Development of Unified Network Controller PureFlow WSX

Kenji Anzai, Koichi Ryu, Atsushi Saegusa, Takayuki Sato, Tatsuya Ono, Syoya Fujii, Junya Oda

[Summary] We have developed the Unified Network Controller PureFlow WSX NF7600 series with a high-performance TCP acceleration function and bandwidth control engine to optimize data center and carrier infrastructure. To suppress drops in TCP communications speeds caused by transmission delays over long-distance network and by decreased communications quality, in addition to developing a new TCP acceleration engine, high-speed retransmission function, and new TCP-FEC algorithm, we also added a precision bandwidth control function. This new Unified Network Controller helps eliminate long delays on global networks and improves throughput in environments, such as the Internet, with relatively high packet loss by implementing TCP acceleration supporting high rates of 10 Gbit/s for transmitting large data traffic.

1 Introduction

As data traffic generated by business activities increases, as well as establishing On Premises data management systems to analyze, manage and share so-called big data, businesses using Off Premises resources, such as Cloud services, are starting to appear.

Cloud services provide access to servers in remote data centers via networks. However, the current long-distance network communications environment suffers from the following problems.

(1) The Transmission Control Protocol (TCP) communications method suffers from decreased throughput on networks with larger delay tomes. At global long-distance communications, the presence of many repeaters in the network path causes distance-related communications delays. During TCP communications, to assure data integrity, a verification response called an ACK is sent for every received packet and the sending side waits until the ACK response is received before sending the next packet. This wait time is known as the Round Trip Time (RTT). When the RTT is long, the time until the subsequent packet is sent is delayed, resulting in a drop in the TCP communications throughput (Figure 1).

(2) Since the communications quality of overseas local network cannot be guaranteed, data integrity can be damaged by intervening communications equipment and paths, causing dropped packets. In this case, the communications speed drops because the data packets are resent after waiting a fixed time.

(3) When using narrow-band dedicated international private line, data packets may accumulate at repeaters as a result of congestion, depending on the volume of data, and the packet overflow may result in dropped packets. When packets are dropped, the sending side detects an ACK timeout and retransmits the packets, meaning that throughput cannot be assured.

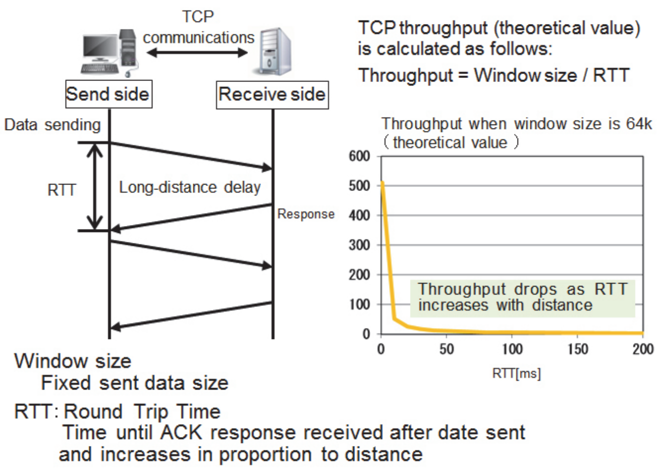

Figure 1 Delay Mechanism at TCP Communications

As a result of these problems, more time is needed for data transmissions, and delays in sharing information between offices worldwide result in decreased business efficiency. Under these circumstances, there is an increasing need to stable high-speed, long-distance communications infrastructure.

Based on our experience in developing bandwidth control equipment, we developed the Unified Network Controller PureFlow WSX NF7600 series (hereafter WSX) to solve the previously described issues related to Wide Area Network (WAN) communications (Figure 2).

Figure 2 External View of PureFlow WSX

The WSX is installed at the send and receive sides, or in other words, at both ends of the network, offering users stable high-speed communications (Figure 3).

This article explains the technologies used by the WSX and its features.

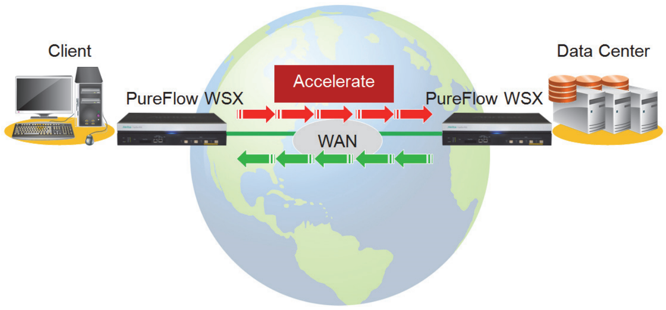

Figure 3 PureFlow WSX Operation Image

2 Design Concept

With the globalized development of business networks, the difference in network quality at each office base causes problems with lower throughput. Additionally, the trend towards increasing number of global office increases the workload for network administrators. Under these business circumstances, we examined the following points with the aim of offering an easily configured high-grade business communications environments.

2.1 TCP Acceleration Function

As described above, the conventional TCP communications protocol suffers increasing communications delay times as the distance between the send and receive sides becomes longer. We know that as the ACK is delayed by the longer RTT, the TCP communications throughput drops.

By using the WSX, a high-speed TCP tunnel is configured between WSX units at each end, stopping drops in throughput due to communications delay times and enabling fast TCP communications even over long distances.

2.2 High-Speed Congestion Control Function

The causes of reduced TCP throughput are increased delay time and dropped packets. The congestion-control algorithm used by conventional TCP decreases the send transmission speed when packets are dropped as a result of network congestion and gradually increases the transmission speed as long as no packets are dropped. As a consequence, time is required until the send speed recovers once the send speed drops, and on a network with high-packet losses, this is known to cause remarkable drops in TCP communications throughput.

To prevent drops in send speed resulting from network congestion and make best use of communications capacity, the network communications speed must be tracked in real-time using adjustment of the TCP send speed and congestion control function.

The WSX "high-speed TCP tunnel" uses a newly developed congestion-control algorithm that responds flexibly to drops in communications speed when packet loss occurs. As a result, the network's full communication performance can be used to the maximum even in environments with high packet losses.

2.3 TCP-FEC Function

As previously described, in TCP communications, throughput drops as a result of retransmission processing performed when packet loss occurs.

The WSX "high-speed TCP tunnel" uses a newly developed TCP-FEC function that appends redundancy data to TCP packets, enabling dropped packets to be recovered without the need for retransmission, thereby preventing throughput drops.

2.4 SMB Acceleration Function

The Server Message Block (SMB) is a protocol for sharing files using shared folders and network drives on Windows networks. Like TCP communications, the SMB protocol is affected by delays due to long-distance communications as the session read/write times become longer, resulting in remarkably delayed data transfer times, which must be improved to support better work efficiency.

The WSX "high-speed TCP tunnel" supports faster file read and write operations by optimizing SMB protocol command communications.

2.5 Precision Bandwidth Control

Network traffic can increase from ms to ms (called microbursts) as a result of instantaneous changes in server performance and network bandwidth. These microbursts can cause packet classes as a result of network traffic congestion and are known to cause drops in TCP performance.

Using the WSX performs high-accuracy control with ms precision of the send packet timing to achieve high-reliability communications networks and support throughput performance of 10 Gbit/s.

2.6 Coordination with Cloud Services

Administrators of cloud service networks launching new services must not only setup virtual servers and virtual networks but must also change the base physical network settings. Previously, network administrators did this work manually, but in an effort to decrease workloads, automation using Web Application Programming Interfaces (WebAPI) is increasing, so the WSX supports WebAPI to meet these needs.

Additionally, WebAPI typically uses two protocols: SOAP (Simple Object Access Protocol) and REST (Representational State Transfer). Previously, only the SOAP protocol was used as a WebAPI but current cloud services are commonly using the REST protocol. As a result, the WSX incorporates a built-in WebAPI using REST.

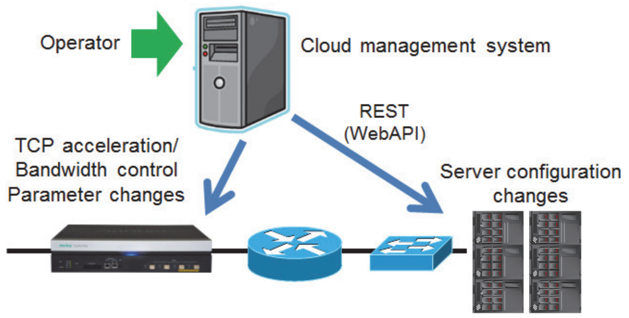

Figure 4 Cloud-Based Management System

3 Equipment Design

3.1 Equipment Configuration

Figure 5 shows the block diagram of the WSX, which is composed of equipment controller, packet processor, power supply units, and fan units. To reduce fault risks, each unit operates independently. Even in the event of a rare fault developing in the equipment controller, the packet processor will continue operating to prevent interruptions to network services.

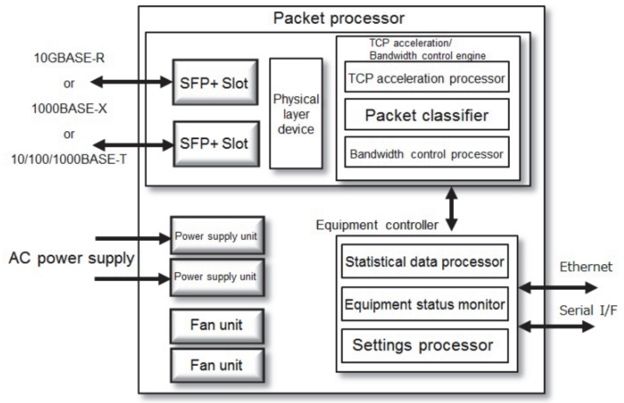

Figure 5 System Block Diagram

3.2 TCP Acceleration/Bandwidth Control Engine

The TCP acceleration/bandwidth control engine in the packet processor classifies received packets into control units called scenarios. The TCP acceleration, SMB acceleration, and bandwidth control are implemented in control units classified in these scenarios. A multi-core CPU is used to achieve both high-performance and high-accuracy processing of these packets. The TCP acceleration processing, packet classification processing, and scheduling processing are performed by the TCP acceleration/bandwidth control engine using a unique WAN Acceleration Relay Pipeline (WARP) technology to execute parallel processing independently by the cores of the multi-core CPU. The WARP technology automatically allocates buffers for storing the optimum amount of communications data per TCP session to achieve high-resolution processing performance and highly TCP acceleration communications.

TCP acceleration is achieved by the WSX units positioned at each side of the network path handling the TCP communications from the sending-side client to the receiving-side server to terminate the TCP protocol processing instead of the server (Figure 6).

For example, at TCP communications, data is sent to the server and the server receiving the data sends a response (ACK). However, since the ACK response is delayed in an environment with large WAN delays, the client side cannot send the data successively, which causes lower performance. At TCP acceleration using the WSX, the data from the client is buffered temporarily by the WSX positioned at the client side of the network and this WSX returns the ACK response instead of the server. This buffered data is transferred over a uniquely developed high-speed TCP tunnel to the other

WSX positioned at the server side. Consequently, the WSX at the server side becomes a substitute client and sends the data to the server by TCP communications. Using this linked processing accelerates the communications with no effects from WAN network delays.

Additionally, TCP acceleration requires various processing for each packet, such as buffering for TCP termination, ACK response substitution, accelerated tunnelling, and bandwidth control. Executing these processes in succession required development of the WARP parallel processing pipeline technology to achieve 10 Gbits/s scale throughput.

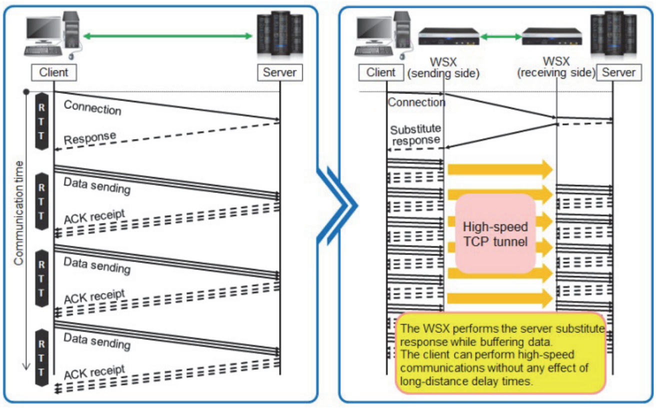

Figure 6 Typical TCP Response Sequence

The WSX TCP ACK response substitution, SMB acceleration processing, high-speed tunnelling, data compression, and bandwidth control processing are divided into finer processing stages using a total of 8-stage pipelining performed in parallel by the multi-core processor to implement the high throughput (Figure 7).

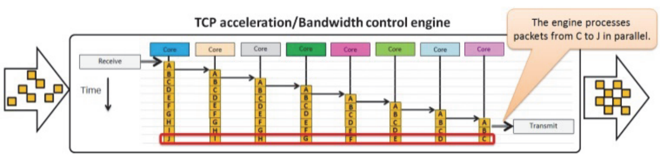

Figure 7 Parallel Processing by Multicore Processor

3.3 High-Speed Congestion Control Function

To support high and stable throughput even on WAN network with packet loss, we developed high-speed TCP tunnel technology between WSX units using High-Speed Adaptive TCP with a unique high-speed congestion control technology. High-speed adaptive TCP supports accelerated TCP communications even in environments with high packet loss since it is independent of network conditions such as packet loss and RTT. Instead of changing the communications method, the slow start algorithm feature of TCP is improved to shorten the time from the start of communications until the optimum transmission rate is reached. Additionally, the transmission rate at which congestion occurs is estimated from the dropped-packet pattern to fine-adjust the transmission rate and prevent it falling more than required.

High-speed adaptive TCP has two conditions: [1] Rapid Start and [2] Optimal Congestion Control (Figure 8). The Rapid Start condition starts the beginning of communications and increases the communication rate with time to detect the upper limit of the communication rate. When the upper limit of the communication capacity is reached, the status transitions to the Optimal Congestion Control condition to fine- adjust the communications bandwidth. In both these statuses, changes in the communication rates are calculated from the dropped-packet pattern. In the Rapid Start state, the acceleration increases and in the Optimal Congestion Control state, the transmission rate is fine-adjusted. If packets are dropped contiguously, the transmission rate is decreased, and when the communications suffer no dropped packets, the transmission rate is increased. As a result, the WSX units achieve greatly accelerated TCP communications.

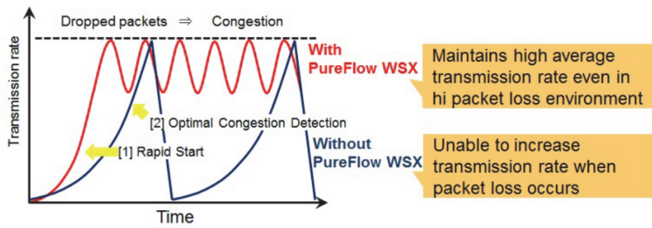

Figure 8 Congestion Control

3.4 TCP-FEC Function

In high-congestion environments, when dropped packets cause packet retransmission, the transmission rate drops remarkably. By using Forward Error Correction (FEC) technology, redundancy is added to the packets at the send side and if dropped packets are detected at the receive side, they can be recovered, which reduces the number of resent packets and stabilizes the transmission rate (Figure 9).

The sending-side WSX configures the data stream by compressing packet of data received from the client. Next, data for the predetermined data blocks size is obtained from the data stream to generate FEC redundancy blocks. Then, the data stream including the FEC data blocks is packetized and sent.

If the receiving-side WSX detects missing data resulting from dropped packets, it recovers the missing data using the FEC redundancy block data and then sends the packets without the FCC redundancy blocks to the server.

By using the FEC function in the high-speed TCP tunnel between the WSX units, retransmission can be suppressed by recovering data even if packets are dropped in the communication path between the WSX units, maintaining the high transmission rate.

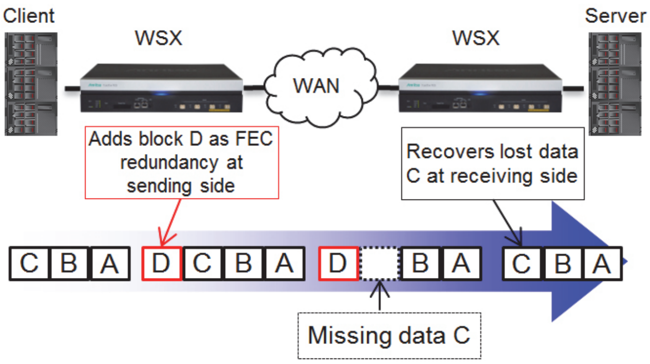

Figure 9 FEC Function

3.5 SMB Acceleration Function

The SMB protocol is used by Windows networks for setting shared folders and network drives; it can also be used when sharing files on file servers.

At communications using the SMB protocol, when the SMB command is requested from the client, the SMB command response is performed from the server. When the client performs a file operation on the server, an Open request requests the access permissions, a QueryInfo request reads the attributes, a DataRead request reads the file contents, a DataWrite request writes the file contents, and a Close request terminates the access. At file reading, the SMB commands are the Open, QueryInfo, DataRead, and Close requests (left side of Figure 10). The time until receiving the SMB command response is affected by this network delay, delaying the file read completion. For example, when the network delay is 100 ms, the number of file operations per second is limited to 2 to 3 files.

To improve this problem, the SMB acceleration function optimizes communications using the SMB protocol to accelerate the file read and write operations (right side of Figure 10). When the Open request is sent from the client, the client-side WSX detects the Open request from the client and notifies the server-side WSX, which, on receiving this notification, predicts the command to request required by the client and sends substitute QueryInfo and DataRead requests to the server. The server-side WSX notifies the client-side WSX about the command response received from the server and the client-side WSX prepares this command response before the QueryInfo and DataRead requests are sent from the client. Using this procedure, when the QueryInfo request is sent from the client, the command response from the client side is returned, eliminating the effect of network delays.

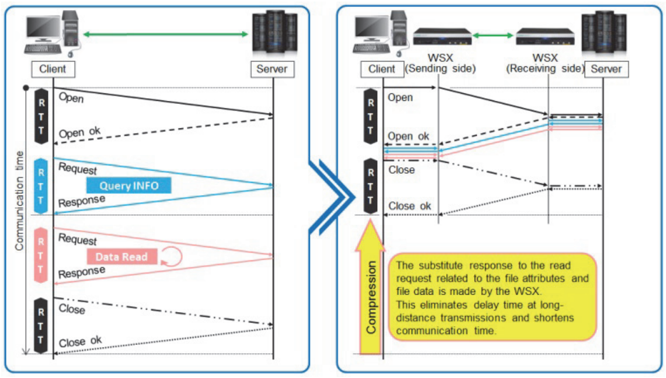

Figure 10 Optimized SMB Protocol

3.6 Precision Bandwidth Control Function

The precision bandwidth control engine demonstrates its effectiveness even for traffic with microbursts. By controlling the send timing for each packet, it optimizes smooth traffic flows with ideal packet intervals (Figure 11).

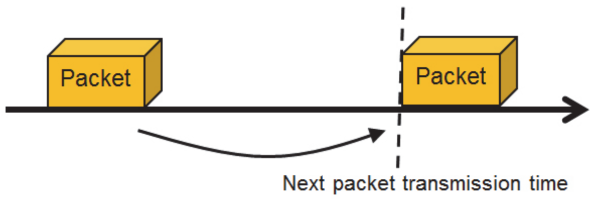

Figure 11 Packet Scheduling

To achieve this processing at high-speed, the bandwidth control for each packet is performed by distributing the processing between multiple CPU cores. Received packets are distributed to multiple cores for packet identification and bandwidth control processing. If the multi-scenario packet send times overlap, the send timing is arbitrated between cores so that there is no overlap (Figure 12).

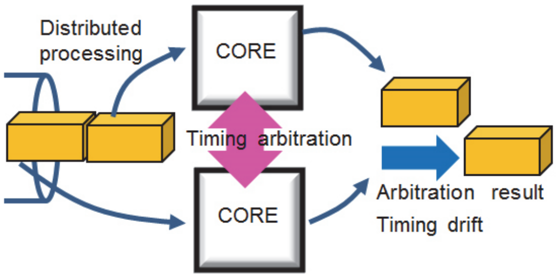

Figure 12 Parallel Scheduling using Multi-core Processor

Moreover, the bandwidth control engine calculates the next send timing so that there is no delay due to arbitration. Figure 13 shows an example of delay caused by arbitration. When the first packet of scenario B is delayed by arbitration, the bandwidth control engine calculates and subtracts the delay due to arbitration from the send timing for the second send packet of scenario B. Accumulated delays are prevented using this type of packet scheduling, and error in bandwidth control is held to less than 1%.

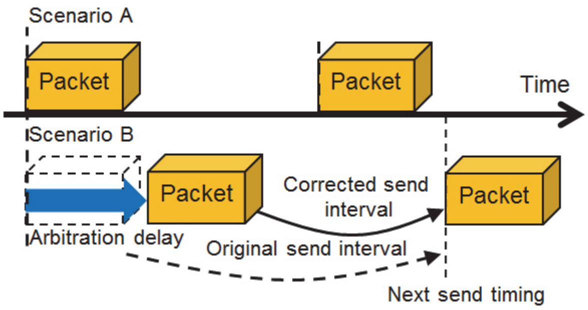

Figure 13 Packet Scheduling Competition

3.7 Equipment Controller

The equipment controller manages the user interface functions for displaying and changing settings and notifying and managing the equipment status. In addition to supporting common network management protocols, such as serial console, TELNET, Secure Shell (SSH), Simple Network Management Protocol (SNMP), Web Graphical User Interface (WebGUI), it also supports WebAPI for linking with cloud services.

4 Effect of Higher Speeds

4.1 Effect of TCP Acceleration

This section described the effect of high-performance TCP acceleration on WAN. We configured, tested and evaluated a virtual test network between data centers. The test network was configured as a Hypertext Transfer Protocol (HTTP) server at the data-center side with an HTTP client at the client side, communicating across a simulated WAN (Figures 14 and 15).

The simulated WAN had a peak transmission rate of 10 Gbit/s and variable delay time. In this test, large 5-GB files were downloaded to the client from the server and the average transmission rate and transmission time were measured.

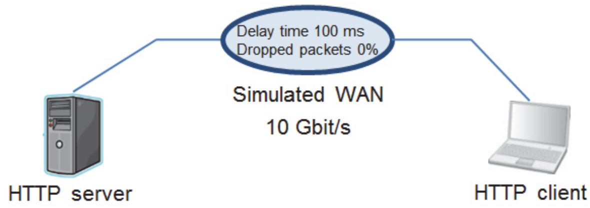

Figure 14 Test Configuration 1 (without TCP acceleration)

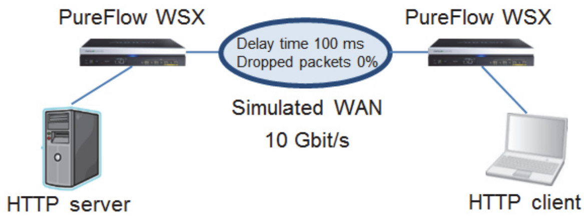

Figure 15 Test Configuration 2 (with TCP acceleration)

Table 1 lists the measured average transmission rate and time until transmission was completed for a WAN with 100 ms delay using 10 HTTP client sessions. Comparing the results with and without TCP acceleration, the average transmission rate with TCP acceleration was about 10 times faster and the time until completion of file download was about 90% shorter.

Table 1 Mean Transmission Rate and Time (RTT 100 ms)

Item	Mean Tx Rate	Tx Time
Without TCP acceleration	851.0 Mbit/s	47.0 s
With TCP acceleration	8.6 Gbit/s	4.6 s

In addition, Figure 16 shows a graph of the RTT delay versus average transmission rate with and without TCP acceleration.

Without TCP acceleration, the actual transmission rate for the simulated WAN with 400 ms delay averaged 316.0 Mbit/s despite the 10 Gbit/s specified rate. The reason for the decreased transmission rate was the long time waiting for ACK responses at the data sending side due to the network RTT delay.

Conversely, with TCP acceleration, the average transmission rate exceeded 8.5 Gbit/s. This was due to the effect of the TCP acceleration ACK substituted response. Since there was an immediate ACK response to the data-sending side, the data-sending wait time was greatly reduced, resulting in the higher transmission rate.

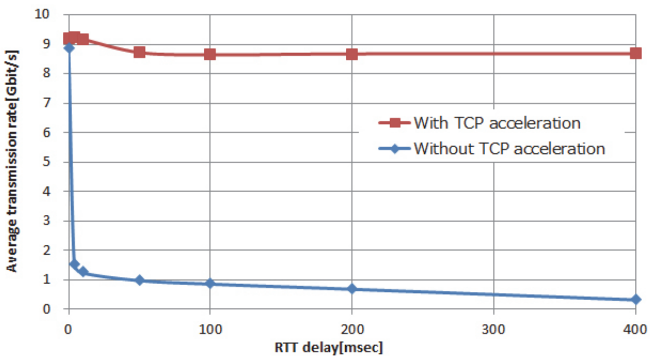

Figure 16 High-Speed TCP Throughput

Next, we examined the effect of TCP acceleration on a network with dropped packets. Figure 17 shows a comparison with and without TCP acceleration (with and without TCP-FEC). The graph shows the change in average transmission rate as the number of dropped packets changed.

Without TCP acceleration, the simulated 10 Gbit/s WAN saw the average transmission rate drop dramatically to about 0.9 Mbit/s with a dropped packet rate of 1%. This was caused by the TCP congestion control and retransmission control operation. Normally, when congestion control operates, temporary packet dropping is evaluated as an error caused by network congestion and the effect is unnecessary control of the transfer rate by the data-sending side. In addition, at the normal retransmission control operation, data can only be resent during the period until the missing data arrives. As a result, substantive data sending is impossible and the transmission rate drops. As a result of this, the low-transmission-rate status continues, causing a huge drop in the overall rate.

As opposed to this, with TCP acceleration (without TCP-FEC), the average transmission rate was about 755 times faster at 680 Mbit/s. This was due to the effect of the accelerated TCP congestion-control algorithm optimizing the TCP transmission rates while identifying dropped packets resulting from temporary packet dropping and excessive band control.

Moreover, when using TCP-FEC (with TCP-FEC), the average transmission rate was 843 times faster at 759 Mbit/s.

This was due to the effect of eliminating packet retransmission control resulting from missing data interpolation by TCP-FEC.

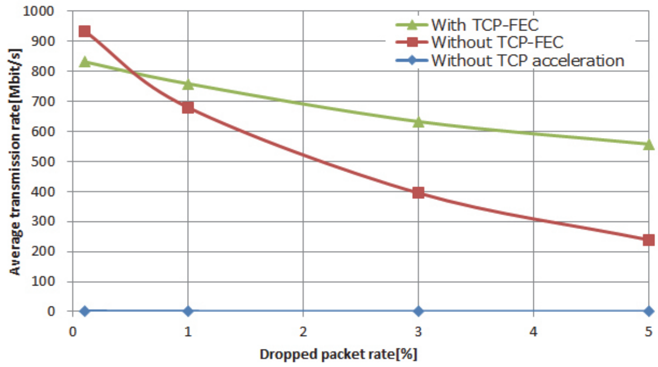

Figure 17 Throughput with Dropped Packets (RTT 100 ms)

4.2 Effect of SMB Acceleration

Next, we measured the average transmission rates and transmission times when downloading a 1 GB file from a server to a clients using the Windows network SMB file sharing protocol.

Figure 18 shows a graph of the average transmission rates versus RTT delay time with and without SMB acceleration.

Without SMB acceleration, on a 10 Gbit/s WAN with 100 ms delay the actual average transmission rate was only 19.8 Mbit/s despite the 10 Gbit/s specified rate. This was due to the network RTT delay causing long command response wait times at the client.

With SMB acceleration, the average transmission rate was 22 times faster at 444.5 Mbit/s even with a 100 ms delay. This was due to the effect of the shorter command response wait time at the client resulting from the WSX returning the server substitute SMB command response to the client.

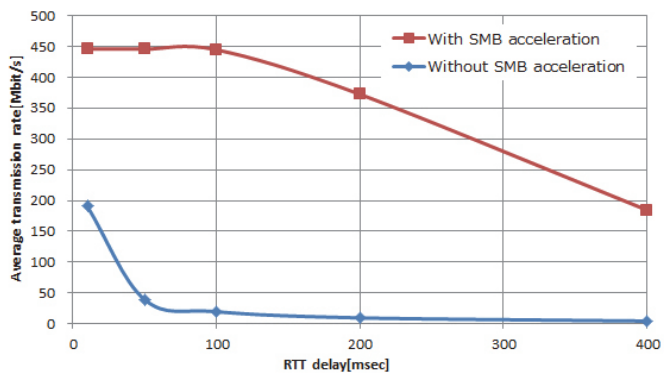

Figure 18 Throughput with SMB Protocol

5 Functions

The main WSX functions are listed in Table 2.

6 Conclusions

We have developed the Unified Network Controller PureFlow WSX with a maximum performance of 10 Gbit/s. The current PureFlow GS series has achieved the top share in the Japanese bandwidth control market due to its high performance and precision control. This new PureFlow WSX helps solve some customers' issues by offering truly global accelerated Communications.

Cloud-based Business is making remarkable progress and, we hope the developed PureFlow WSX will not only be a key device in helping further development of cloud-based business, but will also help stimulate further development of new technologies, such as Software Defined Networks meeting market requirements.

References

1) Alexander Afanasyev, Neil Tilley, Peter Reiher, and Leonard Kleinrock, "Host-to-Host Congestion Control for TCP", July 2010
2) Microsoft Corporation, [MS-SMB2]-v20160926, "Server Message Block (SMB) Protocol Versions 2 and 3", September 2016
3) Microsoft Corporation, [MS-SMB]-v20160714, "Server Message Block (SMB) Protocol", July 2016

* Windows is a registered trademark of Microsoft Corporation in the United States and other countries.

Authors


Kenji Anzai
Development Dept.
Anritsu Networks Co., Ltd.


Koichi Ryu
Development Dept.
Anritsu Networks Co., Ltd.


Atsushi Saegusa
Development Dept.
Anritsu Networks Co., Ltd.


Takayuki Sato
Development Dept.
Anritsu Networks Co., Ltd.


Tatsuya Ono
Development Dept.
Anritsu Networks Co., Ltd.


Syoya Fujii
Development Dept.
Anritsu Networks Co., Ltd.


Junya Oda
Development Dept.
Anritsu Networks Co., Ltd.

Table 2 Specifications

Item			Specification				
Name			Unified Network Controller				
Common Name			PureFlow WSX Lite		PureFlow WSX		
Model			NF7602A	NF7605A	NF7601A	NF7603A	NF7604A
Hardware Bypass			—	Supported	—	Supported	Supported
Bandwidth Control			10 kbit/s to 1 Gbit/s		10 kbit/s to 10 Gbit/s		
Scenarios	Max Scenario No./ Max. Levels		4096/8				
	Scenario Types		Acceleration, Aggregation, Fixed, Discard modes				
Filter	Max. Filter No.		40,000				
Rule List	Max. Group No.		1,024				
	Max. Entry No.		512 (but 64,000 max. total as groups × entries)				
Flow	Max. Flow No.		1,280,000				
Interface	Network Ports		SFP slot × 4 1000BASE-SX/LX, 10/100/1000BASE-T		SFP+/SFP slot × 4 10GBASE-SR/LR, 1000BASE-SX/LX, 10/100/1000BASE-T		
	Bypass Ports	Relevant Standard	—	1000BASE-T	—	1000BASE-SX 10GBASE-SR	1000BASE-LX 10GBASE-LR
	Console Port		RS-232C (RJ-45) (RJ-45/DB9 cable)				
	CF Card Slot		CompactFlash Specification Revision 4.1				
	USB Port		USB 2.0 Type A connector				
	Management Ethernet Port		10/100/1000BASE-T				
Traffic Acceleration	Target Protocol		TCP (IPv4/v6)				
	Max. TCP Session No.		100,000				
	Max. TCP-FEC Session No.		1000 (requires separate option)				
	Data Compression Method		ZIP				
	Congestion Control Method		Acceleration mode (High Speed Adaptive TCP)				
	Connection Configuration		In-Path connection, Out-Of-Path connection				
	Acceleration Bypass Function		Bypass switching at fault detection at 2-way device (RTT measurement, TCP connection error, KeepAlive error, forced)				
	Redundancy Length		secondary peer switching				
SMB Protocol Acceleration	SMB Session No.		10,000				
	SMB Version		SMB 2.0.2, SMB 2.1				
VLAN Support			VLAN Tag (IEEE802.1Q), QinQ (802.1ad)				
QoS Settings			Min. assured bandwidth, Max. bandwidth, buffer size, priority level (8 levels)				
Tos/Cos Remarking Function			Supported				
Max. Frame Length	Network Ports		2,048 or 10,240 bytes				
	Management Ethernet Port		1,518 byte				
Operation Management	Setting		CLI, RADIUS authentication, REST (WebAPI), WebGUI via Console/Telnet/SSHv2				
	Management		CLI, SNMPv1/v2c/v3, EnterpriseMIB, SYSLOG, peak rate monitor, WebGUI via Console/Telnet/SSHv2				
	Other		Traffic monitoring using Monitoring Manager 2				
Fault Recovery			Link Down transmission function, far-end equipment auto-switching, hardware bypass (NF7603A, NF7604A, NF7605A)				
Power Supply/Consumption			100 to 127/200 to 240 Vac, 50/60 Hz ±2 Hz/180 VA max., 140 W max.				
Operating Environment	Temperature/Humidity		0° to 40°C/20% to 80% (no condensation)				
Dimensions/Mass			88 (H) × 436 (W) × 471 (D) mm (excluding projections)/9.5 kg max. (with two PSU units installed)				
Safety Standard			UL60950-1, CSA C22.2 No.60950-1-07, EN60950-1				
Interference Prevention Standards			VCCI-A, FCC-A, EN55022-A, RCM-A, JIS C 61000-3-2				