed information on RTP streams			
RTCP Analyser		⊙	⊙
Output detailed information on RTCP streams			
Reports		⊙	⊙
Create customisable reports from various metrics			
Stateless load generation with multiple load distribution models			
Generic/RAW Load Generator		⊙	⊙
Generic any type of load with extensive stream options			
TCP Client		⊙	⊙
Simulate clients with data streams			
TCP Server		⊙	⊙
Simulate servers with data streams			
RTP / VOIP		⊙	⊙
Simulate audio/video equipment			
UDP		⊙	⊙
Raw UDP traffic with payloads			
DDOS Simulation		⊙	⊙
Simulate extremely stressful DDOS environments			
Audio Visual (AV) Pack			
RTP Filter		⊙	✓
MPEG H.264 Filter		⊙	✓
MPEG H.264 Corruptor		⊙	✓
G.1050 Wizard		⊙	⊙
TIA-921 Wizard		⊙	⊙
Management			
Drag and Drop User Interface		✓	✓
Simple User Interface, allowing user to draw out their target network on screen, drop impairments as required and visualise the network under test			
Open XML for Test Automation		✓	✓
Easy integration with test enviroments to schedule, run and report on emulations without recourse to user interface			
Hardware Capacity Monitoring		✓	✓
Device load, memory usage, hardware status, load history			
Command Line Interface		✓	✓
Powerful CLI (with source code) for Windows, Linux and Mac			
SNMP		⊙	⊙
Operational based SNMP traps and alerts			
Smart Start-Up		✓	✓
Automatically launch previous map on boot			
Multi-User Support		⊙	✓
Unlimited GUI instances, share maps etc			
Hardware NTP / PTPv2 Timestamping		⊙	⊙
Lock hardware and packet timings to accurate internal / external clocks			

Product Range

The JAR:Emulate range offers a variety of WAN Emulators that can be tailored to meet your specific testing requirements.

From the "Lite" range right up to our 40,000 model, from two interfaces to twelve, in a combination of copper and/or fibre.

10,000 and 40,000 Models
The 10,000 series provide 10 Gbps fibre and copper interfaces (up to 12 ports). 40,000 series provides 40 Gbps interfaces (up to 12 ports).

Both series offer the ability to create as many independent emulations as required and includes every tool as standard.

You can have hundreds of impairments in your emulation map all executing at the same time; allowing for complex and un-matchable emulations scenarios.

Lite and Portal Units
The JAR:Emulate Lite series is a range of portable, low cost emulators.

Provide throughput speeds of 2, 10 or 45 Mbps (bi-directionally) to suit your needs and budget.

Ideal for individual developers and testers.

The small sized Lite models are desktop/hand held hardware units that provide accurate virtual network environments for those who do not require the higher throughput of our rack mountable units.

JAR:emulate
Network Emulation

Model: 1 G and 10 G

	PRODUCT RANGE	
	1 G	10 G
Filtering (UDP, TCP, Packet count)		
Maximum Filter	Unlimited*	Unlimited*
Connect multiple filters in any way to create complex filter rules		
IP Source / destination address filtering (impair specific traffic flows)	✓	✓
TCP	✓	✓
Advanced: Source and destination port filtering (including range)		
TCP Packet length filtering		
UDP	✓	✓
Advanced: Source and destination port filtering (including range)		
TCP Packet length filtering		
MAC Address	✓	✓
Src/Dst single or range		
Ethernet Payload	✓	✓
Packet Counting	✓	✓
Fail or Pass filters based on packet count or percentage		
Advanced Filtering		
Generic Filter	✓	✓
Filter on multiple bit / byte values with logic operations		
IP Protocol: Payload Type and Value	✓	✓
MPLS: MPLS Label, QoS Value, TTL value	✓	✓
VLAN: VLAN ID, User Priority	✓	✓
MPEG Video	⊙	⊙
RTP A/V	⊙	⊙
Flow Control (simulation link switching)		
Output Switcher: Up to 4 available outputs	✓	✓
Input Switcher: Up to 4 available inputs	✓	✓
Merge Tool	✓	✓
Network Analysis		
Latency Measurement	⊙	⊙
Measure live hardware and network latency to microsecond levels		
Network Conditions Monitor	✓	✓
Automatically build networks using any wireshark file		
Instancers		
(create unique impairments for hundreds of streams)		
VLAN Instancer	⊙	⊙
MPLS Instancer	⊙	⊙
Reporting		
Live Monitoring	✓	✓
Bandwidth monitoring, packets per second, export to CSV max / average values,- etc		
Wireshark Integration (on up to 200 protocols)	✓	✓
Allows for live traffic capture and root cause analysis; replay 3rd party traffic streams under impairments, record traffic and replay at a later date		
Customisable End of Emulation Reports	⊙	⊙
Reports detailing results of emulation run, such as impairment settings, dropped packets, bandwidth, corruption, jitter etc		

For more information, visit the JAR:Emulate website: www.jaremulate.com

T: +44 (0)28 9023 3322
E: info@jartechnologies.com