



Accelerate Windows Applications with Dolphin SuperSockets™

Network Application Optimization on Windows Platforms

May 13th, 2013

Are you experiencing bottlenecks and lag in your network when running your networked Windows application? Are you looking to solve this problem?

10GigE is an option but the expense is keeping you from trying it and it would not address the base problem that the TCP protocol has a heavy performance cost which is causing the network to work slowly.

Dolphin Interconnect has the answer for you. SuperSockets for Windows will allow you to run your applications with continuous high-bandwidth, between servers and across the network, without ever having to modify the applications.

About SuperSockets™

SuperSockets™ is a plug and play driver module that enables networked user space applications to utilize the 40 Gigabit performance of PCI Express.

It removes system overhead and performance costs in the TCP protocol, resulting in socket latency as low as 1.27 microseconds.

The WinSock2 compliant SuperSockets™ is available for all Windows versions from Windows XP to Windows 8 (32/64 bits). The WinSock2 compliant SuperSockets™ driver supports 32 bit applications running in a 64-bit operating system interface.

Performance Evaluation

Performance evaluations are based on several aspects including both hardware and application performance. The SuperSockets™ software comes as component of the extensive Dolphin software bundle for Windows that also includes a shared memory API (SISCI).

Using the Dolphin evaluation kit, Dolphin tools and test applications enable system designers to:

- Evaluate PCI Express for both latency and data throughput.
- Look at Interconnect Shared Memory bandwidth over:
 - Cabled PCIe
 - Shared memory ping pong latency between two systems
 - PIO write bandwidth
 - Messaging latency
 - Remote system interrupt latency
- Evaluate the performance of multi-root PCI Express Gen 2 and its benefits over interconnect technologies, such as Ethernet, Infiniband and others.

SuperSockets™ includes support for standard TCP/IP WinSock2 interface for running benchmarks and real applications. Application benchmarks come in both binary and source form.

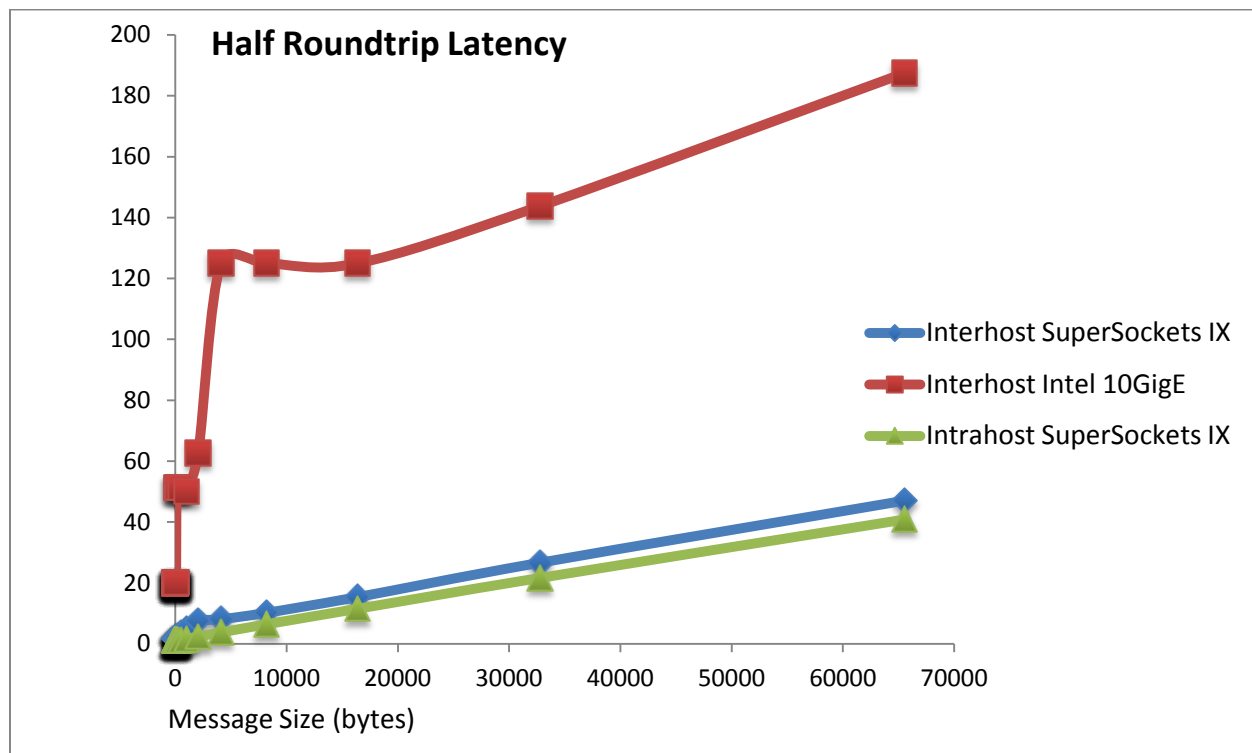
Introduction

This paper presents the *Layered Service Provider* (for short *SuperSockets™*) available with the *Dolphin Express* software stack on Windows. *SuperSockets™* for Windows was introduced in 2001 and provides the fastest available Windows based socket communication on PCI Express technology.

The *SuperSockets™* module addresses the needs of network applications with the following demands:

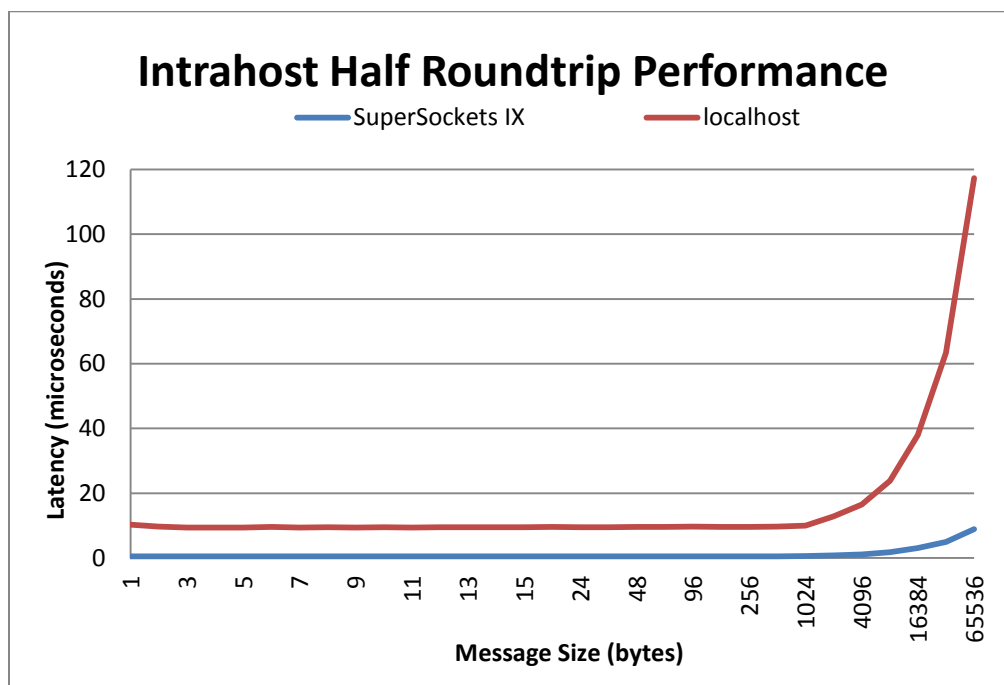
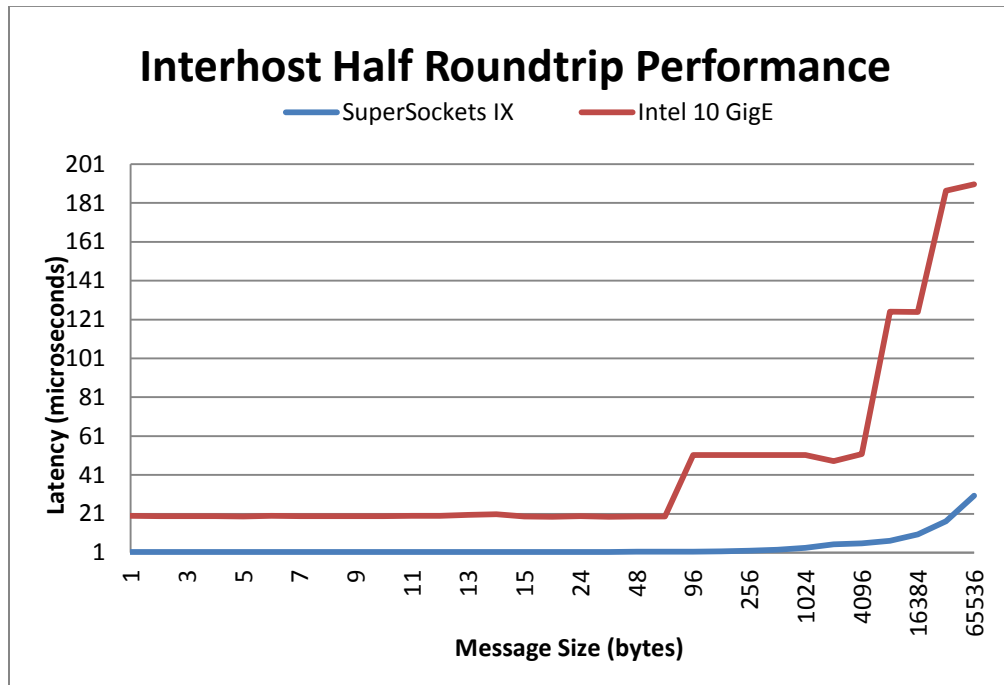
- 1) Extremely Fast Request/Response Times for Small Packets
- 2) High Throughput
- 3) Point-to-Point (`AF_INET`, `SOCK_STREAM`) Socket Family Transfers

After installing and configuring the Dolphin Express network, the *SuperSockets™* module will accelerate applications with no modifications to the application. Standard performance measurements show latency improvements of up to 10 times that of commodity 10 Gigabit Ethernet adapters.



As is shown in the previous chart *SuperSockets™* accelerates both inter-host transfers between separate servers, and intra-host transfers within a host server.

During an Inter-host transfer between host servers the typical half roundtrip *SuperSockets™* latency between the two systems is as low as 1.27 microseconds. The throughput between two hosts can reach 3.27 GBps. On loopback, the throughput can go as high as 2.78 GBps.



Measurements presented in this whitepaper have been performed using Dolphin Express IX
PCIe technology.

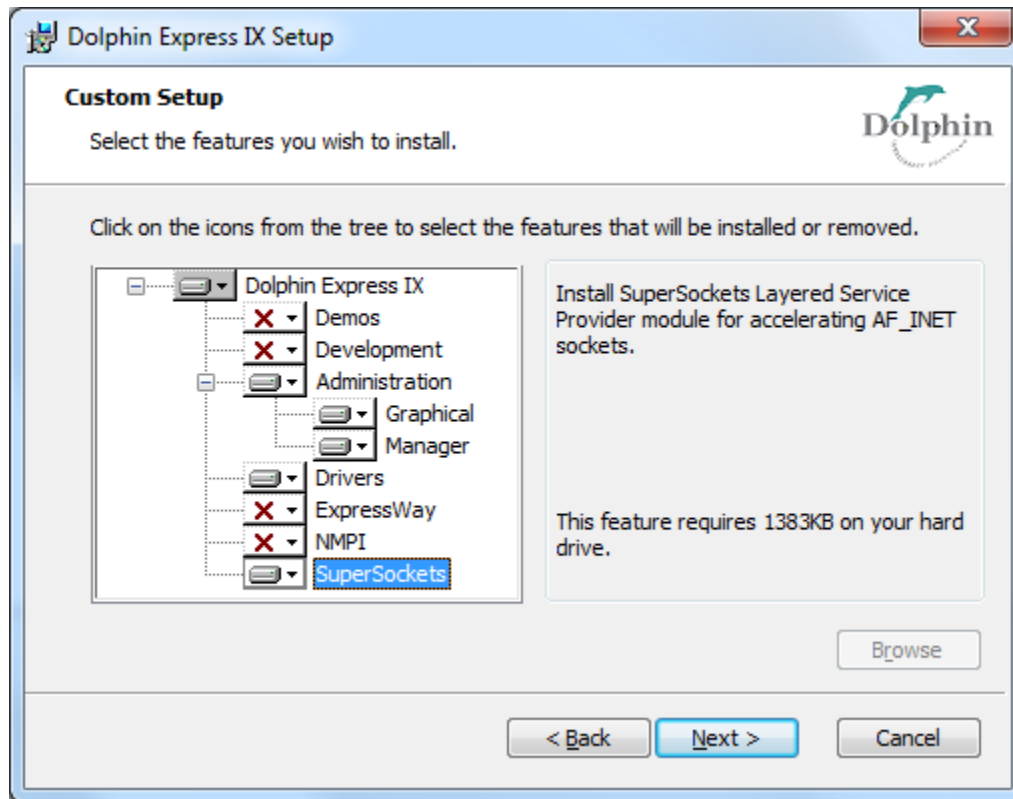
Half Roundtrip Latency Measured with latency_bench

Size (bytes)	Interhost		Intrahost	
	SuperSockets IX	10GigE	SuperSockets IX	Native ::1
1	1.29	19.94	0.47	10.33
2	1.27	19.84	0.46	9.7
3	1.30	19.79	0.46	9.42
4	1.27	19.72	0.46	9.39
5	1.30	19.69	0.46	9.39
6	1.30	19.93	0.46	9.55
7	1.30	19.79	0.47	9.42
8	1.30	19.83	0.47	9.47
9	1.30	19.75	0.46	9.43
10	1.31	19.71	0.46	9.48
11	1.30	19.88	0.47	9.41
12	1.30	19.89	0.46	9.46
13	1.32	20.41	0.46	9.51
14	1.32	20.73	0.47	9.5
15	1.32	19.60	0.47	9.5
16	1.30	19.45	0.46	9.57
24	1.33	19.71	0.46	9.51
32	1.39	19.52	0.47	9.52
48	1.43	19.61	0.47	9.58
64	1.54	19.63	0.47	9.61
96	1.57	51.26	0.51	9.65
128	1.64	51.18	0.48	9.58
256	1.98	51.31	0.52	9.61
512	2.49	51.21	0.54	9.72
1024	3.49	51.26	0.61	9.98
2048	5.28	48.08	0.82	12.82
4096	5.81	51.83	1.12	16.56
8192	7.12	125.06	1.79	23.8
16384	10.46	124.93	3.03	37.94
32768	17.11	187.40	4.91	63.33
65536	30.30	190.65	8.91	117.37

Measurements performed on Dell OptiPlex 790 boxes equipped with
Dolphin IXH610 and Intel 82598EB adapters.

Ease of installation

The *SuperSockets™* module is installed through the MSI setup application distributed for each operating system.



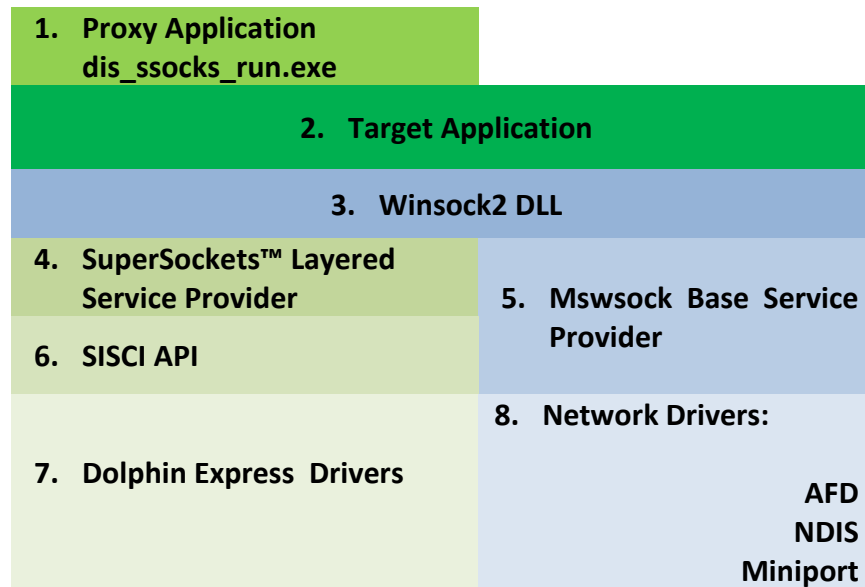
A typical network topology consisting of 2 nodes can be installed and configured in 5 minutes. The main characteristics of the network are node names and socket adapters. Socket adapters represent hostnames, IPv4 or IPv6 addresses that are accelerated by the layered service provider. For more information on configuring the adapters, please refer to the online guide available at:

<http://www.dolphinics.com/support/installation-ix.html>

Once the installation is completed, launch `ixadmin` to confirm that the topology is configured correctly. Click on Cluster menu, Run Application and then select `latency_bench` from the drop-down box to validate the SuperSockets™ installation.

Architecture

The *SuperSockets™ Layered Service Provider* integrates with the operating system as follows:



1. *Proxy Application* `dis_ssocks_run.exe`: notifies the Layered Service Provider that the child process requires acceleration. By itself, the layered service provider is loaded with every process performing a `socket()` or `WSASocket()` call. For reliability reasons, only selected applications use the Dolphin Express modules. Critical operating system processes such as services or the shell will continue to use the standard network stack.
The proxy application is installed in the *SuperSockets\bin* directory available in the installation directory. The default installation directory is *%ProgramFiles%\Dolphin Express IX*.
2. *Target Application*: represents an application consuming Winsock2 calls. Blocking, non-blocking and overlapped socket transfer modes are supported. Applications can be accelerated within limited user accounts.
3. *Winsock2 DLL*: the DLL located in the system directory. Applications link directly to `ws2_32.dll` or indirectly through `wsock32.dll`. `ws2_32.dll` exposes all the socket capabilities such as: socket transfers, socket extensions, helper functions, service provider catalog handling.
4. *SuperSockets™ Layered Service Provider*: the module registered at installation time with the operating system. It resides inside the installation directory in *SuperSockets\LSP\dislsp.dll*. The module is loaded with every `socket()` or `WSASocket()` call made by the `ws2_32.dll`. The *SuperSockets™* module is installed on top of all the other layered service providers. On a typical Dolphin Express network, the

SuperSockets™ module is the only Layered Service Provider installed above the standard *Base Service Provider* implemented by *mswsock.dll*. At runtime, the *SuperSockets™* module loads both the SISC API module and the next service provider. The reason for loading the next provider is that a typical socket handle is accelerated only after a `connect()` call is made. Any socket function called before `connect()` falls through the service provider below.

Keep in mind that this DLL may still reside in memory after it has been unregistered. On Windows Vista and above, the uninstall process may restart the services that have loaded this module.

The *SuperSockets™* module integrates with the operating system performance counters. To view the *Total Bytes Sent* and *Bytes Received* counters, run `perfmon` and look for *Dolphin LSP Statistics* inside the *Performance Object* list.

5. *Mswsock Base Provider*: on a standard system, the *mswsock.dll* is the only socket service provider. Third party products such as antiviruses or virtualization products may install their own service providers. For relevancy, we refer to this DLL only. As a base service provider, *mswsock* implements the user-mode network functionality. It communicates with the kernel mode network stack through file handles and I/O control codes. In Windows, all socket objects are file handles. *Mswsock* also implements socket extensions such as `TransmitFile()`. Some of the extensions are not accelerated by the *SuperSockets™* module.
6. *SISC API*: the module responsible for RDMA (Remote Direct Memory Access) transfers. This is the main user-mode component in the Dolphin Express software stack. Together with the kernel mode drivers, it exposes the adapter resources. The DLL is installed either when the Drivers feature is selected inside the setup application or when the *SocketsProvider* feature is selected. It resides in the System directory.
7. *Dolphin Interconnect Drivers*: The drivers are automatically configured on each node by a service called `dis_nodemgr`. This service is installed and started with the system. It receives commands periodically from an Administration machine which runs the `dis_networkmgr` service. The commands are sent using the TCP protocol. When installing either one of these services, the setup application requests for the user's permission to add it to the operating system Firewall database.
8. *Network Drivers*: these are the kernel drivers which implement the standard network stack. The stack consists of filter and miniport drivers.

Runtime

For TCP/IP consumers scheduled for acceleration, the network configuration requires a stable Ethernet connection in addition to the Dolphin Express PCIe adapter. Before socket transfers are commuted on the PCIe link, a connection is established using the Ethernet link. The bidirectional connection happens when each application calls `connect()` and `accept()`.

The network acceleration occurs when all Dolphin Express adapters are configured and both applications are running as child processes for `dis_ssocks_run.exe`. If one of the applications is not running inside `dis_ssocks_run.exe` environment, then all socket operations go through the standard network stack.

The `dis_ssocks_run.exe` takes the application name and switches as arguments as in:

```
"C:\Program Files\Dolphin Express IX\SuperSockets\bin\dis_ssocks_run.exe"  
C:\netserver.exe -H 192.168.0.1 -f
```

A Windows standalone service is scheduled for acceleration by using the following switches:

```
"dis_ssocks_run.exe -add <application_path>"  
"dis_ssocks_run.exe -remove <application_path>"  
"dis_ssocks_run.exe -list"
```

Please note that child processes are not accelerated by SuperSockets™, unless they are scheduled in advance with `"dis_ssocks_run.exe -add"`.

If an application needs to establish at runtime if a particular socket object has its transfers accelerated, it can call `WSAIoct1()` function with `SIO_GET_EXTENSION_FUNCTION_POINTER` control code and `DISLSP_EXTENSION_GUID` as input argument. The callback function returned by `WSAIoct1()` must be called after the connection has been established. For reference, view the `include\dislspect.h` header file located inside the installation directory. This file is installed when the Development feature is selected from the setup application.

Traditional network statistics such as **Bytes Received** are retrieved using a dedicated performance counter under the **Dolphin LSP Statistics** performance object.

Availability

The *SuperSockets™* module is part of the *Dolphin Express* software stack for Windows. The software is available for Windows XP, Windows Server 2003, Windows Vista, Windows Server 2008, Windows 7, Windows Server 2008 R2, Windows 8, Windows Server 2012 - 32 and 64 bit versions.

SuperSockets™ is also available for all major Linux distributions.

For further information regarding Dolphin Interconnect Solutions software and hardware products, please access <http://www.dolphinics.com> or send an e-mail at sales@dolphinics.com.