

Flexible Workload Acceleration on Intel® Architecture Lowers Equipment Cost

Intel® QuickAssist Technology eliminates the need for various add-on acceleration modules



Acceleration modules built
into the Intel® Platform for
Communications Infrastructure

There will always be classes of workloads and algorithms that push the capabilities of traditional compute platforms. A prime example is computationally-expensive public key encryption algorithms supporting SSL, particularly the handshaking mechanism that allows a server and a client to establish a connection. Consequently, IT professionals often deploy specialty based add-on acceleration modules to increase performance and avoid overloading their servers.

Now, there's an alternative to using specialized ASICs for compute-intensive workloads. Servers based on the Intel® Platform for Communications Infrastructure have new capabilities for accelerating asymmetric and symmetric cryptography, data compression and other workloads. The platform incorporates Intel® QuickAssist Technology, which includes acceleration modules that are accessed via a unified set of industry-

standard application programming interfaces (APIs). They provide consistent conventions and semantics across multiple accelerator implementations, thus future-proofing software investments. The platform also delivers exceptional packet processing performance by running performance-optimized libraries from the Intel® Data Plane Development Kit (Intel® DPDK).

This paper provides an overview of Intel QuickAssist Technology and describes four implementation scenarios that underscore the flexibility available to equipment manufacturers, independent software vendors (ISVs) and IT professionals. Using this workload acceleration technology, it's possible for Telco central offices and datacenters to significantly lower their equipment cost and improve workload balancing across servers.

Intel® QuickAssist Technology Overview

Although Intel QuickAssist Technology supports several mechanisms for Intel® architecture platforms to communicate with accelerators, this paper focuses on the acceleration capabilities within the Intel Platform for Communications Infrastructure, namely through the processors' chipset which may also be used as a PCIe* device for higher performance. They are accessed via the Intel QuickAssist Technology API, which makes calls to hardware acceleration modules in the platform, as depicted in Figure 1. The specialized acceleration modules available in this chipset, execute specific workloads including symmetric and asymmetric cryptography, compression and pattern matching. The API enables the acceleration modules to be easily accessed by open source software or vendor. Therefore, this flexibility can shorten time to market by reducing development complexity and effort.

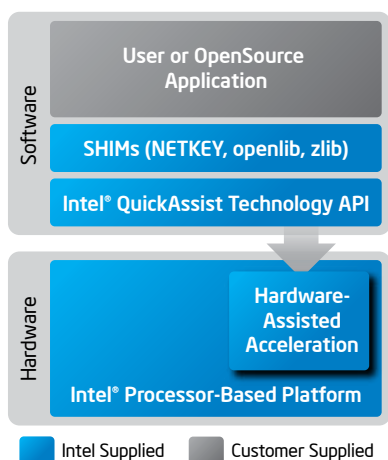


Figure 1. Intel® QuickAssist Technology Accesses Hardware Acceleration Modules

Open Source Software Support

Today, Intel QuickAssist Technology accelerates cryptography and data compression workloads, with others currently in development. The workloads are supported by open source frameworks and applications, as shown in Table 1. The use of open frameworks enables developers to quickly evaluate the performance of the acceleration modules.

Workload	Open Source Framework	Open Source Applications
Cryptography	<ul style="list-style-type: none"> OpenSSL libcrypto Linux* Kernel Crypto API (scatterlist) 	<ul style="list-style-type: none"> IPSec (NETKEY) Apache*
Data Compression	<ul style="list-style-type: none"> zlib Linux Kernel Crypto API (scatterlist) 	<ul style="list-style-type: none"> File compression (minigzip) IPComp (NETKEY)

Table 1: Supported Workloads and Open Source Frameworks and Applications

Benefits

Today, equipment manufacturers and IT professionals can choose from a large assortment of specialty add-on acceleration modules that span a wide range of performance. Compared to many modules, Intel QuickAssist Technology offers comparable or better performance, while providing some other benefits. It is relatively easy to scale performance because the acceleration capacity can be increased by adding processor cores and/or chipsets to the design. For instance, in bladed systems, having integrated acceleration in the processor chipset eliminates the need for an add-on acceleration module, reduces cost, lowers overall power consumption and decreases footprint. The Intel QuickAssist Technology API also preserves software investment over generations of platforms.

Implementation Scenarios

Intel QuickAssist Technology offers a high level of flexibility given that it supports diverse workloads, embraces open source software and runs in both kernel and user space to best accommodate the implementation provided by the open source community. For example, IPSec and IPComp typically run in kernel space, whereas many other open protocols run in user space. Proprietary applications may run in either user or kernel space as well.

- **User space:** The memory area typically assigned to application software by the operating system.
- **Kernel space:** The memory area reserved to run the kernel (the central part of an operating system) and software controlling platform resources, such as device drivers.

Implementation Effort and Performance

Equipment manufacturers and IT professionals can implement Intel QuickAssist Technology in a number of ways, as illustrated in Figure 2. The implementation options support either open source or proprietary applications and whether the protocol processing is done in user or kernel space. A number of open source application patches (user and kernel space) are supplied by Intel engineers; and therefore, the customer software integration effort is

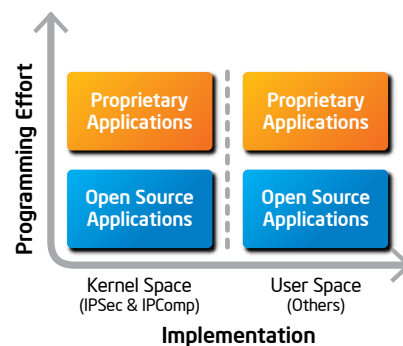
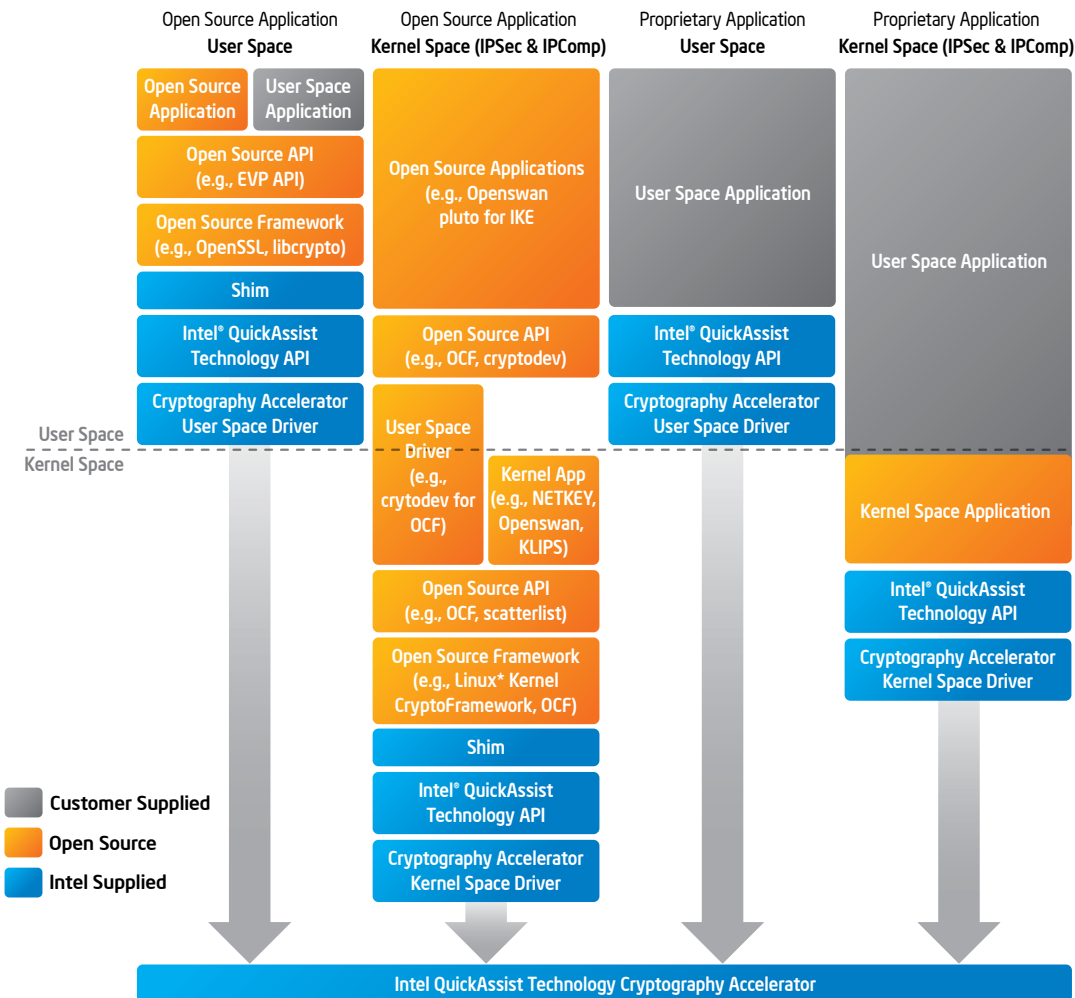


Figure 2. Possible Implementation Scenarios simplified. On the other hand, performance optimizations are possible for proprietary applications, which may require some customization to interface to the Intel QuickAssist Technology APIs. And these performance optimizations can be carried forward in future generations of Intel QuickAssist Technology. Implementation examples for these scenarios are provided in the following and depicted in Figure 3.



Scenario 1. Open source application, user space

Challenge: An ISV wants to set up a cipher to perform encryption and decryption operations in CBC, CFB, OFB or ECB mode.

Solution: Download OpenSSL (open source) and the associated high level API, called EVP. Include openssl/evp.h and link to libcrypto (-lcrypto). Compile with an Intel-provided shim that interfaces to the Intel QuickAssist Technology API through an OpenSSL engine implementation, which in turn calls a user space driver that interacts with the hardware.

Scenario 2. Open source application, kernel space (IPSec & IPComp)

Challenge: An ISV wants to use open source IPSec & IPComp software, where parts of the protocol handling run in kernel space.

Solution: Use an open source API, such as OCF-Linux*, which is a Linux port of the OpenBSD*/FreeBSD* Cryptographic Framework (OCF). This port aims to bring full asynchronous crypto acceleration to the Linux kernel and applications running under Linux. Intel supports kernel space operation with the Intel QuickAssist Technology API, and a kernel-built shim and driver. In this scenario, the packet flow usually stays in kernel space, and the user space code only handles control messages.

Scenario 3. Proprietary application, user space

Challenge: An equipment manufacturer wants to accelerate security functions called by its proprietary user space security application.

Solution: Modify the user space security application to make calls to the Intel QuickAssist Technology API. Some effort is needed to learn the API.

Scenario 4. Proprietary application, kernel space

Challenge: An equipment manufacturer wants to call the Intel acceleration hardware modules from kernel space.

Solution: Split up the proprietary security application so the calls to the hardware accelerators are made from kernel space.

Boost Packet Processing Throughput in User Space

Intel developed the Intel Data Plane Development Kit (Intel DPDK) to accelerate packet processing performance on Intel architecture-based platforms. The Intel DPDK contains a number of libraries, whose source code is available to developers to use and/or modify in a production network element. For select Intel processors and chipsets, examples are available using “run-to-completion” and/or “pipeline” models that enable the equipment manufacturer’s application to maintain complete control.

The Intel DPDK dramatically speeds up packet processing on Intel processors, as much as a seven times improvement, thereby enabling a higher level of throughput.

An Alternative to Add-on Acceleration Modules

Intel QuickAssist Technology introduces a framework for deploying platform-level workload acceleration. It provides the flexibility to allow equipment manufacturers and IT professionals to strike the right balance between programming effort and performance. In cases where Intel QuickAssist Technology eliminates the need for add-on acceleration modules, it’s possible to reduce cost, power consumption and form factor, while increasing the scalability and portability of the workload acceleration solution.

To learn more about Intel solutions for communications and networking, please visit <http://www.intel.com/go/commsinfrastructure>

