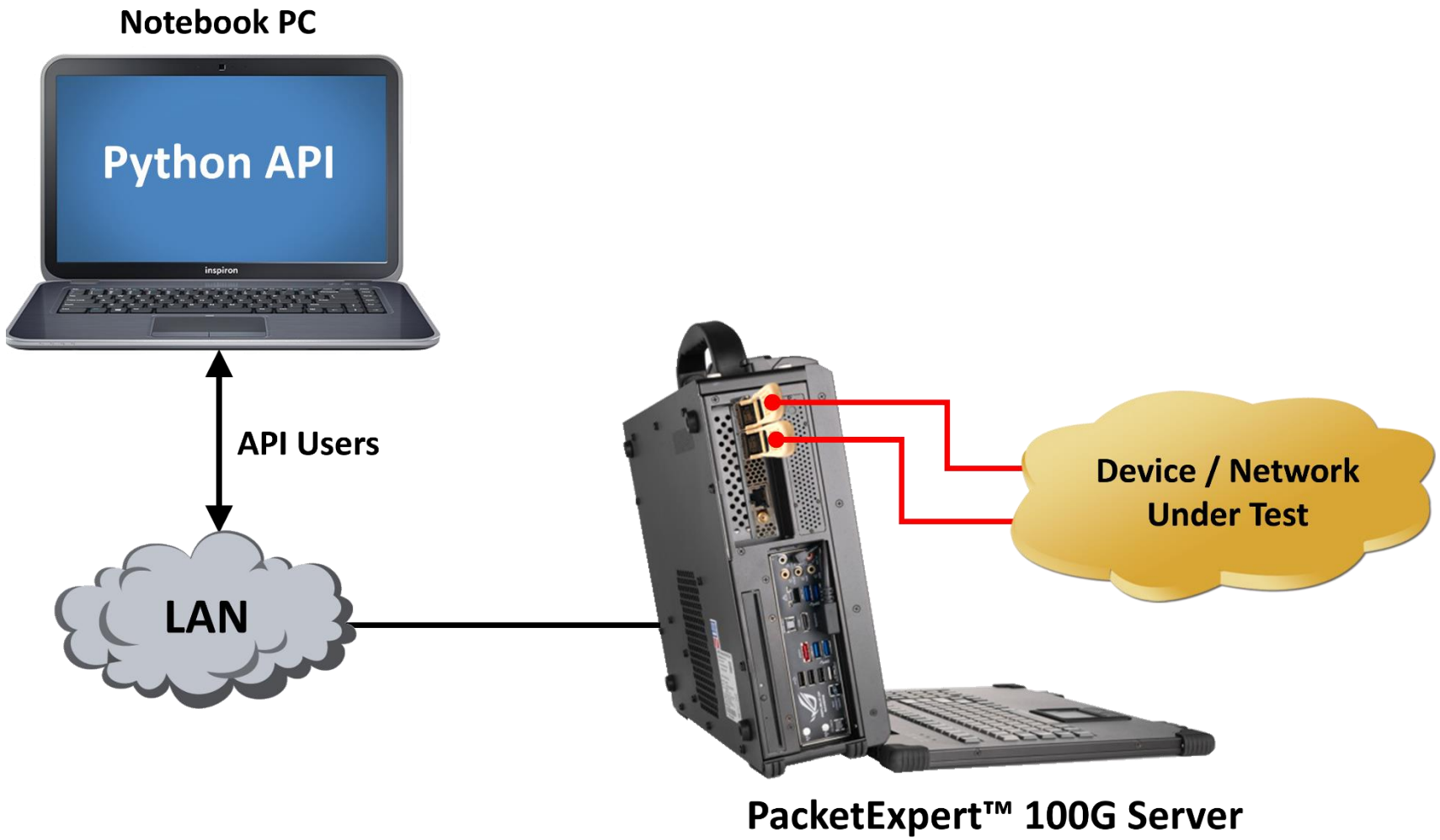
APIs for Test Automation and Remote Access

Up to 100 Gbps

GL Communications Inc.
818 West Diamond Avenue - Third Floor, Gaithersburg, MD 20878
Phone: (301) 670-4784 Fax: (301) 670-9187 Email: info@gl.com
Website: http://www.gl.com

PacketExpert™ APIs for Test Automation and Remote Access



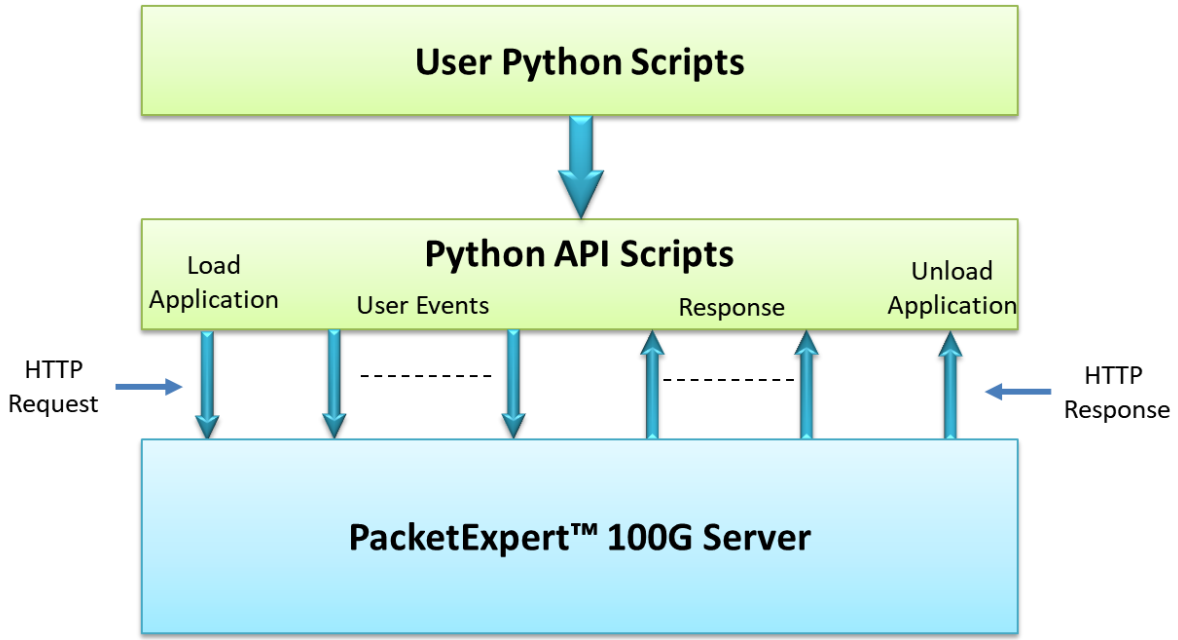
Features

- With additional licensing, PacketExpert™ 100G enables automation and regression testing through Python scripting and REST APIs
- Users can remotely access features like All Port BERT, Loopback, RFC 2544, ExpertSAM™, and Multi-Stream Traffic Generation and Analysis using a Python Client architecture
- Scripts for traffic generation at Ethernet, VLAN, MPLS, IP and UDP layers up to 100 Gbps
- Multiple PacketExpert™ 100G can be controlled remotely from single client application via PacketExpert™ 100G server

Applications

- Test and verify QoS Parameters of network devices like Switches/Routers etc.
- End to end testing of network paths for QoS parameters
- In-depth troubleshooting of the Carrier network in the event of network failures or impairments
- QoS testing of Triple-play services to ensure that they fully qualify SLA parameters
- Terrestrial wireless, satellite, and other WAN technologies network validations
- Test VoIP network in real-time conditions to verify if it meets the quality requirements before you deploy
- Testing video on IP networks by emulating the loss and congestion characteristics
- SFP support can be used for Broadband aggregation applications, Metro edge switching, Metro and access multi-service platforms, and are suitable for Fast Ethernet applications

Python Functional Modules

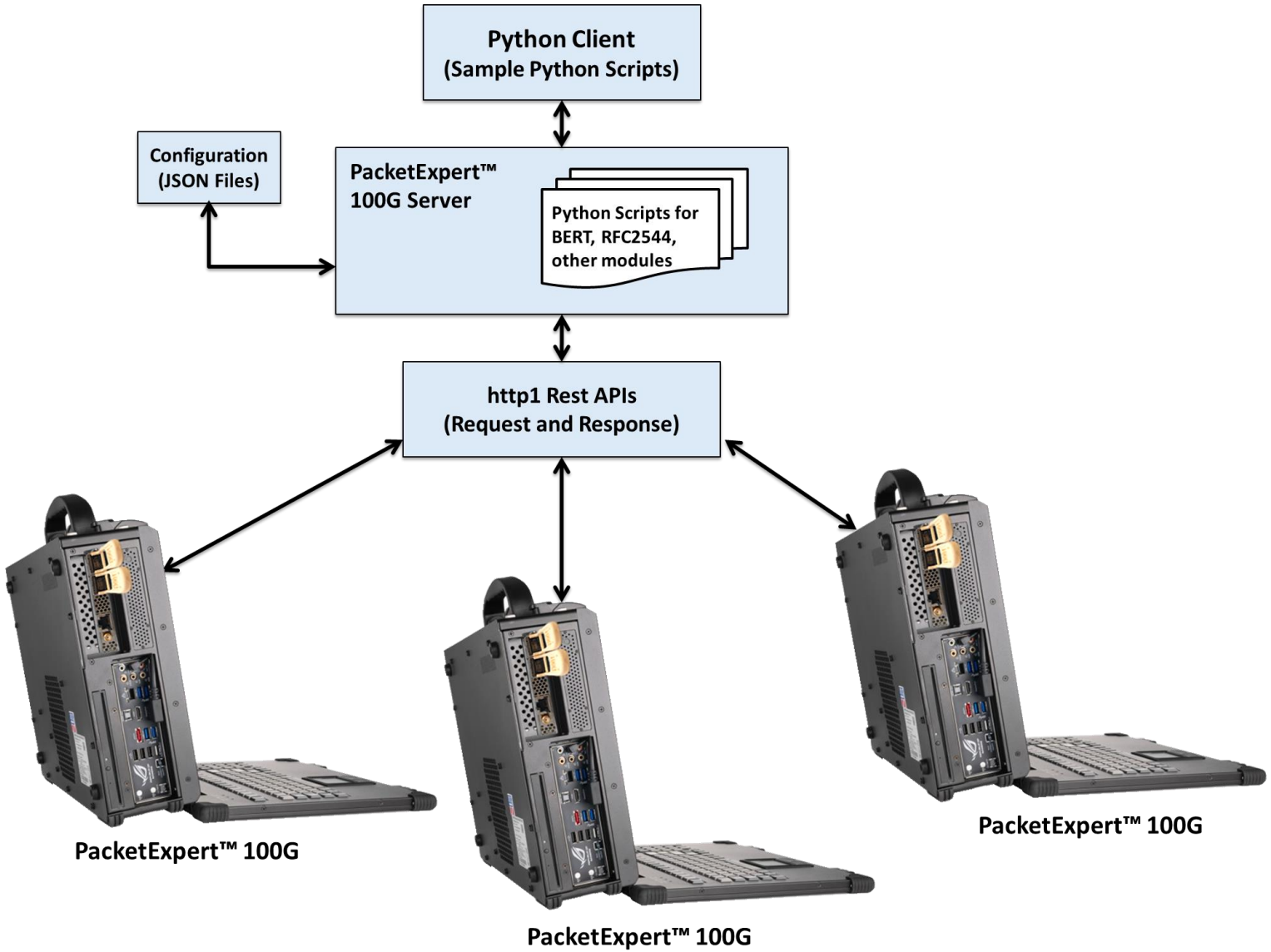


The application consists of 2 functional modules. These modules interacts with each other to perform as a single entity

- **Python API Scripts** - Acts as User Interface. The API scripts are predefined python scripts, that can be used by users to access various PacketExpert functionalities
- **PacketExpert™ 100G Server** - The PacketExpert™ 100G is a web-accessible appliance with multiple 100G, 50G, 40G, 25G, 10G and 1G ports, controlled via a REST Server on an integrated PC, enabling browser-based management through REST APIs. Users can interact with the system using standard HTTP GET/POST requests in JSON format, allowing flexible control and monitoring of the hardware devices

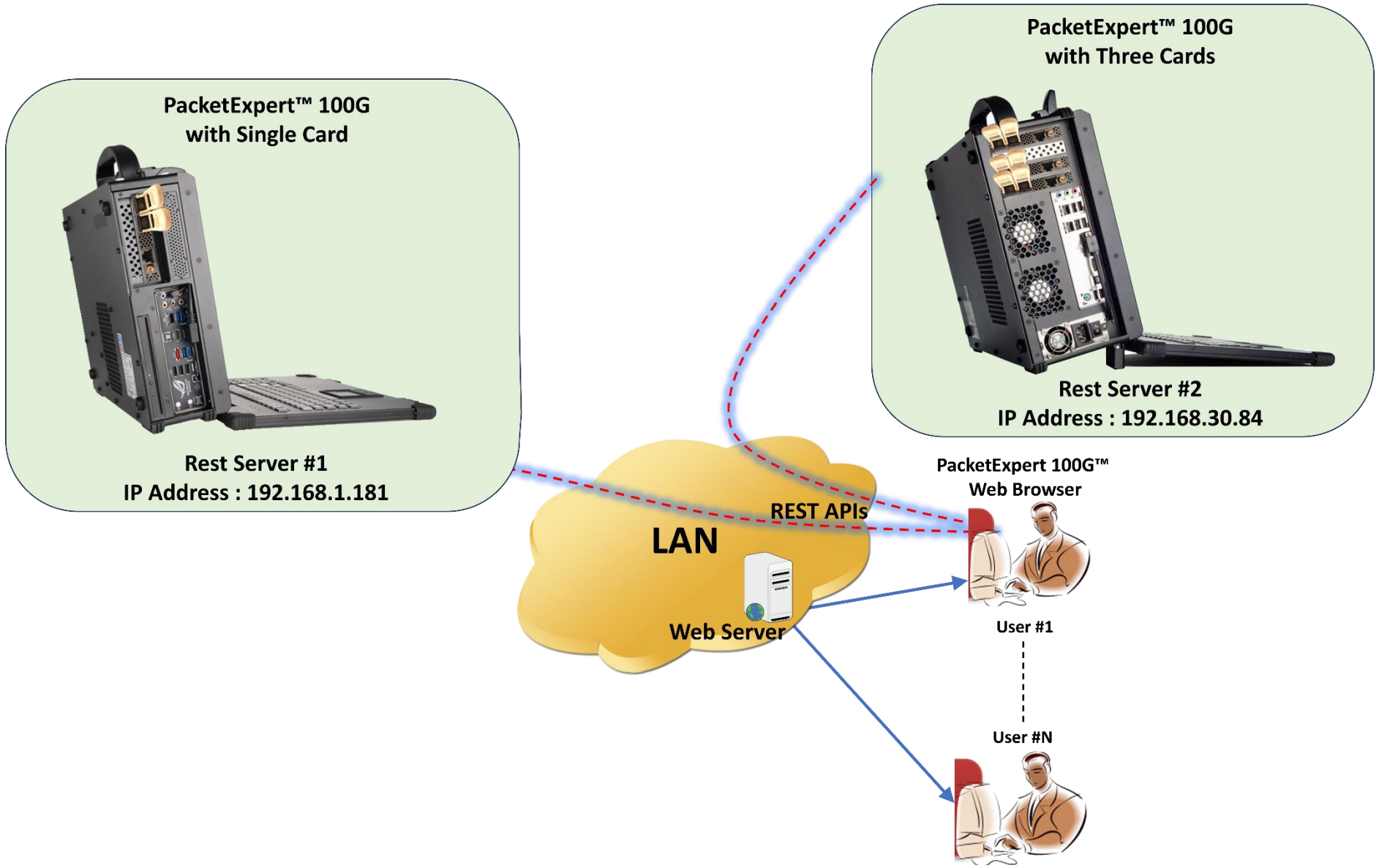
GL Communications

Working Principle of Python Client

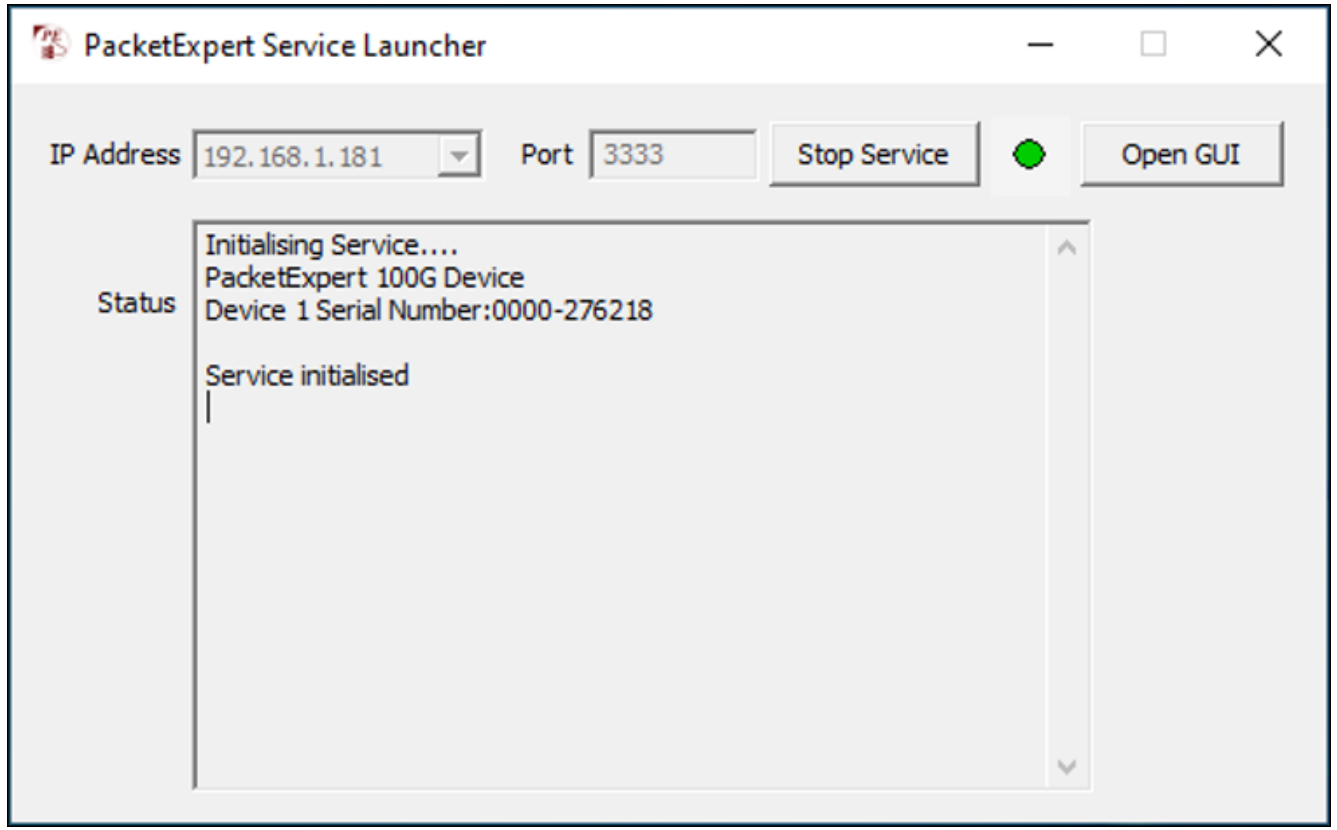


GL Communications

PacketExpert™ 100G Multi Server



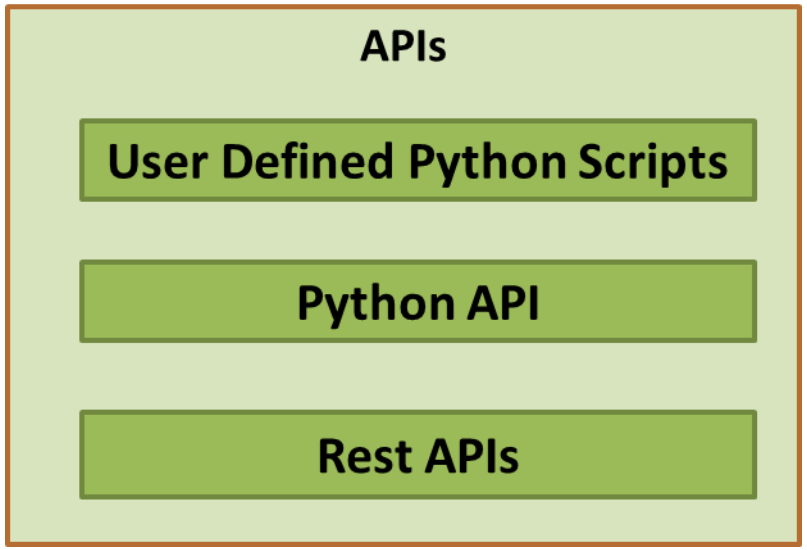
PacketExpert™ 100G Server



GL Communications

Python Client and Scripting

- The Python Client consists of following components:
- Python API scripts, that provide High Level APIs, using which all the PacketExpert™ functionalities are accessible to the users
- These APIs in turn use http1 Rest APIs to communicate with the PacketExpert™ 100G server



Python Client Sample Script and Result

AllPortBert.py

```python
from Core.Utils import *
from PacketExpertTests import *


1 usage
def main():
    # Specify server details and test configuration
    global time
    server_ip = "127.0.0.1"
    server_port = 3333
    device_list = [1]
    port_list = [1, 2]

    err, device_test_configuration = set_device_traffic_config(device_list)

    # Configure Bert Test Parameters

    device_test_configuration[1].port_mode = PortMode.6bps10  # Link Speed Selection for 100G, Set the
    device_test_configuration[1].start_error_rate = 4  # Bit error insertion rate 10^-4

    test_duration = 10

    result_file_name = "Bert_Results"

    generate_report_info = GenerateReport()
    generate_report_info.test_conducted_by = "GLIndia"
    generate_report_info.filename = "Bert_Report"
    generate_report_info.title = "All Port Bert"
    generate_report_info.init_selected_ports(device_list, port_list, AppName.AllPortBERT)

    enable_generate_report = True

    # Uncomment the following section to set default paths
    # default_json_path = 'C:\\ProgramFiles\\GLCommunicationsInc\\PXXPythonClient\\JSON\\'
    # result_file_path = 'C:\\ProgramFiles\\GLCommunicationsInc\\PXXPythonClient\\Log\\'
    # set_default_config_path(default_json_path)
    # set_default_result_path(result_file_path)
```

```
BERT Result for Device1_Port1 :

Bit error Status  : No Error
Sync loss Status : InSync
Out of sequence Status : No Error
Bit error Count  : 0
Sync loss Count : 0
Out of sequence Count : 0

BERT Result for Device1_Port2 :

Bit error Status  : No Error
Sync loss Status : InSync
Out of sequence Status : No Error
Bit error Count  : 0
Sync loss Count : 0
Out of sequence Count : 0
```

GL Communications

Hardware Specifications – Portable Platforms

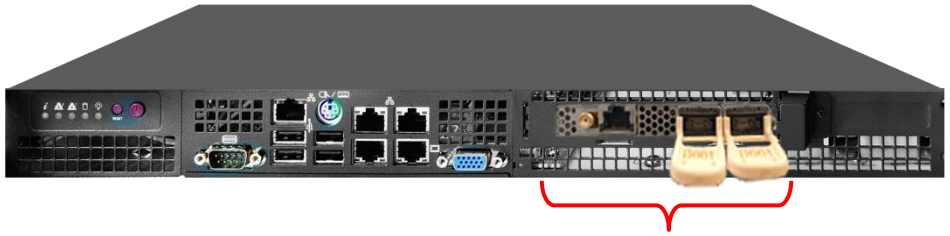
Portable PacketExpert™ 100G

Portable PacketExpert™ 100G

Portable PacketExpert™ 100G

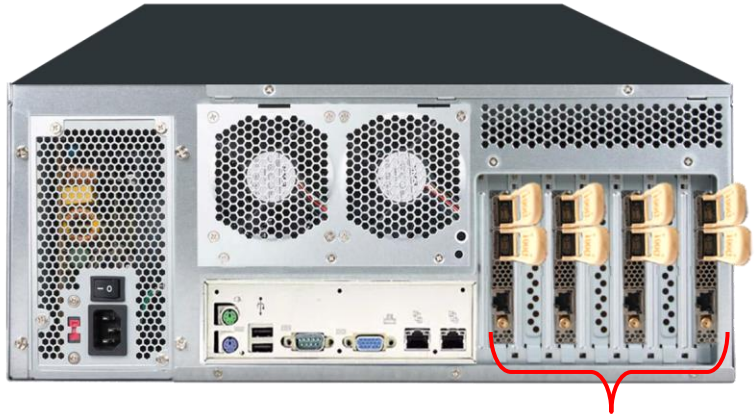
GL Communications

Hardware Specifications – Rack-mount Platforms



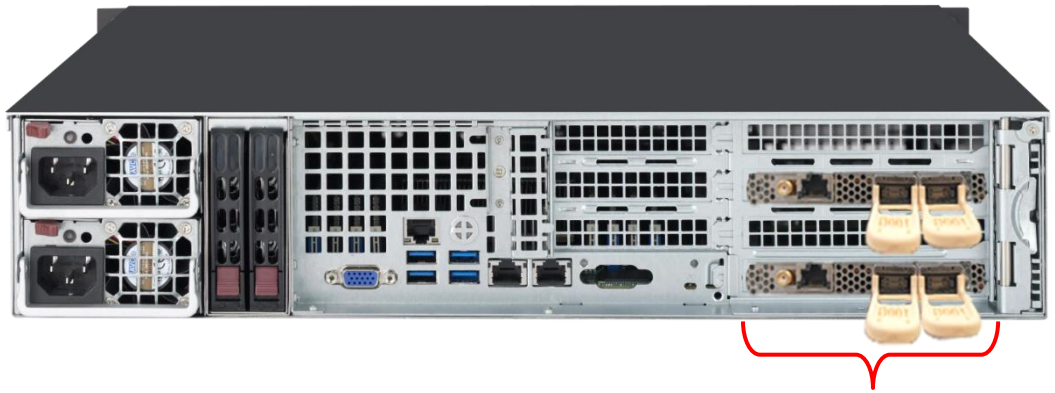
2x1G/10G/25G/40G/50G/100G

PacketExpert™ 100G – 1U Rack-mount



4x(2x1G/10G/25G/40G/50G/100G)

PacketExpert™ 100G – 4U Rack-mount



2x(2x1G/10G/25G/40G/50G/100G)

PacketExpert™ 100G – 2U Rack-mount

GL Communications

Thank You