ARISTA WHITE PAPER

Arista 7050X Switch Architecture *('A day in the life of a packet')*

Arista Networks 7050 Series has become the mainstay fixed configuration 10GbE and 40GbE platform in many of the world's largest data centers.

The introduction of the Arista 7050X Series extends the family with increased performance, scalability, density and features designed for software defined networking and Universal Cloud Networks. The 7050X series adds support for next generation features and protocols, while combining a 2 fold increase in port count, table sizes and system forwarding capacity, without making any compromises to the system availability, reliability or functions.

This white paper provides an overview of the switch architecture and the packet forwarding characteristics of the Arista 7050X Series.

SWITCH OVERVIEW

The Arista 7050X Series are purpose built 10GbE/40GbE data center switches in compact and energy efficient form factors with wirespeed layer 2 and layer 3 forwarding, combined with advanced features for software defined cloud networking.

The Arista 7050X switches are a family of fixed configuration 1RU and 2RU systems, supporting a wide variety of interface options.

The 7050X Series are designed specifically to meet the challenges of dense 10 Gigabit and 40 Gigabit Ethernet switching. Featuring a flexible combination of 10GbE (SFP+) and 40GbE (QSFP+) interfaces in compact, low latency, power efficient form factors, supporting all the way up to 32 x 40GbE with a choice of systems.

The following table highlights the key technical specifications.

Table 1: Arista 7050X Series overview

Characteristic	7050QX-32	7050SX-128
Switch Height (RU)	1 RU	2 RU
SFP+ Ports	--	96
QSFP+ Ports	32	8
Maximum System Density 10GbE ports	96	96
Maximum System Density 40GbE ports	32	8
Maximum System Throughput (Tbps)	2.56Tbps	
Maximum Forwarding Rate (PPS)	1.44Bpps	
Latency	550ns	
Packet Buffer Memory	12MB	

THE 7050X SERIES AT A GLANCE

Increased adoption of 10 Gigabit Ethernet servers coupled with applications using higher bandwidth is accelerating the need for dense 10 and 40 Gigabit Ethernet switching. The 7050X Series supports a flexible combination of 10GbE and 40GbE in a highly compact form factor that allows customers to design flexible leaf and spine networks that accommodate both east-west traffic patterns and a requirement for low latency and power efficiency.

Figure 1: Left to Right: 7050QX-32, 7050SX-128

Each product within the 7050X series supports low-latency forwarding from just 550ns in cut-through mode. This is coupled with an extremely efficient forwarding pipeline that delivers minimal jitter. To ensure no performance hit during congestion or microbursts, each port-asic has access to a 12MB buffer which can be dynamically shared between ports that actively need it, while incorporating mechanisms to prevent buffer starvation due to elephant flows on one or more ports.

All models within the 7050X family share a common system architecture built upon the same system on chip (SOC) silicon. Varying only in interface type and quantity provided all models share a common set of software and hardware features, and key capabilities for high availability and reversible airflow options.

With typical power consumption of under 5 watts per 40GbE port the 7050X Series provides an industry leading power efficiency coupled with power supplies rated at platinum level. All models offer a choice of airflow direction to support deployment in either hot aisle / cold aisle top of rack environments, middle and end of row designs or at the network spine layer. An optional built-in SSD enables advanced capabilities for example long term logging, data captures and other services that are run directly on the switch.

Built on top of the same industry defining EOS image that runs on the entire Arista product portfolio, the 7050X Series delivers advanced features for big data, cloud, virtualized and traditional network designs.

7050X ASIC CONFIGURATIONS

At a high level the 7050X series are classified as “single chip” switches. The single chip switches refers to a System on a Chip (SoC) solution, where all hardware forwarding actions are controlled by a single chip. Advances in switching ASIC technology enables the 7050X Series to significantly increase the number of line rate interfaces delivered in a single chip, while still maintaining high throughput and low power usage. The 7050QX-32 and 7050SX-128 are both single chip systems.

7050X ARCHITECTURE

All stages of the packet forwarding pipeline are performed entirely in the hardware/dataplane. The forwarding pipeline is a closed system integrated on the packet processor (PP) of each SoC. The packet processors are capable of providing both the ingress and egress forwarding pipeline stages for packets that arrive on or are destined to ports located on that packet processor.

7050QX-32

The 7050QX-32 is a 1 RU solution, with 32 QSFP+ interfaces.

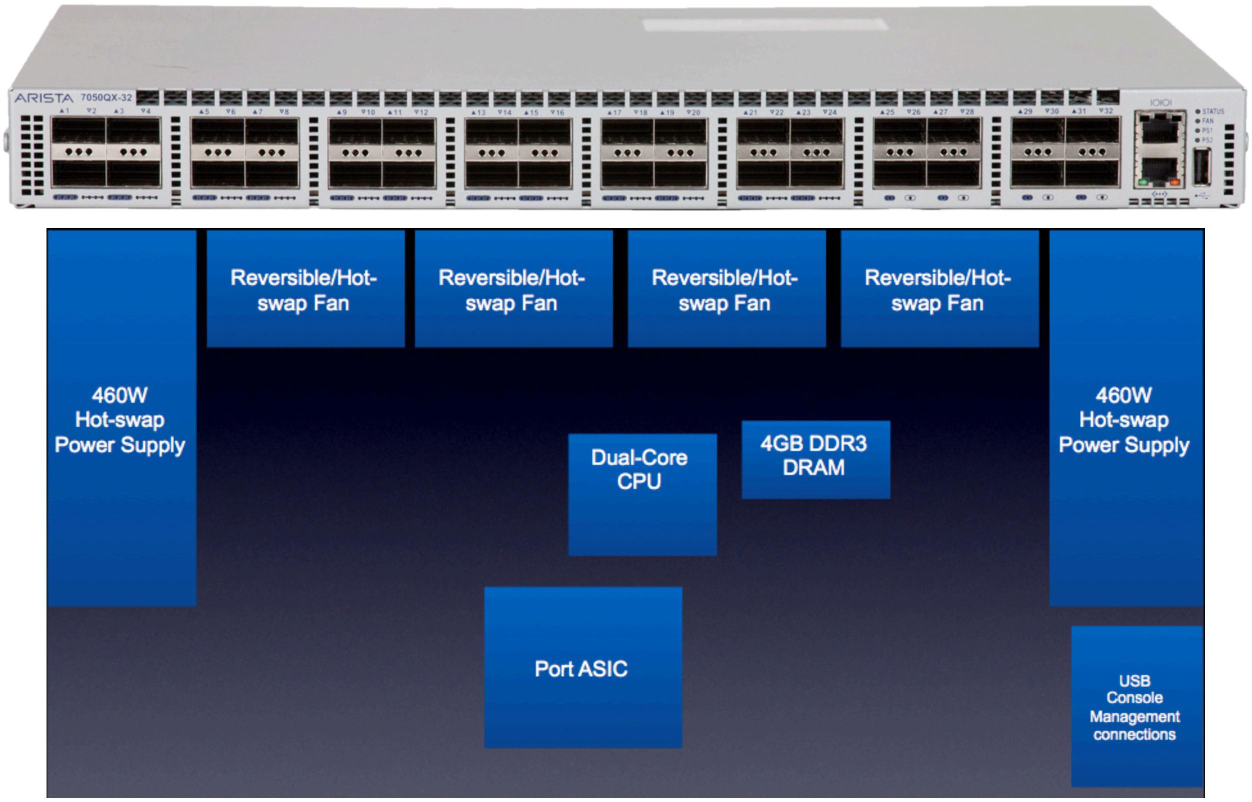

Figure 2: Arista 7050QX-32 (**above**: Physical Image, **below**: Logical device layout)

The 7050QX-32 supports two forwarding modes; a performance mode and a 10GbE low latency mode.

Performance mode is ideal for deployments where network scale is the objective. This mode enables all front panel interfaces and packets switching between interfaces both running at 40GbE the switching behavior is cut-through forwarding, whereas in contrast switching between two interfaces at 10GbE or mixed speeds (40GbE to 10GbE or vice versa) forwarding operates in a store-and-forward mode. In performance mode each of the QSFP+ interfaces 1-24 can be broken out into four 10GbE interfaces through the use of suitable transceivers and a splitter cable or copper cables, and the upper eight QSFP+ interfaces remain in 40GbE mode providing a wide combination of 10GbE and 40GbE ports in one system. Enabling all of the 24 QSFP+ ports in 10GbE mode allows a configuration of 96 x 10GbE and 8 x 40GbE interfaces.

10GbE latency mode is best suited to deployments where 10GbE latency is a primary concern. Enabling 10GbE latency mode will allow switching between interfaces at the same speeds (10GbE-10GbE and 40GbE-40GbE) to take place in a cut-through mode. As in performance mode, interfaces Ethernet 1-24 can be broken out on a per interface basis into four x 10GbE, however in 10GbE low latency mode the remaining eight QSFP+ interfaces are disabled providing a range of between 96 10GbE or 24 40GbE.

The 7050QX-32 platform uses the same power supplies and fan trays as the Arista 7050 and 7150 series switches for ease of sparing, reduced complexity and simpler operations.

7050SX-128

The 7050SX-128 is a 2RU solution with 96 SFP+ and 8 QSFP+ interfaces supporting 96 x 10GbE and 8 x 40GbE.

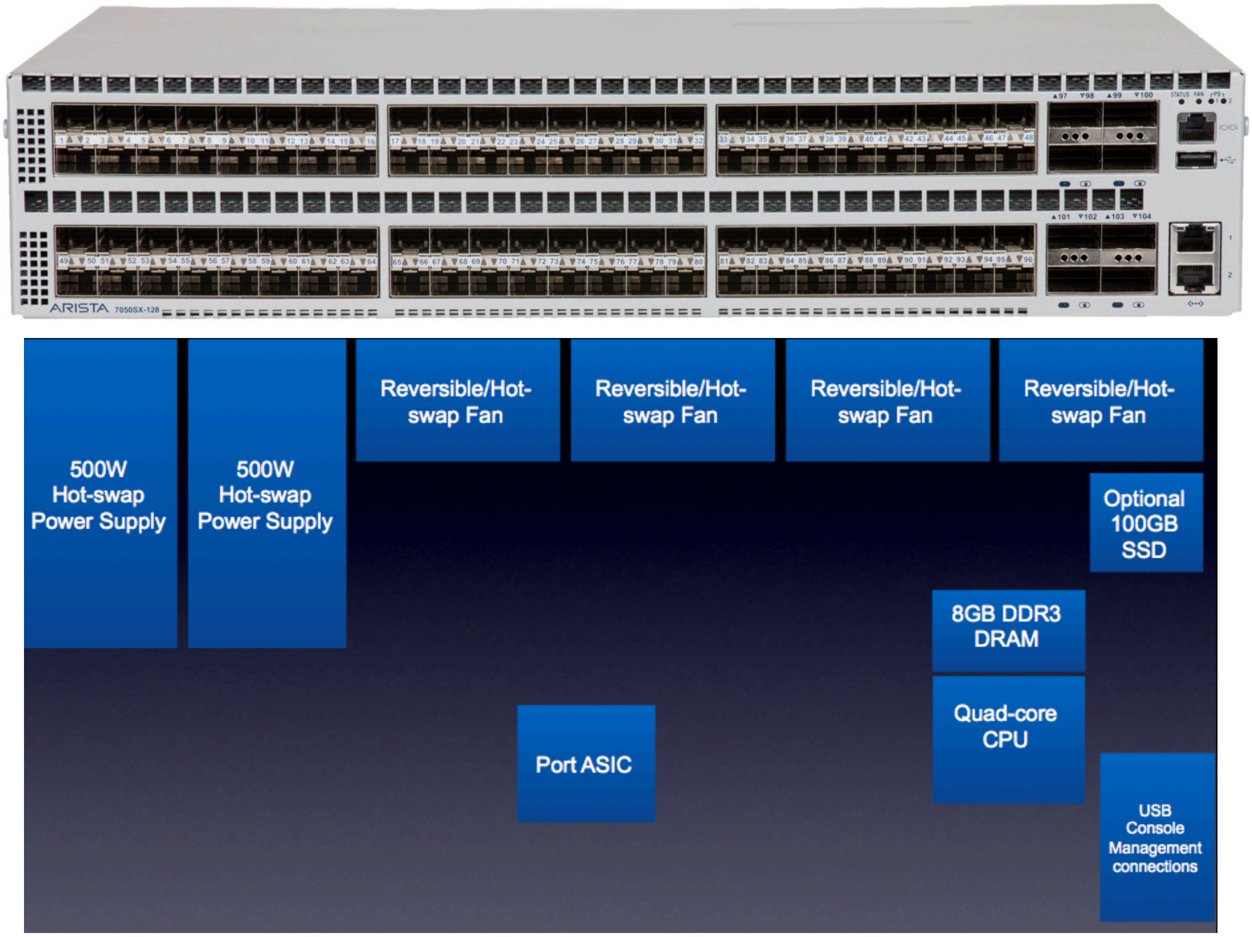

Figure 4: Arista 7050SX-128 (**above**: Physical Image, **below**: Logical device layout)

As with the 7050QX-32, the 7050SX-128 can be run in two switching modes, performance and 10GbE low latency mode:

Performance mode is primarily for deployments where network scalability is the focus. It enables all of the interfaces, with 40GbE to 40GbE in cut-through forwarding, while 10GbE to 10GbE and mixed speeds operates in store-and-forward. In performance mode all SFP+ interfaces can be used in 10GbE mode, while the QSFP+ interfaces remain in 40G mode. This results in a total of 96 x 10GbE and 8 x 40GbE interfaces.

10GbE latency mode is best suited to deployments where latency is the primary concern. Enabling 10GbE latency mode allows 10GbE to 10GbE switching in cut-through mode. All 96 SFP+ interfaces are active in 10GbE mode, and the remaining 8 QSFP+ ports are disabled.

SCALING THE DATA PLANE

In addition to increasing the port density available on fixed configuration platforms, the 7050X series also makes significant increases in both forwarding table density and flexibility. While traditional switches statically allocate resources to specific functions such as MAC Address tables, or IPv4 Host routes, recognizing that no two deployments are identical the 7050X supports a more flexible approach.

Forwarding table flexibility on the 7050X Series is delivered through the Unified Forwarding Table (UFT). Each L2 and L3 forwarding element has a dedicated table and can additionally have the tables sizes augmented by allocating a portion of the UFT. The UFT contains 256K entries from 4 banks, where each bank can be allocated to forwarding tables. Much wider deployment flexibility is achieved by being able to dedicate the entire UFT to expand the MAC address tables in dense L2 environments, or a balanced approach achieved by dividing the UFT between MAC Address and Host route scale. The UFT can also be leveraged to support the expansion of the longest prefix match (LPM tables) – (future).

Table 2: Arista 7050X Series Table Scale with UFT

Linecard Port Characteristics	DCS-7050X
MAC Address Table	288K
IPv4 Host Routes	208K
IPv4 LPM Routes	128K *
IPv4 Multicast Routes	104K
IPv6 Host Routes	104K
IPv6 LPM Routes	77K *
IPv6 Multicast Routes	4000
Packet Buffers	12MB
ACLs	4K Ingress 1K Egress

*Roadmap Scale

SCALING THE CONTROL PLANE

The CPU on the Arista 7050X Series is used exclusively for control-plane and management functions; all data-plane forwarding logic occurs at the packet processor/Port ASIC level.

Arista EOS®, the control-plane software for all Arista switches executes on multi-core x86 CPUs with multiple gigabytes of DRAM. As EOS is multi-threaded, runs on a Linux kernel and is extensible, the large RAM and fast multi-core CPUs provide for operating an efficient control plane with headroom for running 3rd party software, either within the same Linux instance as EOS or within a guest virtual machine.

Out-of-band management is available via a serial console port and/or the 10/100/1000 Ethernet management interface. The 7050X Series also offer a USB2.0 interface that can be used for a variety of functions including the transferring of images or logs.

PACKET FORWARDING PIPELINE

Each packet processor is a System on Chip (SoC) that provides all the ingress and egress forwarding pipeline stages for packets to or from the front panel ports. Forwarding always occurs in the data-plane and never falls back to software for forwarding.

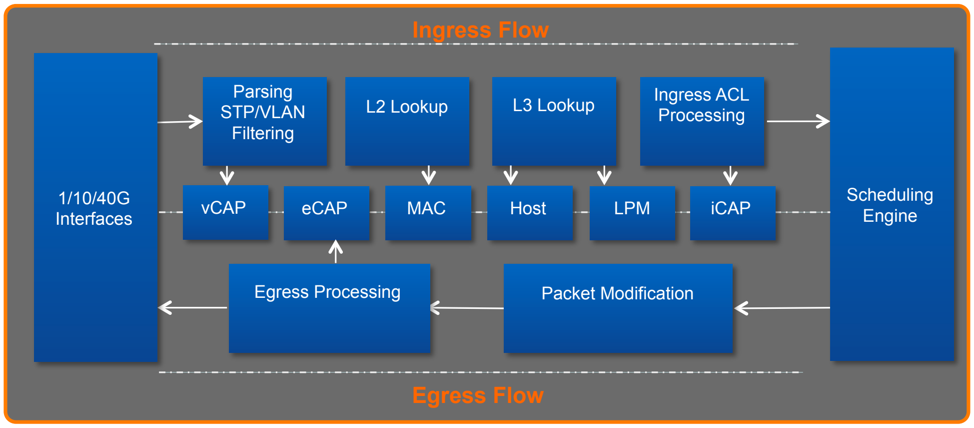

Figure 5: Packet forwarding pipeline stages inside a single chip Arista 7050X

The forwarding pipeline can be separated into two phases, the ingress flow and egress flow. The ingress flow is responsible for the majority of switching functions, including address learning, VLAN assignment, L2 and L3 forwarding lookups, QoS classification and Ingress ACL processing. The Egress flow provides packet buffering, the packet rewrite and egress ACL processing.

STAGE 1: NETWORK INTERFACE (INGRESS)

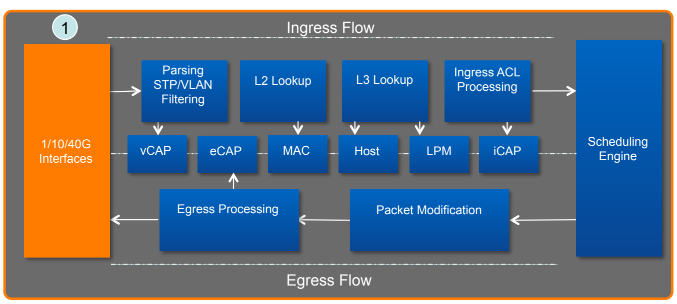

Figure 6: Packet Processor stage 1: Network Interface (Ingress)

When packets/frames enter the switch, the first block they arrive at is the Network Interface stage. This block is responsible for implementing the Physical Layer (PHY) interface and Ethernet Media Access Control (MAC) layer on the switch.

The PHY layer is responsible for transmission and reception of bit streams across physical connections including encoding, multiplexing, synchronization, clock recovery and serialization of the data on the wire for whatever speed/type of Ethernet interface is configured. Operation of the PHY is in compliance with the IEEE 802.3 standard. The PHY layer transmits/receives the electrical signal to/from the transceiver where the signal is converted to light in the case of an optical port/transceiver. In the case of a copper (electrical) interface, e.g., Direct Attach Cable (DAC), the signals are converted into differential pairs.

If a valid bit stream is received at the PHY then the data is sent to the MAC layer. On input, the MAC layer is responsible for turning the bit stream into frames/packets: checking for errors (FCS, Inter-frame gap, detect frame preamble) and find the start of frame and end of frame delimiters.

STAGE 2: INGRESS PARSER

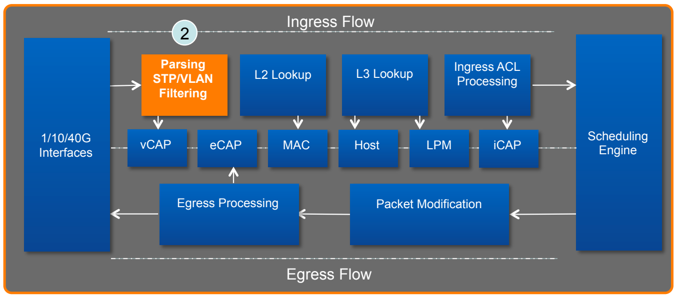

Figure 7: Packet Processor stage 2: Ingress Parser

The Ingress Parser represents the first true block of the forwarding pipeline. While the entire packet is received at the Mac/Phy layer only the packet header is sent through the forwarding pipeline itself.

The first step is to parse the headers of the packet and extract all of the key fields required to make a forwarding decision. The headers extracted by the parser depend on the type of packet being processed. A typical IPv4 packet would extract a variety of L2, L3 and L4 headers including the source MAC address, destination MAC address, Source IP, Destination IP and Port numbers).

The Parser will then determine the VLAN ID of the packet, if the packet arrived on a trunk port this can be determined based on the contents of the VLAN header. If the packet arrived on an access port, or arrived untagged the VLAN ID is determined based on the port configuration.

Once the Parser is aware of the VLAN ID and ingress interface it must verify the STP port state for the receiving VLAN. If the port STP state is discarding or learning, the packet is dropped. If the port STP state is forwarding no action is taken.

As a final ingress check the Parser will compare the packet against any configured Port ACLs by performing a lookup in the vCAP, the first of the three ACL TCAMs. If the packet matches a DENY statement it will be dropped. If the packet matches a PERMIT statement, or no port ACL is applied, the packet is passed to the next block of the pipeline.

STAGE 3: L2 LOOKUP

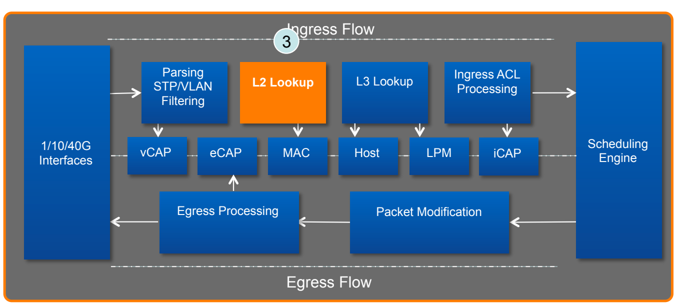

Figure 8: Packet Processor stage 3: L2 Lookup

The L2 Lookup block will access the MAC address-table (an exact-match table) and perform two parallel lookups.

The first lookup is performed with the key (VLAN, source mac-address), to identify if it matches an entry already known to the switch and therefore present in the mac-address table. There are three possible outcomes to this lookup:

- MAC address unknown, trigger a new MAC learn, mapping the source MAC to this port.
- MAC address known but attached to another port, triggering a MAC move and a reset of the entry's age.

- MAC address known and attached to this port, triggering a reset of the entry's age.

The second lookup is performed with the key (VLAN, Destination MAC address) this lookup has four possible outcomes:

- If the destination MAC address is a well known or IEEE MAC, trap the packet to the CPU. The system uses a series of hardware rate-limiters to control the rate at which traffic can be trapped or copied to the CPU.
- If the destination MAC address is either a physical MAC address or a Virtual (VRRP/VARP) MAC address owned by the switch itself, the packet is routed.
- If neither of the above is true but the MAC address-table contains an entry for the destination MAC address, the packet is bridged out of the interface listed within the entry.
- If neither of the above is true and the MAC address-table does not contain an entry for that MAC address, the packet is flooded out of all ports in an STP forwarding state within the ingress VLAN, subject to storm-control thresholds.

STAGE 4: L3 LOOKUP

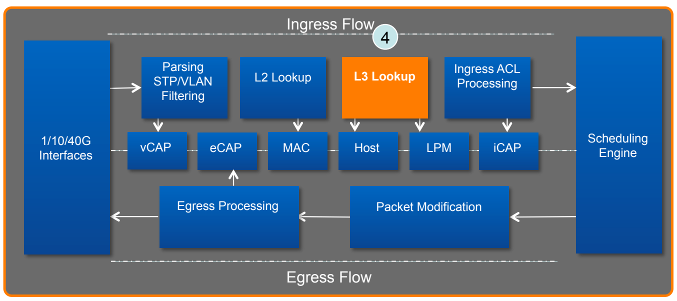

Figure 9: Packet Processor stage 4: L3 Lookup

The L3 Lookup stage performs sequential accesses into two distinct tables, each access includes up to two lookups. The first table is an exact-match table which contains /32 v4 and /128 v6 host routes. The second table is a longest-prefix match (LPM) table which contains all v4 routes and v6 routes shorter than /32 and /128 lengths respectively.

The first lookup into both the host route and LPM tables is based on the key (VRF, Source IP Address), this lookup is designed to verify that the packet was received on the correct interface (the best interface towards the source of the packet), if received on any other interface the packet may be dropped depending on user configuration. This lookup takes place only if uRPF is enabled.

The second lookup takes place initially in the host route table; the lookup is based on the key (VRF, Destination IP address) the purpose is to attempt to find an exact match for the destination IP address. This is typically seen if the destination is a host in a directly connected subnet. If an exact match is found in the host route table the result provides a pointer to an egress physical port, L3 interface and packet rewrite data.

If there is no match for the lookup in the host table, another lookup with an identical key is performed in the LPM table to find the best or longest prefix-match, with a default route being used as a last resort. This lookup has three possible outcomes:

- If there is no match, including no default route, then the packet is dropped.
- If there is a match in the LPM and that match is a directly connected subnet, but there was no entry for the destination in the host route table, the packet is punted to the CPU to generate an ARP request.

- If there is a match in the LPM table, and it is not a directly connected subnet it will resolve to a next-hop entry which will be located in the Host Route table. This entry provides an egress physical port, L3 interface and packet rewrite data.

The logic for multicast traffic is virtually identical, with multicast routes occupying the same tables as the unicast routes. However instead of providing egress port and rewrite information, the adjacency points to a Multicast ID. The Multicast ID indexes to an entry in the multicast expansion table to provide a list of output interfaces.

STAGE 5: INGRESS ACL PROCESSING

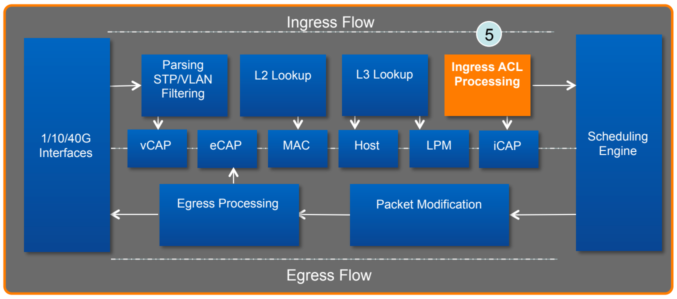

Figure 10: Packet Processor stage 5: Ingress ACL Processing

The Ingress ACL processing block functions as a matching and policy enforcement engine. All policy and matching logic is stored in the iCAP TCAM.

Routed traffic is checked against any router ACLs configured on the ingress direction of the receiving L3 interface. If the packet matches a DENY statement it will be dropped. However if the packet matches a PERMIT statement, or no router ACL is applied to the source interface, the traffic will continue through the forwarding pipeline.

The packet is also checked against any quality of service (QoS) policies contained on the ingress interface, if the packet is matched by a class within a policy-map it is subject to any actions defined within that class. Typical actions include policing/rate-limiting, remarking the CoS/DSCP or manually setting the traffic-class/queue of the packet to influence queuing further in the pipeline.

Finally the Ingress ACL Processing block applies any packet filters, such as storm-control and IGMP Snooping.

STAGE 6: SCHEDULING ENGINE

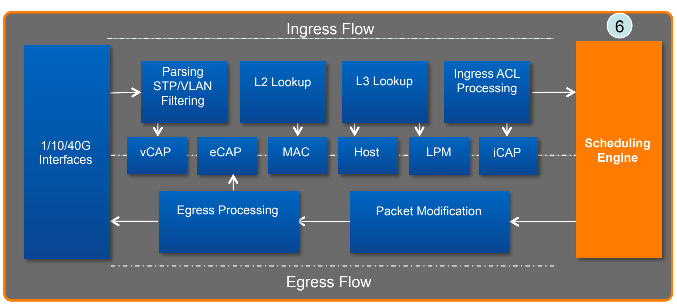

Figure 11: Packet Processor stage 6: Scheduling Engine

The Scheduling Engine or Memory Management Unit (MMU) performs the packet buffering and scheduling functions of the ASIC. The scheduler is made up of two components:

- The ingress MMU allocates available memory segments to packets that must be buffered.

- The egress MMU replicates and de-queues packets resident in system buffers, making those buffers made available for other packets.

The packet processor has 12MB of on chip packet buffer memory. This memory is divided into fixed segments, 208 bytes in size, this ensures the system contains a finite but predictable number of packet buffer segments. These segments are then distributed among the various memory pools. There are three types of memory pool:

- Headroom pools, buffers used exclusively for in-flight packets.
- Private Pools, buffers dedicated exclusively to a particular system queue.
- The Shared Pool, a single large pool is available to store packets once a particular system queue's private pool has been exhausted. The shared pool is significantly larger than the headroom or private pools.

If packet buffering is required the ingress MMU ascertains if there is memory available for this packet and in which pool the packet should be stored (based on the system fair-use policy). While a large packet may consume multiple buffer segments it is not possible for multiple packets to be located in a single segment.

Each physical port will have 8 unicast queues which map internally to the 8 supported traffic-classes. Therefore a system queue can be uniquely identified by the combination of Egress Port and Traffic class, or (EP, TC). Each system queue will have a pool of dedicated (private) buffers that cannot be used by any other system queue.
If a packet arrives at the scheduling engine and must be en-queued (i.e. if the egress port is congested), several steps take place. In the first instance the Ingress MMU will attempt to en-queue this packet into the private buffers for the destination system queue.

If the are no private buffers for that (EP,TC) available in the appropriate private pool, two further checks are made:

- Are any packet buffers available in the shared buffer pool?
- Is the system queue occupying less than its permitted maximum number of buffer segments in the shared pool? (i.e. the queue-limit).

If both of the above statements are true, the packet will be en-queued on buffers from the shared pool. If either of the above statements is false the packet will be dropped.

If a packet arrived and no congestion was encountered then the packet would be held in 'headroom buffers' used exclusively for in-flight packets, the packet would remain here only long enough for the header to pass through the forwarding pipeline and be serialized out of the egress interface.

Once a system queue contains 1 or more segments the egress MMU will attempt to de-queue these segments. The egress MMU will attempt to forward packets out of an Egress Port on a per Traffic Class basis.   The rate at which this occurs is based on the queuing configuration and any configured egress packet shaping. By default the MMU will be configured with hierarchical strict priority queues, this ensures packets in traffic-class 5 are processed only when the higher priority classes 6 and 7 are empty, while packets in traffic-class 4 are processed only when classes 5, 6 and 7 are empty etc.

STAGE 7: PACKET MODIFICATION

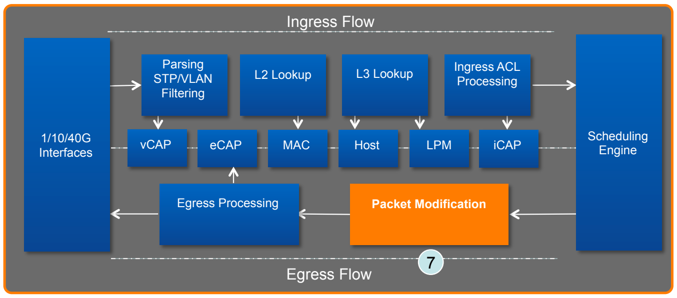

Figure 12: Packet Processor stage 7: Packet Modification

All previous blocks in the forwarding pipeline performed actions, some of these actions resulted in a requirement to make changes to the packet header, however no actual rewrites took place. Each block in the pipeline appended any changes to the packet header as meta-data.

The packet modification block takes the meta data added by previous blocks in the pipeline, and performs the appropriate rewrite of the packet header. The exact data rewritten depends on the packet type and if the packet was routed or bridged, rewritten data typically includes changing the source and destination MAC address and decrementing the TTL for routed traffic and rewriting the CoS value.

STAGE 8: EGRESS PROCESSING

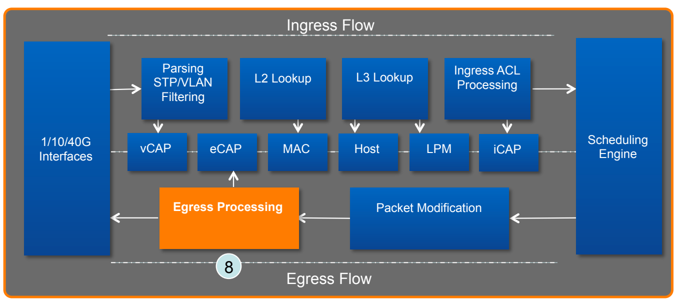

Figure 13: Packet Processor stage 8: Egress Processing

The Egress ACL processing block enables packet-filtering functionality in the egress direction by performing a mask-based lookup in the eCAP, the third of the ACL TCAMs.

If a packet has been routed, it will be compared against any Router ACLs applied in the outbound direction of the egress L3 Switched VLAN Interface (SVI) and any Port ACLs applied in the outbound direction of the egress physical interface. If a packet has been bridged it will be compared only against Port ACLs applied in the outbound direction of the egress physical interface.

As with the previous TCAM lookups, if the packet matches a DENY statement it will be dropped. However if the packet matches a PERMIT statement, or no ACL is applied to the destination SVI/interface, the traffic will continue through the forwarding pipeline.

EOS features that require egress filtering, such as MLAG, to prevent duplication of flooded packets, also use the eCAP.

STAGE 9: NETWORK INTERFACE (EGRESS)

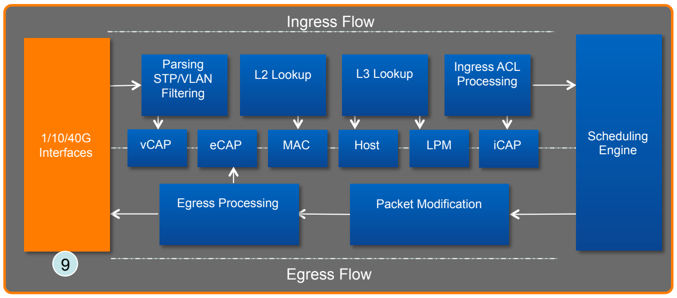

Figure 14: Packet Processor stage 9: Network Interface (Egress)

Just as packets/frames entering the switch went through the Ethernet MAC and PHY layer with the flexibility of multi-speed interfaces, the same mechanism is used on packet/frame transmission. Packets/frames are transmitted onto the wire as a bit stream in compliance with the IEEE 802.3 standards.

ARISTA EOS: A PLATFORM FOR SCALE, STABILITY AND FLEXIBILITY

Arista Extensible Operating System, or EOS®, is the most advanced network operating system in the world. It combines modern-day software and O/S architectures, transparently restartable processes, open platform development, a Linux kernel, and a stateful publish/subscribe database model.

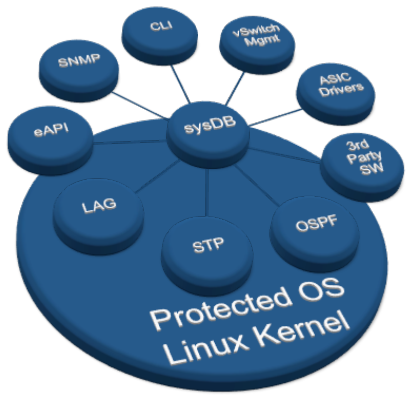

Figure 30: Arista EOS Software Architecture showing some of the Agents

At the core of EOS is the System Data Base, or SysDB for short. SysDB is machine generated software code based on the object models necessary for state storage for every process in EOS. All inter-process communication in EOS is implemented as writes to SysDB objects. These writes propagate to subscribed agents, triggering events in those agents. As an example, when a user-level ASIC driver detects link failure on a port it writes this to SysDB, then the LED driver receives an update from SysDB and it reads the state of the port and adjusts the LED status accordingly. This centralized database approach to passing state throughout the system and the automated way SysDB code is generated reduces risk and error, improving software feature velocity and provides flexibility for customers who can use the same APIs to receive notifications from SysDB or customize and extend switch features.

Arista's software engineering methodology also benefits customers in terms of quality and consistency:

- Complete fault isolation in the user space and through SysDB effectively convert catastrophic events to non-events. The system self-heals from more common scenarios such as memory leaks. Every process is separate, no IPC or shared memory fate sharing, endian-independent, and multi-threaded where applicable.
- No manual software testing. All automated tests run 24x7 and with the operating system running in emulators and on hardware Arista scales protocol and unit testing cost effectively.
- Keep a single system binary across all platforms. This improves the testing depth on each platform, improves time-to-market, and keeps feature and bug compatibility across all platforms.

EOS, and at its core SysDB, provide a development framework that enables the core concept - Extensibility. An open foundation, and best-in-class software development models deliver feature velocity, improved uptime, easier maintenance, and a choice in tools and options.

CONCLUSION

The 7050X Series combined with the Arista leaf-spine design methodology and the Arista 7000 Family provides flexible design solutions for network architects looking to deliver market-leading performance driven networks, which scale from several hundred hosts all the way up several hundred thousand hosts.

The performance focused 7050X Series delivers up to 32 x 40GbE or 96 x 10GbE and 8 x 40GbE ports designed specifically to operate in a real world deployment. With up to 2.56Tbps or 1.44Bpps of forwarding capacity, the 7050X Series provides the port density, table scale, feature set and forwarding capacity essential in today's data center environments.

All Arista products including the 7050X Series run the same Arista EOS software binary image, simplifying network administration with a single standard across all switches. Arista EOS is a modular switch operating system with a unique state sharing architecture that cleanly separates switch state from protocol processing and application logic. Built on top of a standard Linux kernel, all EOS processes run in their own protected memory space and exchange state through an in-memory database. This multi-process state sharing architecture provides the foundation for in-service-software updates and self-healing resiliency.

Combining the broad functionality with the diverse form factors make the Arista 7050X Series ideal for building reliable, low latency, cost effective and highly scalable data center networks, regardless of the deployment size and scale.

Santa Clara—Corporate Headquarters
5470 Great America Parkway
Santa Clara, CA 95054
Tel: 408-547-5500
www.aristanetworks.com

San Francisco—R&D and Sales Office
1390 Market Street Suite 800
San Francisco, CA 94102

India—R&D Office
Eastland Citadel
102, 2nd Floor, Hosur Road
Madiwala Check Post
Bangalore - 560 095

Vancouver—R&D Office
Suite 350, 3605 Gilmore Way
Burnaby, British Columbia
Canada V5G 4X5

Ireland—International Headquarters
Hartnett Enterprise Acceleration Centre
Moylish Park
Limerick, Ireland

Singapore—APAC Administrative Office
9 Temasek Boulevard
#29-01, Suntec Tower Two
Singapore 038989

ABOUT ARISTA NETWORKS

Arista Networks was founded to deliver software-defined cloud networking solutions for large data center and computing environments. The award-winning Arista 10 Gigabit Ethernet switches redefine scalability, robustness, and price-performance. More than one million cloud networking ports are deployed worldwide. The core of the Arista platform is the Extensible Operating System (EOS®), the world's most advanced network operating system. Arista Networks products are available worldwide through distribution partners, systems integrators, and resellers.

Additional information and resources can be found at www.aristanetworks.com.